INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING, VOL. 38, 1655–1679 (1995)

REPRODUCING KERNEL PARTICLE METHODS FOR STRUCTURAL DYNAMICS

WING KAM LIU*, SUKKY JUN†, SHAOFAN LI†, JONATHAN ADEE† AND TED BELYTSCHKO*

Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208, U.S.A.

SUMMARY

This paper explores a Reproducing Kernel Particle Method (RKPM) which incorporates several attractive features. The emphasis is away from classical mesh generated elements in favour of a mesh free system which only requires a set of nodes or particles in space. Using a Gaussian function or a cubic spline function, flexible window functions are implemented to provide refinement in the solution process. It also creates the ability to analyse a specific frequency range in dynamic problems reducing the computer time required. This advantage is achieved through an increase in the critical time step when the frequency range is low and a large window is used. The stability of the window function as well as the critical time step formula are investigated to provide insight into RKPMs. The predictions of the theories are confirmed through numerical experiments by performing reconstructions of given functions and solving elastic and elastic–plastic one-dimensional (1-D) bar problems for both small and large deformation as well as three 2-D large deformation non-linear elastic problems. Numerical and theoretical results show the proposed reproducing kernel interpolation functions satisfy the consistency conditions and the critical time step prediction; furthermore, the RKPM provides better stability than Smooth Particle Hydrodynamics (SPH) methods. In contrast with what has been reported in SPH literature, we do not find any tensile instability with RKPMs.

KEY WORDS: smooth particle hydrodynamics; wavelets; elastic-plastic large deformation; tensile instability; correction function; aliasing control

1. INTRODUCTION

There is always a drive to find new, more advantageous ways to analyse problems using numerical methods. Typical finite elements use linear or quadratic shape functions to define the response within each element. For large deformation or high frequency problems, the elements must be very small to predict accurately the characteristic response. To avoid these problems, the Reproducing Kernel Particle Method (RKPM) advantages are exploited.

There is no explicit mesh, so mesh creation time is saved. Since a mesh is not required, there is no problem due to mesh entanglements allowing for large deformations and unrestrained movement of nodes. Not only is mesh creation time saved, but mesh recreation time is eliminated since to refine the problem in an area of interest one needs only to add points in the interesting region. Nayroles *et al.*[1] developed a meshless method called diffuse element method with application to heat conduction. Belytschko *et al.*[2–4] developed an Element Free Galerkin Method (EFGM) based on the moving least-squares interpolant of Lancaster and Salkauskas[5] which satisfies the patch test. The Smooth Particle Hydrodynamics (SPH) method, developed by

* Professor of Mechanical and Civil Engineering
† Graduate student

CCC 0029–5981/95/101655–25

Received 30 December 1993
Revised 13 July 1994

Gingold and Monaghan,[6] Monaghan[7] and others, also provides a mesh free environment but it has some difficulties creating accurate solutions on the boundaries or when a small number of particles is used.[8] The SPH method is similar in basic construction to the RKPM to be presented here, but it lacks key features of the RKPM.

In this reproducing kernel particle method, the approximate solution is constructed by a continuous reproducing kernel function. It can also be viewed as a continuous least-squares polynomial.[9-11,2-4] However, by exploiting the moment definitions of the flexible window function, the reproducing kernel can be reduced to a simplified form so its properties can be investigated. Second, a localized flexible window function is incorporated by translating the function across the entire domain to reproduce the response. The window function which generates the shape function is controlled by two parameters. The two parameters allow a greater problem-solving ability than typical finite elements which only have one. The flexible window allows for response frequencies or wave numbers to be selectively reproduced in the numerical approximation. Essential to the development of this method is an understanding of the stability limits of the flexible window function, as well as the critical time step in dynamic analyses. The aforementioned properties of the flexible window function and reproducing kernel create interesting characteristics for the kernel stability. Explicit criteria are developed for both of these stability limits.

The reproducing kernel in this derivation is also similar to the SPH Lagrangian particle method[12] with one major difference: the development of a correction function for boundary effects. With this correction function, the so-called *tensile instability*, which has recently been explained by Attaway *et al.*,[13] Libersky *et al.*[14] and Johnson,[15] has been *completely* eliminated. The primary motivation behind the correction function is to provide accurate solutions at the boundaries, but by careful integration techniques it is also possible for the correction function to provide stability to the solution. SPH Lagrangian methods provide accurate solutions in the interior of the problem when the number of particles is large, but they do not provide a means to get an accurate solution near the boundary. The correction functions of RKPMs are relatively dormant in the interior and then provide correction on demand at the boundaries.

Through the implementation of a window function and the exploitation of the Fourier transform, it is possible to develop a new type of shape function that can still be used in the usual Galerkin formulation. The derivative of the shape function, and thus reproducing kernel, can be obtained by direct differentiation. The development of the proposed shape function will be derived in detail later, but for now we describe its characteristics. The two parameters in the shape function provide the ability to translate and dilate the window function. Translation is required to move the window function around the domain since the window functions themselves have a compact support. The ability to translate replaces the need to define elements. The dilation parameter is used to provide refinement. The larger the dilation parameter, the smaller the frequency band is in the solution, and the larger the critical time step becomes in dynamic analyses. The refinement parameter transformation between the time and frequency domain (or space and wave number) controls the solution space. This introduces the ability to choose the size of the frequency or wave number range in the calculation.

2. DEVELOPMENT OF THE REPRODUCING KERNEL

There are many examples of reproducing kernels; the most noteworthy may be the Fourier transform. The Fourier transform motivates this study since it originates the concept of analysing specific frequency bands. This capability is also incorporated in this reproducing kernel method.

A reproducing kernel is a class of operators that reproduce the function itself through integration over the domain. The Fourier transform is an excellent example of a reproducing kernel. The Fourier transform $\hat{f}(\omega)$, of a function, $f(x)$ is defined by

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} e^{-ix\omega} f(x)\,dx \tag{1}$$

and the inverse Fourier transform is defined by

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{ix\omega}\hat{f}(\omega)\,d\omega \tag{2}$$

Bounds of the transform can be easily prescribed if the spectrum of interest is limited to a single frequency band, $-\Omega \leqslant \omega \leqslant \Omega$. The bounds of the Fourier transform, $\hat{f}(\omega)$, are given by

$$\begin{aligned} \hat{f}(\omega) &\neq 0 \quad \text{if } |\omega| \leqslant \Omega \\ \hat{f}(\omega) &= 0 \quad \text{if } |\omega| > \Omega \end{aligned} \tag{3}$$

It is now shown that the inverse transform on the transform modified by equation (3) is a reproducing kernel representation if $f(x) \neq 0$ only in the domain $x \in [0, L]$:

$$\begin{aligned} f(x) &= \frac{1}{2\pi}\int_{-\Omega}^{\Omega} e^{ix\omega}\left[\int_{-\infty}^{\infty} e^{-iy\omega} f(y)\,dy\right] d\omega \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty}\left[\int_{-\Omega}^{\Omega} e^{i(x-y)\omega}\,d\omega\right] f(y)\,dy \\ &= \int_{-\infty}^{\infty} \frac{\sin[\Omega(x-y)]}{\pi(x-y)} f(y)\,dy \\ &= \int_{0}^{L} \frac{\sin[\Omega(x-y)]}{\pi(x-y)} f(y)\,dy = \int_{0}^{L} \Phi(x-y) f(y)\,dy \end{aligned} \tag{4}$$

This example of a reproducing kernel using a sinc function is presented by Chui[16] and Liu *et al.*[17] In Liu's analysis, the multiple scale analysis predicts a critical time step according to the following equation:

$$\Delta t \leqslant \frac{2\pi}{\Delta\omega} \tag{5}$$

This is used as a guideline since equation (5) bounds the sampling rate needed to prevent aliasing when using the reproducing kernel for a given frequency band, $\Delta\omega$. It is believed that after discretizing the continuous reproducing kernel, a similar form exists for the stability condition of the reproducing kernel in space (see Section 4).

2.1. *Window function selection*

The *ideal* window function, $\Phi(x)$, is chosen such that the following two conditions are satisfied. Its integral over the domain, R_x, must be unity, and $\Phi(x)$ must also be orthogonal.

$$\int_{R_x} \Phi(\mathbf{x})\,dR_x = 1 \tag{6}$$

$$\int_{R_x} \Phi(\mathbf{x}-\mathbf{a})\Phi(\mathbf{x}-\mathbf{b})\,dR_x = \delta_{ab} \tag{7}$$

Possibility for window functions includes splines, wavelets, or Gaussian functions, but none of them satisfy both of these conditions. The integration of orthogonal wavelets does not produce unity over the entire domain, rather the value is zero. Furthermore, most wavelets cannot exactly reproduce constant or linear terms, i.e. they cannot reproduce the simplest functions very well. For these reasons, only scaling functions, which are used to generate wavelets, are considered here, and the appealing characteristics from the wavelet analysis[16,18] are used. Splines are always an option since we can custom tailor them, but they can produce undesirable irregularities. The Gaussian function is not orthogonal, but it is used in this analysis because of its special properties. It is also important to keep in mind that the integral on the interior of the domain will be less than the integral near the boundaries.

