INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2004; **59**:1007–1020 (DOI: 10.1002/nme.999)

Moving particle finite element method with global smoothness

Su Hao, Wing Kam Liu*,† and Ted Belytschko

Department of Mechanical Engineering, Northwestern University, Evanston, IL 60208, U.S.A.

SUMMARY

We describe a new version of the moving particle finite element method (MPFEM) that provides solutions within a C^0 finite element framework. The finite elements determine the weighting for the moving partition of unity. A concept of 'General Shape Function' is proposed which extends regular finite element shape functions to a larger domain. These are combined with Shepard functions to obtain a smooth approximation. The Moving Particle Finite Element Method combines desirable features of finite element and meshfree methods. The proposed approach, in fact, can be interpreted as a 'moving partition of unity finite element method' or 'moving kernel finite element method'. This method possesses the robustness and efficiency of the C^0 finite element method while providing at least C^1 continuity. Copyright © 2004 John Wiley & Sons, Ltd.

KEY WORDS: moving particle finite element; meshfree; reproducing condition; partition of unity; finite element

1. INTRODUCTION

The finite element method (FEM) is the most popular and successful computational method [1–4] in engineering analysis. The conventional finite element method approximates the solution with a piecewise continuous function so that discontinuities of the gradient occur on the element boundaries. These discontinuities reduce the accuracy of the gradient of the approximated solution. Thus, the finite element method may require a very fine mesh in problems with high gradients, or with distinct localized features, which can be computationally expensive. Also under large deformations, mesh distortion may either abort the calculation or result in a dramatic deterioration of accuracy. These issues hamper the application of FEM for some engineering problems.

Meshless, or meshfree, methods such as the Element Free Galerkin method (EFG) [5, 6], reproducing kernel particle method (RKPM) [7], *h-p* clouds [8, 9], and the stress-point method

*Correspondence to: Wing Kam Liu, Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208, U.S.A.
†E-mail: w-liu@northwestern.edu

Contract/grant sponsor: National Science Foundation

Received 30 December 2002
Revised 8 May 2003
Accepted 12 September 2003

[10], are a new class of smooth particle methods (SPH) [11] that have been developed in the last decade. A distinguishing feature of these new meshfree methods, as compared to SPH, is their ability to satisfy the reproducing conditions [12–14], see the reviews [6, 15, 16]. Meshless approximations can also be constructed by the moving least-square approximation in a finite element [17, 18]. All of these methods demonstrate robustness in handling large deformation problems with good accuracy when the domain of influence covers enough nodes. However, meshfree methods also demonstrate several disadvantages. For example, most meshfree approximations are not interpolants, which necessitates special treatment to handle essential boundary conditions; e.g. see References [15, 19]. Furthermore, nodal-based meshfree algorithms may have spurious modes [20], whereas a background mesh is needed for Gauss quadrature; in any case, the Galerkin weak form is difficult to integrate accurately, especially near a boundary [21]. Focusing on these challenges for both meshfree methods and FEM, in the past decade continuous endeavors have been made in developing innovative ideas and approaches; e.g. References [9, 20–33].

In this paper we propose a method that combines finite element approximations with meshfree kernel weights. The method has the following properties:

1. it provides continuity of both the approximated solution and its gradient
2. it is an interpolant, i.e. it satisfies so-called Kronecker-delta property, so no special treatment for essential boundary conditions is needed
3. only a compact domain of influence is required
4. it can be integrated easily in Galerkin formulations and
5. it is robust for large deformation simulations.

This paper is organized as follows: Section 2 introduces the main concept and the structure of the interpolation function. Numerical examples are presented in Section 3. Discussions of extensions are given in Section 4. Section 5 contains a summary and conclusions.

The following notation is used throughout. Boldface symbols denote a tensor, the order of which is indicated by the context. Plain symbols denote scalars or a component of a tensor when a subscript is attached; repeated lower case indices indicate summations.

2. PROPOSED APPROACH

2.1. General shape function

Consider a domain Ω with boundary $\partial\Omega$ and let the Ω be subdivided into n_e elements so that $\Omega = \bigcup_{e=1}^{n_e} \Omega^e$. Each element has NEN nodes, but the method is also applicable to meshes with elements that have different numbers of nodes.

We first introduce the concept of general shape function: *the general shape function $\hat{N}_J^e(\mathbf{x})$ is the finite element shape function $N_J^e(\mathbf{x})$ extended over the entire domain.*

The above definition can be made more precise as follows.

Let S^e be the set of all nodes of element e and $P_J^e(\mathbf{x})$ be a set of polynomial interpolants associated with the element e, so that

$$P_J^e(\mathbf{x}_I^e) = \delta_{IJ} \quad \forall I, J \in S^e \tag{1}$$

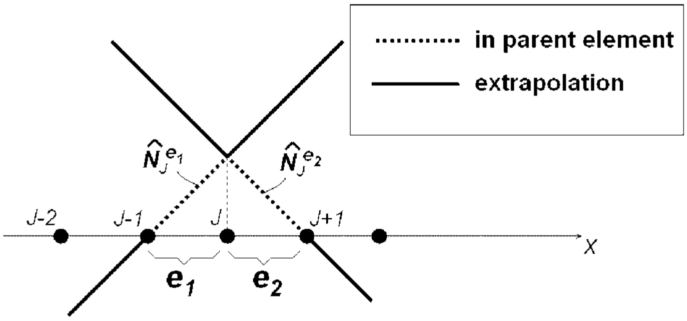

Figure 1. The general shape functions associated with node J for a one-dimensional linear finite element.

where $\mathbf{x}_I^e$ are the nodes of element e. The general shape functions for element e are then

$$\hat{N}_J^e(\mathbf{x}) = P_J^e(\mathbf{x}) \quad \forall \mathbf{x} \in \Omega \tag{2}$$

We call element e the 'parent element' of the general shape function $\hat{N}_J^e(\mathbf{x})$.

