Meshfree and particle methods and their applications

Shaofan Li

Department of Civil & Environmental Engineering, University of California, Berkeley CA 94720; li@ce.berkeley.edu

Wing Kam Liu

Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Rd, Evanston IL 60208; w-liu@northwestern.edu

Recent developments of meshfree and particle methods and their applications in applied mechanics are surveyed. Three major methodologies have been reviewed. First, smoothed particle hydrodynamics (SPH) is discussed as a representative of a non-local kernel, strong form collocation approach. Second, mesh-free Galerkin methods, which have been an active research area in recent years, are reviewed. Third, some applications of molecular dynamics (MD) in applied mechanics are discussed. The emphases of this survey are placed on simulations of finite deformations, fracture, strain localization of solids; incompressible as well as compressible flows; and applications of multiscale methods and nano-scale mechanics. This review article includes 397 references. [DOI: 10.1115/1.1431547]

1 INTRODUCTION

Since the invention of the finite element method (FEM) in the 1950s, FEM has become the most popular and widely used method in engineering computations. A salient feature of the FEM is that it divides a continuum into discrete elements. This subdivision is called discretization. In FEM, the individual elements are connected together by a topological map, which is usually called a mesh. The finite element interpolation functions are then built upon the mesh, which ensures the compatibility of the interpolation. However, this procedure is not always advantageous, because the numerical compatibility condition is not the same as the physical compatibility condition of a continuum. For instance, in a Lagrangian type of computations, one may experience mesh distortion, which can either end the computation altogether or result in drastic deterioration of accuracy. In addition, FEM often requires a very fine mesh in problems with high gradients or a distinct local character, which can be computationally expensive. For this reason, adaptive FEM has become a necessity.

Today, adaptive remeshing procedures for simulations of impact/penetration problems, explosion/fragmentation problems, flow pass obstacles, and fluid-structure interaction problems *etc* have become formidable tasks to undertake. The difficulties involved are not only remeshing, but also mapping the state variables from the old mesh to the new mesh. This process often introduces numerical errors, and frequent remeshing is thus not desirable. Therefore, the so called Arbitrary Lagrangian Eulerian (ALE) formulations have been developed (see, *eg* [1–4]). For a complete description on this subject, readers may consult Chapter 7 of the book by Belytschko, Liu, and Moran [5]. The objective of the ALE formulation is to make the mesh independent of the material so that the mesh distortion can be minimized. Unfortunately, in computer simulations of very large deformation and/or high-speed mechanical and structural systems, even with the ALE formulation, a distorted mesh introduces severe errors in numerical computations. Furthermore, the convective transport effects in ALE often lead to spurious oscillation that needs to be stabilized by artificial diffusion or a Petrov-Galerkin stabilization. In other cases, a mesh may carry inherent bias in numerical simulations, and its presence becomes a nuisance in computations. A well known example is the simulation of the strain localization problem, which is notorious for its mesh alignment sensitivity [6,7]. Therefore, it would be computationally efficacious to discretize a continuum by only a set of nodal points, or particles, without mesh constraints. This is the *leitmotif* of contemporary meshfree Galerkin methods.

The advantages of the meshfree particle methods may be summarized as follows:

1) They can easily handle very large deformations, since the connectivity among nodes is generated as part of the computation and can change with time;
2) The methodology can be linked more easily with a CAD database than finite elements, since it is not necessary to generate an element mesh;
3) The method can easily handle damage of the components, such as fracture, which should prove very useful in modelings of material failure;

Transmitted by Associate Editor JN Reddy

4) Accuracy can be controlled more easily, since in areas where more refinement is needed, nodes can be added quite easily (h-adaptivity);
5) The continuum meshfree methods can be used to model large deformations of thin shell structures, such as nanotubes;
6) The method can incorporate an enrichment of fine scale solutions of features, such as discontinuities as a function of current stress states, into the coarse scale; and
7) Meshfree discretization can provide accurate representation of geometric object.

In general, particle methods can be classified based on two different criteria: physical principles, or computational formulations. According to the physical modeling, they may be categorized into two classes: those based on deterministic models, and those based on probabilistic models. On the other hand, according to computational modelings, they may be categorized into two different types as well: those serving as approximations of the strong forms of partial differential equations (PDEs), and those serving as approximations of the weak forms of PDEs. In this survey, the classification based on computational strategies is adopted.

To approximate the strong form of a PDE using a particle method, the partial differential equation is usually discretized by a specific collocation technique. Examples are smoothed particle hydrodynamics (SPH) [8–12], the vortex method [13–18], the generalized finite difference method [19,20], and many others. It is worth mentioning that some particle methods, such as SPH and vortex methods, were initially developed as probabilistic methods [10,14], and it turns out that both SPH and the vortex method are most frequently used as deterministic methods today. Nevertheless, the majority of particle methods in this category are based on probabilistic principles, or used as probabilistic simulation tools. There are three major methods in this category: 1) molecular dynamics (both quantum molecular dynamics [21–26] and classical molecular dynamics [27–32]); 2) direct simulation Monte Carlo (DSMC), or Monte Carlo method based molecular dynamics, such as quantum Monte Carlo methods [33–41]) (It is noted that not all the Monte Carlo methods are meshfree methods, for instance, a probabilistic finite element method is a mesh-based method [42–44]); and 3) the lattice gas automaton (LGA), or lattice gas cellular automaton [45–49] and its later derivative, the Lattice Boltzmann Equation method (LBE) [50–54]. It may be pointed out that the Lattice Boltzmann Equation method is not a meshfree method, and it requires a grid; this example shows that particle methods are not always meshfree.

The second class of particle methods is used with various Galerkin weak formulations, which are called meshfree Galerkin methods. Examples in this class are Diffuse Element Method (DEM) [55–58], Element Free Galerkin Method (EFGM) [59–63], Reproducing Kernel Particle Method (RKPM) [64–72], h-p Cloud Method [73–76], Partition of Unity Method [77–79], Meshless Local Petrov-Galerkin Method (MLPG) [80–83], Free Mesh Method [84–88], and others.

There are exceptions to this classification, because some particle methods can be used in both strong form collocation as well as weak form discretization. The particle-in-cell (PIC) method is such an exception. The strong form collocation PIC is often called the finite-volume particle-in-cell method [89–91], and the weak form PIC is often called the material point method [92], or simply particle-in-cell method [93–95]. RKPM also has two versions as well: a collocation version [96] and a Galerkin weak form version [66].

In areas such as astrophysics, solid state physics, biophysics, biochemistry and biomedical research, one may encounter situations where the object under consideration is not a continuum, but a set of particles. There is no need for discretization to begin with. A particle method is the natural choice in numerical simulations. Relevant examples are the simulation of formation of a star system, the nano-scale movement of millions of atoms in a non-equilibrium state, folding and unfolding of DNA, and dynamic interactions of various molecules, *etc*. In fact, the current trend is not only to use particle methods as discretization tools to solve continuum problems (such as SPH, vortex method [14,15,97] and meshfree Galerkin methods), but also to use particle methods as a physical model (statistical model, or atomistic model) to simulate continuum behavior of physics. The latest examples are using the Lattice Boltzmann method to solve fluid mechanics problems, and using molecular dynamics to solve fracture mechanics problems in solid mechanics [98–103].

This survey is organized as follows: The first part is a critical review of smoothed particle hydrodynamics (SPH). The emphasis is placed on the recent development of corrective SPH. The second part is a summary of meshfree Galerkin methods, which includes DEM, EFGM, RKPM, hp-Cloud method, partition of unity method, MLPGM, and meshfree nodal integration methods. The third part reviews recent applications of molecular dynamics in fracture mechanics as well as nanomechanics. The last part is a survey on some other meshfree/particle methods, such as vortex methods, the Lattice Boltzmann method, the natural element method, the particle-in-cell method, *etc*. The survey is concluded with the discussions of some emerging meshfree/particle methods.

2 SMOOTHED PARTICLE HYDRODYNAMICS

2.1 Overview

Smoothed Particle Hydrodynamics is one of the earliest particle methods in computational mechanics. Early contributions have been reviewed in several articles [8,12,104]. In 1977, Lucy [10] and Gingold and Monaghan [9] simultaneously formulated the so-called Smoothed Particle Hydrodynamics, which is known today as SPH. Both of them were interested in the astrophysical problems, such as the formation and evolution of proto-stars or galaxies. The collective movement of those particles is similar to the movement of a liquid, or gas flow, and it may be modeled by the governing equations of classical Newtonian hydrodynamics. Today, SPH is being used in simulations of supernovas [105], col-

lapse as well as formation of galaxies [106–109], coalescence of black holes with neutron stars [110,111], single and multiple detonations in white dwarfs [112], and even in ''Modeling the Universe'' [113]. Because of the distinct advantages of the particle method, soon after its debut, the SPH method was widely adopted as one of the efficient computational techniques to solve applied mechanics problems. Therefore, the term *hydrodynamics* really should be interpreted as *mechanics* in general, if the methodology is applied to other branches of mechanics rather than classical hydrodynamics. To make distinction with the classical hydrodynamics, some authors, *eg* Kum *et al* [114,115], called it *Smoothed Particle Applied Mechanics*.

This idea of the method is somewhat contrary to the concepts of the conventional discretization methods, which discretize a continuum system into a discrete algebraic system. In astrophysical applications, the real physical system is discrete; in order to avoid singularity, a local continuous field is generated by introducing a localized kernel function, which can serve as a smoothing interpolation field. If one wishes to interpret the physical meaning of the kernel function as the probability of a particle's position, one is dealing with a probabilistic method. Otherwise, it is only a smoothing technique. Thus, the essence of the method is to choose a smooth kernel, $W(\mathbf{x},h)$ (h is the smoothing length), and to use it to localize *the strong form* of a partial differential equation through a convoluted integration. Define SPH averaging/localization operator as

$$A_k(\mathbf{x}) = \langle A(\mathbf{x}) \rangle = \int_{\mathbb{R}^n} W(\mathbf{x}-\mathbf{x}',h) A(\mathbf{x}') d\Omega_{\mathbf{x}'} \approx \sum_{I=1}^{N} W(\mathbf{x}-\mathbf{x}_I,h) A(\mathbf{x}_I) \Delta V_I \tag{1}$$

One may derive a SPH discrete equation of motion from its continuous counterpart [12,116],

$$\left\langle \rho \frac{d\mathbf{v}}{dt} \right\rangle_I = -\langle \nabla \cdot \boldsymbol{\sigma} \rangle_I \Rightarrow \rho_I \frac{d\mathbf{v}_I}{dt} \approx -\sum_{J=1}^{N} (\boldsymbol{\sigma}_I + \boldsymbol{\sigma}_J) \cdot \nabla W(\mathbf{x}_I - \mathbf{x}_J, h) \Delta V_J \tag{2}$$

where $\boldsymbol{\sigma}$ is Cauchy stress, ρ is density, $\mathbf{v}$ is velocity, and ΔV_J is the volume element carried by the particle J.

Usually a positive function, such as the Gaussian function, is chosen as the kernel function

$$W(\mathbf{x},h) = \frac{1}{(\pi h^2)^{n/2}} \exp\left[-\frac{\mathbf{x}^2}{h^2} \right], \quad 1 \leqslant n \leqslant 3 \tag{3}$$

where the parameter h is the smoothing length. In general, the kernel function has to satisfy the following conditions:

$$i) \quad W(\mathbf{x},h) \geqslant 0 \tag{4}$$

$$ii) \quad \int_{\mathbb{R}^n} W(\mathbf{u},h) d\Omega_{\mathbf{u}} = 1 \tag{5}$$

$$iii) \quad W(\mathbf{u},h) \to \delta(\mathbf{u}), \quad h \to 0 \tag{6}$$

$$iv) \quad W(\mathbf{u},h) \in C^p(\mathbb{R}^n), \quad p \geqslant 1 \tag{7}$$

The third property ensures the convergence, and the last property comes from the requirement that the smoothing kernel must be differentiable at least once. This is because the derivative of the kernel function should be continuous to prevent a large fluctuation in the force felt by the particle. The latter feature gives rise to the name *smoothed* particle hydrodynamics.

In computations, compact supported kernel functions such as spline functions are usually employed [117]. In this case, the smoothing length becomes the radius of the compact support. Two examples of smooth kernel functions are depicted in Fig. 1.

The advantage of using an analytical kernel is that one can evaluate a kernel function at any spatial point without knowing the local particle distribution. This is no longer true for the latest corrective smoothed particle hydrodynamics methods [66,118], because the corrective kernel function depends on the local particle distribution.

The kernel representation is not only an instrument that can smoothly discretize a partial differential equation, but it also furnishes an interpolant scheme on a set of moving particles. By utilizing this property, SPH can serve as a Lagrangian type method to solve problems in continuum mechanics. Libersky and his co-workers apply the method to solid mechanics [117,119,120], and they successfully simulate 3D thick-wall bomb explosion/fragmentation problem, tungsten/plate impact/penetration problem, *etc*. The impact and penetration simulation has also been conducted by Johnson and his co-workers [121–123], and an SPH option is implemented in EPIC code for modeling inelastic, damage, large deformation problems. Attaway *et al* [124] developed a coupling technique to combine SPH with the finite element method, and an SPH option is also included in PRONTO 2D (Taylor and Flanagan [125]).

SPH technology has been employed to solve problems of both compressible flow [126] and incompressible flow

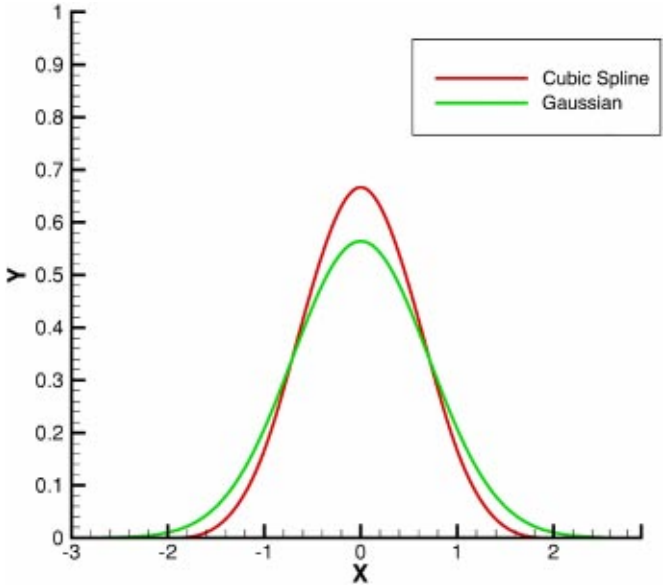

Fig. 1 Examples of kernel functions

[116,127–129], multiple phase flow and surface tension [114,115,129,130,131,132,133], heat conduction [134], electro-magnetic (Maxwell equations) [90,104,135], plasma/fluid motion [135], general relativistic hydrodynamics [136–138], heat conduction [134,139], and nonlinear dynamics [140].

2.2 Corrective SPH and other improvements in SPH formulations

Various improvements of SPH have been developed through the years [104,141–149]. Most of these improvement are aimed at the following shortcomings, or pathologies, in numerical computations:

- tensile instability [150–154];
- lack of interpolation consistency, or completeness [66,155,156];
- zero-energy mode [157];
- difficulty in enforcing essential boundary condition [120,128,131].

2.2.1 Tensile instability

So-called tensile instability is the situation where particles are under a certain tensile (hydrostatic) stress state, and the motion of the particles become unstable. To identify the culprit, a von Neumann stability analysis was carried out by Swegle *et al* [150], and by Balsara [158]. Swegle and his co-workers have identified and explained the source of the tensile instability. Recently, by using von Neumann and Courant stability criterion, Belytschko *et al* [151] revisited the problem in the general framework of meshfree particle methods. In their analysis, finite deformation effects are also considered.

Several remedies have been proposed to avoid such tensile instability. Morris proposed using special kernel functions. While successful in some cases, they do not always yield satisfactory results [152]. Randles and Libersky [120] proposed adding dissipative terms, which is related to conservative smoothing. Notably, Dyka *et al* [153,154] proposed a so-called stress point method. The essential idea of this approach is to add additional points other than SPH particles when evaluating, or sampling, stress and other state variables. Whereas the kinematic variables such as displacement, velocity, and acceleration are still sampled at particle points. In fact, the stress point plays a similar role as the ''Gauss quadrature point'' does in the numerical integration of the Galerkin weak form. This analogy was first pointed out by Liu *et al* [66]. This problem was revisited again recently by Chen *et al* [159] as well as Monaghan [148]. The former proposes a special corrective smoothed-particle method (CSPM) to address the tensile instability problem by enforcing the higher order consistency, and the latter proposes to add an artificial force to stabilize the computation. Randles and Libersky [160] combined normalization with the usual stress point approach to achieve better stability as well as linear consistency. Apparently, the SPH tensile instability is related to the lack of consistency of the SPH interpolant. A 2D stress point deployment is shown in Fig. 2.

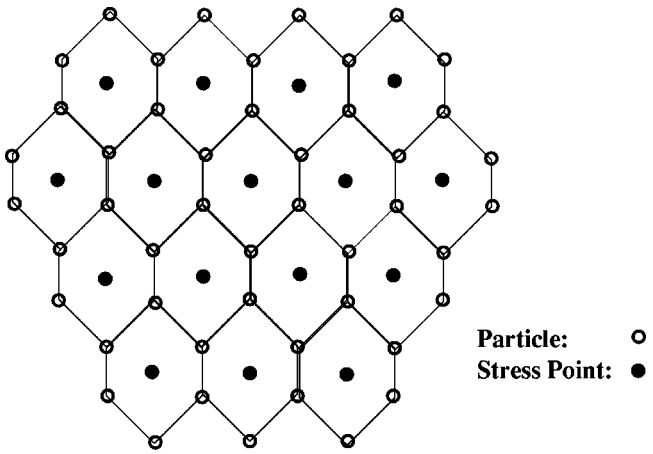

Fig. 2 A 2D Stress point distribution

2.2.2 Zero-energy mode

The zero energy mode has been discovered in both finite difference and finite element computations. A comprehensive discussion of the subject can be found in the book by Belytschko *et al* [5]. The reason that SPH suffers similar zero energy mode deficiency is due to the fact that the derivatives of kinematic variables are evaluated at particle points by analytical differentiation rather than by differentiation of interpolants. In many cases, the kernel function reaches a maximum at its nodal position, and its spatial derivatives become zero. To avoid a zero-energy mode, or spurious stress oscillation, an efficient remedy is to adopt the stress point approach [157].

2.2.3 Corrective SPH

As an interpolation among moving particles, SPH is not a partition of unity, which means that SPH interpolants cannot represent rigid body motion correctly. This problem was first noticed by Liu *et al* [64–66]. They then set forth a key notion, a correction function, which has become the central theme of the so-called corrective SPH. The idea of a corrective SPH is to construct a corrective kernel, a product of the correction function with the original kernel. By doing so, the consistency, or completeness, of the SPH interpolant can be enforced. This new interpolant is named the reproducing kernel particle method [64–66].

SPH kernel functions satisfy zero-th order moment condition (5). Most kernel functions satisfy higher order moment condition as well [104], for instance

$$\int_R xW(x,h)dx=0. \tag{8}$$

These conditions only hold in the continuous form. In general they are not valid after discretization, *ie*

$$\sum_{I=1}^{NP} W(\mathbf{x}-\mathbf{x}_I,h)\Delta x_I \neq 1 \tag{9}$$

$$\sum_{I=1}^{NP} (\mathbf{x}-\mathbf{x}_I)W(\mathbf{x}-\mathbf{x}_I,h)\Delta x_I \neq 0 \tag{10}$$

where NP is the total number of the particles. Note that condition (9) is the condition of partition of unity. Since the

kernel function can not satisfy the discrete moment conditions, a modified kernel function is introduced to enforce the discrete consistency conditions

$$\tilde{W}_h(x-x_I;x)=C_h(x-x_I;x)W(x-x_I,h) \tag{11}$$

where $C_h(x;x-x_I)$ is the correction function, which can be expressed as

$$C_h(x;x-x_I)=b_0(x,h)+b_1(x,h)\frac{x-x_I}{h}+b_2(x,h)\times\left(\frac{x-x_I}{h}\right)^2+\cdots\cdots \tag{12}$$

where $b_0(x),b_1(x),\cdots,b_n(x)$ are unknown functions. We can determine them to correct the original kernel function. Suppose $f(x)$ is a sufficiently smooth function. By Taylor expansion,

$$f_I=f(x_I)=f(x)+f'(x)\left(\frac{x_I-x}{h}\right)h+\frac{f''(x)}{2!}\left(\frac{x_I-x}{h}\right)^2h^2+\cdots\cdots \tag{13}$$

the modified kernel approximation can be written as,

$$\begin{aligned}f^h(x)&=\sum_{I=1}^{NP}\tilde{W}_h(x-x_I;x)f_I\Delta x_I\\&=\left(\sum_{I=1}^{NP}\tilde{W}_h(x-x_I,x)\Delta x_I\right)f(x)h^0\\&\quad-\left(\sum_{I=1}^{NP}\left(\frac{x-x_I}{h}\right)\tilde{W}_h(x-x_I,x)\Delta x_I\right)f'(x)h\\&\quad+\cdots\cdots+\left(\sum_{I=1}^{NP}(-1)^n\left(\frac{x-x_I}{h}\right)^n\tilde{W}_h\times(x-x_I,x)\Delta x_I\right)\frac{f^n(x)}{n!}h^n+\mathcal{O}(h^{n+1}).\end{aligned} \tag{14}$$

To obtain an n-th order reproducing condition, the moments of the modified kernel function must satisfy the following conditions:

$$\left.\begin{aligned}M_0(x)&=\sum_{I=1}^{NP}\tilde{W}_h(x-x_I,x)\Delta x_I=1;\\M_1(x)&=\sum_{I=1}^{NP}\left(\frac{x-x_I}{h}\right)\tilde{W}_h(x-x_I,x)\Delta x_I=0;\\&\vdots\\M_n(x)&=\sum_{I=1}^{NP}\left(\frac{x-x_I}{h}\right)^n\tilde{W}_h(x-x_I,x)\Delta x_I=0;\end{aligned}\right\} \tag{15}$$

Substituting the modified kernel expressions, (11) and (12) into Eq. (15), we can determine the $n+1$ coefficients, $b_i(x)$, by solving the following *moment equations*:

$$\begin{pmatrix}m_0(x)&m_1(x)&\cdots&m_n(x)\\m_1(x)&m_2(x)&\cdots&m_{n+1}(x)\\\vdots&\vdots&\vdots&\vdots\\m_n(x)&m_{n+1}(x)&\cdots&m_{2n}(x)\end{pmatrix}\begin{pmatrix}b_0(x,h)\\b_1(x,h)\\\vdots\\b_n(x,h)\end{pmatrix}=\begin{pmatrix}1\\0\\\vdots\\0\end{pmatrix} \tag{16}$$

It is worth mentioning that after introducing the correction function, the modified kernel function may not be a positive function anymore,

$$K(\mathbf{x}-\mathbf{x}_I)\ngeq 0. \tag{17}$$

Within the compact support, $K(\mathbf{x}-\mathbf{x}_I)$ may become negative. This is the reason why Duarte and Oden refer to it as *the signed partition of unity* [73,74,76].

There are other approaches to restoring completeness of the SPH approximation. Their emphases are not only consistency, but also on cost effectiveness. Using RKPM, or a moving-least-squares interpolant [155,156] to construct modified kernels, one has to know all the neighboring particles that are adjacent to a spatial point where the kernel function is in evaluation. This will require an additional CPU to search, update the connectivity array, and calculate the modified kernel function pointwise. It should be noted that the calculation of the modified kernel function requires pointwise matrix inversions at each time step, since particles are moving and the connectivity map is changing as well. Thus, using a moving least square interpolant as the kernel function may not be cost-effective, and it destroys the simplicity of SPH formulation.

Several compromises have been proposed throughout the years, which are listed as follows:

1) Monaghan's symmetrization on derivative approximation [104,145];
2) Johnson-Beissel correction [123];
3) Randles-Libersky correction [120];
4) Krongauz-Belytschko correction [61];
5) Chen-Beraun correction [139,140,161];
6) Bonet-Kulasegaram integration correction [118];
7) Aluru's collocation RKPM [96].

Since the linear reproducing condition in the interpolation is equivalent to the constant reproducing condition in the derivative of the interpolant, some of the algorithms directly correct derivatives instead of the interpolant. The Chen-Beraun correction corrects even higher order derivatives, but it may require more computational effort in multi-dimensions.

Completeness, or consistency, closely relates to convergence. There are two types of error estimates: interpolation error and the error between exact solution and the numerical solution. The former usually dictates the latter. In conventional SPH formulations, there is no requirement for the completeness of interpolation. The particle distribution is assumed to be randomly distributed and the summations are

Monte Carlo estimates of integral interpolants. The error of random interpolation was first estimated by Niedereiter [162] as being $\propto N^{-1}\log N^{n-1}$ where N is total particle number and n is the dimension of space. This result was further improved by Wozniakowski [163] as being $\propto N^{-1}\log N^{n-1/2}$. According to reference [104], ''this remarkable result was produced by a challenge with a payoff of sixty-four dollars !'' Twenty-one years after its invention, in 1998 Di Lisio *et al* [164] gave a convergence proof of smoothed particle hydrodynamics method for regularized Euler flow equations.

Besides consistency conditions, the conservation properties of a SPH formulation also strongly influence its performance. This has been a critical theme throughout SPH research, see [12,104,120,145,155,165]. It is well known that classical SPH enjoys Galilean invariance, and if certain derivative approximations, or Golden rules as Monaghan puts it, are chosen, the corresponding SPH formulations can preserve some discrete conservation laws. This issue was recently revisited by Bonet *et al* [166], and they set forth a discrete variational SPH formulation, which can automatically satisfy the balance of linear momentum and balance of angular momentum conservation laws. Here is the basic idea. Assume the discrete potential energy in a SPH system is

$$\Pi(\mathbf{x})=\sum_I V_I^0 U(J_I) \tag{18}$$

where V_I^0 is the initial volume element, and $U(J_I)$ is the internal energy density, which is assumed to be the function of determinant of the Jacobian—ratio between the initial and current volume element,

$$J=\frac{V_I}{V_I^0}=\frac{\rho_I^0}{\rho_I} \tag{19}$$

where ρ_I^0 and ρ_I are pointwise density in initial configuration and in current configuration.

For adiabatic processes, the pressure can be obtained from $\partial U_I/\partial J=p_I$. Thus, the stationary condition of potential energy gives

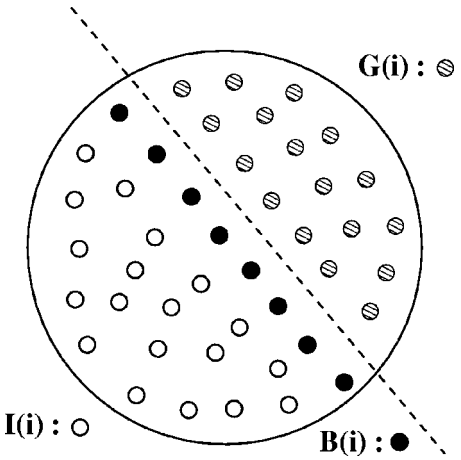

Fig. 3 The *Ghost particle* approach for boundary treatment

$$\delta\Pi=D\Pi[\delta\mathbf{v}]=\sum_I V_I^0 DU_I[\delta\mathbf{v}]$$

$$=\sum_I\left\{\sum_J m_I m_J\left(\frac{p_I}{\rho_I^2}+\frac{p_J}{\rho_J^2}\right)\nabla W_I(\mathbf{x}_J)\right\}\delta\mathbf{v}_I \tag{20}$$

where m_I is the mass associated with particle I.