2.2. Development of window function

Like the Fourier transform pair, we want to be able to reconstruct any function $u(x)$ by a series of window functions, $\Phi(x)$. Since $\Phi(x)$ is a localized function, it is necessary to translate the function to represent the entire response. This is performed by inserting the argument $s - x$ in the function. Now $u(x)$ is presented by the following integral:

$$\int_{R_x} E(s, x)\Phi(s - x)\,dR_x = 1 \tag{8}$$

where $E(s, x)$ is the correction function to be defined later in the construction, Section 2.4. Or it could be an arbitrary constant as in SPH methods.[12]

To the window function argument, it is also necessary to add the dilation or refinement parameter, r. This is incorporated by dividing the window function argument by this refinement parameter. For notational purposes this definition is used for the window function:

$$\frac{1}{r}\Phi\left(\frac{s - x}{r}\right) \equiv \Phi_{rx} \tag{9}$$

$$\int_{R_x} E(s, x)\Phi_{rx}(x)\,dR_x = \int_{R_x} E(s, x)\Phi(x)\,dR_x = 1 \tag{10}$$

When the Gaussian distribution is used for the window function, an intuitive sense of the refinement parameter can be revealed as the *standard deviation*. The additional constant, r^{-1}, scales the window function so that integral over the domain window equals one according to equation (10).[10] It is also useful to derive the moment equations in Section 2.4.1.

$$\int_{R_x} E(s, x)\frac{1}{r}\Phi\left(\frac{s - x}{r}\right)dR_x = 1 \tag{11}$$

This completes the development of the window function to be used in this 1-D analysis.

2.3. The reconstruction equation

The concept that any function can be represented as a sum of linearly independent functions initiates the analysis. Starting with the following definition

$$u(\mathbf{x}) = \mathbf{P}(\mathbf{x})\mathbf{d} \tag{12}$$

where $\mathbf{P}(\mathbf{x}) = [P_1(\mathbf{x}), P_2(\mathbf{x}), \ldots, P_n(\mathbf{x})]$ which is any number of linearly independent functions and $\mathbf{d}^T = [d_1, d_2, \ldots, d_n]$ is the same number of unknown coefficients. For a linear one-dimen-

sional case, $P(x) = [1, x]$, or for a quadratic one-dimensional case, $P(x) = [1, x, x^2]$, etc. This moving least-squares interpolant type of reconstruction[5] has been used by Nayroles *et al.*[1] and Belytschko *et al.*[2-4] for the element free Galerkin method.

It is possible to solve for the unknown coefficients $\mathbf{d}$ by using the window function. The variable $\mathbf{x}$ is changed to $\mathbf{s}$ in equation (12), and then both sides are premultiplied by $\mathbf{P}^{\mathrm{T}}(\mathbf{s})$ and the integral window transform is applied.[10] Both sides of equation (12) are multiplied by the window function, Φ_{rx}, and then integrated over the domain.

$$\int_{R_x} \mathbf{P}^{\mathrm{T}}(\mathbf{s})\, u(\mathbf{s})\, \Phi_{rs}\, \mathrm{d}s = \int_{R_x} \mathbf{P}^{\mathrm{T}}(\mathbf{s})\, \mathbf{P}(\mathbf{s})\, \Phi_{rs}\, \mathrm{d}s\, \mathbf{d} \tag{13}$$

$$\mathbf{C}(x) = \int_{R_x} \mathbf{P}^{\mathrm{T}}(\mathbf{s})\, \mathbf{P}(\mathbf{s})\, \Phi_{rx}\, \mathrm{d}s \tag{14}$$

Equation (14) is used in the next section to complete the construction of the correction function. (In that section the merits of $\mathbf{C}$ are discussed which shows why the term is sometimes referred to as the boundary correction function.) By this definition for $\mathbf{C}$, the solution for $\mathbf{d}$ can be substituted back into (12), obtaining the reconstruction equation in (16) which can also be written as one integral (17):

$$\mathbf{d} = \mathbf{C}^{-1}(\mathbf{x}) \int_{R_x} \mathbf{P}^{\mathrm{T}}(\mathbf{s})\, u(\mathbf{s})\, \Phi_{rx}\, \mathrm{d}s \tag{15}$$

$$u(\mathbf{x}) = \mathbf{P}(\mathbf{x})\mathbf{C}^{-1}(\mathbf{x}) \int_{R_x} \mathbf{P}^{\mathrm{T}}(\mathbf{s})\, u(\mathbf{s})\, \Phi_{rx}\, \mathrm{d}s \tag{16}$$

$$u(\mathbf{x}) = \int_{R_x} \mathbf{P}(\mathbf{x})\, \mathbf{C}^{-1}(\mathbf{x}) \mathbf{P}^{\mathrm{T}}(\mathbf{s})\, u(\mathbf{s})\, \frac{1}{r}\, \Phi\left(\frac{s-x}{r}\right) \mathrm{d}s \tag{17}$$

2.4. Correction function

Comparing the reconstruction equation in (17) to the SPH method reveals that the only difference is the appearance of the $\mathbf{P}(\mathbf{x})\mathbf{C}^{-1}(\mathbf{x})\, \mathbf{P}^{\mathrm{T}}(\mathbf{s})$ term in the RKPM. This term is defined as the correction function $E(x, s)$, and its merits are analysed in this section. For simplicity, the characteristics of this function are derived here in one dimension for linear polynomials $\mathbf{P}(\mathbf{x})$, but it can be shown to be valid for multiple dimensions as well.[10] Expanding equation (14) gives the following:

$$\mathbf{C}(x) = \int_{R_x} \begin{bmatrix} 1 \\ s \end{bmatrix} [1 \;\; s] \frac{1}{r}\, \Phi\left(\frac{s-x}{r}\right) \mathrm{d}s \tag{18}$$

The inverse of this matrix will be computed along with $\mathbf{P}(\mathbf{x})$ and $\mathbf{P}^{\mathrm{T}}(\mathbf{s})$ from the reconstruction formula in equation (17) to form the correction function. The entire term, the $\mathbf{P}(\mathbf{x})\mathbf{C}^{-1}(\mathbf{x})\mathbf{P}^{\mathrm{T}}(\mathbf{s})$ function, simplifies to a scalar regardless of the number of terms used for $\mathbf{P}(\mathbf{x})$ and $\mathbf{P}^{\mathrm{T}}(\mathbf{s})$.

2.4.1. Definition of moments. In order to evaluate equation (18) moments are defined in the following manner:

$$m_0 = \int_{B(x)} \Phi(z)\, \mathrm{d}z \tag{19}$$

$$m_1 = \int_{B(x)} z\,\Phi(z)\,dz \tag{20}$$

$$m_{11} = \int_{B(x)} z^2\,\Phi(z)\,dz \tag{21}$$

These moment equations are integrated over the region $B(x)$, where $B(x)$ is the region where the window function is non-zero. The calculation could be performed over the entire domain; however, many unnecessary calculations are involved. In order to estimate these moments for the Gaussian function, a truncated three standard deviation criterion is enforced which ensures the inclusion of 99·7 per cent of the total area. The values of the moment are given in Table I. It will be shown that these characteristics lead to a correction function that provides an accurate solution near the boundary, and lies relatively dormant in the interior.

Because the correction function is unity in the interior, it can be concluded that it plays no role in the interior. It is now known that in numerical implementation, the correction function and its inverse function can have a profound effect on the stability of the kernel, see Figures 3 and 4. It is noted that the stabilization effect is much more pronounced in the data when the number of nodes is relatively small. In addition, with the correct construction of the *antisymmetric moment* m_1 (equation (20)), the so-called tensile instability[13-15] associated with SPH methods is completely eliminated.

2.4.2. Final correction function form. The expression for $\mathbf{P(x)C^{-1}(x)P^T(s)}$ can be written in the simplified form shown below through manipulation. Equation (18) can be inverted and substituted back into the reconstruction in equation (17) to reveal a continuous reproducing kernel for the function $u(x)$. This is the final reconstruction equation to be used in this analysis. It is noted that the correction function, $E(x,s)$ is simplified into the sum of the two terms which are defined in the following manner:

$$u^h(x) = \int_{R_x} E(x,s)\,u(s)\,\frac{1}{r}\,\Phi\left(\frac{x-s}{r}\right)ds \tag{22}$$

$$E(x,s) = C_1(x) + C_2(x)\left(\frac{x-s}{r}\right) \tag{23}$$

$$C_1 = \frac{m_{11}}{(m_0 m_{11} - m_1^2)} \tag{24}$$

$$C_2 = \frac{m_1}{(m_0 m_{11} - m_1^2)} \tag{25}$$

Table I. The differences of moments between interior region and near boundaries

	Interior region	Near boundaries
m_0	$= 1$	< 1
m_1	$= 0$	$\neq 0$
m_{11}	$= r^2$	$\neq r^2$

The discretized form of the reconstruction equation can be used to reveal certain properties of the RKPM. The discretized form is given by

$$u^h(x) \cong \sum_{J=1}^{NP} \left[C_1(x) + C_2(x)\left(\frac{x - x_J}{r}\right)\right] \frac{1}{r} \Phi\left(\frac{x - x_J}{r}\right) u(x_J)\Delta x_J \tag{26}$$

The discretized reconstruction equation can also be written in a more familiar form in terms of a shape function $N_J(x)$ in equation (27):

$$u^h(x) = \sum_{J=1}^{NP} N_J(x)\, u_J \tag{27}$$

$$N_J(x) = \left[C_1(x) + C_2(x)\left(\frac{x - x_J}{r}\right)\right] r^{-1} \Phi\left(\frac{x - x_J}{r}\right)\Delta x_J \tag{28}$$

The characteristics of this new shape function must be carefully analysed to avoid erroneous results. The most apparent difference is that the shape function does not meet the Kronecker delta indentity since each node is influenced by several shape functions. However, the shape function will meet the consistency condition. This can be proven by the following equation where the reconstruction equation can be substituted to reproduce itself:

$$\begin{aligned} \mathbf{P}(\mathbf{x}) &= \int_{R_x} \mathbf{P}(\mathbf{x})\mathbf{C}^{-1}(\mathbf{x})\mathbf{P}^{\mathrm{T}}(\mathbf{s})\,\Phi\left(\frac{\mathbf{s} - \mathbf{x}}{r}\right)\mathbf{P}(\mathbf{s})\,d\mathbf{s} \\ &= \mathbf{P}(\mathbf{x})\mathbf{C}^{-1}(\mathbf{x})\int_{R_x} \mathrm{P}^{\mathrm{T}}(\mathbf{s})\,\mathrm{P}(\mathbf{s})\,\frac{1}{r}\Phi\left(\frac{\mathbf{s} - \mathbf{x}}{r}\right)d\mathbf{s} \\ &= \mathbf{P}(\mathbf{x}) \end{aligned} \tag{29}$$

It is also pointed out that the integration method used to calculate $\mathbf{C}$ and $\mathbf{C}^{-1}$ must be similar in order to meet this condition. It is also proposed by Liu *et al.*[10] that by using the trapezoidal rule to integrate the moments defined within $\mathbf{C}$, the stability of the kernel is increased. The shape functions will meet the following isoparametric shape function properties.

$$\sum_{J=1}^{NP} N_J(x) = 1 \quad \text{and} \quad \sum_{J=1}^{NP} N_J(x)\, x_J = x \tag{30}$$

By analysing the properties of the moments, in a *continuous* case the values for C_1 and C_2 are found to equal 1 and 0 respectively in the interior of a domain (assuming a sufficiently large number of particles), but definitely not equal to these values on the boundary. In the continuous case, the RKPM will be identical to the SPH only in the interior.