To distinguish the above from standard finite element shape functions $N_J^e(\mathbf{x})$, note that standard finite element shape functions are defined by

$$N_J^e(\mathbf{x}) = \begin{cases} P_J^e(\mathbf{x}) & \forall \mathbf{x} \in \Omega^{\mathrm{e}} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Figure 1 shows the one-dimensional linear general shape functions of node J for the standard 2-node element. The nodes are arranged in order of node number, but this is not necessary. As can be seen, the general shape functions of node J of elements e_1 and e_2 within their parent elements are identical to the standard shape functions, but they are extended beyond the elements.

Thus, while the general shape functions of element e are

$$\hat{N}_1^e(x) = \frac{x - x_1^e}{x_2^e - x_1^e} \quad \forall x, \quad \hat{N}_2^e(x) = \frac{x_2^e - x}{x_2^e - x_1^e} \quad \forall x$$

the standard shape functions are defined as

$$N_1^e(x) = \begin{cases} \dfrac{x - x_1^e}{x_2^e - x_1^e} & \text{for } x_1^e \leqslant x \leqslant x_2^e, \\ 0 & \text{otherwise.} \end{cases} \qquad N_2^e(x) = \begin{cases} \dfrac{x_2^e - x}{x_2^e - x_1^e} & \text{for } x_1^e \leqslant x \leqslant x_2^e \\ 0 & \text{otherwise} \end{cases}$$

The terminology 'general shape function' is borrowed from spectrum analysis in material science [34] and psychobiology [35]. We introduce this concept because, as will be seen, it enables the construction of a smooth interpolation similar to finite elements.

2.2. MPFEM approximation

Consider a kernel $\phi(\mathbf{x})$ defined on a compact support Ω_x:

$$\phi(\mathbf{x}) = \begin{cases} \phi(\mathbf{x}) & \forall \mathbf{x} \in \Omega_x \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

We impose the restriction that the compact support can span at most the domain generated by the union of all elements that contain any node in S^e, i.e. when $\mathbf{x} \in \Omega^e$, then

$$\Omega_x \subset \bigcup_{f \in N} \Omega_f \tag{5}$$

where N is the set of all elements that contain any node in S^e. This kernel $\phi(\mathbf{x})$ is normalized so that

$$\int_{\bar{\Omega}_x} \phi(\tilde{\mathbf{x}} - \mathbf{x})\, \mathrm{d}\tilde{\mathbf{x}} = 1 \quad \text{where } \bar{\Omega}_x = \Omega_x \cap \Omega \tag{6}$$

It can be seen that the above corresponds to continuous forms of the Shepard function and meets the reproducing condition for a constant (a partition of unity). No other reproducing condition need be met by the kernel.

Let E^x be the set of all elements e such that

$$\Omega_x^e = \Omega^e \cap \bar{\Omega}_x \neq 0 \tag{6a}$$

The MPFEM approximation is then given by

$$\mathbf{u}^h(\mathbf{x}) = \sum_{e=1}^{n_e} \int_{\Omega_x^e} \phi(\tilde{\mathbf{x}} - \mathbf{x})\, \mathrm{d}\tilde{\mathbf{x}} \sum_{I \in S^e} \hat{N}_I^e(\mathbf{x}) \mathbf{u}_I \quad \forall \mathbf{x} \in \Omega \tag{7}$$

where $\mathbf{u}_I$ is the nodal parameter of $\mathbf{u}$ at node I. Equation (7) can be written also as

$$\mathbf{u}^h(\mathbf{x}) = \sum_{e \in E^x} \int_{\Omega_x^e} \phi(\tilde{\mathbf{x}} - \mathbf{x})\, \mathrm{d}\tilde{\mathbf{x}} \sum_{I \in S^e} \hat{N}_I^e(\mathbf{x}) \mathbf{u}_I \quad \forall \mathbf{x} \in \Omega \tag{8}$$

The second form (8) is the one used in computations since the sum is over smaller number of elements. Note that the kernel $\phi(\mathbf{x})$ is only integrated over the intersection of the element domain with the support of $\phi(\mathbf{x})$.

For the one-dimensional case, (8) becomes

$$u^h(x) = \sum_{e \in E^x} \int_{\Omega_x^e} \phi(\tilde{x} - x)\, \mathrm{d}\tilde{x} \sum_{I \in S^e} \hat{N}_I^e(x) u_I \quad \forall x \in \Omega \tag{9}$$

We note that (8) can also be written as

$$\mathbf{u}^h(\mathbf{x}) = \sum_{e \in E^x} \omega_e(\mathbf{x}) \sum_{I \in S^e} \overset{\hat{}}{N}{}_I^e(\mathbf{x}) \mathbf{u}_I \quad \forall \mathbf{x} \in \Omega \tag{10}$$

where

$$\omega_e(\mathbf{x}) = \int_{\Omega_x^e} \phi(\tilde{\mathbf{x}} - \mathbf{x})\, \mathrm{d}\tilde{\mathbf{x}} \tag{11}$$