On the other hand,

$$D\Pi[\delta\mathbf{v}]=\sum_I\frac{\partial\Pi}{\partial\mathbf{x}_I}\delta\mathbf{v}_I=\sum_I\mathbf{T}_I\cdot\delta\mathbf{v}_I \tag{21}$$

where $\mathbf{T}$ is the internal force (summation of stress). Then through the variational principle, one can identify,

$$\mathbf{T}_I=\sum_I m_I m_J\left(\frac{p_I}{\rho_I^2}+\frac{p_J}{\rho_J^2}\right)\nabla W_I(\mathbf{x}_J) \tag{22}$$

and establish the discrete SPH equation of motion (balance of linear momentum),

$$m_I\frac{d\mathbf{v}_I}{dt}=-\sum_I m_I m_J\left(\frac{p_I}{\rho_I^2}+\frac{p_J}{\rho_J^2}\right)\nabla W_I(\mathbf{x}_J). \tag{23}$$

2.2.4 Boundary conditions

SPH, and in fact particle methods in general, have difficulties in enforcing essential boundary condition. For SPH, some effort has been devoted to address the issue. Takeda's image particle method [131] is designed to satisfy the no-slip boundary condition; it is further generalized by Morris *et al* [128] to satisfy boundary conditions along a curved boundary. Based on the same philosophy, Randles and Libersky [120] proposed a so-called *ghost particle* approach, which is outlined as follows: Suppose particle **i** is a boundary particle. All the other particles within its support, **N(i)**, can be divided into three subsets:

1) **I(i)**: all the interior points that are the neighbors of **i**;
2) **B(i)**: all the boundary points that are the neighbors of **i**;
3) **G(i)**: all the exterior points that are the neighbors of **i**, *ie*, all the ghost particles.

Therefore $\mathbf{N(i)}=\mathbf{I(i)}\cup\mathbf{B(i)}\cup\mathbf{G(i)}$. Figure 3 illustrates such an arrangement.

In the ghost particle approach, the boundary correction formula for general scalar field f is given as follows

$$f_i=f_{bc}+\frac{\sum_{j\in\mathbf{I(i)}}(f_j-f_{bc})\Delta V_j W_{ij}}{\left(1-\sum_{j\in\mathbf{B(i)}}\Delta V_j W_{ij}\right)} \tag{24}$$

where f_{bc} is the prescribed boundary value at $x=x_i$. One of the advantages of the above formula is that the sampling formula only depends on interior particles.

2.3 Other related issues and applications

Besides resolving the above fundamental issues, there have been some other progresses in improving the performance of SPH, which have focused on applications as well as algorithmic efficiency. How to choose an interpolation kernel to ensure successful simulations is discussed in [167]; how to modify the kernel functions without correction is discussed

in [168,169]; and how to use SPH to compute incompressible flow, and to force incompressibility conditions are studied in [126]. How to use SPH to simulate contact is revisited by Campell *et al* [170], which is critical in SPH impact/fragmentation simulation. In astrophysics, the SPH method is now used in some very complex computations, including simulations of various protostellar encounters [171–174], dissipative formation of elliptical galaxies, supernova feedback, and thermal instability of galaxies [105,175].

By considering a smoothing operator as a filter, it has been found that an adaptive smoothing filter is an efficient tool to resolve large-scale structure (astrophysical problems) as well as small-scale structure (micro-mechanics problems). Owen [176,177] has recently developed an adaptive SPH (ASPH) technique—an anisotropic smoothing algorithm which uses an ellipsoidal kernel function with a tensor smoothing length to replace the traditional isotropic (or spherical) kernel function with a scalar smoothing length. The method has been tested in various computations, *eg* cosmological pancake collapse, the Riemann shock tube, Sedov blast waves, the collision of two strong shock waves. Seto [178] used perturbation theory to adjust adaptive parameters in SPH formulation to count the fluctuations present in a statistical environment.

Much effort has been devoted to develop parallelization of SPH. Dave *et al* [179] developed a parallelized code based on TreeSPH, which is a unification of conventional SPH with the hierarchical tree method [180]. The parallel protocol of TreeSPH is called PTreeSPH. Using a message passing interface (MPI), it is executed through a domain decomposition procedure and a synchronous hypercube communication paradigm to build self-contained subvolumes of the simulation on each processor at every time step. When used on Cray T3D, it can achieve a communications overhead of $\sim 8\%$ and load balanced up to 95%, while dealing with up to 10^7 particles in specific astrophysics simulations. Recently, Lia and Carraro [181] also presented their version of parallel TreeSPH implementation, which has been used in the simulation of the formation of an X-ray galaxy cluster in a flat cold dark matter cosmology. In solid mechanics applications, Plimpton and his co-workers [182] have implemented a parallelization of a multi-physics code PRONTO-3D, which combines transient structural dynamics with smoothed particle hydrodynamics, and they have carried out some simulations of complex impact and explosions in coupled structure/fluid systems.

The traditional Newtonian SPH has been generalized to the form of general relativistic hydrodynamic equations for perfect fluids with artificial viscosity in a given arbitrary space-time background [136,138]. With this formulation, both Chow and Monaghan [136] and Siegler *et al* have simulated [138] ultrarelativistic shocks with relativistic velocities up to 0.9999 the speed of light. On the small scale end, SPH methodology has been used in simulation of cohesive grains. Recently, both Gutfraind *et al* [183] and Oger *et al* [184] used SPH to simulate a broken-ice field floating on water under the influence of wind. The broken-ice field is simulated as a cohesive material with rheology based on the Mohr-Coulomb yield criterion. In comparison with the classical Lagrangian method, it has been found that SPH can eliminate problems of artificial diffusion at the free boundaries of the ice region, and it can handle discontinuities at the free surface and also the cohesive effects between moving particles by proper choice of the kernel functions. Moreover, Gutfraind *et al* [185] have been trying to connect SPH with discrete-element method to make a particle-cohesive model.

Birnbaum *et al* [186] recently tested a coupling technique between SPH with the Lagrangian finite element method as well as with the arbitrary Lagrangian Eulerian finite element method to simulate fluid-structural interaction problems, which is called the SPH-Lagrange coupling technique. Instead of forming smoothed hydrodynamics from strong forms of the governing equation, Fahrenthold and Koo [187] argued that one may form a hydrodynamics directly from the Hamiltonian of the mechanical system. By doing so, one may end up with discrete equations that will have an intrinsic energy conserving property. An example was given in [187] to solve a wall shock problem.

3 MESH-FREE GALERKIN METHODS

There have been several review articles on meshfree Galerkin methods, *eg*, [60,68], and two special issues are devoted to meshfree Galerkin methods (*Computer Methods in Applied Mechanics and Engineering*, Vol 139, 1996; *Computational Mechanics*, Vol. 25, 2000). The focus of this review is placed on the latest developments and perspectives that are different from previous surveys.

3.1 Overviews

Unlike SPH, meshfree Galerkin methods are relatively young. In the early 1990s, there were several research groups, primarily the French group (P Villon, B Nayroles, G Touzot) and the Northwestern group (T Belytschko and W K Liu) who were looking for either meshless interpolants [55,57,58] to relieve the heavy burden of structured mesh generation that is required in traditional finite element refinement process, or interpolants having multiple scale computation capability [64,65,188]. Nayroles *et al* basically rediscovered the moving least square interpolant derived in a landmark paper by Lancaster and Salkauskas [189]. Foreseeing its potential use in numerical computations, they named it the diffuse element method (DEM). Meanwhile, Liu *et al* [64–66,188] derived the so-called reproducing kernel particle interpolant in an attempt to construct a corrective SPH interpolant.

Then in 1994, another landmark paper was published by Belytschko, Lu, and Gu [59], in which the MLS interpolant was used in the first time in a Galerkin procedure. Belytschko *et al* formed a variational formulation to accommodate the interpolant to solve linear elastic problems, specifically the fracture and crack growth problems [63,190–192]. The authors named their method the element free Galerkin method. Meanwhile, Liu and his co-workers used the reproducing kernel particle interpolant, which is an advanced version of the MLS interpolant, to solve structural dynamics problems [66,193].

Meshfree interpolants are constructed among a set of scattered particles that have no particular topological connection among them. The commonly-used meshfree interpolations are constructed by a data fitting algorithm that is based on the *inverse distance weighted* principle. The most primitive one of the kind is the well-known Shepard's interpolant [194]. In the Shepard's method, one chooses a decaying positive window function $w(x)>0$, and interpolate only arbitrary function, $f(x)$, as

$$f^h(x)=\sum_{i=1}^{N} f_i \frac{w(x-x_i)}{\sum_{i=1}^{N} w(x-x_i)} \tag{25}$$

where the decaying positive window function, $w(x-x_i)$, localizes around x_i. The Shepard's interpolant then has the form

$$\phi_i(x)=\frac{w(x-x_i)}{\sum_{i=1}^{N} w(x-x_i)} \tag{26}$$

Obviously, $\Sigma_{i=1}^{N}\phi_i(x)=1$, *ie* Shepard's interpolant is a partition of unity, hence the interpolant reproduces a constant. Note that the partition of unity condition is a discrete summation, which may be viewed as normalized zero-th order discrete moments. To generalize Shepard's interpolant, one needs to normalize higher order discrete moments of the basis function. There are two approaches to generalize Shepard's interpolant: 1) moving least square interpolant by Lancaster and Salkauskas [189]; and 2) moving least square reproducing kernel by Liu, Li and Belytschko [70]. The procedures look alike, but subtleties remain. For instance, without employing the *shifted* basis, ill-conditioning may arise in the stiffness matrix.

The reproducing kernel interpolant may be interpreted as a moving least square interpolant, if one chooses the following shifted local basis

$$f^h(\mathbf{x},\overline{\mathbf{x}})=\sum_{i=1}^{n+1} P_i(\overline{\mathbf{x}}-\mathbf{x})b_i(\mathbf{x})=\mathbf{P}(\overline{\mathbf{x}}-\mathbf{x})\mathbf{b}(\overline{\mathbf{x}}) \tag{27}$$

where $\mathbf{b}=(b_1(\mathbf{x}),b_2(\mathbf{x}),\cdots,b_{n+1}(\mathbf{x}))^T$ and $\mathbf{P}(\mathbf{x})=(P_1(\mathbf{x}),P_2(\mathbf{x}),\cdots,P_{n+1}(\mathbf{x}))$, $P_i(\mathbf{x})\in C^{n+1}(\Omega)$. One may notice that there is a difference between Eq. (27) and the orginal choice of the local approximation by Lancaster and Salkauskas [189] or Belytschko *et al* [59]. To determine the unknown vector $\mathbf{b}(\mathbf{x})$, we minimize the local interpolation error

$$J(\mathbf{b}(\overline{\mathbf{x}}))=\sum_{I=1}^{NP} \Phi(\overline{\mathbf{x}}-\mathbf{x}_I)[\mathbf{P}(\overline{\mathbf{x}}-\mathbf{x}_I)\mathbf{b}(\overline{\mathbf{x}})-f(\mathbf{x}_I)]^2\Delta V_I \tag{28}$$

such that

$$\begin{aligned}\frac{\partial J}{\partial \mathbf{b}}&=2\sum_{I\in\Lambda}\mathbf{P}^T(\overline{\mathbf{x}}-\mathbf{x}_I)\Phi(\overline{\mathbf{x}}-\mathbf{x}_I)\\&\quad\times[\mathbf{P}(\overline{\mathbf{x}}-\mathbf{x}_I)\mathbf{b}(\overline{\mathbf{x}})-f(\mathbf{x}_I)]\Delta V_I\\&=0.\end{aligned} \tag{29}$$

Let

$$\mathbf{M}(\overline{\mathbf{x}}):=\sum_{I\in\Lambda}\mathbf{P}^T(\overline{\mathbf{x}}-\mathbf{x}_I)\Phi(\overline{\mathbf{x}}-\mathbf{x}_I)\mathbf{P}(\overline{\mathbf{x}}-\mathbf{x}_I)\Delta V_I. \tag{30}$$

One can obtain $\mathbf{b}(\overline{\mathbf{x}})=\mathbf{M}^{-1}(\overline{\mathbf{x}})\Sigma_{I\in\Lambda}\mathbf{P}^T(\overline{\mathbf{x}}-\mathbf{x}_I)\Phi(\overline{\mathbf{x}}-\mathbf{x}_I)\Delta V_I f(\mathbf{x}_I)$. Then the modified local kernel function would be $\tilde{\mathbf{W}}(\overline{\mathbf{x}})=\mathbf{P}(\overline{\mathbf{x}}-\mathbf{x})\mathbf{M}^{-1}(\overline{\mathbf{x}})\Sigma_{I\in\Lambda}\mathbf{P}^T(\overline{\mathbf{x}}-\mathbf{x}_I)\Phi(\overline{\mathbf{x}}-\mathbf{x}_I)\Delta V_I$. To this end, only a standard least square procedure has been used, to complete the process, one has to move the fixed point $\overline{\mathbf{x}}$ to any point $\mathbf{x}\in\Omega$; this is why the method is called *moving* least square method. By so doing, the corrective kernel becomes

$$\begin{aligned}\tilde{W}_I(\mathbf{x})&=\lim_{\overline{\mathbf{x}}\to\mathbf{x}}=\mathbf{P}(0)\mathbf{M}^{-1}(\mathbf{x})\mathbf{P}^T(\mathbf{x}-\mathbf{x}_I)\Phi(\mathbf{x}-\mathbf{x}_I)\Delta V_I,\\&\quad I\in\Lambda.\end{aligned} \tag{31}$$

If we let $\mathbf{P}=(1,x,x^2,\cdots,x^{n+1})$, the moving least square interpolant is exactly the same as reproducing kernel interpolant. For comparison, the Lancaster-Salkauskas interpolant is listed as follows

$$K_I(\mathbf{x})=\mathbf{P}(\mathbf{x})\mathbf{M}^{-1}(\mathbf{x})\mathbf{P}^T(\mathbf{x}_I)\Phi(\mathbf{x}-\mathbf{x}_I),\quad I\in\Lambda. \tag{32}$$

Two things are obviously different: 1) Lancaster and Salkauskas did not use the shifted basis, or local basis, and 2) they used $\Delta V_I=1$ for all particles. In our experience, the variable weight is more accurate than the uniform weight, especially along boundaries.

There has been a conjecture that Eqs. (31) and (32) are equivalent. In general, this may not be true, because interpolant (31) can reproduce basis vector $\mathbf{P}$ globally, if only P_i is monomial [70]. For general bases, such as $\mathbf{P}(x)=\{1,\sin(x),\sin(2x)\}$, the global basis may differ from the local basis. To show the global reproducing property of (32) [66], let $\mathbf{f}(\mathbf{x})=\mathbf{P}(\mathbf{x})$

$$\begin{aligned}\sum_{I\in\Lambda}K_I(\mathbf{x})\mathbf{f}_I&=\sum_{I\in\Lambda}K_I(\mathbf{x})\mathbf{P}(\mathbf{x}_I)\\&=\mathbf{P}(\mathbf{x})\mathbf{M}^{-1}(\mathbf{x})\sum_{I\in\Lambda}\mathbf{P}^T(\mathbf{x}_I)\Phi(\mathbf{x}-\mathbf{x}_I)\mathbf{P}(\mathbf{x}_I)\\&=\mathbf{P}(\mathbf{x}).\end{aligned} \tag{33}$$

A variation of the above prescription is that the basis vector $\mathbf{P}$ need not be polynomial, and it can include other independent basis functions as well such as trigonometric functions. Utilizing the reproducing property, Belytschko *et al* [195] and Fleming [196] used the following basis to approximate crack tip displacement field,

$$\begin{aligned}\mathbf{P}(\mathbf{x})=\Big[&1,x,y,\sqrt{r}\cos\frac{\theta}{2},\sqrt{r}\sin\frac{\theta}{2}\\&+\sqrt{r}\sin\frac{\theta}{2}\sin\theta,\sqrt{r}\cos\frac{\theta}{2}\sin\theta\Big].\end{aligned} \tag{34}$$

The same trigonometric basis was used again by Rao and Rahman [197] in fracture mechanics. The similar bases,

$$\mathbf{P}(\mathbf{x})=\{1,\cos(kx),\sin(kx)\} \tag{35}$$

$$\mathbf{P}(\mathbf{x})=\{1,\cos(kx\cos\theta+ky\sin\theta),\sin(kx\cos\theta+ky\sin\theta),\cos(-kx\sin\theta+ky\cos\theta),\sin(-kx\sin\theta+ky\cos\theta)\}, \tag{36}$$

are employed by Liu *et al* [198] in Fourier analysis of RKPM, and it is used in computational acoustics applications by Uras *et al* [199] and Suleau *et al* [200,201]. For given a wave number, k, the meshfree interpolant built upon the above bases reproduces desired mode function, and it is believed to be able to minimize dispersion error. A detailed analysis was performed by Bouillard *et al* [202] to assess the pollution error of EFG, when it is used to solve Helmholtz equations. It is worth mentioning that Christon and Voth [203] performed von Neumann analysis for reproducing kernel semi-discretization of both one and two-dimensional, first- and second-order hyperbolic differential equations. Excellent dispersion characteristics are found for the consistent mass matrix with the proper choice of dilation parameter. In contrast, row-sum lumped mass matrix is demonstrated to introduce lagging phase errors.

3.2 Completeness, convergence, adaptivity, and enrichment

The reproducing property of RKPM interpolant leads to a set of very interesting consistency conditions. Denote $\{K_I^\rho(\mathbf{x})\}$ as the basis of RKPM interpolant, the so-called m-th order consistency condition derived by Li *et al* in [70,204] reads as

$$\sum_I \mathbf{P}\left(\frac{\mathbf{x}-\mathbf{x}_I}{\rho}\right)K_I^\rho(\mathbf{x})=\mathbf{P}(0)\,. \tag{37}$$

If $\mathbf{P}(\mathbf{x})$ is a polynomial basis, the consistency condition is equivalent to reproducing condition,

$$\sum_I \mathbf{P}(\mathbf{x}_I)K_I^\rho(\mathbf{x})=\mathbf{P}(\mathbf{x}). \tag{38}$$

For instance,

$$\sum_I \mathbf{x}_I^m K_I^\rho(\mathbf{x})=\mathbf{x}^m,\quad m=0,1,2,\cdots. \tag{39}$$

Moreover, it has been showed in [70,204] that there is a m-th order consistency condition for the derivatives of meshfree interplant,

$$\sum_I (\mathbf{x}_I-\mathbf{x})^\alpha D_{\mathbf{x}}^\beta K_I^\rho(\mathbf{x})=\alpha!\,\delta_{\alpha\beta} \tag{40}$$

which is equivalent to

$$\sum_I \mathbf{x}_I^\alpha D_{\mathbf{x}}^\beta K_I^\rho(\mathbf{x})=\frac{\alpha!}{(\alpha-\beta)!}\mathbf{x}^{\alpha-\beta}. \tag{41}$$

These consistency conditions firmly establish the basis for the convergence of mesh-free Galerkin methods [70,73,74,204], which is far more systematic than the early convergence study done by Farwig [205,206] for MLS interpolant.

The m-th order consistency for the derivatives of RKPM interpolant has a profound consequence. Based on this condition, one can construct a multiple scale meshfree interpolant on a set of scattered data [207] by enforcing different vanishing moment conditions,

$$\mathbf{M}(\mathbf{x})\mathbf{b}^{(\alpha)}(\mathbf{x})=\{\mathbf{P}^{(\alpha)}(0)\}^t \tag{42}$$

where $\mathbf{P}^{(\alpha)}(0)=(0,\cdots,0,\underbrace{1}_{\ell-\alpha},0,\cdots,0)$.

The procedure resembles the construction of wavelet basis on the regular grid, *eg*, [208,209]. Indeed, Li *et al* [204,207,210] showed that the higher order RKPM interpolants indeed satisfy the primitive definition of wavelet transformation/function. Figure 4 illustrates the build-up of meshfree wavelet function on a set of randomly distributed points. These wavelets functions have been used by Li and Liu [210,211] to calculate reduced wave equation—Helmholtz equation, advection diffusion problem and Stokes flow problems, and used by Günther *et al* [212] to compute compressible flow problems as a stabilization agent. Chen *et al* [213] utilized the meshfree wavelet basis as a numerical regularization agent to introduce an intrinsic length, and consequently stabilize the numerical simulation of strain localization problem.

The m-th consistency condition (37) is further generalized by Wagner and Liu [214], Huerta and Fernández-Méndez [215,216], and Han *et al* [217] for the hybrid finite-element–meshfree refinement, which has been used in either meshfree h-adaptivity [218,215], or to enforce the essential boundary conditions [217]. Denoting finite element basis as $\{N_I^h(\mathbf{x})\}$ and meshfree basis as $\{K_I^\rho(\mathbf{x})\}$, the hybrid interpolation has the following m-order consistency condition

$$\sum_I \mathbf{P}\left(\frac{\mathbf{x}-\mathbf{x}_I}{\rho}\right)K_I^\rho(\mathbf{x})+\sum_I \mathbf{P}\left(\frac{\mathbf{x}-\mathbf{x}_I}{h}\right)N_I^h(\mathbf{x})=\mathbf{P}(0) \tag{43}$$

and the corresponding reproducing property,

$$\sum_I \mathbf{P}(\mathbf{x}_I)K_I^\rho(\mathbf{x})+\sum_I \mathbf{P}(\mathbf{x}_I)N_I^h(\mathbf{x})=\mathbf{P}(\mathbf{x}). \tag{44}$$

This generalized consistency condition is instrumental in the convergence study of mixed hierarchical finite-element/meshfree approximation. In fact, the mixed finite-meshfree enrichment procedure has been a success, which is much easier to implement than the conventional finite element h-type refinement, which may require structured mesh. In practice, one can simply sprinkle particles onto a finite element mesh expecting much improvement in numerical solutions [215].

Another important enrichment is the so-called p-type enrichment. Since moving least square interpolant is a partition of unity, Duarte and Oden [73,74] used Legendre polynomial to construct a first p-version meshfree interpolant, which they named as h-p Clouds. In one dimensional case, it takes the form of

$$u^h(x)=\sum_{I\in\Lambda}\phi_I^{n+1}(x)\left(u_I L_0+\sum_{i=1}^{\ell} b_{iI}L_i(x)\right) \tag{45}$$

where $\phi_I^{n+1}(x)$ is the $n+1$ order moving least square interpolant. In general, $L_i(x)$ may be regarded as the Taylor expansion of $u(x)$ at point x_I. The reason using Legendre polynomial as p-enrichment is its better conditioning; a similar procedure is well established in p-version finite element [219]. An early paper by Liu *et al* [188] proposed an interpolation formula that is aslo similar to Eq. (45); it is called the *multiple-scale spectral finite element method*. The Legendre polynomial enrichment basis is called by Belytschko *et al* [60] as extrinsic basis, and it is attached to the intrinsic basis, $\phi_I^{n+1}(x)$ to form a p-cloud. There is a seldom mentioned belief among the advocates of h-p clouds. That is one can build h-p clouds on the simplest meshless partition of unity—the Shepard interpolant, *ie*, one can pile up higher order polynomial to Shepard interpolant. By so doing, one does not need the matrix inversion when constructing higher order meshfree shape function; one may still be able to enjoy good interpolation convergence.

This line of thinking leads to a more general formulation, for instance, the so-called *partition of unity method* set forth by Babuška and Melenk [77,79]. The essence of the partition of unity method is: take a partition of unity and multiply it with any independent basis to form a new and better basis. This flexibility provides leverage in computation practice. Sometimes the choices of the independent basis can be based on users' prior knowledge and experience about the problem that they are solving. For instance, Babuška and Melenk [79] used the following basis,

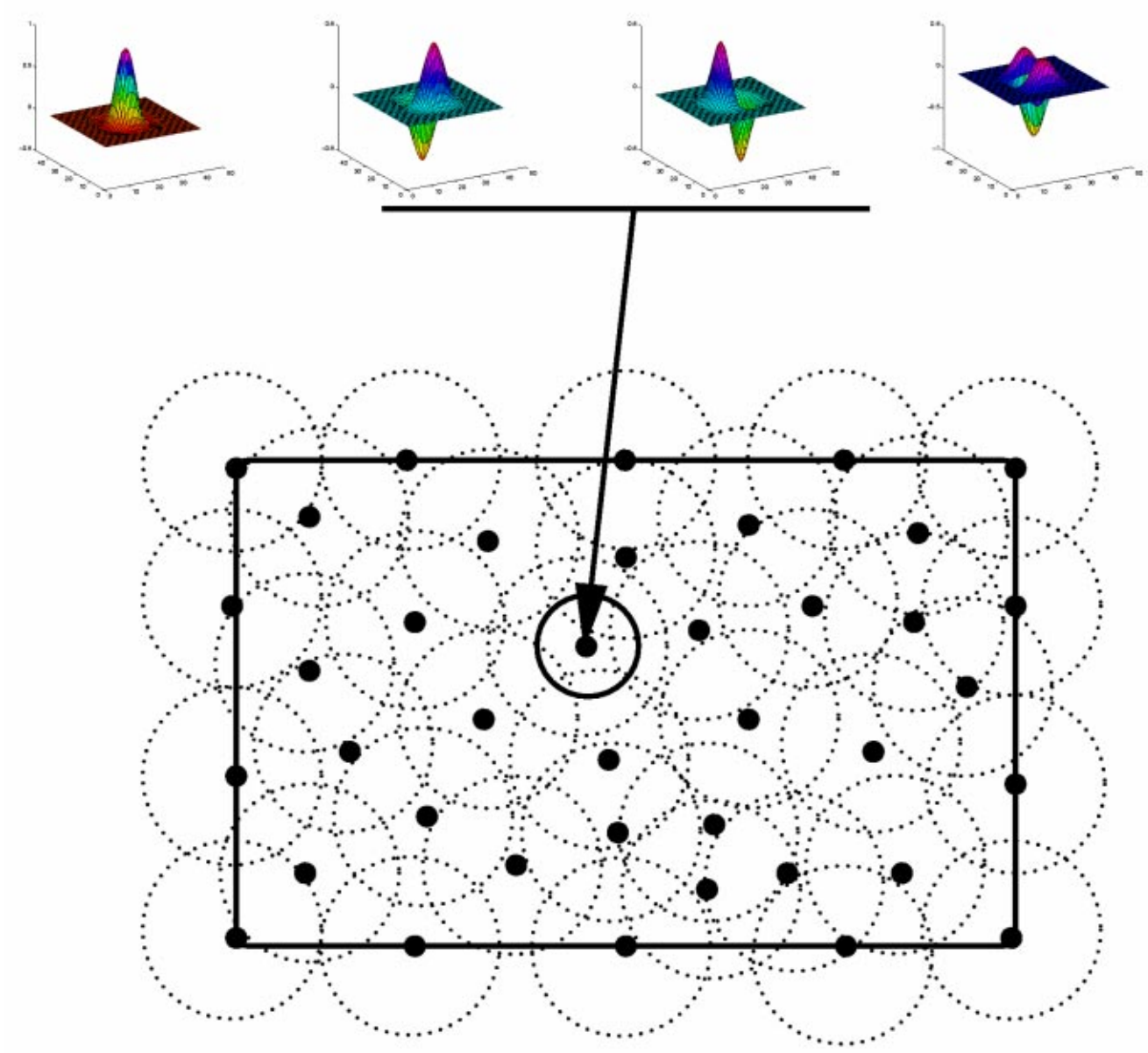

Fig. 4 An illustration of 2D hierarchical partition of unity

$$u^h(x)=\sum_{I\in\Lambda}\phi_I(a_{0I}+a_{1I}x+a_{2I}y+b_{1I}\sin(nx)+b_{2I}\cos(nx)) \quad (46)$$

to solve Helmholtz equation. Dolbow *et al* [220] used the following interpolant to simulate strong discontinuity, *ie* the crack surfaces,

$$\mathbf{u}^h(\mathbf{x})=\sum_I \phi_I\left[\mathbf{u}_I+H(\mathbf{x})\mathbf{b}_I+\sum_J \mathbf{c}_{IJL}F_L(\mathbf{x})\right] \quad (47)$$

where $H(\mathbf{x})$ is the Heaviside function and $F_L(\mathbf{x})$ are asymptotic fields in front of crack tip. If $\phi_I(x)$ is a meshfree interpolant, then the method is a meshfree method; if $\phi_I(x)$ is a finite elment interpolant, the method is called PUFEM, an acronym of *partition of unity finite element method*. Recently, Wagner *et al* [221] used a discontinuous version of PUFEM to simulate rigid particle movement in a Stokes flow. By embedding a discontinuous function to a partition of unity, the interpolant can accurately represent the shape of a finite size particle, and the particle surface need not to conform to the finite element boundary. By doing so, the problem of moving particles in a flow can be simulated without remeshing. A so-called X-FEM technique, a variant of PUFEM, is used by Daux *et al* [222] to model cracks, especially cracks with arbitrary branches, or intersecting cracks.