However, in the discretized form, the effect of the correction function will depend on the integration technique used as well as the number of particles in the integration domain. These factors will determine whether the correction function will only act on the boundaries or enhance the stability of the solution throughout the entire domain. In this paper the trapezoidal rule is used to calculate the moments of which the discretized forms are given in the following:

$$\overline{m}_0(x) = \sum_{J}^{NP} a^{-1}\,\Phi\left(\frac{x_J - x}{a}\right)\Delta x_J \tag{31}$$

$$\overline{m}_1(x) = \sum_{J}^{NP} \left(\frac{x_J - x}{a}\right) a^{-1}\,\Phi\left(\frac{x_J - x}{a}\right)\Delta x_J \tag{32}$$

$$\overline{m}_{11}(x) = \sum_{J}^{NP} \left(\frac{x_J - x}{a}\right)^2 a^{-1}\,\Phi\left(\frac{x_J - x}{a}\right)\Delta x_J \tag{33}$$

It is noted that the above moments $\bar{m}_0$, $\bar{m}_1$ and $\bar{m}_{11}$ possess the approximate characteristics given in Table I.

2.5. *Gaussian reproducing kernel formulation*

It is important at this point to define the Gaussian function used in this analysis at this time. The refinement parameter is defined to contain a measure of normalization so that a given dilation of the window function always contains the same number of nodes, regardless of particle density. The refinement parameter, r can also be recognized as the standard deviation in the Gaussian equation:

$$r^{-1}\Phi\left(\frac{x-s}{r}\right) = \frac{1}{r\sqrt{\pi}} e^{-(x-s)^2/r^2} \tag{34}$$

$$r = 2^j \Delta x \sqrt{\frac{2}{\pi}} \tag{35}$$

Defining the Gaussian function in this way has the advantage of maintaining the same number of nodes for support while changing the distance between nodes, but the window and shape functions will change while changing nodal co-ordinates as well as the j refinement. The parameter j can be any real number but stability limits are set in Section 5. In prior analyses, the definition in Section 5 was found to be optimal,[9] so it is chosen as a starting point for this analysis. In this paper, the Gaussian window function and cubic spline functions are presented in Section 6.

2.6. *Cubic spline reproducing kernel formulation*

Spline functions have been widely used for window-like kernel functions in many particle methods because it is easy to determine the range of support of window function and they lessen the computing time.[14] The cubic spline functions are defined below:

$$\Phi(z) = \frac{2}{3} - z^2\left(1 - \frac{z}{2}\right) \quad \text{for } 0 \leqslant z \leqslant 1 \tag{36}$$

$$\Phi(z) = \frac{1}{6}(2 - z)^3 \qquad \text{for } 1 \leqslant z \leqslant 2 \tag{37}$$

$$z = \frac{|y - x|}{a} \tag{38}$$

in which the dilation parameter is $\dfrac{a}{\Delta x} = 2^j$.

2.7. *Dynamic frequency analysis*

One of the interesting aspects of this method is its ability to predetermine the frequency range studied in the analysis. The frequency range captured can be determined from the Fourier transform of the window function.[19] The window function's shape can be changed by the refinement parameter, r, allowing for adaptability in the solution process. Even though the Fourier transform of the window function is also bell shaped, the cutoff frequency for the banded window has been found to be $1/r$.[10] This method matches the area under the window function with the area of the straight box created by the cutoff frequency. The approximation for the

highest accurately reconstructed frequency inside the window function, $1/r$, enables an understanding of the limitations of this method in its straight form.

Unfortunately for this preliminary implementation of the method, the frequency window is always centred around $\omega = 0$. This means that we cannot selectively consider only higher frequency bands; it is necessary to capture all frequencies below the highest frequency of interest. The ability of this method to capture high frequencies is also limited by the stability of the reproducing kernel itself. Although the number of nodes is any one window function is variable, there must be at least two nodes in the domain of the window functions for stability. This is necessary in order to have connectivity between window functions. Unless the variable connectivity is always greater than two, the response cannot be translated along to the adjacent nodes.

It is hoped that the multiple scale methods of Liu *et al.*[17] can be included in subsequent analyses to shift the interested frequency bands away from the origin, removing the unnecessary calculation of frequencies between the interesting areas. Another approach is to use wavelets to capture the high frequency bands, which is being investigated by Liu *et al.*[9]

2.8. *Multiple-dimensional analysis*

The extension of the RKPM to multiple dimensions (2-D and 3-D) is straightforward. For 2-D and 3-D analysis,

$$\mathbf{P}(\mathbf{x}) = \{1, x_1, x_2\} \quad \text{(2-D)} \tag{39}$$

$$\mathbf{P}(\mathbf{x}) = \{1, x_1, x_2, x_3\} \quad \text{(3-D)} \tag{40}$$

are employed and the reproducing kernel approximation becomes

$$u(x) = \int_{R_x} E(\mathbf{x}, \mathbf{y}) \Phi(\mathbf{x} - \mathbf{y}) u(\mathbf{y}) \mathrm{d}y \tag{41}$$

where $E(\mathbf{x}, \mathbf{y})$ is shown to be

$$E(\mathbf{x}, \mathbf{y}) = C_1(\mathbf{x}, \mathbf{y}) + \mathbf{C}_2(\mathbf{x}, \mathbf{y}) \cdot \left(\frac{\mathbf{x} - \mathbf{y}}{r} \right) \tag{42}$$

in which $C_1(\mathbf{x}, \mathbf{y})$ and $\mathbf{C}_2(\mathbf{x}, \mathbf{y})$ are functions of moments and given by Liu *et al.*[10] Generally the window function $\Phi(\mathbf{x} - \mathbf{y})$ has elliptic support, depending on the dilation parameters of each spatial co-ordinate. For rectangular geometry, a product of 1-D window function can be used for simplicity and efficiency.

3. FUNCTION RECONSTRUCTION

The reproducing kernel is first used to reconstruct a given function as a demonstration and evaluation technique. This is done by using the discretized form of the reconstruction equation (17) to reproduce a known function. The trapezoidal rule is used here for integration. The dilation parameter has a profound effect on the shape function. Examples of shape functions for several values of the dilation parameter are shown below. Note that for $j = -2$ the window function approaches the ordinary finite element shape function.

Figure 1 portrays that as the dilation parameter decreases in magnitude, the Gaussian window function approaches a Dirac delta function. If it were possible to reproduce the Dirac delta function, all frequencies of a function would be reproduced. Unfortunately, without additional

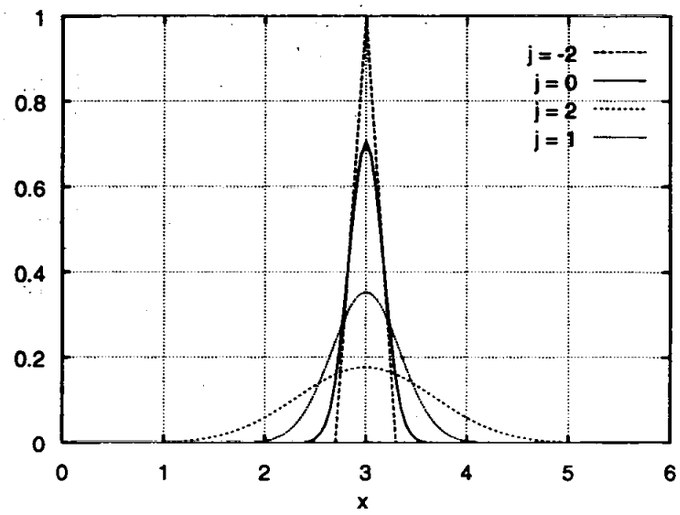

Figure 1. Gaussian shape function dilations for $j = -2, 0, 1, 2$

tools this is not easily possible without a large number of particles, due to the stability limit of the window function itself requiring a minimum of two nodes in its support, Section 5.1.

These reconstructions shown in Figures 2–7 illustrate the intricacies. Notably, it shows the difference between RKPM and SPH methods. The difference arises from the correction function, which enables an accurate approximation throughout the entire domain of the response as well as as increased range of stable operation, expecially for a small number of particles. The accuracy of the correction function can be analysed by using a simple reconstruction of a known function.

It is readily seen in Figure 2 that the SPH reconstruction is not even able to reproduce a simple sinusoidal wave near the boundaries while the RKPM solution is almost identical to exact solution.