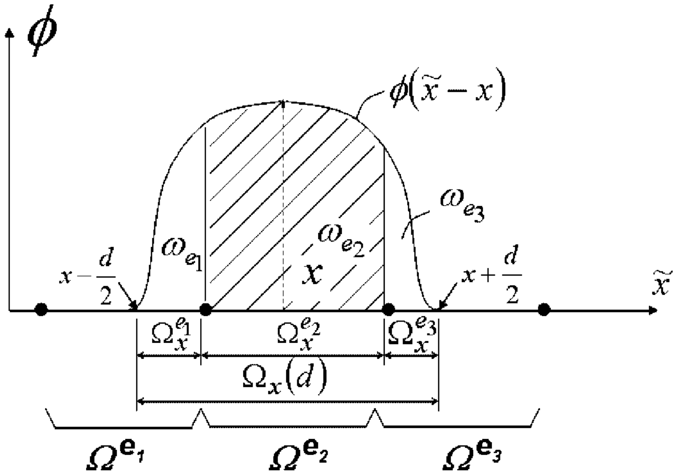

Figure 2. The weights ω_{e_i} in (9)–(12) are determined by the integrals over the intersection of the element domains and the support Ω_x of the kernel $\phi(\tilde{x}-x)$ that is centred at $x=\tilde{x}$.

From the definition of $\omega_e(\mathbf{x})$ given in (11) and the normality condition (6), it follows that

$$\sum_{e=1}^{n_e} \omega_e(\mathbf{x}) = \sum_{e\in E^x} \omega_e(\mathbf{x}) = 1 \quad \forall \mathbf{x}\in\Omega \tag{12}$$

We also require that

$$\omega_e(\mathbf{x}_J) = 0 \quad \text{if } J\notin S^e \tag{13}$$

This corresponds to the previously stated requirement that the support of $\phi(\mathbf{x})$ be small enough so that for any point $\mathbf{x}\in\Omega^e$, $\phi(\mathbf{x}-\mathbf{x}_J)=0$ if node $J\notin S^e$.

The kernel function is shown at a typical point in Figure 2. It can be seen that in one dimension, as many as three values of $\omega_e(x)$ can be non-zero at a point (in this case, $\omega_{e_1}(x)$, $\omega_{e_2}(x)$, and $\omega_{e_3}(x)$). The functions $\omega_e(x)$ serve as weights on the general shape functions. This is illustrated in Figure 3, which shows the approximation over a segment consisting of three elements. In this figure, we have used the definition

$$\hat{u}^e(x) = \sum_{I\in S^e} \hat{N}_I^e(\mathbf{x}) u_I$$

It can be seen that the approximation is smooth and is an interpolant (i.e. it satisfies the Kronecker-delta condition), which will be demonstrated later.

2.3. Kronecker-delta property

We show here that the approximant (8) is an interpolant. If we evaluate (10) at a node P, i.e. at $\mathbf{x}=\mathbf{x}_P$, we obtain:

$$\mathbf{u}^h(\mathbf{x}_P) = \sum_{e\in E^x} \omega_e(\mathbf{x}_P) \sum_{I\in S^e} \hat{N}_I^e(\mathbf{x}_P)\mathbf{u}_I \tag{14}$$

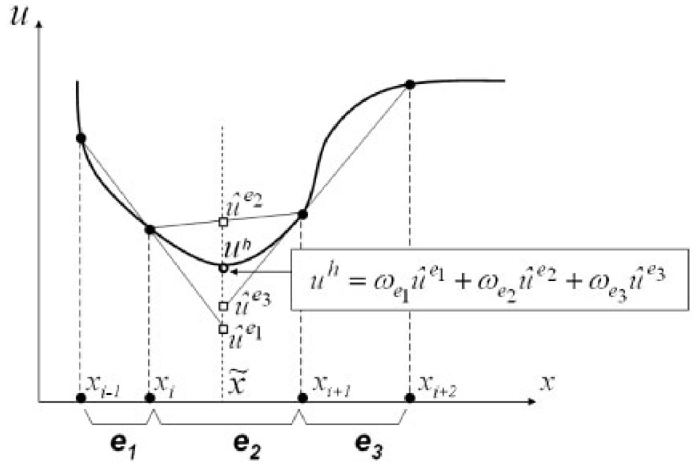

Figure 3. An approximation by MPFEM, it is a weighted sum of the general interpolants $\hat{u}^{e_i}$.

From requirement (13) it follows that the right hand side of (14) vanishes for any node $P \notin S^e$. For any other node P of element e, i.e. for any $P \in S^e$, from (1) and (2) we have

$$\hat{N}_I^e(\mathbf{x}_J) = \delta_{IJ} \tag{15}$$

So (14) becomes

$$\mathbf{u}^h(\mathbf{x}_P) = \sum_{e\in E^x} \omega_e(\mathbf{x}_P) \sum_{I\in S^e} \delta_{IP}\mathbf{u}_I \tag{16}$$

From (12) and (16) we obtain

$$\mathbf{u}^h(\mathbf{x}_P) = \mathbf{u}_P$$

which demonstrates that (8) is an interpolant.

2.4. Boundary conditions

When we evaluate (8) or (10) at a boundary node P, i.e. at $\mathbf{x} = \mathbf{x}_P$, $\mathbf{x}_P \in \partial\Omega$, condition (13) requires that the support of $\phi(\mathbf{x})$ not intersect elements that are not connected to node P. Therefore the general shape function extrapolation and its benefit cannot be exploited for natural boundary conditions. Since MPFEM is an interpolant, it will satisfy essential boundary conditions with the same accuracy as finite elements.

2.5. Consistency and reproducing conditions

In this subsection we prove that the definition of the weights $\omega_e(x)$ given in (6) and (11)–(13) ensure that the approximation (8) satisfies the reproducing condition [6, 7].