A slight modification of the X-FEM technique was used by Wagner [223] to simulate concentrated particulate suspensions on a fixed mesh. In this work, the velocity and pressure function spaces are enriched with the lubrication theory solution for flow between two particles in close proximity. This allows particles to approach each other at distances much smaller than the element size, avoiding the need to refine or adapt the mesh to capture these small-scale flow details. Wagner took advantage of the fact that the lubrication solution is determined completely in terms of the particle motions and pressure gradient across the gap to reduce the number of degrees of freedom by tying the values of the nodes in the lubrication region together; the standard X-FEM approach allows the variation of these nodes for maximum freedom in the solution. Tying the nodes together as done by Wagner allows the entire velocity and pressure solution between two particles to be determined in terms of just eight degrees of freedom for the 2D case. This is a good example of multiple scale analysis. Contrary to PUFEM and XFEM, the fine scale lubrication solution is embedded into the standard PUFEM and X-FEM with only two unknown coefficients of flow rate and pressure, and the remaining six unknown degrees of freedom are the two particles velocities and rotations.

3.3 Enforcement of essential boundary conditions

One of the key techniques of meshfree-Galerkin methods is how to enforce an essential boundary condition because most meshfree interpolants do not possess Kronecker delta property. This means that in general, the coefficients of the interpolant are not the same as the nodal values, that is for $u^h(x)=\Sigma_I N_I(x)d_I$,

$$u^h(x_I)\neq d_I. \quad (48)$$

However, there are exceptions. For instance, if the boundary is piece-wise linear, and the particle distribution can be arranged such that they are evenly distributed along the boundary, one may obtain Kronecker delta property along the boundary. This is because the correction function not only can enforce consistency conditions, but also can correct abnormality due to the finite domain. This is a hardly known fact, which was discussed in a paper by Gosz and Liu [224]. This procedure, nevertheless, is only feasible for certain simple geometries. In general, a systematic treatment is still needed.

3.3.1 Lagrangian multiplier method

In the first EFG paper [59], Belytschko *et al* enforced the essential boundary via Lagrangian multiplier method. Lu *et al* [63] slightly modified the formulation. Consider an elastostatics problem

$$\nabla\cdot\boldsymbol{\sigma}+\mathbf{b}=\mathbf{0},\quad \mathbf{x}\in\Omega \quad (49)$$

with the boundary conditions

$$\boldsymbol{\sigma}\cdot\mathbf{n}=\overline{\mathbf{T}},\ \ \forall\mathbf{x}\in\Gamma_t \quad (50)$$

$$\mathbf{u}=\overline{\mathbf{u}},\ \ \forall\mathbf{x}\in\Gamma_u. \quad (51)$$

To accommodate the non-interpolating shape function, we introduce the reaction force, $\mathbf{R}$, on Γ_u as another unkown variable, which is complementary to the primary unknown, $\mathbf{u}$, the displacement. A weak form of the original problem can be written as,

$$\int_\Omega[(\nabla_s\mathbf{v}^T):\boldsymbol{\sigma}-\mathbf{v}^T:\mathbf{b}]d\Omega-\int_{\Gamma_t}\mathbf{v}^T\cdot\overline{\mathbf{T}}dS-\int_{\Gamma_u}\boldsymbol{\lambda}^T\cdot(\mathbf{u}-\overline{\mathbf{u}})dS-\int_{\Gamma_u}\mathbf{v}^T\cdot\mathbf{R}d\Omega=0,\quad \forall\mathbf{v}\in H^1(\Omega),\ \ \boldsymbol{\lambda}\in H^0(\Omega) \quad (52)$$

where $\mathbf{v}$ and $\boldsymbol{\lambda}$ are identified as $\delta\mathbf{u}$ and $\delta\mathbf{R}$, respectively.

Let

$$\mathbf{u}^h(\mathbf{x})=\sum_{I\in\Lambda}N_I(\mathbf{x})\mathbf{u}_I,\quad \mathbf{v}^h(\mathbf{x})=\sum_{I\in\Lambda}N_I(\mathbf{x})\mathbf{v}_I \quad (53)$$

where $\Lambda=1,2,\cdots,NP$. Define a sub index set Λ_b, $\Lambda_b=\{I|I\in\Lambda,N_I(\mathbf{x})\neq 0,\mathbf{x}\in\Gamma_u\}$. And let

$$\mathbf{R}(\mathbf{x})=\sum_{I\in\Lambda_b}\tilde{N}_I(\mathbf{x})\mathbf{R}_I,\boldsymbol{\lambda}(\mathbf{x})=\sum_{I\in\Lambda_b}\tilde{N}_I(\mathbf{x})\boldsymbol{\lambda}_I,\mathbf{x}\in\Gamma_u \quad (54)$$

where $\tilde{N}_I(\mathbf{x})$ may be different from $N_I(\mathbf{x})$ in order to satisfy the LBB condition. The following algebraic equations may then be derived,

$$\begin{pmatrix}\mathbf{K} & \mathbf{G}\\ \mathbf{G}^T & \mathbf{0}\end{pmatrix}\begin{pmatrix}\mathbf{u}\\ \mathbf{R}\end{pmatrix}=\begin{pmatrix}\mathbf{f}\\ \mathbf{q}\end{pmatrix}. \quad (55)$$

And

$$\mathbf{K}_{IJ}=\int_\Omega\mathbf{B}_I^T\mathbf{D}\mathbf{B}_J d\Omega \quad (56)$$

$$\mathbf{G}_{IK}=-\int_{\Gamma_u}N_I\tilde{\mathbf{N}}_K d\Gamma, \quad (57)$$

$$\mathbf{f}_I=\int_{\Gamma_t}N_I\overline{\mathbf{t}}d\Gamma+\int_\Omega N_I\mathbf{b}d\Omega \quad (58)$$

$$\mathbf{q}_K = -\int_{\Gamma_u} \tilde{N}_K \overline{\mathbf{u}} d\Gamma \tag{59}$$

where $\mathbf{D}$ is elasticity matrix, and

$$\mathbf{B}_I = \begin{bmatrix} N_{I,x}, & 0 \\ 0, & N_{I,y} \\ N_{I,y}, & N_{I,x} \end{bmatrix} \tag{60}$$

$$\tilde{\mathbf{N}}_I = \begin{bmatrix} \tilde{N}_{I,x}, & 0 \\ 0, & \tilde{N}_{I,y} \end{bmatrix}. \tag{61}$$

The Lagrangian multiplier method may run into a stability problem, if one chooses shape functions without discretion.

3.3.2 Penalty method

The penalty method is another alternative to impose essential boundary conditions, which was first proposed by Belytschko *et al* [190]. A detailed illustration is given by Zhu *et al* [225] for the case of 2D linear elastostatics. Consider the same problem Eqs. (49)–(51). One has the Lagrangian,

$$\Pi_\varrho = \frac{1}{2}\int_\Omega (\boldsymbol{\epsilon}^h)^T \cdot \mathbf{D} \cdot \boldsymbol{\epsilon}^h d\Omega - \int_\Omega (\mathbf{u}^h)^T \cdot \mathbf{b} d\Omega - \int_{\Gamma_t} (\mathbf{u}^h)^T \cdot \overline{\mathbf{T}} dS + \frac{\alpha}{2}\int_{\Gamma_u} (\mathbf{u}^h - \overline{\mathbf{u}})^T \cdot (\mathbf{u}^h - \overline{\mathbf{u}}) dS. \tag{62}$$

Taking $\delta\Pi_h = 0$, we have the following algebraic equations,

$$(\mathbf{K} + \alpha \mathbf{K}^u)\mathbf{U} = \mathbf{f} + \alpha \mathbf{f}^u. \tag{63}$$

The additional terms due to essential boundary conditions are

$$\mathbf{K}^u_{IJ} = \int_{\Gamma_u} N_I \mathbf{S} N_J dS \tag{64}$$

$$\mathbf{f}^u_I = \int_{\Gamma_u} N_I \mathbf{S} \overline{\mathbf{u}} dS \tag{65}$$

where

$$\mathbf{S} = \begin{bmatrix} S_x, & 0 \\ 0, & S_y \end{bmatrix},$$

$$S_i = \begin{cases} 1 & \text{if } u_i \text{ is prescribed on } \Gamma_u, \\ 0 & \text{if } u_i \text{ is not prescribed on } \Gamma_u, \end{cases} \quad i = 1,2. \tag{66}$$

In computations, the penalty parameter is taken in the range $\alpha = 10^3 \sim 10^7$.

3.3.3 Transformation method

The most efficient method to impose essential boundary conditions for meshfree methods is the transformation method. It was first proposed by Chen *et al* [71], and it has been reiterated by many authors [226–228]. There are two versions of it: full transformation method (see: [71]) and boundary transform method [226,227]. An efficient boundary transformation algorithm is proposed by Günther *et al* [229] based on the intuitive argument of d'Alembert principle. The version of transformation method described here has been used by the Northwestern Group since 1994. All the particles are separated into into two sets: boundary set marked with superscript b and interior set marked with nb (non-boundary particle). We distribute N_b number of particles on the boundary Γ^u, and the number of interior particles are: $N_{nb} := NP - N_b$. The essential boundary condition provides N_b constraints,

$$u_i^h(\mathbf{x}_I, t) = u_i^0(\mathbf{x}_I, t) =: g_i(\mathbf{x}_I, t), \quad I = 1, \cdots, N_b \tag{67}$$

denote $g_{iI}(t) := g_i(\mathbf{x}_I, t)$, $I = 1, \cdots\cdots, N_b$.

$$\begin{aligned} u_i^h(\mathbf{x}, t) &= \sum_{I=1}^{NP} N_I(\mathbf{x}) d_{iI}(t) \\ &= \sum_{I=1}^{N_b} N_I^b(\mathbf{x}) d_{iI}^b(t) + \sum_{I=1}^{N_{nb}} N_I^{nb}(\mathbf{x}) d_{iI}^{nb}(t) \\ &= \mathbf{N}^b(\mathbf{x})\mathbf{d}_i^b(t) + \mathbf{N}^{nb}(\mathbf{x})\mathbf{d}_i^{nb}(t). \end{aligned} \tag{68}$$

Let $\mathbf{D}^b := \{N_I^b(\mathbf{x}_J)\}^{N_b \times N_b}$, and $\mathbf{D}^{nb} := \{N_I^{nb}(\mathbf{x}_J)\}^{N_b \times N_{nb}}$. Thus the enforced discrete essential conditions, (67), become

$$\mathbf{D}^b \mathbf{d}_i^b(t) = \mathbf{g}_i(t) - \mathbf{D}^{nb}\mathbf{d}_i^{nb}(t) \tag{69}$$

after inversion $\mathbf{d}_i^b(t) = (\mathbf{D}^b)^{-1}\mathbf{g}_i(t) - (\mathbf{D}^b)^{-1}\mathbf{D}^{nb}\mathbf{d}_i^{nb}(t)$, a transformed interpolation is obtained,

$$u_i^h(\mathbf{x}, t) = \mathbf{N}^b(\mathbf{x})(\mathbf{D}^b)^{-1}\mathbf{g}_i(t) + (\mathbf{N}^{nb}(\mathbf{x}) - \mathbf{N}^b(\mathbf{x}) \times (\mathbf{D}^b)^{-1}\mathbf{D}^{nb})\mathbf{d}_i^{nb}(t). \tag{70}$$

Obviously, for $\mathbf{x}_I \in \Gamma^u$, $I = 1, \cdots, N_b$,

$$u_i^h(\mathbf{x}_I, t) = g_{iI}(t); \quad \delta u_i^h(\mathbf{x}_I, t) = 0, \quad I = 1,2, \cdots, N_b. \tag{71}$$

This result can also be interpreted as a new interpolant, *ie*

$$\begin{aligned} u_i^h(\mathbf{x}, t) &= \sum_{I=1}^{N_b} W_I^b(\mathbf{x}) u_{iI}(t) + \sum_{I=1}^{N_{nb}} W_I^{nb} d_{iI}(t) \\ &= \mathbf{W}^b(\mathbf{x})\mathbf{u}_i + \mathbf{W}^{nb}(\mathbf{x})\mathbf{d}_i^{nb} \end{aligned} \tag{72}$$

where $\mathbf{W}^b(\mathbf{x}) := \mathbf{N}^b(\mathbf{x})(\mathbf{D}^b)^{-1}$, and $\mathbf{W}^{nb}(\mathbf{x}) := [\mathbf{N}^{nb}(\mathbf{x}) - \mathbf{N}^b(\mathbf{x})(\mathbf{D}^b)^{-1}\mathbf{D}^{nb}]$. One may notice that the new shape functions in (72) possess the Kronecker-delta, or interpolation property at the boundary.

3.3.4 Boundary singular kernel method

The idea of using singular kernel function to enforce the Kronecker delta property should be credited to Lancaster and Salkauskas [189], which they called the *interpolating moving least square interoplant*. Some authors later used it in computations, *eg* Kaljevic and Saigal [230] and Chen and Wang [227]. The idea is quite simple. Take a set of positive shape function $\{\Phi_h(\mathbf{x} - \mathbf{x}_I)\}_{I=1}^N$. Suppose $\mathbf{x}_J$ is on the boundary Γ_u; we modify the shape function basis as,

$$\tilde{\Phi}_h(\mathbf{x} - \mathbf{x}_I) = \begin{cases} \dfrac{\Phi_h(\mathbf{x} - \mathbf{x}_I)}{|\mathbf{x} - \mathbf{x}_I|^p}, & \forall I \in \Gamma_u, \quad p > 0 \\ \Phi_h(\mathbf{x} - \mathbf{x}_I), & \forall I \notin \Gamma_u \end{cases} \tag{73}$$

and then build a new shepard basis on $\{\Phi_h(\mathbf{x} - \mathbf{x}_I)\}$ as

$$\Psi_h(\mathbf{x}-\mathbf{x}_I)=\frac{\tilde{\Phi}_h(\mathbf{x}-\mathbf{x}_I)}{\sum_I \tilde{\Phi}_h(\mathbf{x}-\mathbf{x}_I)} \tag{74}$$

one may verify that for the boundary nodes $\mathbf{x}_J$, $\Psi_h(\mathbf{x}_I-\mathbf{x}_J)=\delta_{IJ}$. In real computations, the procedure works in certain range of dilation parameter, h, but when h is too large, the convergence of interpolation deteriorates rapidly [227].

3.3.5 Coupled finite element and particle approach

Another approach is to couple finite element with particles close to the boundary and necklace the particle domain with a FEM boundary layer and apply essential boundary conditions to the finite element nodes (see Krongauz and Belytschko [231] and Liu *et al* [218]). In this approach, all the boundary and its neighborhood are meshed with finite element nodal points, and there is a buffer zone between the finite element zone and the particle zone, which is connected with the so-called ramp functions. Denote the finite element basis as $\{N_i(x)\}$, particle basis as $\{\Phi_i(x)\}$, and ramp function as $R(x)$. The interpolation function in the buffer zone is the combination of FEM and particle interpolant

$$\tilde{\Phi}_i(x)=\begin{cases}(1-R(x))\Phi_i(x)+R(x)N_i(x) & x\in\Omega_{fem}\\ \Phi(x) & x\in\Omega_p\end{cases} \tag{75}$$

where the ramp function is chosen as $R(x)=\Sigma_i N_i(x)$, $x_i\in\partial\Omega_{fem}$. Recently, this approach was used again by Liu and Gu in a meshfree local Petrov-Galerkin (MLPG) implementation [232].

Although the method works well, it compromises the intrinsic nature of being meshfree, and subsequently loses the advantages of particle methods. For example, in shear band simulations, the mesh alignment sensitivity due to the finite element mesh around a boundary could pollute the entire numerical simulation. To enforce the Dirichlet boundary condition while still retaining the advantage of a particle method, a so-called hierarchical enrichment technique is developed to enforce the essential boundary condition [214,217], which is a further development of the work [218]. The idea is as follows. Around the boundary, one first deploy a layer of finite element nodes, and all the nodes on the boundary are finite element nodes. Right within the boundary the meshfree particles are blended with the finite element nodes, and there is no buffer zone. Denote the finite element shape function as $N_I(\mathbf{x})$ $I\in B$; and denote meshfree shape function as $\Phi_I(\mathbf{x}), I\in A$. One can view that particle discretization as enrichment of finite element discretization at the boundary.

$$u^h(\mathbf{x})=\sum_{I\in B} N_I(\mathbf{x})a_I+\sum_{I\in A}\tilde{\Phi}_I(\mathbf{x})d_I \tag{76}$$

where $\tilde{\Phi}_I(\mathbf{x})$ is complementary to the finite element basis, *ie*

$$\tilde{\Phi}_I(\mathbf{x})=\Phi_I(\mathbf{x})-\sum_{J\in B} N_J(\mathbf{x})\Phi_I(\mathbf{x}_J) \tag{77}$$

It is easy to verify that for a boundary particle $\mathbf{x}_I$, $I\in B$, $u^h(\mathbf{x}_I)=a_I$. Thus Dirichlet boundary condition can be specified directly. In [217], Han *et al* elegantly proved the convergence of the method.

In fact, one can also utilize the idea of partition of unity finite element (PUFEM) to enforce essential boundary condition. The procedure is as follows. Deploying a few–layer finite element mesh around desired boundary and choosing Lagrange finite element interpolant as extrinsic basis, $L_{JI}(x)$, such that $L_{JI}(x_K)=\delta_{JK}$. A PUFEM shape function is constructed as follows

$$\Phi_I(x)=\sum_{J:\{x_I\in\Omega_J\}} K_J(x)L_{JI}(x) \tag{78}$$

where $K_J(x)$ is a meshfree interpolant. One can show that $\Phi_I(x_J)=\delta_{IJ}$.

It is worth mentioning that even though meshfree interpolants have no difficulties in enforcing natural boundary conditions, the implementation of enforcing natural boundary conditions in meshfree setting is different from those in FEM setting. In finite element procedure, one need only calculate a surface or curved line integral in evaluating traction boundary conditions; whereas in meshfree setting, one has to take into account the influences from the interior particles as well, though this is seldom mentioned in the literature. Pang [233] documented a procedure to treat point loads in an EFG implementation.

3.3.6 Quadrature integration and nodal integration

Most mesh-free Galerkin methods (Fig. 5) used background cell, or background grid to locate the quadrature points to integrate the weak form. Although the background cell need not be structured, and can be easily refined (*eg* the work by Klass and Shepard [234]), there is, nontheless, still a ghost *mesh* present. Moreover, how to place such background cell, or how to place Gauss quadrature points will directly influence the accuracy as well as the invertibility of the stiffness matrix. Early on there were a lot of discussions on patch-test of meshfree Galerkin methods [59,190–192,195,235,236]. The real concern is the stability of quadrature integration. Most meshfree interpolants, for instance MLS interpolant,

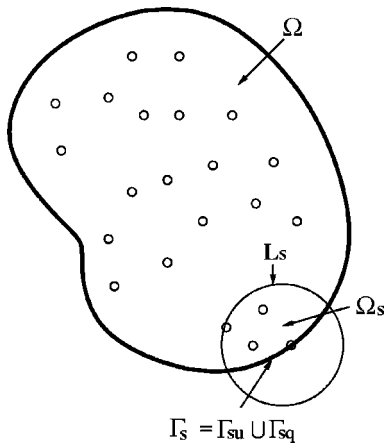

Fig. 5 Local meshfree-Galerkin illustration ($\partial\Omega_s=L_s\cup\Gamma_s$)

are partitions of unity, and in most cases the linear completeness, or consistency is also enforced *a priori*; there is no compatibility issue left to be tested, unlike the incompatible finite element shape function. However, if there are not enough quadrature points in a compact support, or quadrature points are not evenly distributed, spurious modes may occur.

Today, quadrature integration is one of the two major shortcomings (the cost of meshfree methods is the other) left when meshfree methods compared with finite element methods. Beissel and Belytschko [237] proposed a stabilized nodal integration procedure by adding a residual of the equilibrium equation to the potential energy functional to avoid use of quadrature integration. However, adding the additional term in potential energy means sacrificing variational consistency, hence accuracy of the formulation. Gauss quadrature integration error via different set-up of background cells as well as quadrature point distribution is studied in [238]. It is found that if the background cell does not match with the compact support of the meshfree interpolant, considerable integration error may rise.

The simplest remedy is the local, self-similar support integration. Assume the meshfree shape function is compactly supported, and the support for each and every particle is similar in shape, *eg* a circular region in 2D, a sphere in 3D. Take the Element Free Galerkin (EFG) method for example (Belytschko *et al* [59,63]). For linear elastostatics, the stiffness matrix is

$$\mathbf{K}_{IJ}=\int_{\Omega}\mathbf{B}_I^t\mathbf{D}\mathbf{B}_J d\Omega \tag{79}$$

where Ω is the problem domain. If all the shape functions have the same shape of compact support (a 3D sphere in this case), the above integration can be rewritten as

$$K_{IJ}=\int_{\Omega\cap\Omega_I}\mathbf{B}_I^t\mathbf{D}\mathbf{B}_J d\Omega \tag{80}$$

where Ω_I is the support of particle I.

Because all shape functions are compactly supported, the integrals in the rest of domain, *ie* Ω/Ω_I, vanish. And we only need to evaluate K_{IJ} within $\Omega_I\cap\Omega$ and Γ_{Iu}. Since every $\Omega_I,(I=1,\ldots,n)$ has the same shape, once a quadrature rule is fixed for one compact support, it will be the same for the rest of compact supports as well. We can then integrate the weak form locally from one compact support to another compact support. Therefore, it is free of the background cell or any implicit mesh. Note that this is different from the global domain quadrature integration, since in our case compact supports are overlapped with each other.

This local quadrature idea is extended by Atluri and his colleagues to form new meshfree formulations [80–83,239–242]. The first formulation proposed by Atluri *et al* is called the local boundary integral equation (LBIE).

Consider a boundary value problem of Poisson's equation [239]. One can form a boundary integral equation for a chosen subdomain Ω_s (note that Ω_s has nothing to do with a particle's compact support),

$$\alpha u(\mathbf{y})=-\int_{\Omega_s}u(\mathbf{x})\frac{\partial\tilde{u}^*}{\partial n}(\mathbf{x},\mathbf{y})d\Gamma+\int_{\Gamma_s}\frac{\partial u}{\partial n}(\mathbf{x})\tilde{u}^*(\mathbf{x},\mathbf{y})d\Gamma-\int_{\partial\Gamma_s}\tilde{u}^*(\mathbf{x},\mathbf{y})p(\mathbf{x})d\Omega \tag{81}$$

where $\tilde{u}^*$ is the Green's function

$$\tilde{u}^*(\mathbf{x},\mathbf{y})=\frac{1}{2\pi}\ln\frac{r_0}{r}. \tag{82}$$

For each particle in the domain Ω, one can form a local boundary integral equation (81). Letting $u^h(\mathbf{x})=\Sigma_i\phi_i(\mathbf{x})d_i$, one may obtain the following algebraic equations

$$\alpha_i u_i=\sum_{j=1}^{N}K_{ij}^*d_j+f_i^* \tag{83}$$

where

$$K_{ij}^*=\int_{\Gamma_{su}}\tilde{u}^*(\mathbf{x},\mathbf{y}_i)\frac{\partial\phi_j}{\partial n}d\Gamma-\int_{\Gamma_{sq}}\phi_j\frac{\partial\tilde{u}^*}{\partial n}(\mathbf{x}.\mathbf{y}_i)d\Gamma-\int_{L_s}\phi_j\frac{\partial\tilde{u}^*}{\partial n}(\mathbf{x},\mathbf{y}_i)d\Gamma \tag{84}$$

$$f_i^*=\int_{\Gamma_{sq}}\tilde{u}^*(\mathbf{x},\mathbf{y}_i)\overline{q}d\Gamma-\int_{\Gamma_{su}}\overline{u}\frac{\partial\tilde{u}^*}{\partial n}(\mathbf{x},\mathbf{y})d\Gamma-\int_{\Omega_s}\tilde{u}^*(\mathbf{x},\mathbf{y}_i)p(\mathbf{x})d\Omega. \tag{85}$$

Those local boundary integrals and local domain integrals can be integrated by fixed quadrature rules. Sladek *et al* presented a detailed account on how to deal with singularity in numerical integrations [243]. The obvious advantage of this formulation is that it does not need to enforce the essential boundary condition. Nevertheless, this formulation relies on a Green's function, and it is limited to a handful of linear problems.

Subsequently, Atluri *et al* [80,81] formed a local Petrov-Galerkin formulation (MLPG) with meshfree interpolant in the same local region Ω_s For linear elastostatics problem (49), they form N local petrov-Galerkin weak forms. Each of them around a distinct particle I is,

$$\sum_{j=1}^{N}K_{ij}d_j=f_j \tag{86}$$

where

$$\mathbf{K}_{IJ}=\int_{\Omega_s}(\mathbf{B}_v^I)^T\mathbf{D}\mathbf{B}_J d\Omega+\alpha\int_{\Gamma_{su}}\mathbf{v}^I\phi_J d\Gamma-\int_{\Gamma_{su}}\mathbf{v}^I\mathbf{N}\mathbf{D}\mathbf{B}_J d\Gamma \tag{87}$$

$$f_I=\int_{\Gamma_{st}}\mathbf{v}^I\overline{\mathbf{t}}d\Gamma+\alpha\int_{\Gamma_{su}}\mathbf{v}^I\overline{\mathbf{u}}d\Gamma+\int_{\Omega_s}\mathbf{v}^I\mathbf{b}d\Omega \tag{88}$$

Again, Ω_s is not the compact support Ω_I, however, certain conditions must be imposed to Ω_s, such that $K_{ij}\neq 0$ at least for some $j\neq i$. In practical implementation, the trial func-

tion's support is Ω_I whereas the I-th weighting function's support is denoted as Ω_{Is}. Note that all the integrations here are local; no background cell is needed. The term Petrov here indicates that one uses different trial and test (weighting) function (even though they may be the same function but they have different support size, $\Omega_I \neq \Omega_{Is}$). This will result an unsymmetric stiffness matrix in general. Let $\Omega_{Is}=\Omega_I$, and the trial function be the same as the weighting function. Then the above Petrov-Galerkin formulation becomes the conventional Bubnov-Galerkin formulation. In that case, it returns the local quadrature integration scheme we presented at the beginning. It is worth mentioning that if one choose Ω_{Is} as an n-dimensional sphere, the numerical integration may be carried out by Cubature, which is recently documented in details by De and Bathe [244].