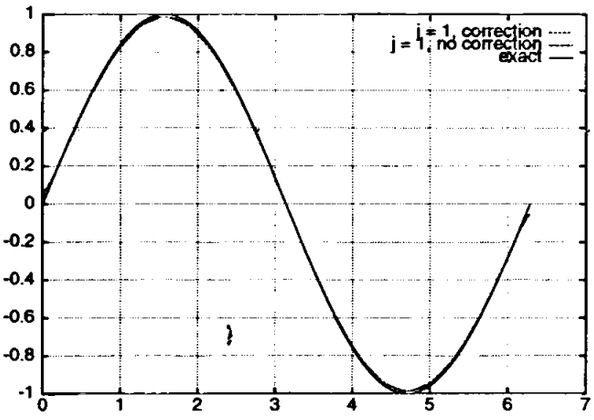

Figure 2. Reconstruction of $\sin(\omega x)$ where $\omega = 1$ with 21 nodes, demonstrating the advantage of the RKPM over SPH near the boundaries

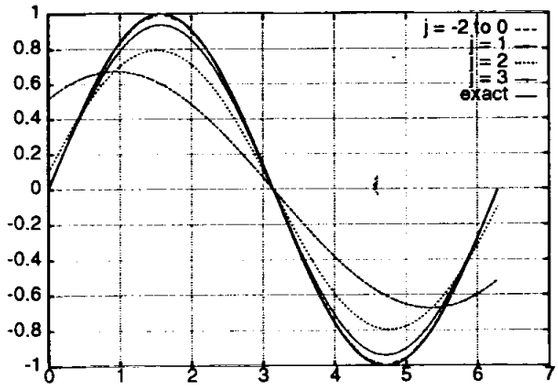

Figure 3. $\sin(x)$ reconstruction using the RKPM, 21 nodes.

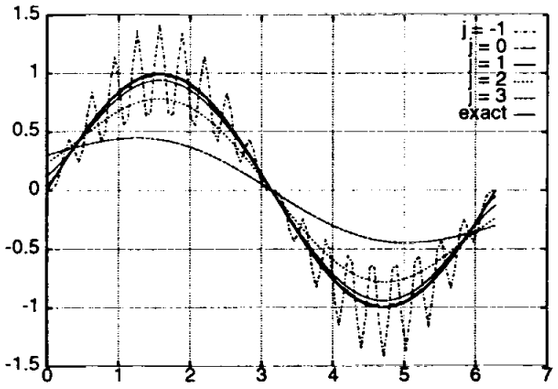

Figure 4. SPH reconstruction for $\sin(x)$ using 21 nodes

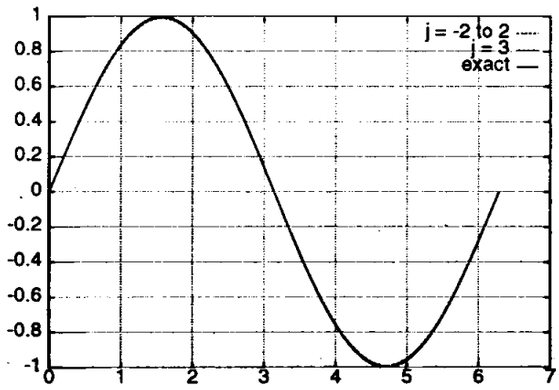

Figure 5. $\sin(x)$ reconstruction using the RKPM, 201 nodes

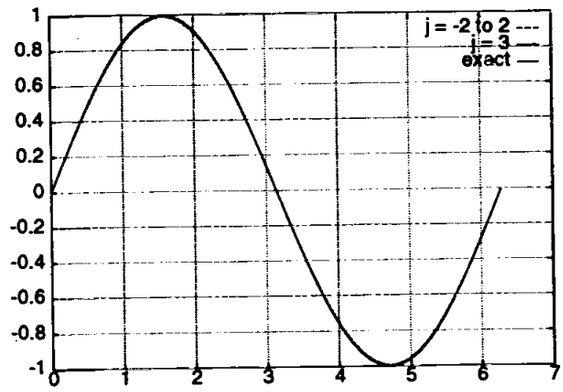

Figure 6. SPH reconstruction for sin(x) using 201 nodes

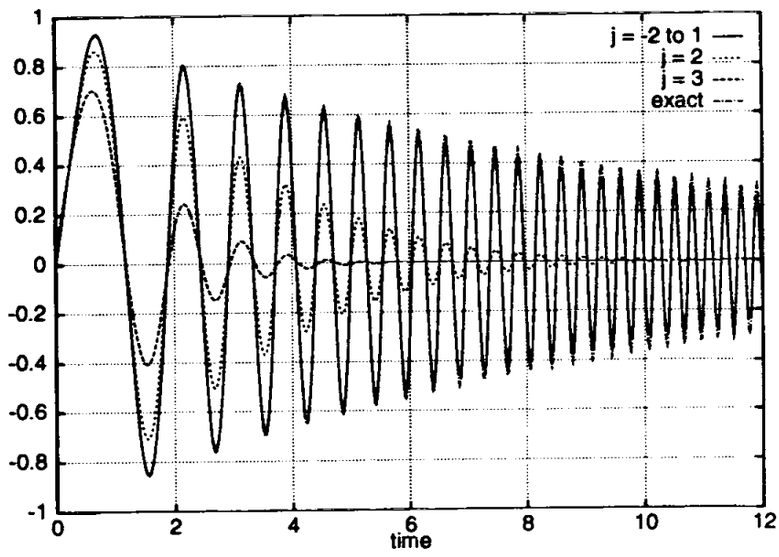

Figure 7. The reconstruction of a sine wave with increasing frequency and damped oscillations for several values of j with 201 nodes

In order to provide a feel for the performance of the dilation parameter, Figures 3 and 4 depict a sinusoidal reconstruction with and without the correction function. Figure 4, which does not contain the boundary correction function, provides an evaluation of the SPH method. The instability of the SPH method proves that the RKPM correction function enhances stability in addition to correcting the reconstruction near the boundaries.

It is important to note that $j = -2$ was not plotted in Figure 4 since the instability of the SPH method would have required a much larger scale. This occurs because the SPH kernel does not satisfy the consistency condition for a small number of nodes; however, for large numbers of nodes the SPH method performs adequately (Figure 6).

Knowing the behaviour or the dilation parameter, namely the effect of the j parameter, our discussion is turned toward a brief look at the frequency content of the window function. By dilating the Gaussian window function (Figure 1), it is easily seen how the function changes shape with the parameter j. It is this dilation that enables the types of analyses discussed in Section 2.7.

A reconstruction is now performed on a sinusoidal wave that has been augmented to increase its frequency as well as dampen its amplitude:

$$f(x) = \sin\left(\frac{\pi}{2}(x + 0{\cdot}6\,x^2)\right)e^{-0{\cdot}1x} \tag{43}$$

The results in Figure 7 clearly shows the ability of the window function to capture the frequency content. The larger the values for the refinement parameter, the more quickly the reconstruction attempt fails.

4. GALERKIN FORMULATION FOR REPRODUCING KERNEL PARTICLE METHODS

The reproducing kernel can be implemented into a Galerkin formulation in a similar way as typical finite elements. The major difference in construction is the loop which occurs over nodes instead of elements, but the formulation is almost identical, beginning from the weak form of the momentum equation. The following variables are used: $\mathbf{t}$ is the traction, $\mathbf{b}$ is the body force, σ is the stress tensor, and $\ddot{u}$ denotes acceleration:

$$\int_{R_x} \rho\,\delta u_i \ddot{u}_i \,dR_x + \int_{R_x} \delta u_{i,j}\sigma_{ij}\,dR_x = \int_{R_x} \delta u_i b_i\,dR_x + \int_{\partial R_x^t} \delta u_i t_i\,d\Gamma_x \tag{44}$$

The element matrices are

$$\{\mathbf{m}_{ab}^e\} = \left\{\int_{R_x^e} \rho_0^e \mathbf{N}_a \mathbf{N}_b\,1\,dR_x^e\right\}, \quad a,b = 1,\ldots,\text{NEN} \tag{45a}$$

$$\{\mathbf{m}_{ab}^e\} = \left\{\int_{R_x^e} \rho_0^e \mathbf{N}_a\,dR_x^e\right\}\delta_{ab}, \quad a,b = 1,\ldots,\text{NEN} \tag{45b}$$

$$\mathbf{f}^{\text{int}}(\mathbf{x},\sigma)^e = \{(\mathbf{f}_a^{\text{int}})^e\} = \left\{\int_{R_x^e} \sigma^e \mathbf{N}_{a,x}\,dR_x^e\right\}, \quad a = 1,\ldots,\text{NEN} \tag{46}$$

$$\mathbf{f}^b(\mathbf{x},\mathbf{b})^e = \{(\mathbf{f}_a^b)^e\} = \left\{\int_{R_x^e} N_a \mathbf{b}^e\,dR_x^e\right\}, \quad a = 1,\ldots,\text{NEN} \tag{47}$$

$$\mathbf{f}^t(\mathbf{x},\mathbf{t})^e = \{(\mathbf{f}_a^t)^e\} = \left\{\int_{\partial R_x^t \cap \partial R_x^e} N_a \mathbf{t}^e\,d\Gamma_x^e\right\}, \quad a = 1,\ldots,\text{NEN} \tag{48}$$

including a *row sum* lumped mass matrix (cf. equation (30), $\sum_J^{\text{NP}} N_J(x) = 1$), but it is important to keep in mind that the matrices have a variable connectivity depending on the number of nodes in the support of the window function, NEN.

The natural boundary conditions are simply entered into the force vector; however, it is difficult to implement the essential boundary conditions since several shape functions can be present at the node dictating the necessary response. In this analysis the essential boundary condition is satisfied by substituting a sum on all the shape functions in place of the row in the stiffness matrix

that corresponds to the constrained nodes. The shape function and its derivative are derived with the correction function in the next two sections.