We assume that the general shape functions satisfy the linear reproducing condition in the entire domain, so

$$\sum_{I\in S^e} \hat{N}_I^e(\mathbf{x}) = 1, \quad \sum_{I\in S^e} \hat{N}_I^e(\mathbf{x}) x_{iI} = x_i, \quad i = 1,2,3 \quad \forall \mathbf{x} \in \Omega \tag{17}$$

where x_{iI} are the nodal values of co-ordinates x_i at node I. Let the nodal values of $u(\mathbf{x})$ be obtained from a linear function of $\mathbf{x}$:

$$u(\mathbf{x}) = a_0 + a_j x_j \tag{18}$$

The nodal values of $u(\mathbf{x})$ are then given by

$$u_I = a_0 + a_j x_{jI} \tag{19}$$

Substituting (19) into (10) yields

$$u^h(\mathbf{x}) = \sum_{e \in E^x} \omega_e(\mathbf{x}) \sum_{I \in S^e} \hat{N}_I^e(\mathbf{x})(a_0 + a_j x_{jI}) \tag{20}$$

From (17) it follows that the above becomes

$$u^h(\mathbf{x}) = \sum_{e \in E^x} \omega_e(\mathbf{x})(a_0 + a_j x_j) \tag{20a}$$

Then using (11), it follows that

$$u^h(\mathbf{x}) = a_0 + a_j x_j \tag{20b}$$

which demonstrates that the reproducing conditions are met. The linear reproducing conditions are necessary for first order consistency.

2.6. *Continuity*

When the general shape functions are polynomial interpolants, then the general shape functions and their derivatives of any order are continuous. The continuity of the approximation (8) then depends strictly on the continuity of the kernel $\phi(\mathbf{x})$. For a C^0 kernel, the approximation is C^1, i.e. continuously differentiable.

The gradient of the approximation (8) is

$$\partial_i \mathbf{u}^h(\mathbf{x}) = \sum_{e \in E^x} \sum_{I \in S^e} \lfloor \hat{N}_I^e(\mathbf{x}) \partial_i \omega_e(\mathbf{x}) + \omega_e(\mathbf{x}) \partial_i \hat{N}_I^e(\mathbf{x}) \rfloor \mathbf{u}_I \tag{21}$$

where $\partial_i = \frac{\partial}{\partial x_i}$.

The derivatives of standard C^0 finite element shape functions have discontinuities at element interfaces. The general shape function eliminates this discontinuity by extending the definition of the shape function to the entire domain. If $\phi(\mathbf{x})$ is a piecewise continuously differentiable function, i.e. a global C^0 function defined by (4), then because of the continuity of the general shape function (see (2)) the RHS of (21) is also a C^0 function.

2.7. *An assumed strain formulation with arbitrary order of continuity*

When the kernel $\phi(\mathbf{x})$ is C^0, according to (11) the secondary derivative of (8) is C^{-1}. As an alternative to (21), we propose an 'assumed strain' [36] approximation of the n-th order derivatives

$$\partial_1^{n_1} \partial_2^{n_2} \partial_3^{n_3} \mathbf{u}^h(\mathbf{x}) = \sum_{e \in E^x} \omega_e(\mathbf{x}) \sum_{I \in S^e} (\partial_1^{n_1} \partial_2^{n_2} \partial_3^{n_3} \hat{N}_I^e(\mathbf{x}))\mathbf{u}_I \quad \text{for } n = 0, 1, \ldots, \tag{22}$$

where $\partial_i^n = \partial^n / \partial x_i^n, n_1, n_2, n_3$ are non-negative integers and $n = n_1 + n_2 + n_3$.

‘Assumed strain’ approximations are commonly used in multi-field methods, such as the Hu-Washizu method or Petrov-Galerkin method as in the assumed strain scheme proposed for EFG in References [37, 38].

It can be shown that when the n-th derivative of the general shape function reproduces any k-th order polynomial, (22) also correctly reproduces the derivative of a k-th order polynomial. For example, from (17), we see that the linear general shape functions satisfy:

$$\sum_{I\in S^e} \partial_i \hat{N}_I^e(\mathbf{x}) = 0, \quad \sum_{I\in S^e} \partial_i \hat{N}_I^e(\mathbf{x}) x_{jI} = \delta_{ij} \tag{23}$$

By substituting (23) into (22) and then applying (12), we can verify that the first derivatives of a linear function are reproduced correctly:

$$\sum_{e\in E^x} \omega_e(\mathbf{x}) \sum_{I\in S^e} \partial_i \hat{N}_I^e(\mathbf{x}) = 0 \tag{24a}$$

and

$$\sum_{e\in E^x} \omega_e(\mathbf{x}) \sum_{I\in S^e} \partial_i \hat{N}_I^e(\mathbf{x}) x_{jI} = \delta_{ij} \tag{24b}$$

For the assumed strain approach, any square integrable C^{-1} function can be used as the kernel $\phi(t)$ if it satisfies the normality condition (6). Note that meeting the reproducing conditions does not suffice to insure satisfaction of the patch test [38].