In order to completely eliminate quadrature points, Chen *et al* [72] proposed a so-called stabilized conforming nodal integration for meshfree Galerkin method. They first identify that for linear exactness in the Galerkin approximation, the shape functions have to be linearly consistent, and the domain integration has to be able to integrate the derivatives of shape functions to nullity for interior nodes and to meet traction equilibrium. The argument made by Chen *et al* is that for meshfree solution of a nodally integrated weak form to be stable and convergent, two conditions need to be satisfied: 1) derivatives of meshfree shape functions evaluated at the nodal point must be avoided and 2) nodal integration must satisfy integration constraints. It is shown in their study [245] that a direct integration introduces numerical instability due to rank deficiency in the stiffness matrix. To stabilize the nodal integration, they proposed a so-called smoothing stabilization technique. The basic idea is that one first integrates strain in a chosen neighborhood of the particle I, say Ω_I, to replace the strain at point I with the average strain, as illustrated in Fig. 6, provided the general triangulation is possible. Note that here Ω_I is not the compact support of the particle I ($\text{supp}(\Psi_I)$), it is the Voronoi cell that contains the particle I. Then divergence theorem is used to replace the area, or volume integration around particle I by a contour integration of the Voronoi cell boundary. The contour integration is carried out by sampling the values at the vertices of the Voronoi cell. In the implementation [72,245],

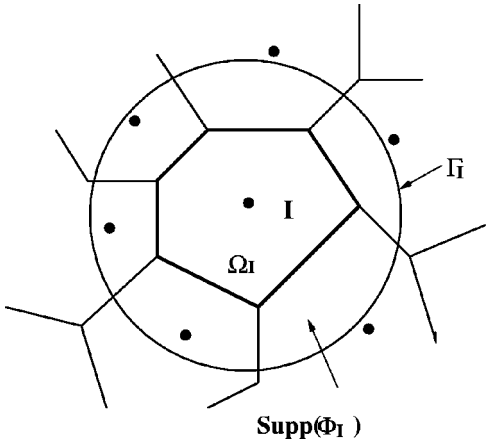

Fig. 6 Geometry definition of a representative nodal domain

$$\bar{\epsilon}_{ij}^h(\mathbf{x}_I)=\int_\Omega \epsilon_{ij}^h(\mathbf{x})\Phi(\mathbf{x}-\mathbf{x}_I)d\Omega \tag{89}$$

where $\Phi(\mathbf{x})$ is the characteristic function of small area Ω_I

$$\Phi(\mathbf{x}-\mathbf{x}_I)=\begin{cases}\dfrac{1}{A_I} & \mathbf{x}\in\Omega_I\\ 0 & \mathbf{x}\notin\Omega_I\end{cases} \tag{90}$$

where $A_I=\text{meas}(\Omega_I)$. Therefore,

$$\bar{\epsilon}_{ij}^h(\mathbf{x}_I)=\frac{1}{2A_I}\int_{\Omega_I}\left(\frac{\partial u_i^h}{\partial x_j}+\frac{\partial u_j^h}{\partial x_i}\right)d\Omega$$

$$=\frac{1}{2A_I}\int_{\Gamma_I}(u_i^h n_j+u_j^h n_i)dS. \tag{91}$$

Finally, employing an assumed strain method and integrating the weak form by a nodal integration, the meshfree discrete equation is obtained. It is shown that if linear basis functions are used in the construction of shape function, the strain smoothing of Eq. (91) in conjunction with the nodal integration of weak form will result the linear exactness in the Galerkin approximation. The main virtue of this approach is that it completely eliminates Gauss quadrature points, which is especially attractive in inelastic large deformation calculation with a Lagrangian formulation.

3.4 Applications

One of the early incentives to develop meshfree Galerkin methods was its ability to simulate crack growth—a critical issue in computational fracture mechanics. Belytschko and his co-workers have systematically applied the EFG method to simulate crack growth/propagation problems [60,63,190–192,235,246,247]. Special techniques, such as the visibility criterion, are developed in modeling a discontinuous field [60,246]. Subsequently, a partition of unity method is also exploited in crack growth simulation [220]. It is fair to say that at least in 2D crack growth simulation meshfree Galerkin procedure offers considerable advantages over the traditional finite element methods, because remeshing is avoided. Meshfree simulation has been conducted by Li *et al* to simulate failure mode transition [248,249]. The simulation has successfully replicated failure mode transition observed in Zhou-Rosakis-Ravichadran experiment [250], which is related to the early Kalthoff problem [251,252]. Figure 7 shows a crack growth from a shear band.

Another area where meshfree Galerkin methods have clear edge over finite element computations is its ability to handle large deformation problems. (See Fig. 8.) Chen and his co-workers proposed a concept of Lagrangian kernel and have been using RKPM to simulate several large deformation problems, such as metal forming, extrusion [253,254], large deformation of rubber materials [255,256], soil mechanics problem [257], shape design sensitivity and optimization, *etc* [71,258]. Li [226,259,260] and Jun [261] developed an explicit RKPM code to compute large deformation

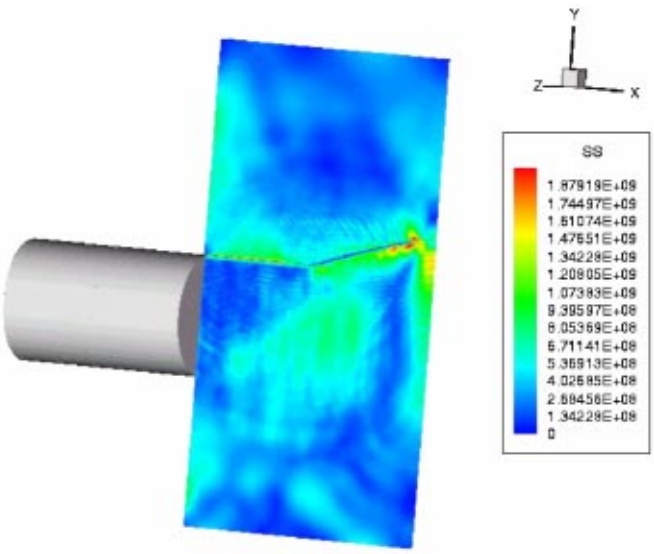

Fig. 7 Asymmetrical impact problem (effective stress contour)

problems as well. The explicit RKPM code has been extended into a 3D parallel code, which has been used to simulate 3D large deformations of thin shell structures, shear band propagation [248], crack growth. The main advantages of using meshfree methods in large deformation simulation are *a*) no remeshing; *b*) relief of volumetric locking for suitable choice of support size of shape function (which has been discussed by several authors [59,71,236,262,263]); and *c*) no complicated mixed formulations.

There are three approaches in numerical simulation of thin plates and shells structures [5]:

1) linear/nonlinear plate and shell theory approach;
2) degenerated continuum approach;
3) three-dimensional (3D) continuum approach.

Among these three approaches, the 3D continuum direct approach is the simplest and most accurate one in principle. Nonetheless, it is the least popular one in practice because the continuum approach requires deployment of multiple el-

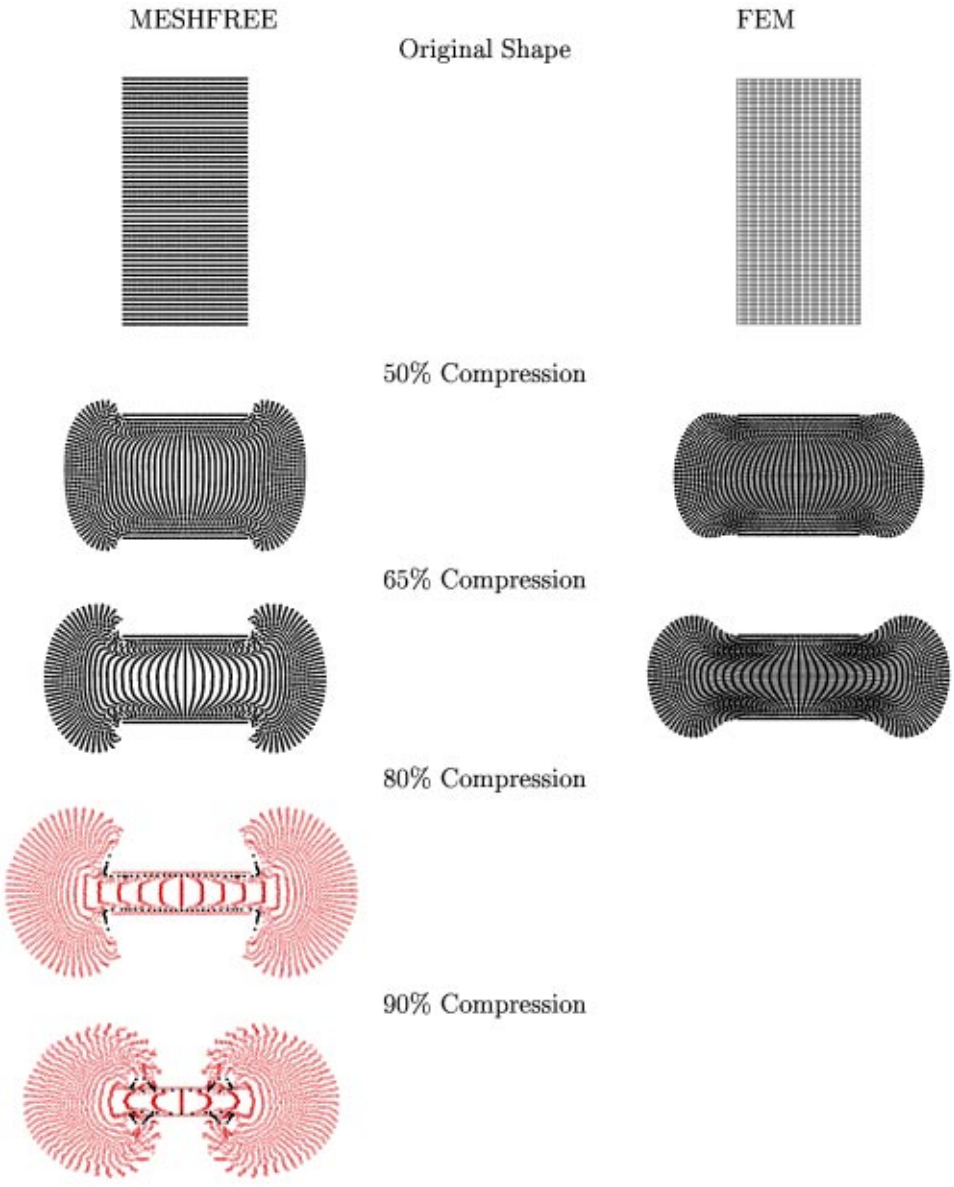

Fig. 8 Comparison of the deformations at different time stages for a block of hyperelastic material under compression by using MESHFREE and FEM when $\Delta t = 1 \times 10^{-6}$ (s)

ements in the thickness direction of a thin shell structure in order to acquire a reasonable gradient field. This degrades the conditioning of the discrete system (the discrete system becomes too stiff) and then accuracy of the numerical solution. On the other hand, the degenerated continuum approach as well as shell theory approach have the drawback of either shear/membrane locking, or difficulties in embedding inelastic constitutive relations. Krysl and Belytschko [264] first applied EFG method to thin plate/shell analysis as MLS interpolant can easily produce C^1 interpolation field. On the other hand, Donning and Liu [265] used a spline based particle method, Noguchi [266] used EFG method, and Garcia *et al* [267] used hp-Clouds to compute deformation of Mindlin plate problems. The problem is revisited again by Noguchi *et al* [268], who used a mapping technique to map the curvature surface to a flat 2D space, and discretization is being done on this 2D mapped space. In their formulation, a convected co-ordinate system is utilized in moving least square procedure. Good convergence results have been reported in those reports. In [226], Li *et al* found that one can use a meshfree interpolant in 3D direct continuum approach, because the smoothness of meshfree interpolant, one can accurately capture the gradient in thickness direction with 2 $\sim$4 layers of particles while avoiding both shear locking as well as volumetric locking in reasonable parameter range. In Fig. 9, large deformation of a pinched cylinder simulated by using meshfree interpolant is displayed [226]. Li *et al* [269] utilized the moving least square principle to devise a meshfree contact algorithm, which has been used in 3D metal forming applications by Qian *et al* [50,270,271].

Meshfree methods have been extensively used by Li *et al* [211,226,248,259,260,269], and others (*eg* [213,272]) in simulations of strain localization problems. By using a meshfree interpolant, one can effectively reduce the notorious mesh alignment sensitivity in strain localization simulation, since there is no mesh involved in meshfree discretization, whereas in finite element simulations the numerical shear band tends to grow along a finite element boundary instead of real physical paths. Chen *et al* [213] introduced an intrinsic length scale based on reproducing kernel approximation, which can be used as regularization in simulations of strain localization problems. It is shown that with proper choice of reproducing conditions, the method can reproduce a numerical gradient theory without introducing additional higher order boundary conditions that are required in all physical gradient theory.

Figure 10 presents a comparison between finite element computations and meshfree (RKPM) computations. The problem is a thin plate with 31 randomly distributed holes under uniaxial tension. In both finite element and meshfree computations, the same nodal/particle distributions have been used, one with a mesh, the other without. The nodal/particle distributions are a) evenly distributed, b) dense in the Y direction, and c) dense in the X direction. One can clearly observe the mesh alignment sensitivity in finite element computation, and the relief of such sensitivity in meshfree computation.

Using meshfree interpolants to conduct multiple scale computation can be rewarding as well. (See Fig. 11) Liu and his co-workers [68,198,273,274,275] were the first to use meshfree interpolant in multiple scale computations. Because reproducing kernel functions may be viewed as filters with different length scales, by choosing different dilation parameters, or different kernel functions (*eg* RKPM wavelets), one can formulate multiple scale formulations. This multiple scale meshfree method has been used in many applications from acoustics, wave propagation/scattering [199,273], wavelet adaptive refinement [211,218,275], fluid dynamics [274,276,277], large-eddy simulation [278], large deformation [275], strain localization [211], and damage [279,280].

Recently, Lee *et al* [281] used a two−scale meshfree method to calculate a 3D stress concentration problem. The RKPM meshfree interpolant provides both error indicator (low/high filter) as well as excellent frequency responses in multiple scale computations. Saigal and Barry suggested a slices based element free Galerkin formulation, which, they believed, can be used in solving problems with multiscale geometry, such as a bone block [282].

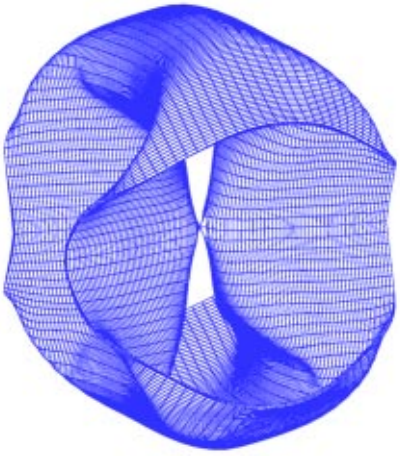

Fig. 9 Large deformation of a pinched cylinder

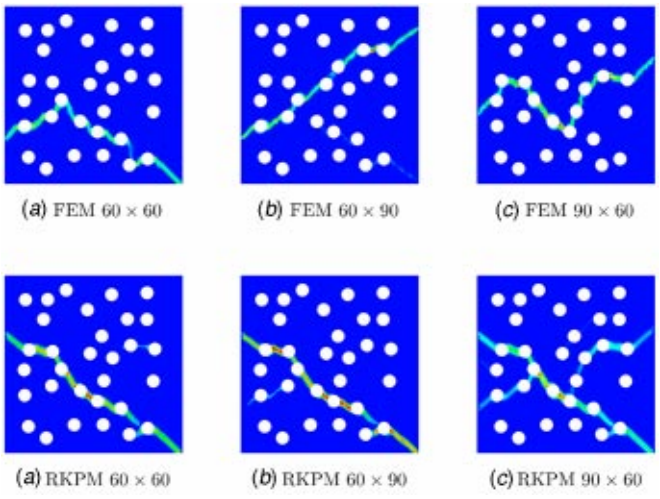

Fig. 10 Shear band pathes obtained via FEM and RKPM with different spatial aspect ratios in mesh/particle distribution

It is worth mentioning that Hao *et al* [280] used RKPM combined with finite element method in a micro-mechanics damage simulation. In that study, a multiple scale RKPM is used to simulate a ductile fracture process involving damage evolution, and multiresolution analysis has also been performed on shear bands formation. The numerical results show that the multiple scale RKPM possesses a strong ability to capture physical phenomena such as shear band, large deformation, and the material instability during damage evolution. Zhang *et al* [283] used EFG to model the jointed rock structures; Aluru [96] used RKPM to analyze microelectromechanical system. Danielson *et al* [284,285] has been developing a new communication scheme for parallel implementation of RKPM formulation. They have tested a quarter million particle computations in Cray T3E supercomputer in simulations of shear band and fracture. Recently, Zhang *et al* [286] have developed a parallel version of 3D RKPM code in implicit CFD calculation, which has the capacity to deal with more than one million particles. A novel procedure of implementing the essential boundary condition by using the bridging scale hierarchical enrichment, and the associated parallel communication with different processors is presented in that paper.

A simple illustration of the multiresolution meshfree method is given in Figs. 12 and 13. Figure 12 depicts the analysis of large deformation solids and the plastic deformation of a notched bar. The high scale solution (Fig. 12*c*) is an extraction from the total solution (Fig. 12*b*). It shows the crack tip field and the localized shear bands. The quantitative experimental result is given in Fig. 12*a*. Similarly, Fig. 13*a* depicts the high scale solution (obtained by wavelets decomposition of the total scale solution) for the pressure from the analysis of the compressible flow-structure interaction. This figure, labeled *High scale* clearly indicates the shock location and this solution can be used as an error indicator to guide the adaptivity which is simply implemented by addition of appropriately placed particles in the meshfree method. The total solution is given in Fig. 13*b*.

Due to the difficulties in imposing essential boundary conditions, a special meshfree contact algorithm is needed

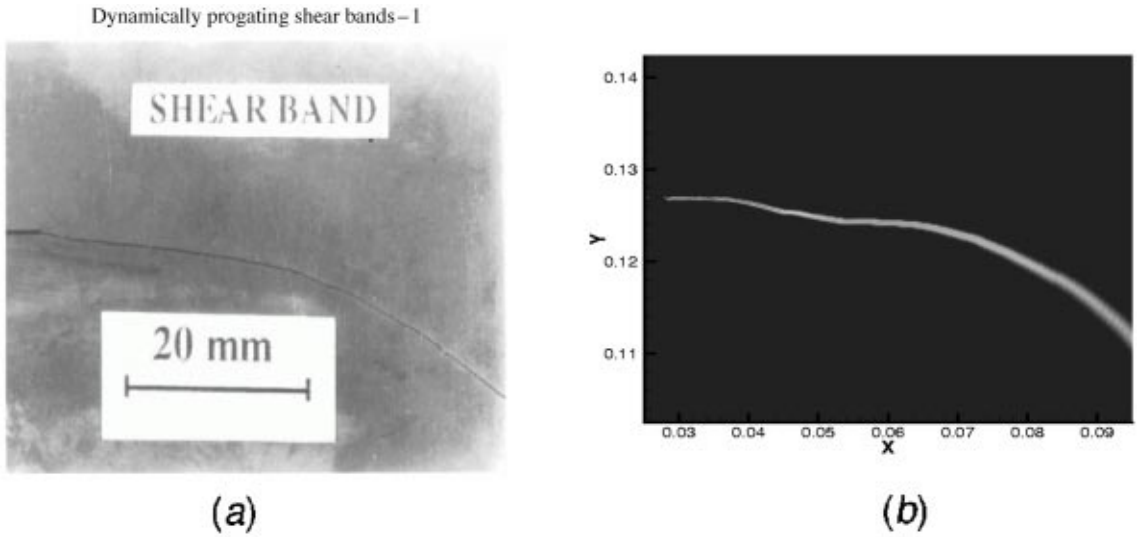

Fig. 11 Meshfree simulation of curved dynamic shear band: *a*) experimental observation; *b*) meshfree calculation [248]

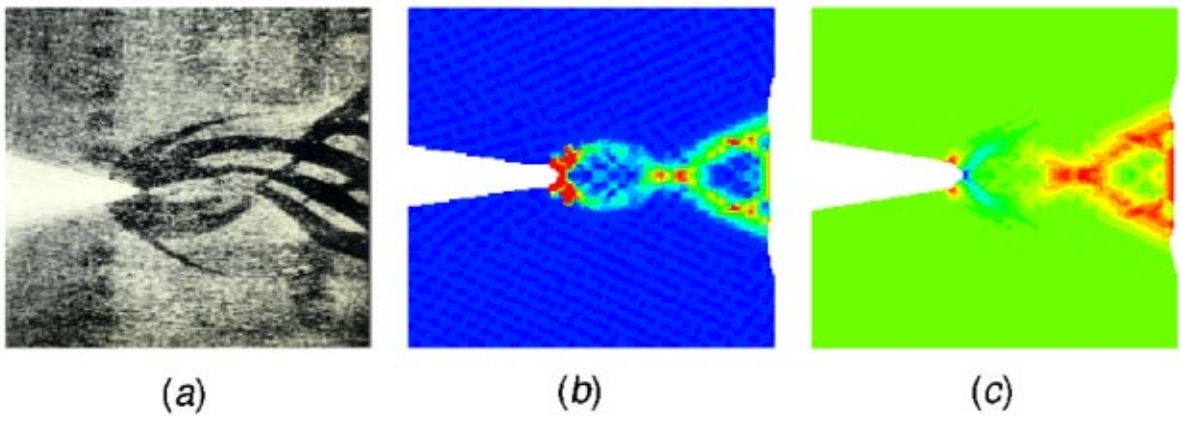

Fig. 12 Multi-scale Meshfree Simulation of strain localization of three point bending test

when solving problems such as impact, sheet metal forming, *etc*. Chen *et al* proposed a static smooth meshfree contact algorithm, in which the contact surface is represented by a reproducing kernel approximation using a parametric coordinate. This approach removes slope discontinuity in the C^0 finite element approximation and significantly improves the iteration convergence in large sliding contact problems [287]. This method has been used in shape sensitivity design, as well as sheet metal forming [254,256,287]. A dynamic meshfree contact algorithm is implemented by Li [269], in which a novel meshfree contact detection algorithm is presented. It has been used in computations of both impact problems and 3D sheet metal forming problems [270,271].

Recently, Hao *et al* [288] have developed a new particle method—the moving particle finite element method (MPFEM). The MPFEM developed out of the desire to combine the advantages of both finite element method (FEM) and meshfree method. In doing so, MPFEM has the ability to handle essential boundary conditions without recourse to special methods, it needs no background mesh to integrate the weak form, and the cost of computing shape functions is comparable to the FEM. As demonstrated in [288], the MPFEM approximation is computed point-wise by enforcing certain reproducing conditions. Any degree of polynomial can be reproduced by simply using more points to construct the approximation. The MPFEM has been shown to be effective in relieving locking in incompressible media problems and also in simulating large deformation penetration problems.

Figure 14 displayed the meshfree simulation of penetration: contours of damage. Due to the symmetry, a quarter of the nine projectiles (almost rigid) penetrating the target was

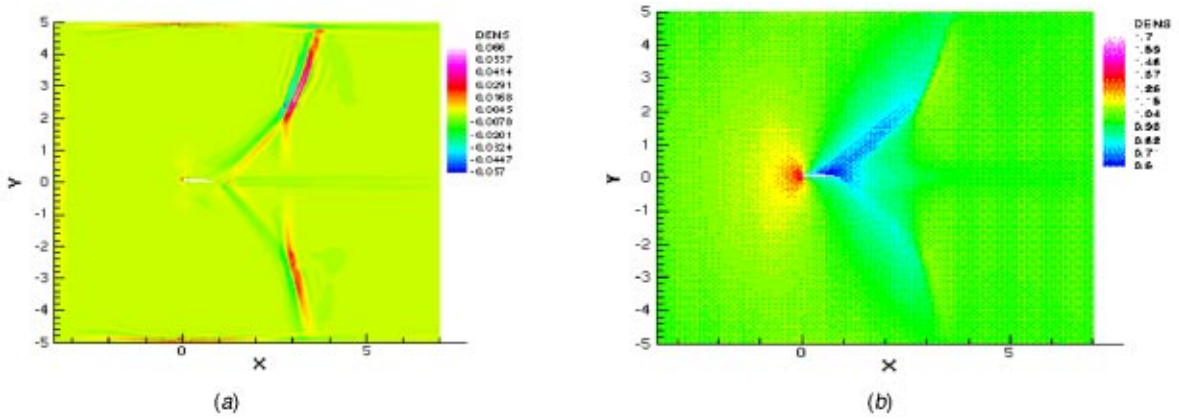

Fig. 13 Meshfree Galerkin Simulation of flow past an airfoil

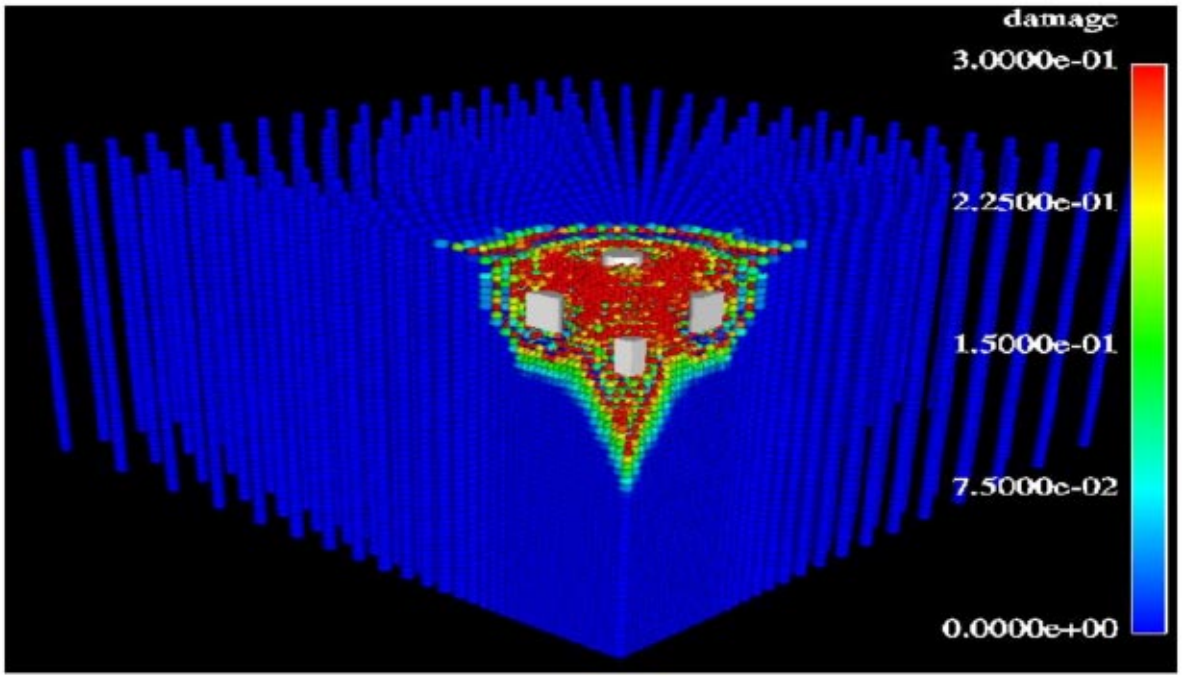

Fig. 14 Penetration of a concrete block

modeled with one-quarter projectile at the center, two half-projectiles at 90°, and a single projectile at 45°. It is noted that we do not use an erosion algorithm to get rid of the damage material and the size of the crater compares well with experimental observation.

4 AB INITIO METHODS AND MOLECULAR DYNAMICS

Molecular dynamics is probably the most important and most widely used particle method in scientific and engineering fields [27–32,289]. There are two types of molecular dynamics: the-first-principle-based molecular dynamics, or *ab initio* molecular dynamics; and semi-empirical molecular dynamics. Recently, both molecular dynamics have been applied to traditional engineering areas such as mechanical engineering, aerospace engineering, electrical engineering, and environmental engineering, among others. One fresh example is the large scale molecular dynamics simulations of fracture in solids at atomistic scale.

4.1 Ab initio methods

Based on our view of the hierarchical structure of the universe, it is believed that if one can understand the mechanics of a small length scale, then one can understand the mechanics at all scales. Though this fool-proof philosophy may be debatable, its simplicity is attractive, especially as we have entered into a new era of super-computing. According to our current knowledge, there are four forces in the universe,

i) strong interaction (nuclear force);
ii) Coulomb force (electrostatic force);
iii) weak interaction (the force related to β decay); and
iv) gravitational force.

Forces *i* and *iii* are short-ranged. They can be neglected in conventional engineering applications. The so-called first-principle calculations, or *ab initio* calculations only take into account of forces *ii* and *iv* in the framework of non-relativistic quantum mechanics. Technically speaking, *ab initio* methods are used to determine the electron density distribution, and the atomic structures of various materials. By so doing, one may be able to predict the various properties of a material at the atomic level.

Comparing to continuum mechanics, atomic scale simulation is indeed *ab initio*. However, non-relativistic quantum mechanics may not be the ultimate theory; besides, there are often many approximations involved in simulations of the quantum state of many-electron systems. The connotation of first-principle is used within a specific context. Ultimately, as Ohno *et al* [289] put it, ''only God can use the true methodology represented by the term, 'first principle methods'; humans have to use a methodology which is fairly reliable but not exact.''