In a large deformation formulation, the internal force matrix is discretized as below:

$$\mathbf{f}^{\text{int}} = \sum_{k=1}^{\text{NP}} \sigma(\mathbf{x}_k)\, \mathbf{N}_{a,x}(\mathbf{x}_k)\, \Delta x_k \tag{49}$$

where NG is the total number of Gaussian quadrature points. The integration weight Δx_k is replaced by $\Delta x_k = \Delta M_k/\rho_k$ in which ΔM_k and ρ_k are respectively the nodal mass and density associated with an integration point x_k. Prior to updating Δx_k and x_k, the nodal spatial co-ordinate is updated via

$$\mathbf{x}_J = \mathbf{X}_J + \mathbf{u}_J, \quad J = 1, \ldots, \text{NP} \tag{50}$$

and the shape functions (Equation 27)) are used to update the Gaussian quadrature points and the mass densities ρ_k as

$$\mathbf{x}_k = \sum_{J=1}^{\text{NP}} \mathbf{N}_J(\mathbf{x}_k)\mathbf{x}_J, \quad k = 1, \ldots, \text{NG} \tag{51}$$

$$\rho_k = \sum_{J=1}^{\text{NP}} \mathbf{N}_J(\mathbf{x}_k)\rho_J, \quad k = 1, \ldots, \text{NG} \tag{52}$$

The nodal densities ρ_J are computed using the continuity equation.

4.1. *Shape function derivatives*

Unlike the moving least-squares methods, diffuse element methods and element free Galerkin methods, the derivative of the RKPM shape function (Equation (28)) can simply be obtained by direct differentiation. It is necessary to consider the correction terms as well as the window function itself to obtain the derivative of the shape function:

$$N_{J,x}(x) = \left\{C_1'(x) + C_2'(x)\left(\frac{x - x_J}{r}\right) + \frac{C_2(x)}{r}\right\} r^{-1}\Phi\left(\frac{x - x_J}{r}\right)\Delta x_J + \left\{C_1(x) + C_2(x)\left(\frac{x - x_J}{r}\right)\right\}\frac{1}{r}\Phi'\left(\frac{x - x_J}{r}\right)\Delta x_J \tag{53}$$

$$C_1' = \frac{m_{11}'}{m_0 m_{11} - m_1^2} - \frac{m_{11}(m_0' m_{11} + m_0 m_{11}' - 2m_1 m_1')}{(m_0 m_{11} - m_1^2)^2} \tag{54}$$

$$C_2' = \frac{m_1'}{m_0 m_{11} - m_1^2} - \frac{m_1(m_0' m_{11} + m_0 m_{11}' - 2m_1 m_1')}{(m_0 m_{11} - m_1^2)^2} \tag{55}$$

If a Gaussian window function is considered, the derivative of the window function is

$$\Phi'\left(\frac{x - s}{r}\right) \equiv \frac{2}{r\sqrt{\pi}}\left(\frac{x - s}{r}\right)\exp\left(-\left(\frac{x - s}{r}\right)^2\right) \tag{56}$$

Now it is necessary to calculate the derivatives of the moments which can be combined with derivative of the Gaussian window function to reveal the final derivative in equation (60):

$$m_0'(x) = \int_{B(x)} \frac{2}{r^2\sqrt{\pi}}\left(\frac{x - s}{r}\right)\exp\left(-\left(\frac{x - s}{r}\right)^2\right) ds \tag{57}$$

$$m'_1(x) = \int_{B(x)} \frac{1}{r^2\sqrt{\pi}}\left[2\left(\frac{x-s}{r}\right)^2 - 1\right]\exp\left(-\left(\frac{x-s}{r}\right)^2\right)ds \tag{58}$$

$$m'_{11}(x) = \int_{B(x)} \frac{2}{r^2\sqrt{\pi}}\left(\frac{x-s}{r}\right)\left[\left(\frac{x-s}{r}\right)^2 - 1\right]\exp\left(-\left(\frac{x-s}{r}\right)^2\right)ds \tag{59}$$

$$N_{J,x}(x) = \left\{C'_1(x) + C'_2(x)\left(\frac{x-s}{r}\right) + \frac{C_2(x)}{r}\right\}r^{-1}\Phi\left(\frac{x-s}{r}\right)\Delta x_J$$
$$-\frac{2}{r}\left\{C_1(x) + C_2(x)\left(\frac{x-s}{r}\right)\right\}\left(\frac{x-s}{r}\right)r^{-1}\Phi\left(\frac{x-s}{r}\right)\Delta x_J \tag{60}$$

5. STABILITY ANALYSIS

It is interesting to note that the stability of this type of analysis is twofold. First there is the stability of the reproducing kernel itself, and then there is the stability of the time integration method. In this analysis an explicit Newmark beta predictor/corrector algorithm is implemented.[20]

5.1. Reproducing kernel stability

The stability of the kernel is mainly a function of the number of nodes encompassed by the window function. (Theoretically the number of nodes covered is the number of nodes in the analysis, since the function has an infinite domain.) The number of significant nodes in a shape function is controlled by the refinement parameter, r. From the aforementioned definitions for the shape of the window function the following equation can be derived to estimate the number of nodes under any given window function. If the radius of a given window function, Δx_c, is defined as the distance from the centre to the edge of the window's significant support, then a ratio can be defined to relate the height of the window function at the peak to the small value where it is safely approximated to be zero. For a Gaussian window function,

$$R = \frac{\exp(-\Delta x_c^2/r^2)}{\exp(-\Delta x_0^2/r^2)} \tag{61}$$

$$\Delta x_c = \Delta x\left(\frac{n-1}{2}\right) \tag{62}$$

Using these definitions it is possible to derive the kernel stability stated in equation (63) where the variables are as follows: n is the number of nodes covered by the Gaussian function, j is the parameter controlling the dilation and R is the ratio between the value at a node in the tails of the function to the significance of the peak of the Gaussian function.

$$n = 1 + \sqrt{\frac{-2\ln R}{\pi}} * 2^{j+1} \tag{63}$$

Theoretically the number of nodes should be at least two in order to maintain the variable connectivity window arrays. This is verified analytically through solutions that were run with values as low as $-2{\cdot}5$ for j when the entire domain is used as the support of the Gaussian function, and as low as $-2{\cdot}2$ when the support of the Gaussian function is limited. Solutions for the number of nodes in the support of a window function for various height ratios are shown in the Table II.

Table II. The numbers of nodes in the support of a Gaussian window function

j	$n(2\sigma)$	$n(3\sigma)$
−2·2	1·69	2·04
−2	1·80	2·20
−1	2·60	3·39
0	4·19	5·79
1	7·38	10·57
2	13·77	20·15
3	26·53	39·30

Note: 2σ and 3σ correspond to $R = 1{\cdot}83\text{E} - 2$ and $R = 1{\cdot}23\text{E} - 4$ respectively

If we examine equations (31)–(33) carefully, the stability condition is governed by the determinant of **C**:

$$\text{Det} = \bar{m}_0 \bar{m}_{11} - \bar{m}_1^2 > 0 \tag{64}$$

Therefore, when the trapezoidal rule is used to integrate the moments, instability occurs when

$$\text{Det} = \bar{m}_0 \bar{m}_{11} - \bar{m}_1^2 \leqslant 0 \tag{65}$$

Equations (64) and (65) provide a very non-linear stability relationship between r and Δx. Our numerical experiments have confirmed these stability conditions. The numerical values of Δx also coincide with those predicted from equation (5) which is explained by Liu *et al.*[10]

5.2. *Critical time step*

In order to perform analyses on the structural dynamic class of problems, it is very important to understand the relationship between this new shape function and the critical time step. This is determined here by performing the standard eigenvalue analysis, solving the determinant of $[\mathbf{K} - \lambda\mathbf{M}]$. This calculation only needs to be performed for each quadrature point since the eigenvalue of a matrix sum is bounded by the eigenvalues of the constituent matrices. The critical time step was found to depend on several parameters including the dilation parameter and the boundary correction function.

Using symbolic manipulation this determinant was solved for RKPM shape functions containing several different numbers of nodes. By evaluating the determinant at each Gaussian point it was found that there is only one non-zero eigenvalue, regardless of the number of nodes in the support of the shape function. It was possible to simplify the result for equally spaced nodes to the following terms:

$$D(x,s) = C_1(x) + C_2(x)\left(\frac{x-s}{r}\right) \tag{66}$$

$$D'(x,s) = C_1'(x) + C_2'(x)\left(\frac{x-s}{r}\right) + \frac{C_2(x)}{r} \tag{67}$$

$$\lambda_{\max} = \left(\frac{c^2}{l^2}\right)\frac{\Delta x}{r}\sum_{i}^{\text{NP}} \frac{[D_i\Phi'_{xi} + D'_i\Phi_{xi}]^2}{D_i\Phi_{xi}}\Delta x_i \tag{68}$$

Table III. The critical time steps calculated using equation (70) for cubic spline window functions

j	Δt (boundary)	Δt (interior)	Δt (Numeric)
0·0	$0{\cdot}89 \times 10^{-5}$	$1{\cdot}51 \times 10^{-5}$	$2{\cdot}7 \times 10^{-5}$
0·5	$0{\cdot}89 \times 10^{-5}$	$2{\cdot}78 \times 10^{-5}$	$3{\cdot}3 \times 10^{-5}$
1·0	$1{\cdot}75 \times 10^{-5}$	$3{\cdot}89 \times 10^{-5}$	$4{\cdot}2 \times 10^{-5}$
1·5	$2{\cdot}73 \times 10^{-5}$	$5{\cdot}55 \times 10^{-5}$	$5{\cdot}7 \times 10^{-5}$

where Δx is the length between nodes, Δx_i is the integration weight using trapezoidal rule and NP is the number of nodes in the shape function's support. The critical time step is shown for a central difference time integration scheme in equation (69). By substituting the maximum eigenvalue corresponding to the maximum frequency, the critical time step can be calculated:

$$\Delta t \leqslant \frac{2}{\omega_{\max}} \tag{69}$$

$$\Delta t \leqslant \frac{2l}{c}\left(\frac{r}{\Delta x}\right)^{1/2}\left[\sum_{i}^{\mathrm{NP}} \frac{[D_i\Phi'_{xi} + D'_i\Phi_{xi}]^2}{D_i\Phi_{xi}}\Delta x_i\right]^{-1/2} \tag{70}$$

Results for the critical time step were calculated using equation (70) for cubic spline window functions. The results are shown in Table III for the problem in Figure 8 with 21 equispaced nodes. The critical time steps obtained near boundaries are approximately half of those associated with interior because of the correction function $D(x, x_j)$. The finite element critical time step for a linear displacement element is $\Delta t \leqslant l/c$ is $2{\cdot}455 \times 10^{-5}$ s.