3. ONE-DIMENSIONAL NUMERICAL EXAMPLE

3.1. MPFEM shape function

An alternative expression for (8) is

$$\mathbf{u}^h(\mathbf{x}) = \sum_{J\in S_U} N_J^{\mathrm{MPF}}(\mathbf{x}) \mathbf{u}_J, \quad S_U \subseteq \bigcup_{e=1}^{n_e} S^e \tag{25}$$

where S_U is the set of nodes that belong to the elements intersected with $\bar{\Omega}_X$, the MPFEM shape function N_J^{MPF} is defined by

$$N_J^{\mathrm{MPF}}(\mathbf{x}) = \sum_{e\in E^x} \omega_e(\mathbf{x}) \sum_{I\in S^e} \delta_{IJ} \hat{N}_I^e(\mathbf{x}) \tag{26}$$

For the one-dimensional example shown in Figure 3, the interpolation at x involves a total 4 nodes, i.e., nodes $i-1$, i, $i+1$, and $i+2$. The corresponding MPFEM shape functions are:

$$\begin{aligned} N_{i-1}^{\mathrm{MPF}}(x) &= \omega_{e_1} \hat{N}_{i-1}^{e_1}(x) \\ N_{i}^{\mathrm{MPF}}(x) &= \omega_{e_1} \hat{N}_{i}^{e_1}(x) + \omega_{e_2} \hat{N}_{i}^{e_2}(x) \\ N_{i+1}^{\mathrm{MPF}}(x) &= \omega_{e_2} \hat{N}_{i+1}^{e_2}(x) + \omega_{e_3} \hat{N}_{i+1}^{e_3}(x) \\ N_{i+2}^{\mathrm{MPF}}(x) &= \omega_{e_3} \hat{N}_{i+2}^{e_3}(x) \end{aligned} \tag{27}$$

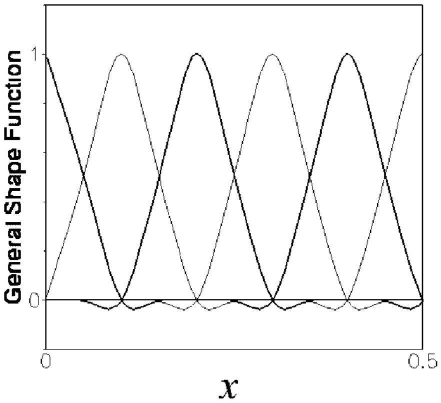

Figure 4. One-dimensional MPFEM shape functions.

For the example shown in Figure 2, let the nodes be uniformly distributed and let the kernel be

$$\phi(t)=\begin{cases} A\cos(\mathrm{t}) & |t|\leqslant \dfrac{d}{2} \\ 0 & |t|>\dfrac{d}{2} \end{cases} \tag{28}$$

The corresponding MPFEM shape functions are plotted in Figure 4. Note that they are quite similar to those in meshfree methods [5, 7] and the natural element method [24].

3.2. A numerical example

Consider the following one-dimensional boundary value problem:

$$\frac{\partial^2 u}{\partial x^2}+g(x)=0 \quad \forall x\in[0,1], \quad u_{,x}n=\bar{t} \quad \text{on } \Gamma_t \quad \text{and} \quad u=\bar{u} \text{ on } \Gamma_u \tag{29}$$

Assume that $g(x)$, the 'body force', has the form

$$g(x)=6x+\left(\frac{2}{\alpha^2}-\left(\frac{2x-2x_0}{\alpha^2}\right)^2\right)\exp\left(-\left(\frac{x-x_0}{\alpha}\right)^2\right) \tag{30}$$

and

$$u(0)=\exp\left(-\frac{x_0^2}{\alpha^2}\right) \tag{31}$$

$$u_{,x}(1)=-3-2\left(\frac{1-x_0}{\alpha^2}\right)\exp\left(-\left(\frac{1-x_0}{\alpha}\right)^2\right) \tag{32}$$

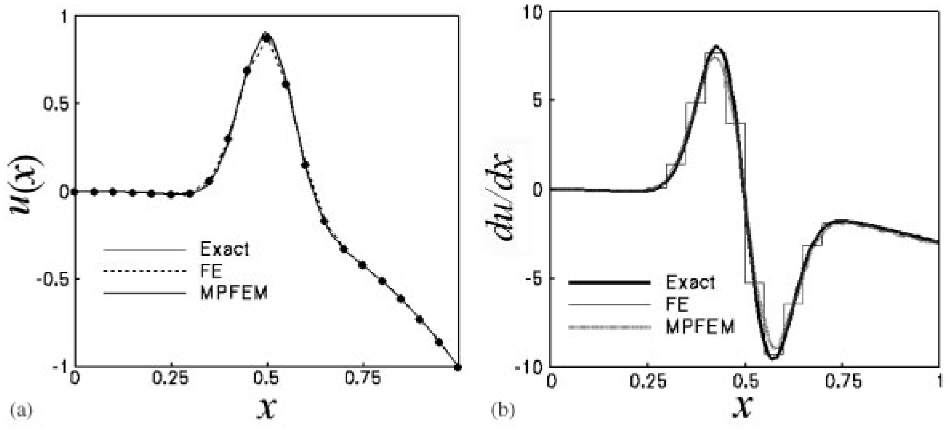

Figure 5. Comparison of displacement solution (a) and its derivative (b) for MPFEM, FEM, with the exact solution (29).

where α is chosen to be small (e.g. $\alpha = 0.01$) and $x_0 \in [0, 1]$. This example has been studied in Reference [21]. The solution to (29) is

$$u(x) = -x^3 + \exp\left(-\left(\frac{x - x_0}{\alpha}\right)^2\right) \tag{33}$$

Numerical studies with the proposed method are reported next. The domain $[0, 1]$ has been partitioned into equal sub-domains separated by particles with spacings from 0.1 (11 particles) to 0.002 (501 particles). Guass quadrature with 1 to 6 points per element are used, but the improvement after 3 quadrature points is small. The L_2 and H_1 norms are defined in [3] by:

$$\|u - u^h\|_{L_2} = \left[\int_0^1 (u - u^h)^2 \,\mathrm{d}x\right]^{1/2} \tag{34}$$

and

$$\|u - u^h\|_{H_1} = \left[\int_0^1 (u_{,x} - u^h_{,x})^2 \,\mathrm{d}x\right]^{1/2} \tag{35}$$