4.1.1 Quantum mechanics of a many-electron system

In quantum mechanics, the state of an N-electron particle system can be described by its wave functions (*eg*, [290–292]). Denoting the Hamiltonian of the system as H, and its ξ-th eigenfunction (wavefunction) as $\Psi_\xi(1,2,\cdots,N)$, if we write the Hamiltonian for the i-th electron as H_i, the total Hamiltonian reads

$$H=H_1+H_2+\cdots+H_N \tag{92}$$

which may be explicitly written as

$$H=-\frac{1}{2}\sum_{i=1}^{N}\nabla_i^2+\sum_{i>j}^{N}\frac{1}{|\mathbf{r}_i-\mathbf{r}_j|}+\sum_{i=1}^{N}v(\mathbf{r}_i). \tag{93}$$

Note that the atomic units of $(e=\hbar=m=1)$ is used in (93). The first term in (93) represents the electron kinetic energy, the second term is due to the electron-electron Coulomb interaction, and the third term $v(\mathbf{r}_i)$ denotes the Coulomb potential caused by the nuclei. The electron distribution can be determined by solving the following steady state Schrödinger equation

$$\begin{aligned}&H\Psi_{\lambda_1,\lambda_2,\cdots,\lambda_N}(1,2,\cdots,N)\\&\quad=E_{\lambda_1,\lambda_2,\cdots,\lambda_N}\Psi_{\lambda_1,\lambda_2,\cdots,\lambda_N}(1,2,\cdots,N)\end{aligned} \tag{94}$$

where $E_{\lambda_1,\lambda_2,\cdots,\lambda_N}=\epsilon_{\lambda_1}+\epsilon_{\lambda_2}+\cdots+\epsilon_{\lambda_N}$ and ϵ_{λ_i} is the eigenvalue of the one electron Schrödinger equation $H_i\psi_{\lambda_i}(i)=\epsilon_{\lambda_i}\psi_{\lambda_i}(i)$. In most cases, the exact solution of the above system is almost impossible. Two approximations are commonly used in *ab initio* calculations: the Hartree-Fock approximation and the density functional theory.

4.1.2 Hartree-Fock approximation

The Hartree-Fock approximation [293–295], is a Ritz variational approximation. Since the exact solution of (94) is obtained by setting the following quadratic functional to minimum:

$$\begin{aligned}\langle\Psi|H|\Psi\rangle&=\sum_{s_1}\sum_{s_2}\cdots\sum_{s_N}\int\Psi^*(1,2,\cdots,N)\\&\quad\times H\Psi(1,2,\cdots,N)d\mathbf{r}_1d\mathbf{r}_2\cdots d\mathbf{r}_N\\&=\min\{E\}=E_0.\end{aligned} \tag{95}$$

The Hartree-Fock approximation is to solve the following one electron form of the Hartree-Fock equation instead of Eq. (95),

$$\begin{aligned}&H_0\psi_\lambda(i)+\left[\sum_{\nu=1}^{N}\sum_{s_i}\int\psi_\nu^*(j)U(i,j)\psi_\nu(j)d\mathbf{r}_j\right]\psi_\lambda(i)\\&\quad-\left[\sum_{\nu=1}^{N}\sum_{s_i}\int\psi_\nu^*(j)U(i,j)\psi_\lambda(j)d\mathbf{r}_j\right]\psi_\mu(i)\\&\quad=\epsilon_\lambda\psi_\lambda(i).\end{aligned} \tag{96}$$

Here $\psi_{\lambda_i}(i)$ is a one-electron solution of one-electron Schrödinger equation, $H_0(i)=-\frac{1}{2}\nabla_i^2+v(\mathbf{r}_i)$, and

$$U(i,j)=\frac{1}{|\mathbf{r}_i-\mathbf{r}_j|}; \tag{97}$$

$$v(\mathbf{r}_i)=-\sum_i\frac{Z_j}{|\mathbf{r}_i-\mathbf{R}_j|} \tag{98}$$

where Z_j is the nucleus charge of the j-th atom, and $\mathbf{R}_j$ is the spatial coordinate of the j-th atom.

In [296], the accuracy of large-scale (10,000 basis size) *ab initio* Hartree-Fock calculation is assessed. There is a large body of literature on Hartree-Fock quantum molecular dynamics simulations [297–299]. A good survey on research work done at the IBM Research Laboratory is presented by Clementi [300], who has done pioneering work in this field.

4.2 Density functional theory

An alternative method to solving an N-particle electron system is the Density Functional Theory [301–303]. The idea is similar to SPH—instead of studying a discrete N-body particle system, one assumes that there is a continuous electron density cloud, $\rho(\mathbf{r})$, such that the system's thermodynamic potential can be expressed as

$$\Omega = \int v(\mathbf{r})\rho(\mathbf{r})d\Omega + T[\rho(\mathbf{r})] + U[\rho(\mathbf{r})] + E_{xc}[\rho(\mathbf{r})] - \mu \int \rho(\mathbf{r})d\mathbf{r} \tag{99}$$

where $v(\mathbf{r})$ is the external potential, $T[\rho(\mathbf{r})]$ is the electron kinetic energy, $U[\rho(\mathbf{r})]$ the Coulomb potential, $E_{xc}[\rho(\mathbf{r})]$ is the exchange-correlation energy functional, and μ is the chemical potential. Based on this continuous representation, one may be able to solve the N-electron system by determining the solution of the following effective one-electron Schrödinger equation—Kohn-Sham equation

$$\left\{ -\frac{1}{2}\nabla^2 + v(\mathbf{r}) + \int \frac{\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}' + \mu_{xc}[\rho](\mathbf{r}) \right\} \psi_\lambda(\mathbf{r}) = \epsilon_\lambda \psi_\lambda(\mathbf{r}) \tag{100}$$

where $\mu_{xc}[\rho](\mathbf{r}) = \delta E_{xc}/\delta\rho(\mathbf{r})$.

There are other *ab initio* methods such as pseudo–potential approach, APW approach, Green's function method, *etc*. One may consult the monograph by Ohno *et al* [289] for detailed discussions.

4.3 Ab initio molecular dynamics

As a particle method, *ab initio* molecular dynamics is used to study material's properties at atomic coordinate level. In *ab initio* molecular dynamics, one needs to compute the wavefunctions of electrons as well as the movement of the nuclei. The velocity and the position of an atom is primarily determined by the position of the nucleus, which is not only influenced by the nuclei of other atoms surrounding it, but also by the electrons surrounding it. In addition, the wavefunction of an electron is also influenced by the presence of the nuclei nearby.

In most *ab initio* molecular dynamics, the so-called Born-Oppenheimer (BO) adiabatic approximation [304] is used. The approximation assumes that the temperature is very low, and hence only the ground state of electrons is considered, and in addition, the interaction between nuclei and electrons is neglected. In fact, up to today, *ab initio* molecular dynamics can only deal with the systems that obey the Born-Oppenheimer condition. In electron-nuclear system, nuclei behave like Newtonian particles, but the wavefunction of an electron is governed by the Schrödinger equation. A popular algorithm is the Car-Parrinello method [305]. Imagine that a small fictitious mass is attached to each electron; the steady state Schrödinger equation will become a hyperbolic equation. Then one can find both the electron wave function, ψ_λ, as well as the atomic coordinates, $\mathbf{R}_i$, by integrating the Newtonian equation of motion. When the fictitious mass attached to each electron approaches zero, the solution should converge to the solution of the coupled electron-nucleus many-body system. The computational task is to integrate the following equations

$$\begin{cases} \mu \dfrac{d^2}{dt^2}\psi_\lambda = -H\psi_\lambda + \sum_\nu \Lambda_{\lambda\nu}\psi_\nu, & (a) \\ M_i \dfrac{d^2}{dt^2}\mathbf{R}_i = -\nabla_i E, & (b) \end{cases} \tag{101}$$

where $\nabla_i E$ is the force acting the nucleus, which is determined by density functional theory as

$$-\nabla_i E = -\nabla_i \sum_{j\neq i} \frac{Z_i Z_j}{|\mathbf{R}_i - \mathbf{R}_j|} - \int \rho(\mathbf{r})\nabla_i v_i(|\mathbf{r}-\mathbf{R}|)d\mathbf{r} - \int \frac{\delta E\{\rho\}}{\delta\rho}\nabla_i\rho(\mathbf{r})d\mathbf{r}. \tag{102}$$

The time integration of the electron wave function is carried out by the following predictor-corrector algorithm:

$$\psi_\lambda^{n+1} = \phi^{n+1} + \frac{(\Delta t)^2}{\mu}\sum_\nu \Lambda_{\lambda\nu}\psi_\nu^n \tag{103}$$

$$\psi^{n+1} = 2\psi^n - \psi^{n-1} + \frac{(\Delta t)^2}{\mu}H\psi_\lambda^n \tag{104}$$

where n is the time step number. The unknown Lagrangian multiplier $\Lambda_{\lambda\nu}$ can be obtained from the orthogonality condition by solving nonlinear algebraic equations. This method is called the Ryckaert method [306]. Equation (101b) can be integrated using either leapfrog or Verlet method [307].

A brief review of quantum molecular dynamics on the simulation of nucleic acids can be found in [299]. A parallelization of general quantum mechanical molecular dynamics (QMMD) is presented in [25]. Simulations on liquid chemicals are reported in [308,309].

4.4 Classical molecular dynamics

At present, *ab initio* methods are restricted to simulations of several hundreds of atoms within the time scale of nanosecond. To simulate any systems larger than that is beyond the limit of current computation technology. In order to study real systems with large numbers of atoms for a longer time duration (or time scale), a simpler dynamics model that can represent most features of micromechanics at atomic length scale is desirable.

Classical molecular dynamics can simulate a system of one million to 1 billion atoms. In classical molecular dy-

namics, one does not calculate electron distribution anymore, the forces acting on each atom are determined by a potential function, *ie*,

$$m_i \frac{d^2}{dt^2}\mathbf{R}_i = -\nabla V \tag{105}$$

which is determined from either empirical knowledge, or from *ab initio* computations.

For example, in a polar molecule system of ionic crystal, or polar molecule system, the potential is mainly due to electrostatic interaction, thus

$$V(R_{ij}) = \sum_j \frac{q_n(\mathbf{R}_j)}{|\mathbf{R}_i - \mathbf{R}_j|} \tag{106}$$

where $R_{ij} = |\mathbf{R}_i - \mathbf{R}_j|$ and q_n is the charge distribution.

The most well-known potential, originally proposed for inert-gas elements, is the Lennard-Jones (LJ) potential [310,311], which is a typical Van der Waals potential. For a pair of atoms i and j located at $\mathbf{R}_i$ and $\mathbf{R}_j$, the potential energy is

$$V_{ij} = 4\epsilon_0 \left[\left(\frac{R_0}{R_{ij}} \right)^{12} - \left(\frac{R_0}{R_{ij}} \right)^{6} \right] \tag{107}$$

where ϵ_0 and R_0 are the minimum energy and collision diameter between the two atoms. respectively. The corresponding force between the two atoms is given by

$$F_{ij} = -\frac{\partial V(R_{ij})}{\partial R_{ij}} = 24\frac{\epsilon_0}{R} \left[2\left(\frac{R}{R_{ij}} \right)^{13} - \left(\frac{R}{R_{ij}} \right)^{7} \right]. \tag{108}$$

The Lennard-Jones (LJ) potential has been used by Falk and Langer [101,312,313] to simulate fracture as well as shear band in noncrystalline or amorphous solids.

In general, for simulation of anisotropic crystalline solid, the LJ potential, or pair potential, is not accurate anymore, and more complex potentials are needed, because the LJ potential is unable to represent specific interaction patterns due to specific lattice structures. To remedy this inadequacy, the embedded-atom potential method (EAM) has been used in simulations. The embedded-atom potential (Daw and Baskes [314]) consists of two sources: 1) the embedding energy for each atom to be introduced to the system, and 2) the short range core-to-core repulsion between nucleus pairs. Thus, its potential has the form,

$$V = \sum_i F_i(\rho_{h,i}) + \frac{1}{2}\sum_i \sum_j \phi_{ij}(R_{ij}) \tag{109}$$

where $\Phi(R_{ij})$ represents the pair potential, and $F_i(\rho_{h,i})$ represents the embedding energy of atom i, and $\rho_{h,i}$ is the density of the host at the position of R_i but without atom i. For example, in simulation of semiconductors, the fourfold coordinated Stillinger-Weber potential is adopted [315,316], which consists of a two-body part of LJ type

$$f_2(R_{ij}) = A(BR_{ij}^{-4} - 1)\exp[(R_{ij} - a)^{-1}] \tag{110}$$

and a three-body part

$$f_3(R_{ij}, R_{ik}, \theta_{ijk}) = \lambda \exp[\gamma(R_{ij} - a)^{-1} + \gamma(R_{ik} - a)^{-1}] \times [\cos\theta_{jik} + 1/3]^{-2}. \tag{111}$$

Between *ab initio* methods and classical molecular dynamics, there are other semi-empirical methods, such as the Tight-Binding Method [317–319]. The Tight-Binding method is a quantum mechanics method, because the forces acting on each atom are based on quantum mechanics, but it uses empirical parameters in the construction of the Hamiltonian. Those parameters can be obtained from either experiments or *ab initio* simulations.

4.5 Applications

4.5.1 Mechanics of nanotubes filled with fullerenes

The recent resurgence of molecular dynamics, both quantum and classical, is largely due to the emergence of nanotechnology. Materials at the nanoscale have demonstrated impressive physical and chemical properties, thus suggesting a wide range of areas for applications. For instance, carbon nanotubes are remarkably strong, and have better electrical conductance, as well as heat conducivity than copper at room temperature. Moreover, nanotubes are such light weight and high-strength (TPa) materials that they eventually will play an important role in reinforced fiber composites, and as both devices and nanowires. In particular, nanotubes having fullerenes inside could have different physical properties compared to empty nanotubes. Such structures also hold promise for use in potential functional devices at nanometer scale: nano-pistons, nano-bearings, nano-writing devices, and nano-capsule storage system.

Modeling of nanotubes filled with fullerenes has two aspects: 1) the bonded interaction between fullerenes and nanotubes; 2) the bonded interactions among the carbon atoms of the nanotubes. Recently, Qian *et al* [320] used combined molecular dynamics and meshfree Galerkin approach to simulate interaction between fullerenes and a nanotube. In the non-bonded interaction, the nanotube is modeled as a continuum governed by the Cauchy-Born rule (*eg* Tadmor *et al* [321] and Milstein [322]). For the bonded interaction, a modified potential is used to simulate interactions among carbon atoms. Specifically, Tersoff-Brenner model (Tersoff [323], Brenner 1990 [324]) is used in simulation,

$$\Phi_{ij}(R_{ij}) = \Phi_R(R_{ij}) - \overline{B}_{ij}\Phi_A(R_{ij}) \tag{112}$$

where Φ_R and Φ_A represent the repulsive and attractive potential respectively,

$$\Phi_R(R_{ij}) = f(R_{ij}) \frac{D_{ij}^{(e)}}{(S_{ij} - 1)} \exp\{-\sqrt{2S_{ij}}\beta_{ij}(R_{ij} - R_{ij}^{(e)})\} \tag{113}$$

$$\Phi_A(R_{ij}) = f(R_{ij}) \frac{D_{ij}^{(e)} S_{ij}}{(S_{ij} - 1)} \exp\{-\sqrt{2/S_{ij}}\beta_{ij}(R_{ij} - R_{ij}^{(e)})\}. \tag{114}$$

For carbon-carbon bonding, $D_{ij}^{(e)} = 6.0\,\text{eV}$, $S_{ij} = 1.22$, $\beta_{ij} = 2.1\,A^{-1}$, $R_{ij}^{e} = 1.39\,A$, and

$$f(r)=\begin{cases}1 & r<R_{ij}^{(1)}\\ \frac{1}{2}\left(1+\cos\left[\frac{\pi(r-R_{ij}^{(1)})}{R_{ij}^{(2)}-R_{ij}^{(1)}}\right]\right) & R_{ij}^{(1)}\leqslant r\leqslant R_{ij}^{(2)}\\ 0 & r>R_{ij}^{(2)}.\end{cases} \quad (115)$$

The effect of bonding angle is taken into account in term $\overline{B}_{ij}$ (see Brenner [324] and Qian *et al* [320]). In Fig. 15, the length of the nanotubes are $L=129\,\text{Å}$, and the diameter of the nanotube is 6.78 Å (5,5), which is close to the diameter of C_{60}.

4.5.2 Atomistic simulations of fracture

During past few years, molecular dynamics simulations have been used extensively in fracture and crack simulation at atomic scale, which is largely promoted and publicized by Bulatov *et al* [325]. The current research in this direction is often associated with the name of multi-scale simulation and multi-physics modeling, which is pioneered by the work done by Clementi and his co-workers [326–329]. Starting in the late 1980s, they have been systematically using supercomputers to carry out *ab initio* modeling, molecular dynamics modeling, Monte Carlo modeling, and phenomenological modeling in a single simulations. They mixed quantum molecular dynamics with continuum mechanics in a single simulation having multiple length scales.

Abraham and his co-workers have conducted extensive simulations ranging from brittle fracture [98,330–332] to ductile fracture [330,333,334] and brittle to ductile transition [335–337]. They have used both classical molecular dynamics and *ab initio* molecular dynamics to simulate crack growth [338]. The current effort is on using multiple scale simulations, or concurrent simulations by combining quantum electron distribution (*ab initio* method), classical atom dynamics (molecular dynamics), and the continuum solid (finite element simulation of solid mechanics) [99]. They developed a method called MAAD that dynamically couples continuum mechanics far from the crack, empirical potential MD near the crack, and quantum tight-binding (TB) dynamics at the crack tip to simulate fracture in silicon [99]. The method couples molecular dynamics with the finite element method in a so-called *handshake region* between MD and finite elements.

Gumbsch and his co-workers also systematically used both MD and *ab initio* methods to simulate brittle fracture [339–343]. They placed emphasis on the atomistic mechanisms of the fracture. Farkas and co-workers have extensively used molecular dynamics with the embedded-atom method (EAM) potential to study the atomistic aspect of fracture mechanics [344–348]. The atomistic simulations conducted by Farkas have been focused on crack propagation along a grain boundary, dislocations emitted from a crack tip, and ductile-to-brittle transitions. Falk and Langer [101,312,313,349] used classical MD with the LJ potential to simulate fracture and shear transformation zone (STZ) in noncrystalline solids.

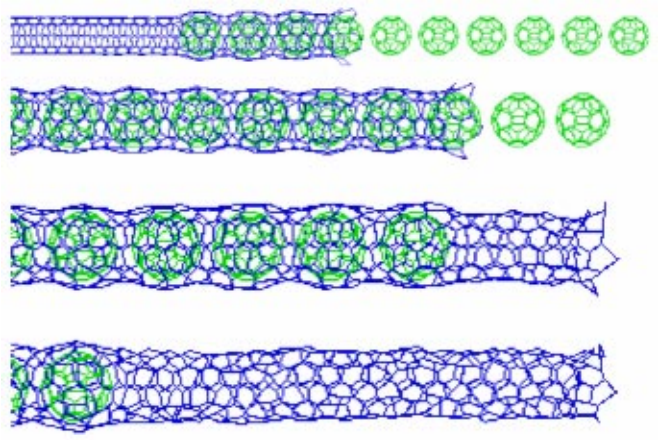

Fig. 15 Molecular dynamics simulations of C_{60} passing through nanotube [320]

5 OTHER PARTICLE METHODS

Many particle methods have been proposed during the past three decades. Each of these particle methods has its own merits, and so far it has not been found that there is a method that is suitable ''for all seasons.'' Research on developing new particle methods is still active. A few representatives of particle methods are worth mentioning. A very important one is the vortex method in fluid mechanics [97,14,15,17,18,350–352].

5.1 Vortex method

In computational fluid mechanics, most of the numerical algorithms for the Navier-Stokes equations are based on the velocity-pressure formulation. An alternative to velocity-pressure formulation is the vorticity-velocity formulation:

$$\frac{\partial\boldsymbol{\omega}}{\partial t}+(\mathbf{u}\cdot\nabla)\boldsymbol{\omega}=(\boldsymbol{\omega}\cdot\nabla)\mathbf{u}+\nu\Delta\boldsymbol{\omega} \quad (116)$$

$$\Delta\mathbf{u}=-\nabla\times\boldsymbol{\omega} \quad (117)$$

where vorticity $\boldsymbol{\omega}=\nabla\times\mathbf{u}$.

The Lagrangian form of the above equations are

$$\frac{d\mathbf{x}_I}{dt}=\mathbf{u}(\mathbf{x}_I,t) \quad (118)$$

$$\frac{d\boldsymbol{\omega}}{dt}=[\nabla\mathbf{u}(\mathbf{x}_I,t)]\boldsymbol{\omega}_I+\nu\Delta\boldsymbol{\omega}(\mathbf{x}_I,t) \quad (119)$$

where the velocity field can be obtained from the Poisson's equation (117). It can be expressed by the Biot-Savart integral,

$$\mathbf{u}(\mathbf{x},t)=\int G(\mathbf{x}-\mathbf{y})\times\boldsymbol{\omega}d\mathbf{y} \quad (120)$$

where the Green's function is

$$G(\mathbf{z})=\begin{cases}-\frac{1}{2\pi}\frac{z}{|z|^2} & 2D\\ \frac{1}{4\pi}\frac{z}{|z|^3} & 3D\end{cases}. \quad (121)$$

The essence of the vortex method is to discretize above the Lagrangian description by the finite number of moving material particles. Following the movement of these particles, one may construct or evaluate the velocity field as well as the vorticity field.

In early approaches, a point (singular) vortex method was employed to represent the vorticity field,

$$\boldsymbol{\omega}(\mathbf{x}) = \sum_I \Gamma_I \delta(\mathbf{x}-\mathbf{x}_I). \tag{122}$$

For example, the 2D discrete velocity field is

$$\frac{d\mathbf{x}_I}{dt} = \frac{1}{2\pi}\sum_J \frac{(\mathbf{x}_I-\mathbf{x}_J)\times \mathbf{e}_z\Gamma_J}{|\mathbf{x}_I-\mathbf{x}_J|^2}. \tag{123}$$

Today, most researchers use vortex blob or smooth vortex methods. It implies that a smoothing kernel function is used to eliminate singularities so that the algorithm may be more stable. The resulting equation becomes,

$$\boldsymbol{\omega}_\rho(\mathbf{x}) = \sum_I \Gamma_I \boldsymbol{\omega}_I \gamma_\rho(\mathbf{x}-\mathbf{x}_I) \tag{124}$$

where $\gamma_\rho(\mathbf{x}) = \rho^{-d}\gamma(\mathbf{x}/\rho)$ is the smoothing kernel. It may be noted that the idea of the vortex blob method is very similar to that of SPH or RKPM. When using the vortex blob method, the velocity field in 2D may be written as

$$\frac{d\mathbf{x}_I}{dt} = -\frac{1}{2\pi}\sum_J \frac{(\mathbf{x}_I-\mathbf{x}_J)\times \mathbf{e}_z\Gamma_J g(|\mathbf{x}_I-\mathbf{x}_J|/\rho_J)}{|\mathbf{x}_I-\mathbf{x}_J|^2} \tag{125}$$

where $G(y) = 2\pi\int_0^y \gamma(z) z\,dz$.

The vortex method was first used in computations of incompressible and inviscid flow, *eg* [97,351]. Later, it was applied to solve viscous flow problems [14,353,354], and show that the method has the ability to provide accurate simulation of complex high Reynolds number flows [13,352,355]. Two versions of vortex methods were used in early implementation: Chorin's random walk [14,15] and Leonard's core spreading technique [17,18]. Today, most people use the following re-sampling scheme:

$$\frac{d\mathbf{x}_I}{dt} = \sum_J V_J K_\rho(\mathbf{x}_I-\mathbf{x}_J)\times\boldsymbol{\omega}_J \tag{126}$$

$$\frac{d\boldsymbol{\omega}_I}{dt} = \left[\sum_J V_J \nabla K_\rho(\mathbf{x}_I-\mathbf{x}_J)\times\boldsymbol{\omega}_J\right] + \nu\rho^{-2}\sum_J V_J[\boldsymbol{\omega}_J-\boldsymbol{\omega}_I]\gamma_\rho(|\mathbf{x}_I-\mathbf{x}_J|). \tag{127}$$

5.2 Particle-in-cell method

Like the vortex-in-cell approach, the particle-in-cell method is a dual description (Lagrangian and Eulerian) method. The main idea is to trace the motions of a set of material points, which carry the information of all the state variables, in a Lagrangian manner; whereas the spatial discretization, hence the displacement interpolation, is made with respect to spatial coordinate detached from the material body as an Eulerian description. At the beginning of each time step, one may first find the velocities and accelerations at each spatial nodal point based on the information of surrounding material points. In the same manner, internal and external forces on a specific spatial nodal point at each time step are calculated by summing up the contribution from the surrounding material points. The method was first used in computational fluid dynamics by Brackbill [93,94,356–358]. It was reformulated by Sulsky and co-workers for solid mechanics applications. Some very good illustrations such as the Taylor bar impact problem and ring collision problem have been shown by Sulsky *et al* [92,95,359].

In the particle-in-cell method, the total mass or total volume of the continuum is divided among N particles

$$\rho(\mathbf{x},t) = \sum_I M_I \delta(\mathbf{x}-\mathbf{X}_I(t)). \tag{128}$$

Consider a weak formulation of the momentum equation

$$\int_\Omega \rho\mathbf{w}\cdot\mathbf{a}\,d\Omega = -\int_\Omega \rho\boldsymbol{\sigma}:\nabla\mathbf{w}\,d\Omega + \int_{\partial\Gamma_t}\mathbf{w}\cdot\mathbf{t}\,dS + \int_\Omega \rho\mathbf{w}\cdot\mathbf{b}\,d\Omega. \tag{129}$$

Substituting (128) into (129), a Lagrangian type of discretization can be achieved

$$\sum_{I=1}^{N_p} M_I\mathbf{w}(\mathbf{X}_I(t),t)\cdot\mathbf{a}(\mathbf{X}_I(t),t) = -\sum_{I=1}^{N_p} M_I\boldsymbol{\sigma}(\mathbf{X}_I(t),t):\nabla\mathbf{w}(\mathbf{x},t)\big|_{\mathbf{x}=\mathbf{X}_I(t))} + \int_{\Gamma_t}\mathbf{w}\cdot\mathbf{t}\,dS + \sum_{I=1}^{N_p} M_I\mathbf{w}(\mathbf{X}_I(t),t)\cdot\mathbf{b}(\mathbf{X}_I(t),t). \tag{130}$$

Since the kinemetic variables are discretized in an Eulerian grid, the accelerations are governed by the discrete equation of motion at spatial nodal points,

$$\sum_{j=1}^{N_n} m_{ij}\mathbf{a}_j = \mathbf{f}_i^{int} + \mathbf{f}_i^{ext}. \tag{131}$$

The exchange of information between the particles and spatial nodal points is described in [359]. The main advantage of the particle-in-cell method is to avoid using a Lagrangian mesh and to automatically track material boundaries. Recent applications of the particle-in-cell method are plasma physics (such as magneto-hydrodynamics, Maxwell-Lorentz equations), astrophysics, and shallow-water/free-surface flow simulations [89,90,360,361].

5.3 Lattice Boltzmann method

There have been several excellent reviews on the Lattice Boltzmann method (LBM) [52,54,362]. The discussion presented here is intended to put the method in comparison with its *peers*, and look at it from a different perspective. The ancestor of LBM is the Lattice Gas Cellular Automaton (LGCA) method, which is also regarded as a special case of molecular dynamics [27]. LBM is designed to improve its statistical *resolution*.

Currently, LBM is a very active research front in computational fluid dynamics because of its easy implementation and parallelization. The LBM technology has been used in simulations of low Mach number combustion [363], multiphase flow and Rayleigh-Taylor instability [364], flow past a cylinder [365], flow through porous media [366], turbulent flow, and thermal flow. One may also find some related references in [367–370] and a convergence study of LBM in [371].