The discretized form of the critical time step was perceived to have a simple translation to the continuous form. The discretized form contains the derivative of the reproducing kernel squared. Then the largest eigenvalue and thus the critical time step is as follows:

$$\lambda = \left(\frac{c^2}{l^2}\right)\frac{\Delta x}{r}\int_{B(x)} \frac{\left[D(x,y)\,\Phi'\left(\dfrac{x-y}{r}\right) + D'(x,y)\,\Phi\left(\dfrac{x-y}{r}\right)\right]^2}{D(x,y)\,\Phi\left(\dfrac{x-y}{r}\right)}\mathrm{d}y \tag{71}$$

$$\lambda \equiv \left(\frac{c^2}{l^2}\right)\frac{\Delta x}{r}\int_{B(x)} \frac{k_{,x}(x,y)^2}{k(x,y)}\mathrm{d}y \tag{72}$$

where $k(x, y)$ is the reproducing kernel defined by equation (73):

$$k(x,y) = D(x,y)\,\Phi\left(\frac{x-y}{r}\right) \tag{73}$$

6. NUMERICAL EXPERIMENTS

A one-dimensional wave propagation problem was analysed using the reproducing kernel particle method. The problem is shown in Figure 8 with material properties (for steel) $E = 3 \times 10^7$ psi, $\rho = 7{\cdot}24 \times 10^{-4}$ slugs/in^3 and $A = 1$ in^2.

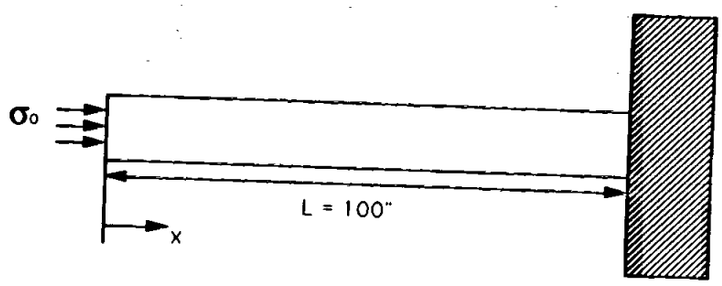

Figure 8. 1-D bar problem with step input

The solution for the linear elastic deforming rod was obtained with the Gaussian shape function and an explicit Newmark beta predictor/corrector algorithm. The results correspond accurately to the closed form solution. For non-linear and large deformation problems, the cubic spline function was selected as the window function.

In order to visualize the effect of the dilation parameter, the shape functions are plotted in Figure 10 along with the nodes which depict the number of significant nodes providing the support of each shape function in each dilation (technically every node is in the support of a Gaussian shape function). This is shown directly in Figure 11 of the plot for velocity versus time which incorporated the shape functions in Figure 10 to obtain this response. Figure 9 also shows the displacements for the node 11 for different values of j. Figures 9 and 11 are obtained when the external load is compression; 21 nodes were used for the elastic problem. For completeness the stresses for the case of tensile load rather than compression are also plotted in Figures 12 and 13. For the results of the RKPM, the stress is calculated at the midpoint of the element 10.

Although the high frequency components of the solution in Figure 14 are filtered away, it is interesting to note that the problem was run at 1·35 times the FEM critical time step.

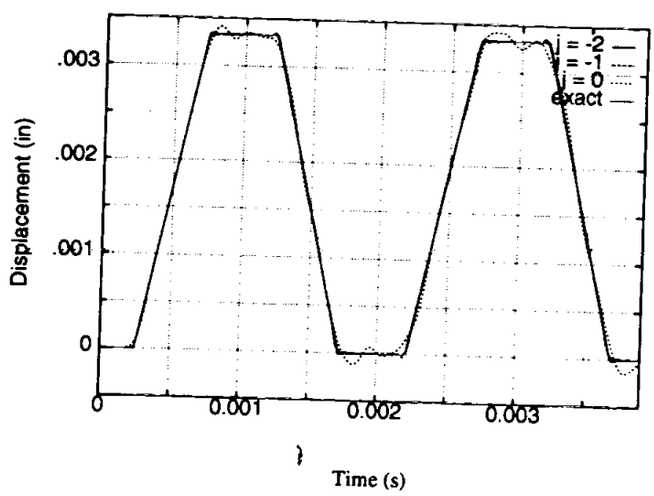

Figure 9. Displacement for node 11 (centre node) showing a range for the dilation parameter and the exact solution for the elastic problem

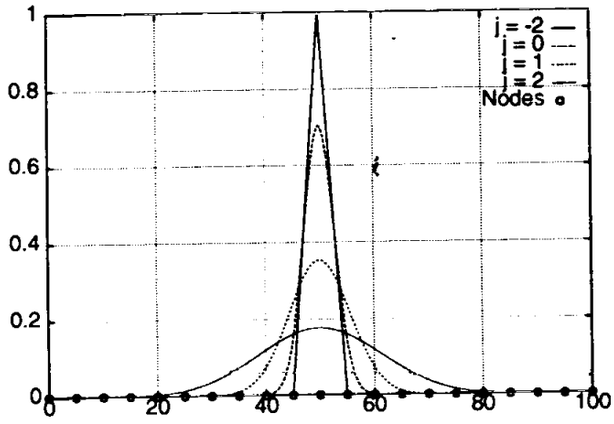

Figure 10. Gaussian shape functions used to calculate the wave propagation, with nodal support

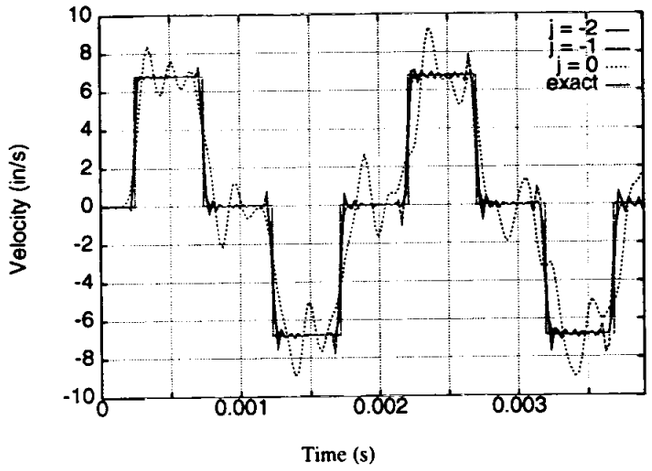

Figure 11. Velocity for node 11 (centre node) showing a range for the dilation parameter and the exact solution for the elastic problem

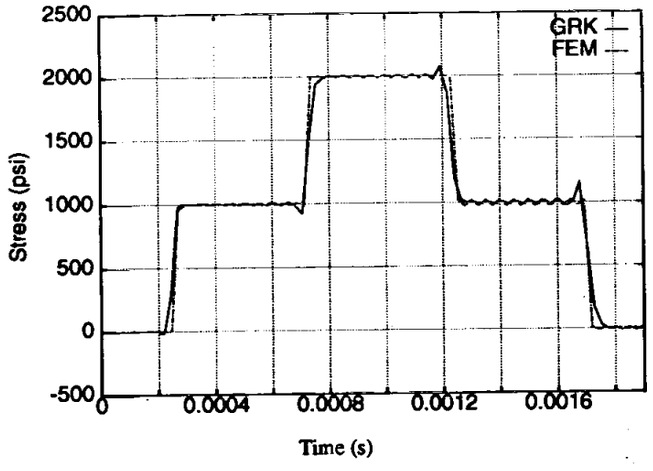

Figure 12. The axial stress at element 10 is plotted versus time for tensile load

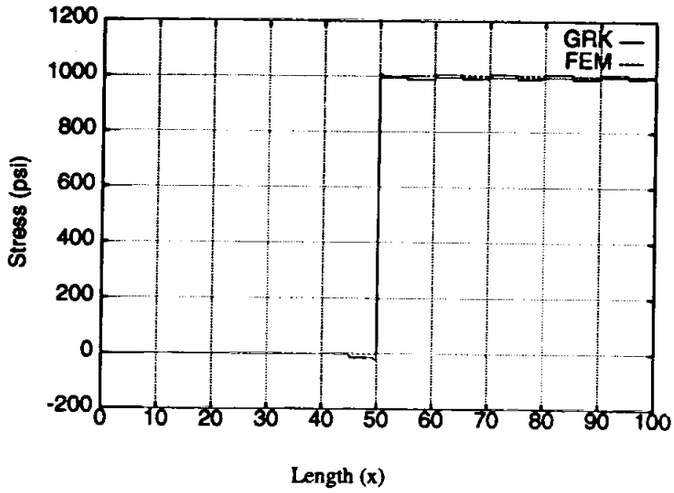

Figure 13. Axial stress at the tenth time step as a function of position for tensile load