In the above, u denotes the exact solution and u^h is the numerical solution. Plotted in Figure 5 are the solution u and its derivative computed by the linear finite element, MPFEM with linear general shape functions, and the exact solution for the case of $\alpha = 0.02$, $x_0 = 0.5$. It demonstrates that the MPFEM smoothes out the discontinuities between elements. Figures 6(a) and (b) are the convergence rates of the L_2 and H_1 norms, computed from the finite element simulation, the MPFEM defined by (8) for u and (22) for $\mathrm{d}u/\mathrm{d}x$. The first diagram shows that the L_2 norms for the two approaches yield the same convergence rate of two. However, the absolute error of the MPFEM scheme is lower than that of the finite element solution. This improvement is much more marked in the H_1 norm. The MPFEM approach demonstrates a remarkably good convergence rate between 1.5 and 2 in the H_1 norm.

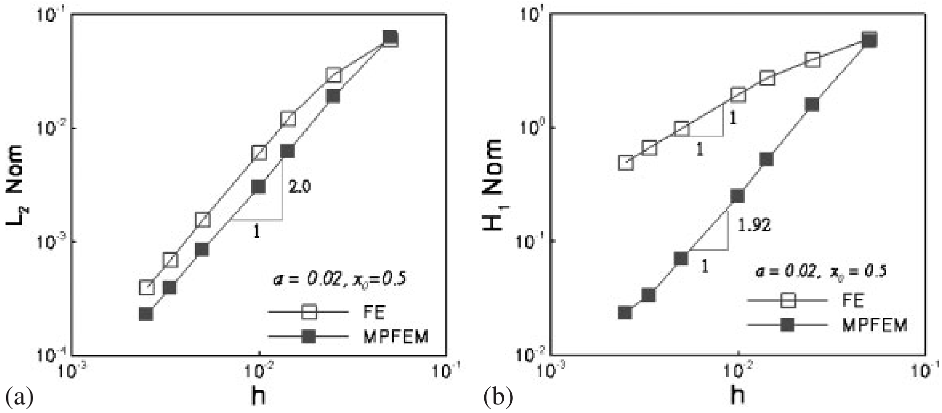

Figure 6. Convergence of MPFEM for the problem defined by (32).

4. DISCUSSION

4.1. Violation of Kronecker-delta property

The Kronecker-delta property does not hold when the compact support of the kernel is too large, i.e. when it intersects any element that is not connected to node P. This occurs because for a general shape function from a parent element that is not connected to node P, Equation (15) does not hold if (13) is not met. This drawback can be avoided in the assumed strain approach by defining a general function to vanish outside its parent element but with derivatives that are extended to the entire domain.

4.2. Multi-dimensional case

We remark here on some additional considerations for multi-dimensional approximations. In two dimensions, the kernel $\phi(\mathbf{r}-\mathbf{x})$ is a function with a compact support Ω_x that is centered at $\mathbf{x}$ with boundary $\partial\Omega_x$; e.g. a circle centred at $\mathbf{x}$ as shown in Figure 7. The union of elements e for which $\Omega^e \cap \Omega_x \neq 0$ forms the domain of influence for the point $\mathbf{x}$. This domain is illustrated by the polygon indicated by bold solid lines in Figure 7(a).

In order to ensure the continuity of the approximation (8) and its first and second order partial derivatives, $\phi(\mathbf{r}-\mathbf{x})$ is required to be a globally C^0 or C^1 function, respectively. When ϕ is defined locally on a compact support Ω_x, it follows that $u^h(\mathbf{x})$ is C^2 if $\phi \in C^1$ on Ω_x and

$$\phi(\mathbf{x}) = 0 \quad \text{and} \quad \partial_i \phi(\mathbf{x}) n_i = 0 \quad \forall \mathbf{x} \in \partial\Omega_x \quad \text{or} \quad \mathbf{x} \notin \Omega_x \tag{36}$$

For the assumed strain approach, the approximation (22) is continuous for any order of derivatives if $\phi(\mathbf{r}-\mathbf{x})$ is a square integrable C^{-1} function in Ω_x.

Plotted in Figure 8 is a two-dimensional example of $N_I^{\mathrm{MPF}}(\mathbf{x})$ and its derivatives for the assumed strain approach. The gradients were constructed by (22) with the general bi-linear shape function and Haar's kernel on a rectangular compact domain:

$$\phi(\mathbf{x}) = \begin{cases} \dfrac{1}{a^2} & \text{for} |x| \leqslant \dfrac{a}{2} \text{ and } |y| \leqslant \dfrac{a}{2} \\ 0 & \text{otherwise} \end{cases}$$

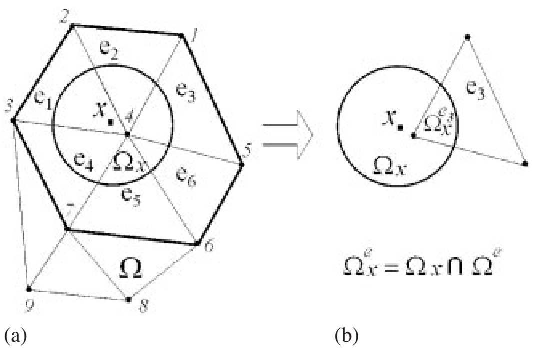

Figure 7. (a) The compact support Ω_x that is centered at $\mathbf{x}$; and (b) definition of Ω_x^e: the intersection of Ω_x and element e.