The basic equation, or the kinetic equation, of the lattice Boltzmann method is

$$f_i(\mathbf{x}+\mathbf{e}_i\Delta\mathbf{x},t+\Delta t)-f_i(\mathbf{x},t)=\Omega_i(f(\mathbf{x},t)),\quad i=0,1,2,\cdots,M \tag{132}$$

where f_i is the particle velocity distribution function along the i-th direction, and Ω_i is the collision operator that represents the rate change of f_i during the collision.

Note that in the lattice Boltzmann method, for a particle at a given node, there are only a finite number of velocity directions $(\mathbf{e}_i, i=0,1,\cdots,M)$ that the particle can have. Figure 16 illustrates examples of plane lattice, and the discrete velocity paths. Figure 17 shows a 3D lattice with the associated discrete velocity set. Viewing Eq. (132) as a discrete meso-scale model, one can average (sum) the particle distribution over the discrete velocity space to obtain the macro-scale particle density at nodal position i,

$$\rho=\sum_{i=1}^{M} f_i\,. \tag{133}$$

The particle velocity momentum at macro-scale can also be obtained by averaging the meso-scale variables

$$\rho\mathbf{u}=\sum_{i=1}^{M} f_i\mathbf{e}_i\,. \tag{134}$$

Unlike most of the other particle methods, the lattice Boltzmann method is a mesh based method. In the LBM, the spatial space is discretized in a way that it is consistent with the kinetic equation, *ie* the coordinates of the nearest points around $\mathbf{x}$ are $\mathbf{x}+\mathbf{e}_i$. Therefore, it requires not only grid, but also the grid has to be uniform. This actually causes problems at general curved boundaries. Recently, efforts have been made to extend LBM to irregular grids [372,373], and specific techniques are developed to enforce boundary conditions [367,374]. During a simulation, particle moves from one lattice node to another. Most likely, there is a probability that the next node is also occupied by other particles. The non-zero density of particle distribution at that point indicates the possibility of collision.

There are two approaches to choosing collision operator Ω_i. Using the Chapman-Enskog expansion or multi-scale singular perturbation [375], one may find that the following continuum form of the kinetic equation,

$$\frac{\partial f_i}{\partial t}+\mathbf{e}_i\cdot\nabla\cdot f_i+\epsilon\left(\frac{1}{2}\mathbf{e}_i\mathbf{e}_i:\nabla\nabla f_i\mathbf{e}_i\nabla\frac{\partial f}{\partial t}+\frac{1}{2}\frac{\partial^2 f_i}{\partial t^2}\right)=\frac{\Omega_i}{\epsilon} \tag{135}$$

is consistent with the discrete kinetic equation (132) up to the second order of ϵ—a small number proportional to the Knudsen number. By choosing a proper collision operator, for instance using the lattice BGK theory (after Bhatnagar, Gross, and Krook in continuum kinetic theory [376]),

$$\frac{\Omega_i}{\epsilon}=-\frac{\delta_{ij}}{\epsilon\tau}(f_j-f_j^{eq}). \tag{136}$$

Equation (135) may recover Navier-Stokes hydrodynamics equations, provided the equilibrium state of particle density is well defined, *eg* that of Qian *et al* [50],

$$f_i^{eq}=\rho w_i\left(1+3\mathbf{e}_i\cdot\mathbf{u}+\frac{9}{2}(\mathbf{e}_i\cdot\mathbf{u})^2-\frac{3}{2}u^2\right). \tag{137}$$

The alternative is to consider Eq. (132) as the discrete version of the continuum Boltzmann equation, and one may derive the discrete collision operator by discretizing the Maxwell-Boltzmann equilibrium distribution [54,362]. The resulting difference equations may reproduce Navier-Stokes hydrodynamic equations in the limit of small Knudsen number, *ie* particle mean-free path much smaller than typical macroscopic variation scales [370].

In principle, the Lattice Boltzmann method is a *bona fide* computational meso-mechanics paradigm. It has both *micro-*

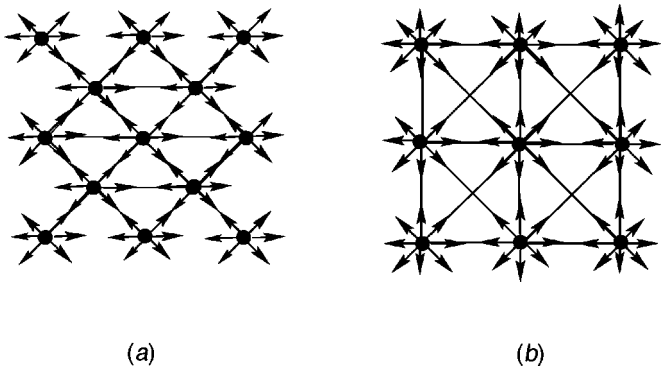

Fig. 16 Lattice and velocity directions: *a*) triangular lattice; *b*) square lattice

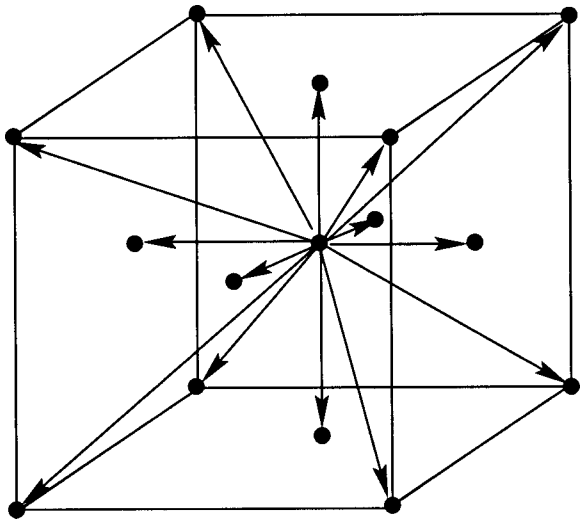

Fig. 17 Cubic Lattice with 15 molecular speeds (D3Q15)

mechanics part, the statistical movement of the molecules—Boltzmann equation, and the *homogenization* part, the assemble or averaging in the phase (velocity) space. In fact, to extend the Boltzmann Lattice method to irregular lattice, or quasi-lattice structure is the current research topic. In 1997, Succi [362] wrote:

> "*Most of the excitement behind LGCA was driven by the 'Grand-dream'*:
> *LGCA: Turbulence=Ising Model:Phase Transitions. Ten years later, all reasonable indications are that the 'Grand-dream' has turned into a 'Grand-illusion' (but, who knows the future ?).*
> *LBE was born on a much less ambitious footing: just provide a useful tool to investigate fluid dynamics and, maybe mesoscopic phenomena, on parallel machines. And in that respect, it appears hard to deny that, even though much remains to be done, the method has indeed lived up to the initial expectations. . . .*"

This assessment has been both accurate and modest, considering the recent development of LBM.

5.4 Natural element method

The natural element method (NEM) was first proposed by Braun and Sambridge [377,378], and was used for geophysical applications. Traversoni [379] proposed the method independently, and he used it in hydraulic engineering application. The natural element or natural neighbor method is based on the so-called Sibson co-ordinates to construct its interpolation function [380,381], which relies on the concept of the Voronoi diagram and Delaunay triangulations.

Sukumar *et al* [382–384] have systematically used the natural neighbor method to solve the solid mechanics problems. Bueche *et al* [385] studied the dispersive properties of the natural element when using it to solve wave and reduced wave equations. Cueto *et al* [386] modified it by means of density-scaled α-shapes to impose essential boundary condition over non-convex boundaries. Recently, Belikov *et al* [387] presented a non-Sibsonian interpolation scheme, which claims to have several advantages over the Sibsonian interpolation schemes.

5.5 Other meshfree methods

In a series of papers, Oñate *et al* proposed a so-called finite point method, mesh-free point method [388–391], which is a gridless numerical procedure based on the combination of weighted least square interpolations on a cloud of points with point collocation for evaluating the approximation integrals. It has been used to solve advection-diffusion problems and fluid flow mechanics.

An interesting meshfree proposal has been made recently by Pardo [392], who is seeking a middle ground between continuum mechanics formulation and statistical formulation. The intention is to solve continuum mechanics problems by actually solving a *mimic Feynman* path integral formulation, an analogy to the Lattice Boltzmann method (instead of solving the Euler equation, one solves a discrete Boltzmann equation). The well-known Feynman path integral of quantum mechanics is equivalent to the Schrödinger equation statistically, whereas the proposed *mimic Feynman* integral is equivalent to the Navier equation with the second order accuracy. The discrete mimic path integral is built on a set of infinitesimal propagators of local supports, and the discretization is truly meshfree.

Besides SPH, one of the early contributions of strong form collocation meshfree methods was Liszka and Orkisz's generalized finite difference method [19,20]. Another important contribution is Yagawa and Furukawa's free mesh method [86–88]. The free mesh method is a meshless FEM, which sounds paradoxical from its name. The idea here is to abandon the global finite element mesh, and it does not require connectivity information between element and nodes. The stiffness matrix is assembled node by node. For each node, at each time step, there may be several satellite nodal points surrounding it to form a temporary mesh, which will allow one to build shape function at that particular node. After that, one can move to the next nodal point. Although one still relies on Delaunay triangulation to set up the initial mesh, the topological data structure here is very different from conventional FEM and it is suitable for massively parallel computations, especially using domain decomposition. This is because the moving element scheme (cf, moving particle FEM [288]) is an element-by-element scheme and matrix-storage free formulation.

6 CONCLUSIONS AND DISCUSSION

In this survey, particle methods and their applications in applied mechanics have been reviewed. Most of the methods discussed here are based on approximations that do not require a mesh structure, and therefore they are called meshfree methods. Modeling with these methods only requires a set of unstructured points that cover the domain of interest. Since meshfree/particle methods have simple topological data structures, they allow easy adaptive refinement, easy parallelization, and flexible interpolation in a deformable domain. It has been shown that many problems that currently cannot be solved by finite element or finite difference methods are tractable by meshfree methods. This class of methods show great potential to meet the demands of modern software, error estimators, hp adaptivity, multiresolution analysis, sampling approximations, edge detection, *etc*. These are the traits that represent the future generation of computational methods, and will benefit applications in the many branches of engineering and physical sciences.

Although much has been achieved in the past decade, there are still many tasks and challenges remaining. These challenges include the cost-effective meshfree-Galerkin method; scalable implementation of essential boundary conditions; accurate nodal integration strategies, and stabilization schemes for both discretized weak form as well as collocated strong form formulations. Besides the algorithmic improvements, it is believed that meshfree particle methods will play a significant role in the next generation computational meso-mechanics, or computational micro-mechanics, which is the integrated part of nano-technology and supercomputing technology.

Today, computational meso-mechanics is still in its infant stage. Few paradigms are available. An outstanding example in CFD is the Lattice Boltzmann Method. A future direction may be developing a Boltzmann-type method without a lattice. In solid mechanics applications, the most ambitious project in computational micro-mechanics has been the multiple scale method, which combines quantum molecular dynamics, classical molecular dynamics, and continuum mechanics in a single simulation. It has been extensively used in large scale simulations (involving 10 to 1 billion atoms) of fracture and crack growth. The current multiple scale computation is a coupling between particle methods and finite element methods, which is a *mechanical bridging* of various length scales with different physics models. A future direction, we believe, is to develop multiple scale method of pure particle methods at all scales, which might have better numerical data structure.

The computational meso-mechanics models in solid mechanics, such as Needleman-Xu-Ortiz's cohesive finite element model [393–396], Tadmor's quasi-continuum model [321], and Gao and Klein's virtual internal bond (VIB) model [397], are all built upon finite element discretization, and all of them have been reported to have mesh alignment sensitivities in numerical simulations. An immediate task is to develop a computational meso-mechanics model based on particle methods, or meshfree methods. A number of researchers have been working towards this direction *eg* [185,392].

7 ACKNOWLEDGMENT

The authors would like to thank D Qian, G Wagner, and H Park for careful reading of the manuscript and providing help with the figures. The authors also would like to acknowledge the financial support from the Army Research Office, the National Science Foundation, and the Tull Family Endowment.

REFERENCES

[1] Liu WK, Belytschko T, and Chang H (1986), An arbitrary Lagrangian-Eulerian finite element method for path-dependent materials. *Comput. Methods Appl. Mech. Eng.* **58**, 227–246.
[2] Liu WK, Chang H, Chen JS, and Belytschko T (1988), Arbitrary Lagrangian and Eulerian Petrov-Galerkin finite elements for nonlinear problems, *Comput. Methods Appl. Mech. Eng.* **68**, 259–310.
[3] Huerta A and Liu WK (1988), Viscous flow with large free surface motion, *Comput. Methods Appl. Mech. Eng.* **9**, 277–324.
[4] Liu WK, Chen JS, Belytschko T, and Zhang YF (1991), Adaptive ALE finite elements with particular reference to external work rate on frictional interface, *Comput. Methods Appl. Mech. Eng.* **93**, 189–216.
[5] Belyschko T, Liu WK, and Moran B (2000), *Nonlinear Finite Elements for Continua and Structures*, John Wiley and Sons, New York.
[6] Needleman A (1988), Material rate dependent and mesh sensitivity in localization problems, *Comput. Methods Appl. Mech. Eng.* **67**, 68–85.
[7] Needleman A (1989), Dynamic shear band development in plane strain, *ASME J. Appl. Mech.* **56**, 1–9.
[8] Benz W (1990), Smooth particle hydrodynamics: a review. In: *Numerical Modeling of Non-linear Stellar Pulsation: Problems and Prospects*, Kluwer Academic, Boston.
[9] Gingold RA and Monaghan JJ (1977), Smoothed particle hydrodynamics: theory and application to non-spherical stars, *Mon. Not. R. Astron. Soc.* **181**, 375–389.
[10] Lucy LB (1977), A numerical approach to the testing of the fission hypothesis, *Astrophys. J.* **82**, 1013.
[11] Monaghan JJ (1982), Why particle methods work (Hydrodynamics), *SIAM (Soc. Ind. Appl. Math.) J. Sci. Stat. Comput.* **3**, 422–433.
[12] Monaghan JJ (1985), Particle methods for hydrodynamics, *Comput. Phys. Rep.* **3**, 71–124.
[13] Bernard PS (1995), A deterministic vortex sheet method for boundary layer flow, *J. Comput. Phys.* **117**, 132–145.
[14] Chorin AJ (1973), Numerical study of slightly viscous flow, *J. Fluid Mech.* **57**, 785–796.
[15] Chorin AJ (1973), Discretization of a vortex sheet, with an example of roll-up, *J. Comput. Phys.* **13**, 423–429.
[16] Chorin AJ (1978), Vortex sheet approximation of boundary layers, *J. Comput. Phys.* **27**, 428–442.
[17] Leonard A (1980), Vortex methods for flow simulation, *J. Comput. Phys.* **37**, 289–335.
[18] Leonard A (1985), Computing three-dimensional incompressible flows with vortex elements, *Annu. Rev. Fluid Mech.* **17**, 523–529.
[19] Liszka T, and Orkisz J (1980), The finite difference method at arbitrary irregular grids and its application in applied mechanics, *Comput. Struct.* **11**, 83–95.
[20] Liszka T (1984), An interpolation method for an irregular net of nodes, *Int. J. Numer. Methods Eng.* **20**, 1599–1612.
[21] Feldneier H and Schnack J (2000). Molecular dynamics for fermions. Technical report, Gesellschaft fur Schwerionenforschung mbH.
[22] Kobrak MN, and Bittner ER (2000), Quantum molecular dynamics study of polaron recombination in conjugated polymers, *Phys. Rev. B* **62**, 11473–11486.
[23] Krumrine JR, Jang S, Alexander MH, and Voth GA (2000), Quantum molecular dynamics and spectral simulation of a boron impurity in solid para-hydrogen, *J. Chem. Phys.* **113**, 9079–9089.
[24] Kihe C, Yildirim T, Mehrez H, and Ciraci S (2000), A first-principles study of the structure and dynamics of C_8H_8, Si_8H_8, and Ge_8H_8 moleculars, *J. Phys. Chem. A* **104**, 2724–2728.
[25] Hedman F, and Laaksonen A (2000), Parallel aspects of quantum molecular dynamics simulations of liquids, *Comput. Phys. Commun.* **128**, 284–294.
[26] Hong J and Zhao XS (2000), New propagators for quantum-classical molecular dynamics simulations, *J. Chem. Phys.* **113**, 930–935.
[27] Rapaport DC (1995), *The Art of Molecular Dynamics Simulation*, Cambridge Univ Press, Cambridge, UK.
[28] Allen MP and Tildesley DJ (1987), *Computer Simulation of Liquids*, Oxford Univ Press, Oxford, UK.
[29] Allen MP and Tildesley DJ ed, (1993) *Computer Simulation of Chemical Physics*, Kluwer Academic Pub, Dordrecht.
[30] Catlow CRA, Parker SC, and Allen MP ed, (1990), *Computer Modelling of Fluids Polymers and Solids*, Kluwer Academic Pub, Dordrecht.
[31] Ciccotti G and Hoover WG (eds) (1986), *Molecular Dynamics Simulation of Statistical Mechanical Systems*, North-Holland, Amsterdam.
[32] Ciccotti G, Frenkel D, and McDonald IR (eds) (1987), *Simulation of Liquids and Solids. Molecular Dynamics and Monte Carlo Methods in Statistical Mechanics*, North-Holland, Amsterdam.
[33] Bird GA (ed) (1994), *Molecular Gas Dynamics and the Direct Simulation of Gas Flow*, Oxford Univ Press, Oxford, UK.
[34] Oran ES, Oh CK, and Cybyk BZ (1998), Direct simulation Monte Carlo: Recent advances and applications, *Annu. Rev. Fluid Mech.* **30**, 403–441.
[35] Tunon I, Martins-Costa MTC, Millot C, Ruiz-Lopez MF, and Rivail JL (1996), A coupled density functional-molecular mechanics Monte Carlo simulation: the water molecule in liquid water, *J. Comput. Chem.* **17**, 19–29.
[36] Gross WJ, Vasileska D, and Ferry DK (1999), A novel approach for introducing the electron-electron and electron-impurity interactions in particle-based simulations, *IEEE Electron Device Lett.* **20**, 463–465.
[37] Drovetsky BY, Chu JC, and Mak CH (1998), Computer simulations of self-avoiding polymerized membranes, *J. Chem. Phys.* **108**, 6554–6557.
[38] Acioli PH (1997), Review of quantum monte carlo methods and their applications, *J. Mol. Struct.* **394**, 75–85.
[39] Binder K (ed) (1988), *The Monte Carlo Method in Condensed Matter Physics*, Springer, Berlin, Heidelberg.
[40] Binder K (ed) (1992), *The Monte Carlo Simulation in Statistical Physics*, Springer, Berlin, Heidelberg.
[41] Baer R (2000), Ab-initio molecular deformation barriers using auxiliary-field quantum Monte Carlo with application to the inversion barrier of water, *Chem. Phys. Lett.* **324**, 101–107.
[42] Liu WK, Belytschko T, and Mani A (1986), Probabilistic finite elements for nonlinear structural dynamics, *Comput. Methods Appl. Mech. Eng.* **56**, 61–81.
[43] Liu WK, Belytschko T, and Mani A (1986), Random field finite elements, *Int. J. Numer. Methods Eng.* **23**, 1831–1845.

[44] Liu WK, Chen YJ, and Belytschko T (1996), Three reliability methods for fatigue crack growth, *Eng. Fract. Mech.* **53**, 733–752.
[45] Frisch U, Hasslacher B, and Pomeau Y (1986), Lattice gas cellular automata for the Navier-Stokes equations, *Phys. Rev. Lett.* **56**, 1505.
[46] Kadanoff L (1986), On two levels, *Phys. Today* **39**, 7–9.
[47] Kadanoff L, McNamara GR, and Zanetti G (1987), A Poiseuille viscometer for lattice gas automata, *Complex Syst.* **1**, 791.
[48] Kadanoff L, McNamara GR, and Zanetti G (1989), From automata to fluid flow: comparisons of simulation and theory, *Phys. Rev. A* **40**, 4527.
[49] Henon M (1987), Viscosity of a lattice gas, *Complex Syst.* **1**, 763.
[50] Qian YH, d'Humiéres D, and Lallemand P (1992), Lattice BGK models for the Navier-Stokes equation, *Europhys. Lett.* **17**, 479–484.
[51] Qian YH and Orszag SA (1993), Lattice BGK models for the Navier-Stokes equation: Nonlinear deviation in compressible regimes, *Europhys. Lett.* **21**, 255–259.
[52] Qian YH, Succi S, and Orszag SA (2000), Recent advances in lattice Boltzmann computing, In: *Annual Reviews of Computational Physics, Volume III*, D Stauffer (ed) World Scientific, Singapore, 195–242.
[53] Chen S, Wang Z, Shan XW, and Doolen GD (1992), Lattice Boltzmann computational fluid dynamics in three dimensions, *J. Stat. Phys.* **68**, 379–400.
[54] Chen S and Doolen GD (1998), Lattice Boltzmann method for fluid flows, *Annu. Rev. Fluid Mech.* **30**, 329–364.
[55] Nayroles B, Touzot G, and Villon P (1992), Generalizing the finite element method: Diffuse approximation and diffuse elements, *Computational Mech., Berlin* **10**, 307–318.
[56] Breitkopf P, Touzot G, and Villon P (1998), Consistency approach and diffuse derivation in element free methods based on moving least squares approximation, *Comp. Assist. Mech. Eng. Sci.* **5**, 479–501 ISSN:1232-308X.
[57] Breitkopf P, Touzot G, and Villon P (2000), Double grid diffuse collocation method. *Computational Mech., Berlin* **25**, 199–206.
[58] Breitkopf P, Rassineux A, Touzot G, and Villon P (2000), Explicit form and efficient computation of MLS shape function and their derivatives, *Int. J. Numer. Methods Eng.* **48**, 451–466.
[59] Belytschko T, Lu YY, and Gu L (1994), Element free galerkin methods. *Int. J. Numer. Methods Eng.* **37**, 229–256.
[60] Belytschko T, Krongauz Y, Organ D, Fleming M, and Krysl P (1996), Meshless methods: An overview and recent developments, *Comput. Methods Appl. Mech. Eng.* **139**, 3–48.
[61] Belytschko T, Krongauz Y, Dolbow J, and Gerlach C (1998), On the completeness of meshfree particle methods, *Int. J. Numer. Methods Eng.* **43**, 785–819.
[62] Belytschko T, Organ D, and Gerlach C (2000), Element-free Galerkin methods for dynamic fracture in concrete, *Comput. Methods Appl. Mech. Eng.* **187**, 385–399.
[63] Lu YY, Belytschko T, and Tabbara M (1995), Element-free Galerkin method for wave propagation and dynamic fracture, *Comput. Methods Appl. Mech. Eng.* **126**, 131–153.
[64] Liu WK, Adee J, and Jun S (1993), Reproducing kernel and wavelets particle methods for elastic and plastic problems, In: *Advanced Computational Methods for Material Modeling*, AMD 180/PVP 268 ASME, 175–190.
[65] Liu WK and Oberste-Brandenburg C (1993), Reproducing kernel and wavelets particle methods, In: *Aerospace Structures: Nonlinear Dynamics and System Response*, AD 33 ASME, 39–56.
[66] Liu WK, Jun S, and Zhang YF (1995), Reproducing kernel particle methods, *Int. J. Numer. Methods Eng.* **20**, 1081–1106.
[67] Liu WK, Jun S, Li S, Adee J, and Belytschko T (1995), Reproducing kernel particle methods for structural dynamics, *Int. J. Numer. Methods Eng.* **38**, 1655–1679.
[68] Liu WK, Chen Y, Chang CT, and Belytschko T (1996), Advances in multiple scale kernel particle methods, *Computational Mech., Berlin* **18**, 73–111.
[69] Liu WK, Chen Y, Jun S, Chen JS, Belytschko T, Uras RA, and Chang CT (1996), Overview and applications of the reproducing kernel particle methods, *Arch. Comput. Mech. Eng.: State of Rev.* **3**, 3–80.
[70] Liu WK, Li S, and Belytschko T (1997), Moving least square reproducing kernel method Part I: Methodology and convergence, *Comput. Methods Appl. Mech. Eng.* **143**, 422–453.
[71] Chen JS, Pan C, Wu CT, and Liu WK (1996), Reproducing kernel particle methods for large deformation analysis of nonlinear structures, *Comput. Methods Appl. Mech. Eng.* **139**, 195–227.
[72] Chen JS, Wu CT, Yoon S, and You Y (2001), A stabilized conforming nodal integration for Galerkin meshfree methods, *Int. J. Numer. Methods Eng.* **50**, 435–466.
[73] Duarte CA, and Oden JT (1996), hp Clouds—an hp meshless method, *Numer. Methods Partial Diff. Eqs.* **12**, 673–705.
[74] Duarte CA, and Oden JT (1996), An hp adaptive method using clouds, *Comput. Methods Appl. Mech. Eng.* **139**, 237–262.
[75] Liszka T, Duarte CAM, and Tworzydlo WW (1996), hp-meshless cloud method, *Comput. Methods Appl. Mech. Eng.* **139**, 263–288.
[76] Oden JT, Duarte CAM, and Zienkiewicz OC (1998), A new Cloud-based hp finite element method, *Comput. Methods Appl. Mech. Eng.* **153**, 117–126.
[77] Babuška I and Melenk JM (1997), The partition of unity method, *Int. J. Numer. Methods Eng.* **40**, 727–758.
[78] Babuška I and Zhang Z (1998), The partition of unity method for the elastically supported beam, *Comput. Methods Appl. Mech. Eng.* **152**, 1–18.
[79] Melenk JM and Babuška I (1996), The partition of unity finite element method: Basic theory and applications, *Comput. Methods Appl. Mech. Eng.* **139**, 289–314.
[80] Atluri SN, and Zhu T (1998), A new meshless local Petrov-Galerkin (MLPG) approach to nonlinear problems in computer modeling and simulation, *Comput. Model. Simul. Eng.* **3**, 187–196.
[81] Atluri SN, Kim HG, and Cho JY (1999), A critical assessment of the truly meshless local Petrov-Galerkin (MLPG) and local boundary integral equation (LBIE) methods, *Computational Mech., Berlin* **24**, 348–372.
[82] Atluri SN, Cho JY, and Kim HG (1999), Analysis of thin beams, using the meshless local Petrov-Galerkin method, with generalized moving least square interpolations, *Computational Mech., Berlin* **24**, 334–347.
[83] Atluri SN and Zhu T (2000), The meshless local Petrov-Galerkin (MLPG) approach for solving problems in elasto-statics, *Computational Mech., Berlin* **25**, 169–179.
[84] Furukawa T, Yang C, Yagawa G, and Wu CC (2000), Quadrilateral approaches for accurate free mesh method, *Int. J. Numer. Methods Eng.* **47**, 1445–1462.
[85] Shirazaki M and Yagawa G (1999), Large-scale parallel flow analysis based on free mesh method: A virtually meshless method, *Comput. Methods Appl. Mech. Eng.* **174**, 419–431.
[86] Yagawa G and Yamada T (1996), Free mesh method: A new meshless finite element method, *Computational Mech., Berlin* **18**, 383–386.
[87] Yagawa G and Yamada T (1998), Meshless method on massively parallel processors with application to fracture mechanics, *Key Eng. Mater.* **145–149**, 201–210.
[88] Yagawa G and Furukawa T (2000), Recent development of free mesh method, *Int. J. Numer. Methods Eng.* **47**, 1419–1417.
[89] Cushman-Roisin B, Esenkov OE, and Mathias BJ (2000), A particle-in-cell method for the solution of two-layer shallow-water equations, *Int. J. Numer. Methods Fluids* **32**, 515–543.
[90] Munz CD, Schneider R, Sonnendrücker E, Stein E, Voss U, and Westermann T (1999), A finite-volume particle-in-cell method for the numerical treatment of Maxwell-Lorentz equations on boundary-fitted meshes, *Int. J. Numer. Methods Eng.* **44**, 461–487.
[91] Munz CD, Schneider R, and Voss U (1999), A finite-volume particle-in-cell method for the numerical simulation of devices in pulsed-power technology, *Surv. Math. Ind.* **8**, 243–257.
[92] Bardenhagen SG, Brackbill JU, and Sulsky D (2000), The material-point method for granular materials, *Comput. Methods Appl. Mech. Eng.* **187**, 529–541.
[93] Brackbill JU and Ruppel HM (1986), FLIP: A method for adaptively zoned, particle-in-cell calculations in two dimensions, *J. Comput. Phys.* **65**, 314–343.
[94] Brackbill JU (1987), On modeling angular momentum and velocity in compressible fluid flow, *Comput. Phys. Commun.* **47**, 1.
[95] Sulsky D and Schreyer HL (1996), Axisymmetric form of the material point with applications to upsetting and Taylor impact problems, *Comput. Methods Appl. Mech. Eng.* **139**, 409–429.
[96] Aluru NR (2000), A point collocation method based on reproducing kernel approximations, *Int. J. Numer. Methods Eng.* **47**, 1083–1121.
[97] Anderson C and Greengard C (1985), On vortex methods, *SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal.* **22**, 413–440.
[98] Abraham FF (1996), Parallel simulations of rapid fracture, In: *Fracture-Instability Dynamics, Scaling and Ductile/Brittle Behavior Symp.*, Mater. Res. Soc, Pittsburgh PA, 311–320.
[99] Abraham FF, Bernstein N, Broughton JQ, and Hess D (2000), Dynamic fracture of silicon: Concurrent simulation of quantum electrons, classical atoms, and the continuum solid, *MRS Bull.* **25**, 27–32.
[100] Foiles SM, Baskes MI, and Daw MS (1986), Embedded-atom-method functions for FCC metals Cu, Ag, Au, Ni, Pd, Pt, and their alloys, *Phys. Rev. B* **33**, 7983–7991.
[101] Falk ML and Langer JS (2000), From simulation to theory in the physics of deformation and fracture, *MRS Bull.* **25**, 40–45.
[102] Galli G, Cygi F, and Catellani A (1996), Quantum mechanical simu-