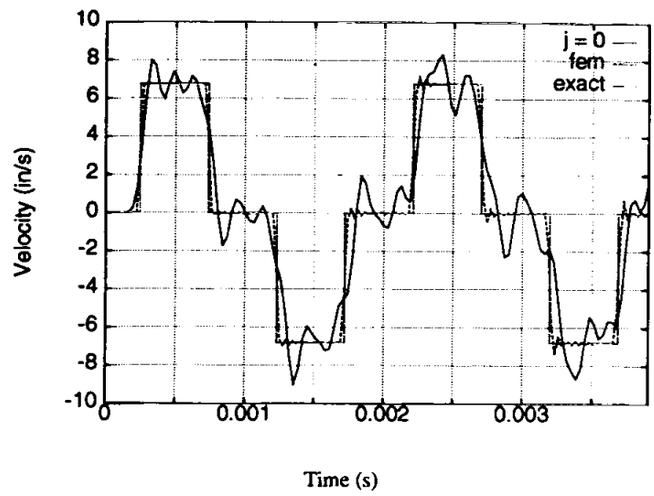

Figure 14. The RKPM run with $j = 0$ and a time step of $3{\cdot}32 \times 10^{-5}$ which is 1·35 times the FEM critical time step for the elastic problem

Figures 15 and 16 depict the stress at $x = 49{\cdot}375$ ins. of the bar which is modelled with the elastic/plastic materials under compression: yield stress $\sigma_y = 3 \times 10^4$ psi; $E = 3 \times 10^7$ psi; $\rho = 7{\cdot}24 \times 10^{-4}$ slugs/in^3; $E_p = E/4$; $\sigma_0 = 5 \times 10^4$ psi. It is also shown in Figure 16 that the RKPM can run with a larger time step ($1{\cdot}0 \times 10^{-5}$ s), 1·626 times the critical time step of FEM ($0{\cdot}615 \times 10^{-5}$ s). For the elastic–plastic problem, the cubic spline functions are selected for the window functions as given in equations (36)-(38) with $j = 0{\cdot}0$ (Figure 15) and $j = 1{\cdot}0$ (Figure 16).

For the large deformation problem, $\sigma_0 = 2{\cdot}5 \times 10^5$ (psi) replaces the previous external stress distribution while other data remain the same. Due to movement of the nodes, a flexible window

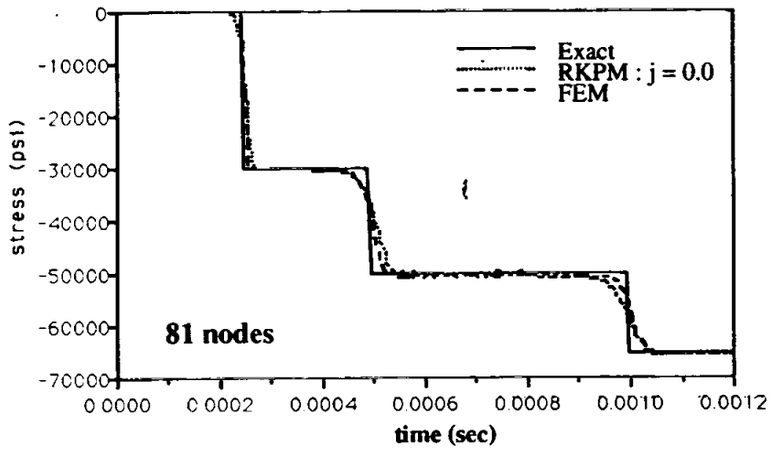

Figure 15. Stress of the elastic–perfectly elastic bar with time step Δt = 0·6E − 5 (s)

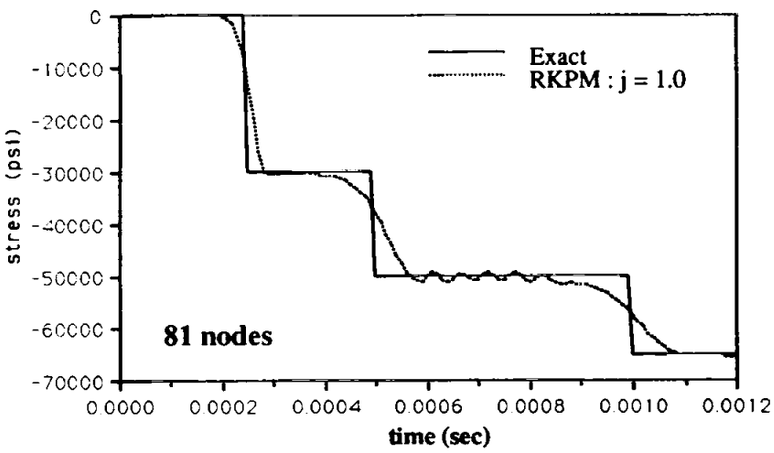

Figure 16. The RKPM with a larger time step Δt = 1·0E − 5 (s) which is 1·626 times the FEM critical time step (Δt = 0·615E − 5)

based on the average nodal space is employed for $j = 0{\cdot}5$. Figures 17 and 18 show the stress responses at x = 49·375 ins. of RKPM using 81 nodes with a time step of $0{\cdot}5 \times 10^{-5}$ s for the cases of compression and tension respectively. FEM results using 641 nodes are included for referential data to compare with.

Two-dimensional large deformation examples are illustrated in Figures 19–21. The materials for the 2-D problems are Mooney–Rivlin rubber materials. An explicit Runge–Kutta method is employed for the time integration. Figure 19 shows results of selected time steps when a 2-D Mooney–Rivlin rubber bar is compressed with a pressure of 200 psi while Figure 20 simulates the bar subject to a tension of 100 psi. For both compression and tension examples, a time step of $0{\cdot}4 \times 10^{-6}$(s) is used and the domain $[x_0, x_L] \times [y_0, y_L]$ is $[0{\cdot}0, 8{\cdot}0] \times [-0{\cdot}5, 0{\cdot}5]$. In these

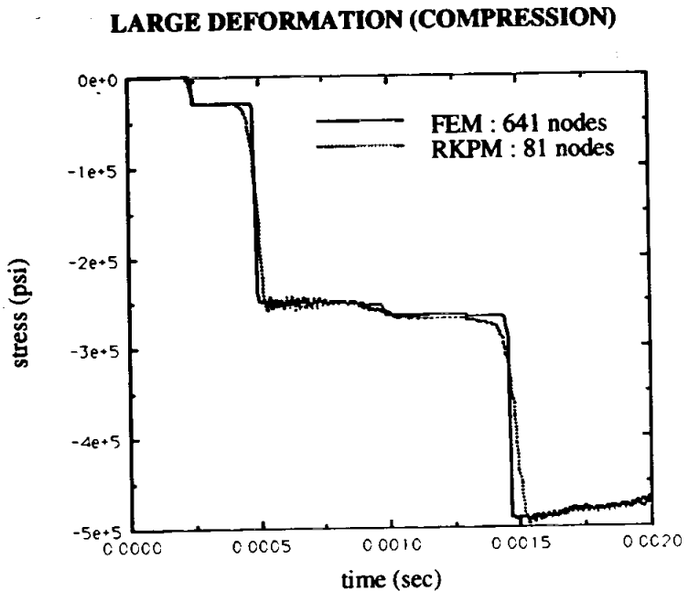

Figure 17. Stress response of RKPM and FEM for the large deformation compression problem of a 1-D bar

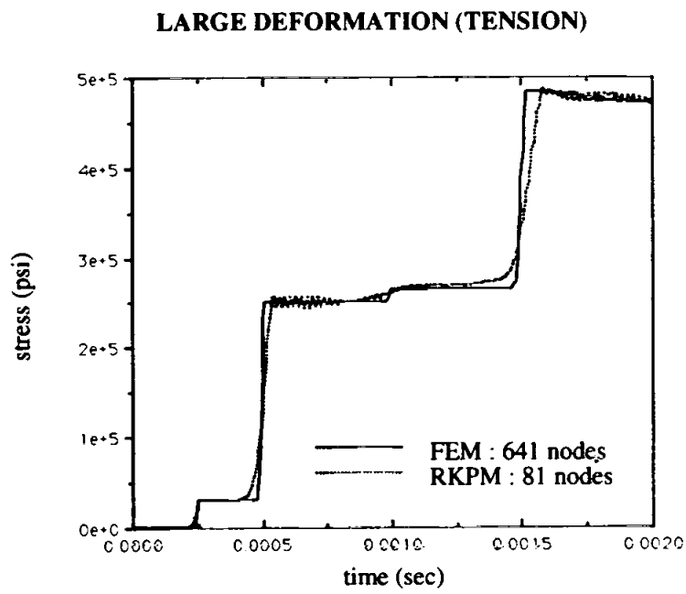

Figure 18. Stress responses of RKPM and FEM for the large deformation tension problem of a 1-D bar

figures, the solid dots depict the positions of the particles. Similar to the 1-D example, we do not observe any tensile instability. Figure 21 is the tunnel-like structure problem subjected to a distributed load of 1000 psi on the top portion of the structure. The time step used is $0{\cdot}5 \times 10^{-6}$ (s). The upper portion of the structure experiences compression while the lower