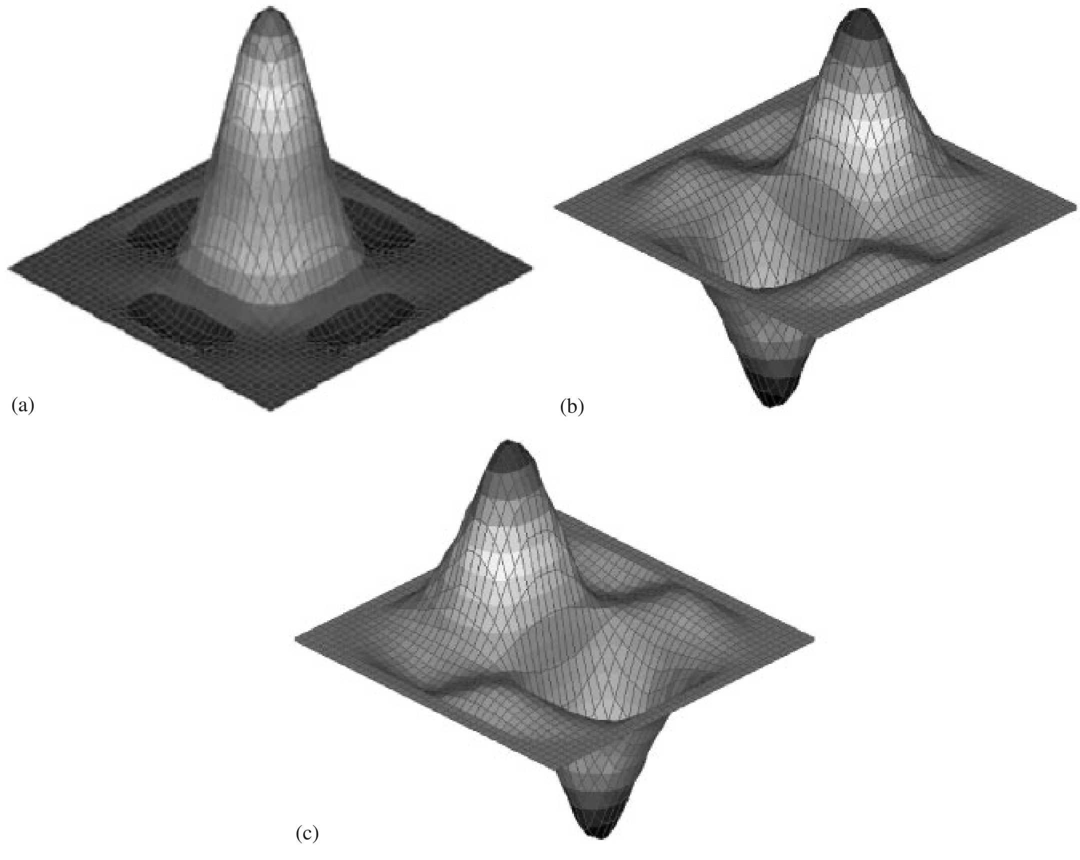

Figure 8. Two-dimensional MPFEM shape function and its derivatives: (a) $N_I^{\mathrm{MPFEM}}(x, y)$; (b) $N_{I,x}^{\mathrm{MPFEM}}(x, y)$; and (c) $N_{I,y}^{\mathrm{MPFEM}}(x, y)$.

4.3. A nodal integration scheme

The integral in (11), which defines the weight $\omega_e(\tilde{x})$, may be tedious to evaluate in multidimensional cases. Here we introduce a nodal integration scheme that is very convenient. By rewriting (22) at node P:

$$\partial_i^n \mathbf{u}^h(\mathbf{x}_P) = \sum_{e\in E^x} \omega_e(\mathbf{x}_P) \sum_{I\in S^e} \partial_i^n \hat{N}_I^e(\mathbf{x}_P)\mathbf{u}_I \quad \text{for } n=0,1,2,\ldots, \tag{37}$$

Instead of (11), we define the weight by

$$\omega_e(\mathbf{x_P}) = \left(\frac{1}{A_e}\right) \Big/ \left(\sum_{a\in E^x} \frac{1}{A_a}\right) \tag{38}$$

where A_a is the area (2D) or the volume (3D) of element a.

5. SUMMARY AND CONCLUSIONS

(1) A moving particle finite element method has been developed. The fundamental idea of this method is to use a kernel approximation in conjunction with general finite element shape functions. As the general shape functions already satisfy certain reproducing conditions, the kernel only needs to be a partition unity. Hence, the proposed method can also be named 'Moving partition-unity finite element method' (MPFEM) or 'Moving kernel finite element method' (MKFEM).

(2) The concept of a 'general shape function' has been proposed, which is the standard finite element shape function but extended to the entire domain.

(3) The proposed MPFEM is an interpolant, i.e. it possesses the Kronecker-delta property.

(4) The proposed MPFEM is C^n if the kernel is globally C^{n-1}.

(5) An assumed strain MPFEM approximation has been proposed for which any order of derivatives is continuous if the kernel is a square integrable function.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the National Science Foundation, the Office of Naval Research and the Army Research Office. The authors would also like to thank Dr L. Zhang, H. Park, and Dr H. Lu for their help in preparing this manuscript. The helpful discussions with Dr G. Wagner of Sandia National Lab. are also appreciated.