lations of microfracture in a complex material, *Phys. Rev. Lett.* **82**, 3476–3479.
[103] Galli G (1996), Linear scaling methods for electronic structure calculations and quantum molecular dynamics simulations, *Curr. Opin. Solid State Mater. Sci.* **1**, 864–874.
[104] Monaghan JJ (1992), Smoothed particle hydrodynamics, *Annu. Rev. Astron. Astrophys.* **30**, 543–574.
[105] Hultman J and Pharayn A (1999), Hierarchical, dissipative formation of elliptical galaxies: Is thermal instability the key mechanism? Hydrodynamical simulations including supernova feedback multi-phase gas and metal enrichment in CDM: Structure and dynamics of elliptical galaxies, *Astron. Astrophys.* **347**, 769–798.
[106] Monaghan JJ and Lattanzio JC (1991), A simulation of the collapse and fragmentation of cooling molecular clouds, *Astrophys. J.* **375**, 177–189.
[107] Berczik P and Kolesnik IG (1993), Smoothed particle hydrodynamics and its application to astrophysical problems, *Kinematics and Physics of Celestial Bodies* **9**, 1–11.
[108] Berczik P and Kolesnik IG (1998), Gasdynamical model of the triaxial protogalaxy collapse, *Astron. Astrophys. Trans.* **16**, 163–185.
[109] Berczik P (2000), Modeling the star formation in galaxies using the chemo-dynamical sph code, *Astron. Astrophys.* **360**, 76–84.
[110] Lee WH (1998), Newtonian hydrodynamics of the coalescence of black holes with neutron stars ii. tidally locked binaries with a soft equation of state, *Mon. Not. R. Astron. Soc.* **308**, 780–794.
[111] Lee WH (2000), Newtonian hydrodynamics of the coalescence of black holes with neutron stars iii. irrotational binaries with a stiff equation of state, *Mon. Not. R. Astron. Soc.* **318**, 606–624.
[112] Garcia-Senz D, Bravo E, and Woosley SE (1999), Single and multiple detonations in white dwarfs, *Astron. Astrophys.* **349**, 177–188.
[113] Monaghan JJ (1990), Modeling the universe, *Proc. Astron. Soc. Aust.* **18**, 233–237.
[114] Kum O, Hoover WG, and Posch HA (1995), Viscous conducting flows with smooth-particle applied mechanics, *Phys. Rev. E* **109**, 67–75.
[115] Posch HA, Hoover WG, and Kum O (1995), Steady-state shear flows via nonequilibrium molecular dynamics and smooth-particle applied mechanics, *Phys. Rev. E* **52**, 1711–1719.
[116] Monaghan JJ and Gingold RA (1983), Shock simulation by the particle method SPH, *J. Comput. Phys.* **52**, 374–389.
[117] Libersky LD, Petschek AG, Carney TC, Hipp JR, and Allahdadi FA (1993), High strain Lagrangian hydrodynamics a three-dimensional SPH code for dynamic material response, *J. Comput. Phys.* **109**, 67–75.
[118] Bonet J and Kulasegaram S (2000), Correction and stabilization of smooth particle hydrodynamic methods with applications in metal forming simulations, *Int. J. Numer. Methods Eng.* **47**, 1189–1214.
[119] Libersky LD and Petschek AG (1991), Smooth particle hydrodynamics with strength of materials, In: Advances in the Free-Lagrange Method, Springer, New York, 248–257.
[120] Randles PW and Libersky LD (1996), Smoothed particle hydrodynamics: Some recent improvements and applications, *Comput. Methods Appl. Mech. Eng.* **139**, 375–408.
[121] Johnson GR, Petersen EH, and Stryk RA (1993), Incorporation of an SPH option into EPIC code for a wide range of high velocity impact computations, *Int. J. Impact Eng.* **14**, 385–394.
[122] Johnson GR, Stryk RA, and Beissel SR (1996), SPH for high velocity impact computations, *Comput. Methods Appl. Mech. Eng.* **139**, 347–374.
[123] Johnson GR and Beissel SR (1996), Normalized smoothing functions for SPH impact computations, *Int. J. Numer. Methods Eng.* **39**, 2725–2741.
[124] Attaway SW, Heinstein MW, and Swegle JW (1994), Coupling of smooth particle hydrodynamics with the finite element method, *Nucl. Eng. Des.* **150**, 199–205.
[125] Taylor LM and Flanagan DP (1987), PRONTO 2D—A two-dimensional transient solid dynamics program, Tech Report SAND 86-0594, Sandia National Labs.
[126] Cummins SJ and Rudman M (1999), An SPH projection method, *J. Comput. Phys.* **152**, 584–607.
[127] Monaghan JJ (1989), On the problem of penetration in particle methods, *J. Comput. Phys.* **82**, 1–15.
[128] Morris JP, Fox PJ, and Zhu Y (1997), Modeling low Reynolds number incompressible flows using SPH, *J. Comput. Phys.* **136**, 214–226.
[129] Monaghan JJ (1994), Simulating free surface flow with SPH, *J. Comput. Phys.* **110**, 399.
[130] Monaghan JJ and Kocharyan A (1995), SPH simulation of multiphase flow, *Comput. Phys. Commun.* **87**, 225.
[131] Takeda H, Miyama SM, and Sekiya M (1994), Numerical simulation of viscous flow by smoothed particle hydrodynamics, *Prog. Theor. Phys.* **116**, 123–134.
[132] Welton WC and Pope SB (1997), PDF model calculations of compressible turbulent flows using smoothed particle hydrodynamics, *J. Comput. Phys.* **134**, 150–168.
[133] Welton WC (1998), Two-dimensional PDF/SPH simulations of compressible turbulent flows, *J. Comput. Phys.* **139**, 410–443.
[134] Cleary PW and Monaghan JJ (1999), Conduction modelling using smoothed particle hydrodynamics, *J. Comput. Phys.* **148**, 227–264.
[135] Bateson W and Hewett DW (1998), Grid and particle hydrodynamics, *J. Comput. Phys.* **144**, 358–378.
[136] Chow E and Monaghan JJ (1997), Ultrarelativistic SPH, *J. Comput. Phys.* **134**, 296–305.
[137] Faber JA and Rasio FA (2000), Post-Newtonian SPH calculations of binary neutron star coalescence: Method and first results, *Phys. Rev. D* **62**, 064012 (1–23).
[138] Siegler S and Riffert H (2000), Smoothed particle hydrodynamics simulations of ultrarelativistic shocks, *Astrophys. J.* **531**, 1053–1066.
[139] Chen JK, Beraun JE, and Jih TC (1999), A corrective smoothed particle method for boundary value problems in heat conduction, *Int. J. Numer. Methods Eng.* **46**, 231–252.
[140] Chen JK and Beraun JE (2000), A generalized smoothed particle hydrodynamics method for nonlinear dynamic problems, *Comput. Methods Appl. Mech. Eng.* **190**, 225–239.
[141] Benz W and Asphaug E (1995), Simulations of brittle solids using smooth particle hydrodynamics, *Comput. Phys. Commun.* **87**, 253–265.
[142] Gingold RA and Monaghan JJ (1982), Kernel estimates as a basis for general particle methods in hydrodynamics, *J. Comput. Phys.* **46**, 429–453.
[143] Monaghan JJ and Lattanzio JC (1985), A refined particle method for astrophysical problems, *Astron. Astrophys.* **149**, 135–143.
[144] Monaghan JJ and Pongracic H (1985), Artificial viscosity for particle methods, *Appl. Numer. Math.* **1**, 187–194.
[145] Monaghan JJ (1988), An introduction to SPH, *Comput. Phys. Commun.* **48**, 89–96.
[146] Monaghan JJ (1997), SPH and riemann solvers, *J. Comput. Phys.* **136**, 298–307.
[147] Monaghan JJ (1999), Implicit SPH drag and dust gas dynamics, *J. Comput. Phys.* **138**, 801–820.
[148] Monaghan JJ (2000), SPH without a tensile instability, *J. Comput. Phys.* **159**, 290–311.
[149] Petschek AG and Libersky LD (1993), Cylindrical smoothed particle hydrodynamics, *J. Comput. Phys.* **109**, 76–83.
[150] Swegle JW, Hicks DL, and Attaway SW (1995), Smoothed particle hydrodynamics stability analysis, *J. Comput. Phys.* **116**, 123–134.
[151] Belytschko T, Guo Y, Liu WK, and Xiao P (2000), A unified stability analysis of meshless particle methods, *Int. J. Numer. Methods Eng.* **48**, 1359–1400.
[152] Morris JP (1996), Stability properties of SPH, *Publ. - Astron. Soc. Aust.* **13**, 97.
[153] Dyka CT and Ingel RP (1995), An approach for tension instability in smoothed particle hydrodynamics, *Comput. Struct.* **57**, 573–580.
[154] Dyka CT, Randles PW, and Ingel RP (1995), Stress points for tensor instability in sph, *Int. J. Numer. Methods Eng.* **40**, 2325–2341.
[155] Dilts GA (1999), Moving least-square particle hydrodynamics I: Consistency and stability, *Int. J. Numer. Methods Eng.* **44**, 1115–1155.
[156] Dilts GA (2000), Moving least-square particle hydrodynamics II: Conservation and boundaries, *Int. J. Numer. Methods Eng.* **48**, 1503–1524.
[157] Vignjevic R, Campbell J, and Libersky L (2000), A treatment of zero-energy modes in the smoothed particle hydrodynamics method, *Comput. Methods Appl. Mech. Eng.* **184**, 67–85.
[158] Balsara DS (1995), Von Neumann stability analysis of smoothed particle hydrodynamics—Suggestions for optimal algorithms, *J. Comput. Phys.* **121**, 357–372.
[159] Chen JK, Beraun JE, and Jih CJ (1999), An improvement for tensile instability in smoothed particle hydrodynamics, *Computational Mech., Berlin* **23**, 279–287.
[160] Randles PW and Libersky LD (2000), Normalized SPH with stress points, *Int. J. Numer. Methods Eng.* **48**, 1445–1462.
[161] Chen JK, Beraun JE, and Jih CJ (1999), Completeness of corrective smoothed particle method for linear elastodynamics, *Computational Mech., Berlin* **24**, 273–285.
[162] Niedereiter H (1978), Quasi-Monte Carlo methods and pseudo-random numbers, *Bull. Am. Math. Soc.* **84**, 957–1041.
[163] Wozniakowski H (1991), Average case complexity of multivariate integration, *Bull. Am. Math. Soc.* **24**, 185–194.
[164] Di Lisio R, Grenier E, and Pulvirenti M (1998), The convergence of

the SPH method, *Comput. Math. Appl.* **35**, 95–102.
[165] Riffert H, Herold H, Flebbe O, and Ruber H (1995), Numerical aspects of the smoothed particle hydrodynamics method for simulating accretion disks, *Comput. Phys. Commun.* **89**, 1–16.
[166] Bonet J and Lok TS (1999), Variational and momentum preservation aspects of smooth particle hydrodynamic formulation, *Comput. Methods Appl. Mech. Eng.* **180**, 97–115.
[167] Capuzzo-Dolcetta R and Di Lisio R (2000), A criterion for the choice of the interpolation kernel in smoothed particle hydrodynamics, *Appl. Numer. Math.* **34**, 363–371.
[168] Hicks DL, Swegle JW, and Attaway SW (1997), Conservative smoothing stabilizes discrete-numerical instabilities in SPH materials dynamics computations, *Appl. Math. Comput.* **85**, 209–226.
[169] Hicks DL (1999), SPH hydrocodes can be stabilized with shape-shifting, *Comput. Math. Appl.* **38**, 1–16.
[170] Campbell J, Vignjevic R, and Libersky L (2000), A contact algorithm for smoothed particle hydrodynamics, *Comput. Methods Appl. Mech. Eng.* **184**, 49–65.
[171] Boffin HMJ, Watkin SJ, Bhattal AS, Francis N, and Whitworth AP (1998), Numerical simulations of protostellar encounters I: Star-disc encounters, *Mon. Not. R. Astron. Soc.* **300**, 1189–1204.
[172] Boffin HMJ, Watkin SJ, Bhattal AS, Francis N, and Whitworth AP (1998), Numerical simulations of protostellar encounters II: Coplanar disc-disc encounters, *Mon. Not. R. Astron. Soc.* **300**, 1205–1213.
[173] Boffin HMJ, Watkin SJ, Bhattal AS, Francis N, and Whitworth AP (1998), Numerical simulations of protostellar encounters. III: Non-coplanar disc-disc encounters, *Mon. Not. R. Astron. Soc.* **300**, 1214–1224.
[174] Marinho EP and Lepine JRD (2000), SPH simulations of chump formation by dissipative collision of molecular clouds I: Non-magnetic case, *Astron. Astrophys., Suppl. Ser.* **142**, 165–179.
[175] Yoshikawa K, Jing JP, and Suto Y (2000), Cosmological smoothed particle hydrodynamic simulations with four million particles: Statistical properties of X-ray clusters in a low-density universe, *Astrophys. J.* **535**, 593–601.
[176] Owen JM, Villumsen JV, Shapiro PR, and Martel H (1996), Adaptive smoothed particle hydrodynamics: Methodology I, *Astrophys. J., Suppl. Ser.* **103**, 269–330.
[177] Owen JM, Villumsen JV, Shapiro PR, and Martel H (1998), Adaptive smoothed particle hydrodynamics: Methodology II, *Astrophys. J., Suppl. Ser.* **116**, 155–209.
[178] Seto N (2000), Perturbative analysis of adaptive smoothing methods in quantifying large-scale structure, *Astrophys. J., Suppl. Ser.* **538**, 11–28.
[179] Dave R, Dubinski J, and Hernquist L (1997), Parallel TreeSPH, *New Astron.* **2**, 277–297.
[180] Hernquist L and Katz N (1989), TREESPH: A unification of SPH with the hierarchical tree method, *Astrophys. J., Suppl. Ser.* **70**, 419–446.
[181] Lia C and Carraro G (2000), A parallel tree SPH code for galaxy formation, *Mon. Not. R. Astron. Soc.* **314**, 145–161.
[182] Plimpton S, Attaway S, Hendrickson B, Swegle J, and Vaughan C (1998), Parallel transient dynamics simulations: algorithms for contact detection and smoothed particle hydrodynamics, *Journal of Parallel and Distributed Computing* **50**, 104–122.
[183] Gutfraind R and Savage SB (1997), Smoothed particle hydrodynamics for the simulation of broken-ice field: Mohr-Coulomb-type rheology and frictional boundary conditions, *J. Comput. Phys.* **134**, 203–215.
[184] Oger L and Savage SB (1999), Smoothed particle hydrodynamics for cohesive grains, *Comput. Methods Appl. Mech. Eng.* **180**, 169–183.
[185] Gutfraind R and Savage SB (1998), Flow of fractured ice through wedge-shaped channels: Smoothed particle hydrodynamics and discrete-element simulations, *Mech. Mater.* **29**, 1–17.
[186] Birnbaum NK, Francis NJ, and Gerber BI (1999), Coupled techniques for the simulation of fluid-structure and impact problems, *Computer Assisted Mechanics and Engineering Science* **16**, 295–311.
[187] Fahrenthold EP and Koo JC (1997), Hamiltonian particle hydrodynamics, *Comput. Methods Appl. Mech. Eng.* **146**, 43–52.
[188] Liu WK, Zhang Y, and Ramirez MR (1991), Multiple scale finite element methods, *Int. J. Numer. Methods Eng.* **32**, 969–990.
[189] Lancaster P and Salkauskas K (1980), Surface generated by moving least square methods, *Math. Comput.* **37**, 141–158.
[190] Belytschko T, Lu YY, and Gu L (1994), Fracture and crack growth by element-free Galerkin methods, *Model. Simul. Sci. Comput. Engrg.* **2**, 519–534.
[191] Belytschko T, Lu YY, and Gu L (1995), Element-free Galerkin methods for static and dynamic fracture, *Int. J. Solids Struct.* **32**, 2547–2570.
[192] Belytschko T, Lu YY, and Gu L (1995), Crack propagation by element-free Galerkin methods, *Eng. Fract. Mech.* **51**, 295–315.
[193] Liu WK (1995), An introduction to wavelet reproducing kernel particle methods, *USACM Bull.* **8**, 3–16.
[194] Shepard D (1968), A two-dimensional interpolation function for irregularly spaced points, In: *Proc of ACM National Conf*, 517–524.
[195] Belytschko T, Krongauz Y, Fleming M, Organ D, and Liu WK (1996), Smoothing and accelerated computations in the element free Galerkin method, *J. Comput. Appl. Math.* **74**, 111–126.
[196] Fleming M, Chu YA, Moran B, and Belytschko T (1997), Enriched element-free Galerkin methods for crack tip fields, *Int. J. Numer. Methods Eng.* **40**, 1483–1504.
[197] Rao BN and Rahman S (2000), An efficient meshless method for fracture analysis of crack, *Computational Mech., Berlin* **26**, 398–408.
[198] Liu WK, Chen Y, Uras RA, and Chang CT (1996), Generalized multiple scale reproducing kernel particle methods, *Comput. Methods Appl. Mech. Eng.* **139**, 91–158.
[199] Uras RA, Chang CT, Chen Y, and Liu WK (1997), Multiresolution reproducing kernel particle methods in acoustics, *J. Comput. Acoust.* **5**, 71–94.
[200] Suleau S and Bouillard Ph (2000), One-dimensional dispersion analysis for the element-free Galerkin method for the Helmholtz equation, *Int. J. Numer. Methods Eng.* **47**, 1169–1188.
[201] Suleau S, Deraemaeker A, and Bouillard Ph (2000), Dispersion and pollution of meshless solution for the Helmholtz equation, *Comput. Methods Appl. Mech. Eng.* **190**, 639–657.
[202] Bouillard Ph and Suleau S (1998), Element-free Galerkin solutions for Helmholtz problems: Formulation and numerical assessment of the pollution effect, *Comput. Methods Appl. Mech. Eng.* **162**, 317–335.
[203] Christon MA and Voth TE (2000), Results of von neumann analyses for reproducing kernel semi-discretizations, *Int. J. Numer. Methods Eng.* **47**, 1285–1301.
[204] Li S (1997), Moving Least Square Reproducing Kernel Methods, PhD thesis, McCormick School of Eng and Applied Science, Northwestern Univ, Evanston IL, May.
[205] Farwig R (1986), Multivariate interpolation of arbitrarily spaced data by moving least squares methods, *J. Comput. Appl. Math.* **16**, 79–93.
[206] Farwig R (1986), Rate of convergence of shepard's global interpolation formula, *Math. Comput.* **46**, 577–590.
[207] Li S and Liu WK (1998), Reproducing kernel hierarchical partition of unity Part I: Formulation and theory, *Int. J. Numer. Methods Eng.* **45**, 251–288.
[208] Daubechies I (1992), *Ten Lectures on Wavelets*, Soc for Indust and Appl Math, Philadelphia.
[209] Chui CK (1992), *An Introduction to Wavelets*, Academic Press, Boston.
[210] Li S and Liu WK (1998), Synchronized reproducing kernel interpolant via multiple wavelet expansion, *Computational Mech., Berlin* **28**, 28–47.
[211] Li S and Liu WK (1998), Reproducing kernel hierarchical partition of unity Part II: Applications, *Int. J. Numer. Methods Eng.* **45**, 289–317.
[212] Günther F, Liu WK, Diachin D, and Christon MA (2000), Multi-scale meshfree parallel computations for viscous compressible flows, *Comput. Methods Appl. Mech. Eng.* **190**, 279–303.
[213] Chen JS, Wu CT, and Belytschko T (2000), Regularization of material instabilities by meshfree approximations with intrinsic length scales, *Int. J. Numer. Methods Eng.* **47**, 1303–1322.
[214] Wagner GJ and Liu WK (2000), Hierarchical enrichment for bridging scales and meshfree boundary conditions, *Int. J. Numer. Methods Eng.* **50**, 507–524.
[215] Huerta A and Fernández-Méndez S (2000), Enrichment and coupling of the finite element and meshless methods, *Int. J. Numer. Methods Eng.* **48**, 1615–1636.
[216] Fernández-Mendez S, Diez P, and Huerta A (2001), Convergence of finite elements enriched with meshless methods, *SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal.* (submitted).
[217] Han W, Wagner GJ, and Liu WK (2002), Convergence analysis of a hierarchical enrichment of dirichlet boundary conditions in a meshfree method, *Int. J. Numer. Methods Eng.* **53**(6), 1323–1336.
[218] Liu WK, Uras RA, and Chen Y (1997), Enrichment of the finite element method with reproducing kernel particle method, *ASME J. Appl. Mech.* **64**, 861–870.
[219] B Szabó and I Babuška (1991), *Finite Element Analysis*, John Wiley & Sons, New York.
[220] Dolbow J, Möse N, and Belytschko T (2000), Discontinuous enrichment in finite elements with a partition of unity method, *Finite Elem. Anal. Design* **36**, 235–260.
[221] Wagner GJ, Möse N, Liu WK, and Belytschko T (2000), The ex-

tended finite element method for rigid particles in stokes flow, *Int. J. Numer. Methods Eng.* **51**, 293–313.
[222] Daux C, Möse N, Dolboaw J, Sukumar N, and Belytschko T (2000), Arbitrary branched and intersecting cracks with the extended finite element method, *Int. J. Numer. Methods Eng.* **48**, 1741–1760.
[223] Wagner GJ (2001), A Numerical Investigation of Particulate Channel Flows, PhD thesis, Northwestern Univ, Evanston IL.
[224] Gosz J and Liu WK (1996), Admissible approximations for essential boundary conditions in the reproducing kernel particle method, *Computational Mech., Berlin* **19**, 120–135.
[225] Zhu T and Atluri SN (1998), A modified collocation method and a penalty formulation for enforcing the essential boundary conditions in the element free Galerkin method, *Computational Mech., Berlin* **21**, 211–222.
[226] Li S, Hao W, and Liu WK (2000), Numerical simulations of large deformation of thin shell structures using meshfree methods, *Computational Mech., Berlin* **25**, 102–116.
[227] Chen JS and Wang HF (2000), New boundary condition treatments in meshfree computation of contact problems, *Comput. Methods Appl. Mech. Eng.* **187**, 441–468.
[228] Wagner GJ and Liu WK (2000), Application of essential boundary conditions in mesh-free methods: A corrected collocation method, *Int. J. Numer. Methods Eng.* **47**, 1367–1379.
[229] Günther F and Liu WK (1998), Implementation of boundary conditions for meshless methods, *Comput. Methods Appl. Mech. Eng.* **163**, 205–230.
[230] Kalijevic I and Saigal S (1997), An improved element free Galerkin formulation, *Int. J. Numer. Methods Eng.* **40**, 2953–2974.
[231] Krongauz Y and Belytschko T (1996), Enforcement of essential boundary conditions in meshless approximations using finite elements, *Comput. Methods Appl. Mech. Eng.* **131**, 133–145.
[232] Liu GR and Gu YT (2000), Meshless local Petrov-Galerkin (MLPG) method in combination with finite element and boundary element approaches, *Computational Mech.* **26**, 536–546.
[233] Pang Z (2000), Treatment of point loads in element free Galerkin method (EFGM), *Commun. in Numer. Methods in Eng.* **16**, 335–341.
[234] Klaas O and Shepard MS (2000), Automatic generation of octree-based three-dimensional discretization for partition of unity methods, *Computational Mech.* **25**, 296–304.
[235] Belytschko T and Tabbara M (1997), Dynamic fracture using element-free Galerkin methods, *J. Comput. Appl. Math.* **39**, 923–938.
[236] Dolbow J and Belytschko T (1999), Volumetric locking in the element-free Galerkin method, *Int. J. Numer. Methods Eng.* **46**, 925–942.
[237] Beissel S and Belytschko T (1996), Nodal integration of the element-free Galerkin method, *Comput. Methods Appl. Mech. Eng.* **139**, 49–74.
[238] Dolbow J and Belytschko T (1999), Numerical integration of the Galerkin weak form in meshfree methods, *Computational Mech.* **23**, 219–230.
[239] Atluri SN, Sladek J, Sladek V, and Zhu T (2000), The local boundary integral equation (LBIE) and its meshless implementation for linear elasticity, *Computational Mech.* **25**, 180–198.
[240] Atluri SN and Zhu T (2000), New concepts in meshless methods, *Int. J. Numer. Methods Eng.* **47**, 537–556.
[241] Zhu T, Zhang J, and Atluri SN (1999), A meshless numerical method based on the local boundary integral equation (LBIE) to solve linear and non-linear boundary value problems, *Eng. Anal. Boundary Elem.* **23**, 375–389.
[242] Zhu T (1999), A new meshless regular local boundary integral equation (MRLBIE) approach, *Int. J. Numer. Methods Eng.* **46**, 1237–1252.
[243] Sladek V, Sladek J, Atluri SN, and Van Keer R (2000), Numerical integration of singularities in meshless implementation of local boundary integral equations, *Computational Mech.* **25**, 394–403.
[244] De S and Bathe KJ (2000), The method of finite spheres, *Computational Mech.* **25**, 329–345.
[245] Chen JS, Wu CT, and Yoon S (2001), Nonlinear version of stabilized conforming nodal integration for Galerkin meshfree methods, *Int. J. Numer. Methods Eng*. In press.
[246] Krysl P and Belytschko T (1996), Element-free Galerkin method: convergence of the continuous and discontinuous shape function, *Comput. Methods Appl. Mech. Eng.* **148**, 257–277.
[247] Krysl P and Belytschko T (1999), The element free Galerkin method for dynamic propagation of arbitrary 3-d cracks, *Int. J. Solids Struct.* **44**, 767–800.
[248] Li S, Liu WK, Rosakis A, Belytschko T, and Hao W (2001), Meshfree Galerkin simulations of dynamic shear band propagation and failure mode transition, *Int. J. Solids Struct.* (in press).
[249] Li S, Liu WK, Qian D, Guduru R, and Rosakis AJ (2001), Dynamic shear band propagation and micro-structure of adiabatic shear band, *Comput. Methods Appl. Mech. Eng.* **191**, 73–92.
[250] Zhou M, Rosakis AJ, and Ravichandran G (1996), Dynamically propagating shear bands in impact-loaded prenotched plates—I. Experimental investigations of temperature signatures and propagation speed, *J. Mech. Phys. Solids* **44**, 981–1006.
[251] Kalthoff JF (1987), Shadow optical analysis of dynamic shear fracture, *Proc. of SPIE, Photomechanics and Speckle Metrology* **814**, 531–538.
[252] Kalthoff JF and Winkler S (1987), Failure mode transition at high rates of shear loading, In: *Impact Loading and Dynamic Behavior of Materials*, CY Chiem, HD Kunze, and LW Meyer (eds), Vol. 1, 185–195.
[253] Chen JS, Pan C, Wu CT, and Roque C (1998), A Lagrangian reproducing kernel particle method for metal forming analysis, *Computational Mechanics* **21**, 289–307.
[254] Chen JS, Roque C, Pan C, and Button ST (1998), Analysis of metal forming process based on meshless method, *J. Mater. Process. Technol.* **80–81**, 642–646.
[255] Chen JS, Pan C, and Wu CT (1997), Large deformation analysis of rubber based on a reproducing kernel particle method, *Computational Mech.* **19**, 153–168.
[256] Chen JS, Pan C, and Wu CT (1998), Application of reproducing kernel particle methods to large deformation and contact analysis of elastomers, *Rubber Chem. Technol.* **7**, 191–213.
[257] Wu CT, Chen JS, Chi L, and Huck F (2001), Lagrangian meshfree formulation for analysis of geotechnical materials, *J. Eng. Mech.* **127**, 140–149.
[258] Kim NH, Choi KK, Chen JS, and Park YH (2000), Meshless shape design sensitivity and optimization for contact problem with friction, *Computational Mech.* **25**, 157–168.
[259] Li S and Liu WK (2000), Numerical simulations of strain localization in inelastic solids using mesh-free methods, *Int. J. Numer. Methods Eng.* **48**, 1285–1309.
[260] Li S, Hao W, and Liu WK (2000), Meshfree simulations of shear banding in large deformation, *Int. J. Solids Struct.* **37**, 7185–7206.
[261] Jun S, Liu WK, and Belytschko T (1998), Explicit reproducing kernel particle methods for large deformation problems, *Int. J. Numer. Methods Eng.* **41**, 137–166.
[262] Chen JS, Yoon S, Wang HP, and Liu WK (2000), An improvement reproducing kernel particle method for nearly incompressible hyperelastic solids, *Comput. Methods Appl. Mech. Eng.* **181**, 117–145.
[263] Askes H, de Borst R, and Heeres OM (1999), Conditions for locking-free elasto-plastic analysis in the element-free Galerkin method, *Comput. Methods Appl. Mech. Eng.* **173**, 99–109.
[264] Krysl P and Belytschko T (1996), Analysis of thin shells by the element-free Galerkin method, *Int. J. Solids Struct.* **33**, 3057–3080.
[265] Donning BM and Liu WK (1998), Meshless methods for shear-deformable beams and plates, *Comput. Methods Appl. Mech. Eng.* **152**, 47–71.
[266] Noguchi H (1997), Application of element free Galerkin method to analysis of Mindlin type plate/shell problems, *Proc of ICE97*, 918–923.
[267] Garcia O, Fancello EA, de Barcellos CS, and Duarte CA (2000), hp-Clouds in Mindlin's thick plate model, *Int. J. Numer. Methods Eng.* **47**, 1381–1400.
[268] Noguchi H, Kawashima T, and Miyamura T (2000), Element free analysis of shell and spatial structures, *Int. J. Numer. Methods Eng.* **47**, 1215–1240.
[269] Li S, Qian D, Liu WK, and Belytschko T (2000), A meshfree contact-detection algorithm, *Comput. Methods Appl. Mech. Eng.* **190**, 7185–7206.
[270] Qian D, Li S, and Cao J (2000). 3D simulation of manufacturing process by a meshfree contact algorithm, *20th Int Congress of IUTAM*, Chicago IL, August.
[271] Song N, Qian D, Cao J, Liu WK, and Li S (2001), Effective model for prediction of springback in flanging, *ASME J. Eng. Mater. Technol.* **123**, 456–461.
[272] Jun S and Im S (2000), Multiple-scale meshfree adaptivity for the simulation of adiabatic shear band formation, *Computational Mech.* **25**, 257–266.
[273] Liu WK and Chen Y (1995), Wavelet and multiple scale reproducing kernel method, *Int. J. Numer. Methods Fluids* **21**, 901–933.
[274] Liu WK, Hao W, Chen Y, Jun S, and Gosz J (1997), Multiresolution reproducing kernel particle methods, *Computational Mech.* **20**, 295–309.