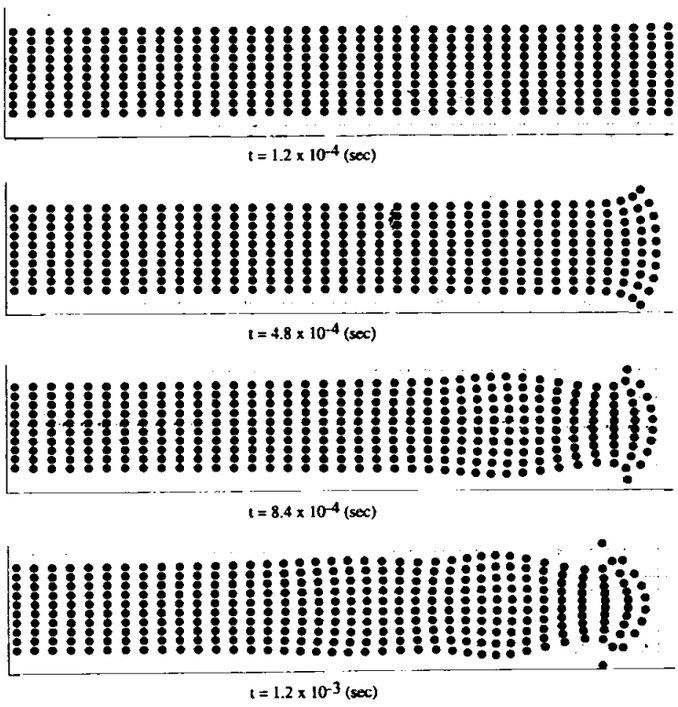

Figure 19. Compression of Mooney–Rivlin rubber bar: 37×10 nodes; $C_1 = 2000{\cdot}59$ (psi); $C_2 = 200{\cdot}367$ (psi); $\lambda = 2{\cdot}0 \times 10^6$ (psi); $\rho = 2{\cdot}4 \times 10^{-4}$ (slug)

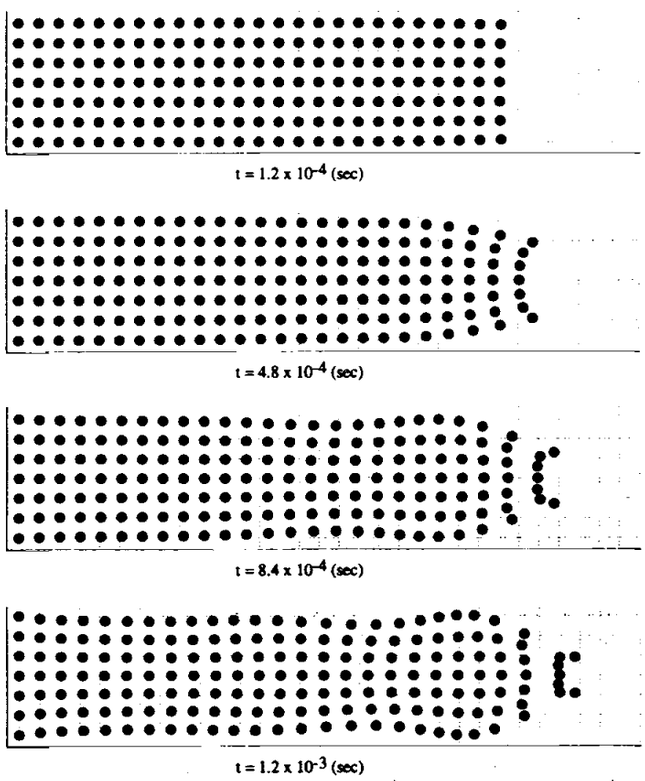

Figure 20. Tension of Mooney–Rivlin rubber bar: 41×10 nodes; $C_1 = 1800{\cdot}35$ (psi); $C_2 = 146{\cdot}8$ (psi); $\lambda = 1{\cdot}468 \times 10^6$ (psi); $\rho = 1{\cdot}4089 \times 10^{-4}$ (slug)

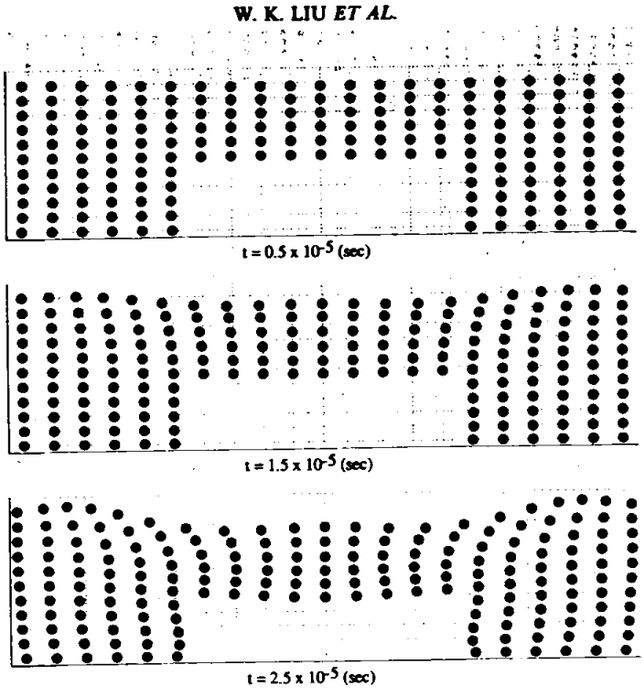

Figure 21. Compression of tunnel structure of Mooney–Rivlin rubber materials: 186 nodes; $C_1 = 1835$ (psi); $C_2 = 146{\cdot}8$ (psi); $\lambda = 1{\cdot}468 \times 10^6$ (psi); $\rho = 1{\cdot}4089 \times 10^{-4}$ (slug)

portion is in tension. This example demonstrates that, with the correction function, the tensile instability phenomenon of SPH methods has been eliminated.

7. CONCLUSIONS

The theory for reproducing kernel particle methods is presented here with numerical experiments performed to confirm the derived equations. The RKPM is proven here to possess the ability to solve a dynamic problem. Results are also presented to verify the supposition that the correction function can provide both boundary correction and reproducing kernel stability. Furthermore the implementation of the flexible window function as frequency control has been initiated.

The results from the numerical experiments not only verified the theory presented, but it produced several encouraging results. Among the most important is the ability of the RKPM to perform at time steps larger than the critical time step for standard finite elements, as predicted by a simple formula derived in this paper. It is also important that results for the correction function proved that it provided boundary correction as well enhancing the stability of the reconstruction equation. The increased stability of the reconstruction equation enables the ability to use very few particles and obtain good accuracy. The correction function increased the stable range of the dilation parameter over the SPH method. In addition, the RKPM removes the so-called tensile instability associated with SPH methods is a large deformation analysis.

ACKNOWLEDGEMENTS

The support of this research by AFSOR grant number F49620-92-J-0505, NSF grant number MSS-9015978 and ONR grant number N00014-93-I-0292 to Northwestern University is gratefully acknowledged.

REFERENCES

1. B. Nayroles, G. Touzot and P. Villon, 'Generalizing the finite element method: diffuse approximation and diffuse elements', *Comput. Mech.*, **10**, 307–318 (1992).
2. T. Belytschko, Y. Y. Lu and L. Gu, 'Element free Galerkin methods', *Int. j. numer. methods eng.*, **37**, 229–256 (1994).
3. T. Belytschko, Y. Y. Lu and L. Gu, 'A new implementation of the element free Galerkin method', *Comput. Methods Appl. Mech. Eng.*, **113**, 397–414 (1994).
4. T. Belytschko, L. Gu and Y. Y. Lu, 'Fracture and crack growth by element free Galerkin Methods', *Modelling Simul. Mater. Sci. Eng.*, **2**, 519–534 (1994).
5. P. Lancaster and K. Salkauskas, 'Surfaces generated by moving least squares methods', *Math. Comput.*, **37**, 141–158 (1981).
6. R. A. Gingold and J. J. Monaghan, 'Smoothed particle hydrodynamics: theory and application to non-spherical stars', *Mon. Not. Roy. Astron. Soc.*, **181**, 375–389 (1977).
7. J. J. Monaghan, 'Why particle methods work', *SIAM J. Sci. Stat. Comput.*, **3**, 422–433 (1982).
8. J. J. Monaghan, 'An introduction to SPH', *Comp. Phys. Comm.*, **48**, 89–96 (1988).
9. W. K. Liu and C. Oberste-Brandenburg, 'Reproducing kernel and wavelet particle methods', in J. P. Cusumano, C. Pierre and S. T. Wu (eds.), *Aerospace Structures: Nonlinear Dynamic and System Response*, AD 33, ASME, 1993, pp. 39–56.
10. W. K. Liu, S. Jun and Y. F. Zhang, 'Reproducing kernel particle methods', *Int. j.numer. methods fluids*, accepted for publication.
11. W. K. Liu, J. Adee and S. Jun, 'Reproducing kernel particle methods for elastic and plastic problems', in D. J. Benson, and R. A. Asaro (eds.), *Advanced Computational Methods for Material Modeling*, AMD 180 and PVP 268, ASME, 1993 pp. 175–190.
12. L. D. Libersky and A. G. Petschek, 'Smooth particle hydrodynamics with strength of materials', in *Advances in the Free-Lagrange Method, Lecture Notes in Physics*, Vol. 395, 1990, pp. 248–257.
13. S. W. Attaway, M. W. Heinstein, F. J. Mello and J. W. Swegle, 'Coupling of smooth particle hydrodynamics with pronto', preprint, 1993.
14. L. D. Libersky, A. G. Petschek, T. C. Carney, J. R. Hipp and F. A. Allahdadi, 'High strain Lagrangian hydrodynamics—a three-dimensional SPH code for dynamic material response', *J. Comput. Phys.*, **109**, 67–75 (1993).
15. G. R. Johnson, private communication, Berlin, 1993.
16. C. K. Chui, *An Introduction to Wavelets*, Academic Press, New York, 1992.
17. W. K. Liu, Y. Zhang and M. R. Ramirez, 'Multiple scale finite element methods'. *Int. j. numer. methods eng.*, **32**, 969–990 (1991).
18. I. Daubechies, *Ten Lectures on Wavelets*, CBMS/NSF Series in Applied Mathematics, No. 61, SIAM, Philadelphia, PA, 1992.
19. W. K. Liu and Y. J. Chen, 'Wavelet and multiple scale reproducing kernel methods', *Int. j. numer. methods fluids*, accepted for publication.
20. T. J. R. Hughes and W. K. Liu, 'Implicit–explicit finite elements in transient analysis', *J. Appl. Mech. ASME*, **45**, 371–378 (1978).