REFERENCES

1. Oden JT. *Finite Elements of Nonlinear Continua*. McGraw-Hill: New York, 1972.
2. Hughes TJR. *The Finite Element Method*. Prentice-Hall: New Jersey, 1987.
3. Belytschko T, Liu WK, Moran B. *Nonlinear Finite Elements for Continua and Structures*. Wiley: New York, 2000.
4. Zienkiewicz OC, Taylor RL. *The Finite Element Method*. vol. 1. McGraw Hill: New York, 1989.
5. Belytschko T, Lu YY, Gu L. Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering* 1994; **37**:229–256.
6. Belytschko T, Krongauz Y, Dolbow J, Gerlach C. On the completeness of meshfree particle methods. *International Journal for Numerical Methods in Engineering* 1998; **43**(5):785–812.
7. Liu WK, Jun S, Zhang YF. Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids* 1995; **20**:1081–1106.
8. Oden JT, Duarte CAM, Zienkiewicz OC. A new cloud-based hp finite element method. *Computer Methods in Applied Mechanics and Engineering* 1998; **153**:117–126.

9. Duarte CA, Oden JT. An h-p adaptive method using clouds. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**(1–4):237–262.
10. Randles P, Libersky L. Smoothed particle hydrodynamics: some recent improvements and applications. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:375–408.
11. Monaghan JJ. Why particle methods work. *SIAM Journal on Scientific and Statistical Computing* 1982; **3**(4):422–433.
12. Chen JS, Pan CH, Wu CT. Application of reproducing kernel particle method to large deformation contact analysis of elastomers. *Rubber Chemistry and Technology* 1998; **71**(2):191–213.
13. Li SF, Liu WK. Synchronized reproducing kernel interpolant via multiple wavelet expansion. *Computational Mechanics* 1998; **21**:28–47.
14. Hao S, Liu WK, Chang CT. Computer implementation of damage models by finite element and meshfree methods. *Computer Methods in Applied Mechanics and Engineering* 2000; **187**(3–4):401–440.
15. Belytschko T, Krongauz Y, Organ D, Fleming M, Krysl P. Meshless methods: an overview and recent developments. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**(1–4):3–47.
16. Li S, Liu WK. Meshfree and particle methods and their applications. *Applied Mechanics Reviews* 2002; **55**:1–34.
17. Lancaster P, Salkauskas K. Surfaces generated by moving least square methods. *Mathematics of Computation* 1980; **37**(155):141–158.
18. Nayroles B, Touzot G, Villon P. Generalizing the finite element method: diffuse approximation and diffuse elements. *Computational Mechanics* 1992; **10**:307–318.
19. Wagner GJ, Liu WK. Application of essential boundary conditions in mesh-free methods: a corrected collocation method. *International Journal for Numerical Methods in Engineering* 2000; **47**:1367–1379.
20. Bonet J, Kulasegaram S. Correction and stabilization of smooth particle hydrodynamics methods with applications in metal forming simulations. *International Journal for Numerical Methods in Engineering* 2000; **47**:1189–1214.
21. Dolbow J, Belytschko T. Numerical integration of the Galerkin weak form in meshfree methods. *Computational Mechanics* 1999; **23**(3):219–230.
22. Belytschko T, Black T. Elastic crack growth in finite elements with minimal remeshing. *International Journal for Numerical Methods in Engineering* 1999; **45**(5):601–620.
23. Nayroles B, Touzot G, Villon P. The diffuse elements method. *Comptes Rendus de l'Academie des Sciences Serie II* 1991; **313**(2):133–138.
24. Sukumar N, Moran B, Belytschko T. The natural element method in solid mechanics. *International Journal for Numerical Methods in Engineering* 1998; **43**(5):839–887.
25. Strouboulis T, Babuska I, Copps K. The design and analysis of the generalized finite element method. *Computer Methods in Applied Mechanics and Engineering* 2000; **181**:43–69.
26. Huerta A, Fernandez-Mendez S. Enrichment and coupling of the finite element and meshless methods. *International Journal for Numerical Methods in Engineering* 2000; **48**(11):1615–1636.
27. Chen JS, Yoon SP, Wu CT. Non-linear version of stabilized conforming nodal integration for Galerkin mesh-free methods. *International Journal for Numerical Methods in Engineering* 2002; **53**(12):2587–2615.
28. Hao S, Liu WK. Moving particle finite element method with global superconvergence. *Proceedings of the 5th World Congress for Computational Mechanics* (*WCCM*), 7–12 July 2002, Vienna, Austria.
29. De SR, Bathe KE. The method of finite spheres with improved numerical integration. *Computers and Structures* 2001; **79**(22–25):2183–2196.
30. Atluri SN, Zhu TL. New concepts in meshless methods. *International Journal for Numerical Methods in Engineering* 2000; **47**(1–3):537–556.
31. Liu WK, Li SF, Belytschko T. Moving least-square reproducing kernel methods. 1. Methodology and convergence. *Computer Methods in Applied Mechanics and Engineering* 1997; **143**(1–2):113–154.
32. Li SF, Liu WK. Moving least-square reproducing kernal method Part II: Fourier analysis. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**(1–4):159–194.
33. Babuska I, Melenk JM. The partition of unity method. *International Journal for Numerical Methods in Engineering* 1997; **40**:727–758.
34. Provder T, Rosen EM. Instrument spreading correction in gpc.1. general shape function using a linear calibration curve. *Separation Science* 1970; **5**(4):437.
35. Warburto D, Greeno JG. General shape function model of learning with applications in psychobiology. *Psychological Review* 1970; **77**(4):348.
36. Simo JC, Hughes TJR. On the variational foundations of assumed strain methods. *Journal of Applied Mechanics* 1986; **53**:51–54.
37. Krongauz Y, Belytschko T. A Petrov-Galerkin diffuse element method (PG DEM) and its comparison to EFG. *Computational Mechanics* 1997; **19**(4):327–333.
38. Krongauz Y, Belytschko T. Consistent pseudo-derivatives in meshless methods. *Computer Methods in Applied Mechanics and Engineering* 1997; **146**:371–386.