[275] Liu WK and Jun S (1998), Multiple scale reproducing kernel particle methods for large deformation problems, *Int. J. Numer. Methods Eng.* **141**, 1339–1362.
[276] Liu WK, Jun S, Sihling DT, Chen Y, and Hao W (1997), Multiresolution reproducing kernel particle method for computational fluid dynamics, *Int. J. Numer. Methods Fluids* **24**, 1391–1415.
[277] Liu WK, Hao S, Belytschko T, Li S, and Chang T (2000), Multi-scale methods. *Int. J. Numer. Methods Eng.* **47**, 1343–1361.
[278] Wagner GJ and Liu WK (2000), Turbulence simulation and multiple scale subgrid models, *Computational Mech.* **25**, 117–136.
[279] Liu WK, Hao S, Belytschko T, Li S, and Chang CT (1999), Multiple scale meshfree methods for damage fracture and localization, *Comput. Mater. Sci.* **16**, 197–205.
[280] Hao S, Liu WK, and Chang CT (2000), Computer implementation of damage models by finite element and mesh-free methods, *Comput. Methods Appl. Mech. Eng.* **187**, 401–440.
[281] Lee SH, Kim HJ, and Jun S (2000), Two scale meshfree method for the adaptivity of 3-d stress concentration problems, *Computational Mech.* **26**, 376–387.
[282] Saigal S and Barry W (2000), A slice based element free Galerkin formulation, *Computational Mech.* **25**, 220–229.
[283] Zhang X, Lu M, and Wagner JL (2000), A 2-D meshless model for jointed rock structures, *Int. J. Numer. Methods Eng.* **47**, 1649–1661.
[284] Danielson KT and Adley MD (2000), A meshless treatment of three-dimensional penetrator targets for parallel computation, *Computational Mech.* **25**, 267–273.
[285] Danielson KT, Hao S, Liu WK, Aziz R, and Li S (2000), Parallel computation of meshless methods for explicit dynamic analysis, *Inter. J. Numer. Methods* **47**, 1323–1341.
[286] Zhang LT, Wagner GJ, and Liu WK (2000), A parallelized meshfree method with boundary enrichment for large-scale CFD, *J. Comput. Phys.* (submitted).
[287] Chen JS and Wang HP (2000), Meshfree smooth surface contact algorithm for sheet metal forming, In: *SAE 2000 World Congress*, Paper No. 2000-01-1103, SAE International, March.
[288] Hao S, Park HS, and Liu WK (2001), Moving particle finite element method, Submitted to Int. J. Numer. Methods Eng.
[289] Ohno K, Esfarjani K, and Kawazoe Y (1999), *Computational Materials Science: from Ab initio to Monte Carlo Methods*, Springer, Berlin.
[290] Landau LD and Lifshitz EM (1965), *Quantum Mechanics: Non-relativistic theory*, Pergmon, Oxford, London.
[291] Feynman RD and Hibbs AR (1965), *Quantum mechanics and path integrals*, McGraw-Hill, New York.
[292] Dirac PAM (1958), *The Principles of Quantum Mechanics*, Oxford Univ Press, London.
[293] Hartree DR (1928), The wave mechanics of an atom with a non-Coulomb central field, Part I, Theory and methods, *Proc. Cambridge Philos. Soc.* **24**, 89.
[294] Hartree DR (1932), A practical method for the numerical solution of differential equations, *Mem and Proc of Manchester Literary and Phil. Soc.* **77**, 91–106.
[295] Fock V (1930), Näherungsmethode zur lösung des quantenmechanischen mehrkörperproblems, *Z. Phys.* **61**, 126.
[296] Takashima H, Kitamura K, Tanabe K, and Nagashima U (2000), Is large-scale Ab initio Hartree-Fock calculation chemically accurate? towards improved calculation of biological molecule properties, *J. Comput. Chem.* **20**, 443–454.
[297] Tu Y and Laaksonen A (2000), Combined Hartree-Fock quantum mechanical and molecular mechanical dynamics simulations of water at ambient and supercritical conditions, *J. Chem. Phys.* **133**, 11264–11269.
[298] Li X, Millam JM, and Sohlegel HB (2000), Ab initio molecular dynamics studies of the photodissociation of formaldehyde, H_2CO-H_2+CO: Direct classical trajectory calculations by MP2 and density function theory, *J. Chem. Phys.* **113**, 10062–10067.
[299] Starikov EB (2000), Nucleic acids as objects of material science: importance of quantum chemical and quantum mechanical studies, *Int. J. Quantum Chem.* **77**, 859–870.
[300] Clementi E (2000), *Ab initio* computations in atoms and molecules, *IBM J. Res. Dev.* **44**, 228–245.
[301] Kohn W and Sham LJ (1965), Self-consistent equations including exchange and correlation effects, *Phys. Rev.* **140**, 1133.
[302] Hohenberg P and Kohn W (1964), Inhomogeneous electron gas, *Phys. Rev.* **136**, B864.
[303] Harris J (1985), Simplified method for calculating the energy of weakly interacting fragments, *Phys. Rev. B* **31**, 1770–1779.
[304] Born M and Oppenheimer JR (1927), Zur quantentheorie, *Ann. Phys. (Leipzig)* **84**, 457.
[305] Car R and Parrinello M (1985), Unified approach for molecular dynamics and density-functional theory, *Phys. Rev. Lett.* **55**, 2471–2474.
[306] Ryckaert JP, Ciccotti G, and Berendsen HJC (1977), Numerical integration of the Cartesian equations of motion of a system with constraints: Molecular dynamics of n-alkanes, *J. Comput. Phys.* **23**, 327–341.
[307] Verlet L (1967), Computer experiments on classical fluids I: Thermodynamical properties of Lennard-Jones molecules, *Phys. Rev.* **159**, 98.
[308] Ishikawa Y, Binning Jr RC, and Shramek NS (1999), Direct ab initio molecular dynamics study of $NO_2^+ + (H_2O)_4$ to $HNO_3(H_7O_3)^+$, *Chem. Phys. Lett.* **313**, 341–350.
[309] Belosludov RV, Sluiter M, Li ZQ, and Kawazoe Y (1999), *Ab initio* and lattice dynamics studies of the vibrational and geometrical properties of the molecular complex of hydroquinone and C_{60}, *Chem. Phys. Lett.* **312**, 299–305.
[310] Jones JE (1924), On the determination of molecular fields I: From the variation of the viscosity of a gas with temperature, *Proc of Royal Society (London)* **106A**, 441–462.
[311] Jones JE (1924), On the determination of molecular fields II: From the equation of state of a gas, *Proc of Royal Society (London)* **106A**, 463.
[312] Falk ML and Langer JS (1998), Dynamics of viscoplastic deformation in amorphous solids, *Phys. Rev. E* **57**, 7192–7205.
[313] Falk ML (1999), Molecular-dynamics study of ductile and brittle fracture in model noncrystalline solids, *Phys. Rev. B* **60**, 7062–7070.
[314] Daw MS and Baskes MI (1984), Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in solids, *Phys. Rev. B* **29**, 6443–6453.
[315] Schuller IK (1988), Molecular dynamics simulation of epitaxial growth, *MRS Bull.* **13**, 23–27.
[316] Baskes M, Daw M, Dodson B, and Foiles S (1988), Atomic-scale simulation in materials science, *MRS Bull.* **13**, 28–35.
[317] Slater JC and Koster GF (1954), Simplified LCAO method for the periodic potential problem, *Phys. Rev.* **94**, 1498.
[318] Anderson PW (1968), Self-consistent pseudo-potentials and ultralocalized functions for energy bands. Phys. Rev. Lett. **21:13**.
[319] Anderson PW (1969), Localized orbitals for molecular quantum theory I: The hückel theory. *Phys. Rev.* **181**, 25.
[320] Qian D, Liu WK, and Ruoff RS (2001), Mechanics of nanotube filled with fullerenes, *J. Phys. Chem. B* **105**, 10753–10758.
[321] Tadmor EB, Ortiz M, and Phillips R (1996), Quasicontinuum analysis of defects in solids. *Philos. Mag. A* **73**, 1529–1563.
[322] Milstein F (1982), Crystal elasticity, In: *Mechanics of Solids*, Pergamon, Oxford, 417–452.
[323] Tersoff J (1988), Empirical interatomic potential for carbon, with application to amorphous carbon, *Phys. Rev. Lett.* **61**, 2879–2882.
[324] Brenner DW (1990), Empirical potential for hydrocarbons for use in simulating chemical vapor deposition of diamond films, *Phys. Rev. B* **42**, 9458–9471.
[325] Bulatov V, Abraham FF, Kubin L, Devincre B, and Yip S (1998), Connecting atomistic and mesoscale simulations of crystal plasticity, *Nature (London)* **391**, 669–672.
[326] Clementi E (1988), Global scientific and engineering simulations on scalar, vector and parallel LCAP-type supercomputer, *Philos. Trans. R. Soc. London, Ser. A* **326**, 445–470.
[327] Clementi E, Chin S, Corongiu G, Detrich JH, Dupuis M, Folsom D, Lie GC, Logan D, and Sonnad V (1989), Supercomputing and supercomputers for science and engineering in general and for chemistry and biosciences in particular, *Int. J. Quantum Chem.* **35**, 3–89.
[328] Given JA and Clementi E (1989), Molecular dynamics and Rayleigh-Benard convection, *J. Chem. Phys.* **90**, 7376–7383.
[329] Hermansson K, Lei GC, and Clementi E (1988), An ab initio pair potential for the interaction between a water molecular and a formate ion, *Theor. Chim. Acta* **74**, 1–10.
[330] Abraham FF (1996), Dynamics of brittle fracture with variable elasticity, *Phys. Rev. Lett.* **77**, 869–872.
[331] Abraham FF (1997), Portrait of a crack: rapid fracture mechanics using parallel molecular dynamics, *IEEE Comput. Sci. Eng.* **4**, 66–77.
[332] Abraham FF, Brodbeck D, Rudge WE, and Xu X (1997), A molecular dynamics investigation of rapid fracture mechanics, *J. Mech. Phys. Solids* **45**, 1595–1619.
[333] Abraham FF, Brodbeck D, Rudge WE, and Xu X (1997), Instability dynamics in three-dimensional fracture: An atomistic simulation, *J. Mech. Phys. Solids* **45**, 1461–71.
[334] Abraham FF, Broughton JQ, and Davidson BN (1997), Large-scale simulation of crack-void and void-void plasticity in metallic fcc crystals under high strain rates, *J. Comput.-Aided Mater. Des.* **5**, 73–80.
[335] Abraham FF (1997), On the transition from brittle to plastic failure in

breaking a nanocrystal under tension (NUT), *Europhys. Lett.* **38**, 103–106.
[336] Abraham FF and Broughton JQ (1997), Large-scale simulations of brittle and ductile failure in fcc crystals, *Comput. Mater. Sci.* **10**, 1–9.
[337] Abraham FF and Gao H (1998), Anomalous ductile-brittle fracture behavior in fcc crystals, *Philos. Mag. Lett.* **78**, 307–312.
[338] Abraham FF, Brodbeck D, Rudge WE, Broughton JQ, Schneider D, Land B, Lifka D, Gerber J, Rosenkrantz M, Skovira J, and Gao H (1998), *Ab initio* dynamics of rapid fracture, *Modell. Simul. Mater. Sci. Eng.* **6**, 639–670.
[339] Gumbsch P and Cannon RM (2000), Atomistic aspects of brittle fracture, *MRS Bull.* **25**, 15–20.
[340] Gumbsch P and Gao H (2000), Driving force and nucleation of supersonic dislocations, *J. Computer-Aided Mat. Des.* **6**, 137–144.
[341] Trebin HR, Mikulla R, Stadler J, Schaaf G, and Gumbsch P (1999), Molecular dynamics simulations of crack propagation in quasicrystals, *Comput. Phys. Commun.* **121–122**, 536–539.
[342] Hartmaier A and Gumbsch P (2000), The brittle-to-ductile transition and dislocation activity at crack tips, *J. Comput.-Aided Mater. Des.* **6**, 145–155.
[343] Perez R and Gumbsch P (2000), An ab initio study of the cleavage anisotropy in silicon, *Acta Mater.* **48**, 4517–4530.
[344] Farkas D (2000), Atomistic theory and computer simulation of grain boundary structure and diffusion, *J. Phys.: Condens. Matter* **12**, R497–516.
[345] Farkas D (2000), Atomistic studies of intrinsic crack-tip plasticity, *MRS Bull.* **25**, 35–38.
[346] Farkas D (2000), Bulk and intergranular fracture behavior of NiAL. *Philos. Mag. A* **80**, 1425–1444.
[347] Farkas D (2000), Mechanisms of intergranular fracture. In: Fracture and Ductile vs Brittle Behavior, GE Beltz, RLB Selinger, K-S Kim, and MP Marder (eds), Mat Res Soc, Warrendale PA, 291–298.
[348] Mishin Y, Farkas D, Mehl MJ, and Papaconstantopoulos DA (1999), Interatomic potentials for Al and Ni from experimental data and ab initio calculations, in: *Multiscale Modeling of Materials*, VV Bulatov, TD de la Rubia, R Phillips, E Kaziras, and N Ghoniem (eds), Mat Res Soc, Warrendale PA, 535–540.
[349] Langer JS (2000), Numerical and analytic routes from microscale to macroscales in theories of deformation and fracture, *J. Comput.-Aided Mater. Des.* **1999**, 89–94.
[350] Monaghan JJ (1994), Vortex particle methods for periodic channel flow, *J. Comput. Phys.* **107**, 152–159.
[351] Beale JT (1986), A convergent 3-D vortex method with grid-free stretching, *Math. Comput.* **46**, 401–424.
[352] Winckelmans GS and Leonard A (1993), Contributions to vortex particle methods for the computation of three-dimensional incompressible unsteady flows, *J. Comput. Phys.* **109**, 247–273.
[353] Fishelov D (1990), A new vortex scheme for viscous flows, *J. Comput. Phys.* **86**, 211–224.
[354] Cotte GH, Koumoutsakos P, and Salihi MLO (2000), Vortex methods with spatially varying cores, *J. Comput. Phys.* **162**, 164–185.
[355] Lin H and Vezza M, (1996), A pure vortex sheet method for simulating unsteady, incompressible, separated flows around static and pitching aerofoils, In: Proc of 20th Congress of Int Council of the Aeronautical Sciences, Sorento, Italy, 2184–2193.
[356] Brackbill JU, Kothe DB, and Ruppel HM (1988), FLIP: A low-dissipation, particle-in-cell method for fluid flow, *Comput. Phys. Commun.* **48**, 25–38.
[357] Brackbill JU (1988), The ringing instability in particle-in-cell calculation of low speed flow, *J. Comput. Phys.* **75**, 469.
[358] Burgess D, Sulsky D, and Brackbill JU (1992), Mass matrix formulation of the FLIP particle-in-cell method, *J. Comput. Phys.* **103**, 1–15.
[359] Sulsky D, Zhou SJ, and Schreyer HL (1995), Application of a particle-in-cell method to solid mechanics, *Comput. Phys. Commun.* **87**, 236–252.
[360] Brackbill JU (1991), FLIP-MHD: A particle-in-cell method of magnetohydrodynamics, *J. Comput. Phys.* **96**, 163–192.
[361] Hockney R and Eastwood J (1988), Computer Simulation Using Particles, Adam Hilger, Bristol.
[362] Succi S (1997), Lattice Boltzmann equation: Failure or success? *Physica A* **240**, 221–228.
[363] Filippova O and Hänel D (2000), A novel lattice bgk approach for low mach number combustion, *J. Comput. Phys.* **158**, 139–160.
[364] He X, Chen S, and Zhang R (2000), A lattice Boltzmann scheme for incompressible multiphase flow and its application in simulation of Rayleigh-Taylor instability, *J. Comput. Phys.* **152**, 642–663.
[365] Mazzocco F and Arrighetti C (2000), Multiscale lattice Boltzmann schemes: A preliminary application to axial turbomachine flow simulations, *Int. J. Mod. Phys.* **11**, 233–245.
[366] van der Sman RGM (1997), Lattice Boltzmann scheme for natural convection in porous media, *Int. J. Mod. Phys.* **8**, 879–888.
[367] Maier RS (1986), Boundary conditions for the lattice Boltzmann method, *Phys. Fluids* **8**, 1788–1801.
[368] McNamara G and Zanetti G (1988), Use of the Boltzmann equation to simulate lattice-gas automata, *Phys. Rev. Lett.* **61**, 2332.
[369] Ziegler DP (1993), Boundary conditions for lattice Boltzmann simulations, *J. Stat. Phys.* **71**, 1171.
[370] Benzi R, Succi S, and Vergassola M (1992), The lattice Boltzmann equation: Theory and applications, *Phys. Rep.* **222**, 145.
[371] Reider MB and Sterling JD (1995), Accuracy of discrete-velocity BGK models for the simulation of the impressible Navier-Stokes equations, *Comput. Fluids* **24**, 459–467.
[372] Karlin IV, Succi S, and Orszag S (1999), Lattice Boltzmann method for irregular grids, *Phys. Rev. Lett.* **26**, 5245–5248.
[373] van der Sman RGM and Ernst MH (2000), Convection-diffusion lattice Boltzmann scheme for irregular lattices, *J. Comput. Phys.* **160**, 766–782.
[374] Mei R, Shyy W, Yu D, and Luo LS (2000), Lattice Boltzmann method for 3-d flows with curved boundary, *J. Comput. Phys.* **161**, 680–699.
[375] Frisch U, d'Humiéres D, Lallemand P, Pomeau Y, and Rivet JP (1987), Lattice gas hydrodynamics in two and three dimensions, *Complex Syst.* **1**, 649–707.
[376] Bhatnagar P, Gross EP, and Krook MK (1954), A model for collision processes in gases I: Small amplitude processes in charged and neutral one-component systems, *Phys. Rev.* **94**, 511.
[377] Braun J and Sambridge MA (1995), A numerical method for solving partial differential equations on highly irregular evolving grids, *Nature (London)* **376**, 655–660.
[378] Braun J, Sambridge MA, and McQueen H (1995), Geophysical parametrization and interpolation of irregular data using natural neighbors, *Geophys. J. Int.* **122**, 837–857.
[379] Traversoni L (1994), Natural neighbor finite elements, In: *Int Conf on Hydraulic Engineering Software Hydrosoft Proc*, Vol 2, Computational Mechanics Publ 291–297.
[380] Sibson R (1980), A vector identity for the Dirichlet tessellation, *Math. Proc. Cambridge Philos. Soc.* **87**, 151–155.
[381] Sibson R (1981), A brief description of natural neighbor interpolation. In: *Interpreting Multivariate Data*, V Barnett (ed) Wiley, Chichester, 21–36.
[382] Sukumar N, Moran B, and Belytschko T (1998), The natural element method in solid mechanics, *Int. J. Numer. Methods Eng.* **43**, 839–887.
[383] Sukumar N and Moran B (1999), c^1 natural neighbor interpolant for partial differential equations, *Numer. Methods for Partial Differential Equations* **15**, 417–447.
[384] Sukumar N, Moran B, Semenov AY, and Belikov VV (2001), Natural neighbor Galerkin methods, *Int. J. Numer. Methods Eng.* **50**, 1–27.
[385] Bueche D, Sukumar N, and Moran B (2000), Dispersive properties of the natural element method, *Computational Mech., Berlin* **25**, 207–219.
[386] Cueto E, Doblaré M, and Gracia L (2000), Imposing essential boundary conditions in natural element method by means of density-scaled α-shapes, *Int. J. Numer. Methods Eng.* **48**, 519–546.
[387] Belikov VV, Ivanov VD, Kontorivich VK, Korytnik SA, and Semenov AY (1997), The non-Sibsonian interpolation: A new method of interpolation of the values of a function on an arbitrary set of points, *Computational Math. and Mathematical Phys.* **37**, 9–15.
[388] Oñate E, Idelsohn S, Zienkiewicz OC, and Taylor RL (1996), A stabilized finite point method for analysis of fluid mechanics problems, *Comput. Methods Appl. Mech. Eng.* **139**, 315–347.
[389] Oñate E, Idelsohn S, Zienkiewicz OC, and Taylor RL (1996), A finite point method in computational mechanics: Application to convective transport and fluid flow, *Int. J. Numer. Methods Eng.* **39**, 3839–3866.
[390] Oñate E and Idelsohn S (1998), A mesh-free point method for advective-diffusive transport and fluid flow problems, *Computational Mech., Berlin* **21**, 283–292.
[391] Taylor RL, Zienkiewicz OC, and Onate E (1997), A hierarchical finite element method based on the partition of unity, *Comput. Methods Appl. Mech. Eng.* **152**, 73–84.
[392] Pardo E (2000), Meshless method for linear elastostatics based on a path integral formulation, *Int. J. Numer. Methods Eng.* **47**, 1463–1480.
[393] Needleman A (1990), An analysis of tensile decohesion along an interface, *J. Mech. Phys. Solids* **38**, 289–324.
[394] Xu XP and Needleman A (1994), Numerical simulations of fast crack

growth in brittle solids, *J. Mech. Phys. Solids* **42**, 1397–1434.
[395] Camacho GT and Ortiz M (1997), Adaptive Lagrangian modeling of ballistic penetration of metallic targets, *Comput. Methods Appl. Mech. Eng.* **142**, 269–301.
[396] Ortiz M and Pandolfi A (1999), Finite-deformation irreversible cohesive elements for three-dimensional crack-propagation analysis, *Int. J. Numer. Methods Eng.* **44**, 1267–1282.
[397] Gao H and Klein P (1998), Numerical simulation of crack growth in an isotropic solid with randomized internal cohesive bonds, *J. Mech. Phys. Solids* **46**, 187–218.

Shaofan Li *received his BS from the East China University of Science and Technology, China (1982); an MS from Huazhong University of Science and Technology, China (1989) and an MS from University of Florida (1993); and his PhD from Northwestern University (1997). After spending three years as a post-doctoral fellow at Northwestern University, he became an Assistant Professor at University of California, Berkeley in 2000. His research interests include finite element methods, meshfree particle methods, micromechanics, structural dynamics, and dynamic fracture/shear band propagation in solids. He is a member of the United States Association for Computational Mechanics.*

Wing Kam Liu *received his BS with the highest honor from the University of Illinois at Chicago (1976); his MS (1977) and PhD (1981) from Caltech. In September 1980, he became an Assistant Professor at Northwestern University. In 1988, he became a Professor of Mechanical and Civil Engineering. His research activities include development of finite element methods, reproducing kernel and wavelet methods and deterministic and probabilistic computational methods, design and manufacturing and nano-mechanics. He is the recipient of the 2001 Computational Structural Award from USACM, 1995 Gustus L Larson Memorial Award from ASME, 1989 Thomas J Jaeger Prize, IASMiRT, 1985 Pi Tau Sigma Gold Medal from ASME, the 1983 Ralph R Teetor Educational Award from ASAE and the 1979 Melville Medal from ASME. He became elected a Fellow of ASME in 1990, ASCE in 1993, and USACM in 1995. Currently, Liu serves as an Associate Editor for the ASME Journal of Pressure Vessel Technology, and the ASME Journal of Applied Mechanics, Managing Editor of Computers and Structures, and Computational Mechanics, and Advisory Editor of Computer Methods in Applied Mechanics and Engineering. He is the current President of the United States Association for Computational Mechanics. In 2001, Wing Kam Liu was listed as one of the 93 most highly cited engineers in the world by the Institute for Scientific Information, (ISI) 93 Most Highly Cited Researchers in Engineering.*