Simple modifications for stabilization of the finite point method

B. Boroomand[1,‡,§], A. A. Tabatabaei[1,¶] and E. Oñate[2,*,†,‖]

[1]*Civil Engineering Department, Isfahan University of Technology, Isfahan, Iran*
[2]*International Center for Numerical Methods in Engineering (CIMNE),
Universidad Politécnica de Cataluña, 08034 Barcelona, Spain*

SUMMARY

A stabilized version of the finite point method (FPM) is presented. A source of instability due to the evaluation of the base function using a least square procedure is discussed. A suitable mapping is proposed and employed to eliminate the ill-conditioning effect due to directional arrangement of the points. A step by step algorithm is given for finding the local rotated axes and the dimensions of the cloud using local average spacing and inertia moments of the points distribution. It is shown that the conventional version of FPM may lead to wrong results when the proposed mapping algorithm is not used.

It is shown that another source for instability and non-monotonic convergence rate in collocation methods lies in the treatment of Neumann boundary conditions. Unlike the conventional FPM, in this work the Neumann boundary conditions and the equilibrium equations appear simultaneously in a weight equation similar to that of weighted residual methods. The stabilization procedure may be considered as an interpretation of the finite calculus (FIC) method. The main difference between the two stabilization procedures lies in choosing the characteristic length in FIC and the weight of the boundary residual in the proposed method. The new approach also provides a unique definition for the sign of the stabilization terms. The reasons for using stabilization terms only at the boundaries is discussed and the two methods are compared.

Several numerical examples are presented to demonstrate the performance and convergence of the proposed methods. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: finite point method; stabilized method; meshless method

*Correspondence to: E. Oñate, International Center for Numerical Methods in Engineering (CIMNE), Universidad Politécnica de Cataluña, 08034 Barcelona, Spain.
†E-mail: onate@cimne.upc.es
‡E-mail: boromand@cc.iut.ac.ir
§Associate Professor in Civil Engineering (IUT).
¶Graduate student.
‖Professor in Civil Engineering (CIMNE).

Copyright © 2005 John Wiley & Sons, Ltd.

1. INTRODUCTION

Advances in computer science and technology in the past three decades have lead to development of robust numerical methods such as finite element (FE) and finite volume (FV) methods in computational mechanics. Although these two methods are still receiving considerable attention and every year thousands of papers are published, the idea of using simpler methods is also the subject of many research works. This is mainly because the mesh generation part of the solution has shown to be a very time consuming challenge especially in application of FE and FV methods to three-dimensional problems. In this way, the idea of developing methods requiring no mesh has led to emerging of a new class of the so-called 'Meshless' methods.

A survey in the literature shows that the first application of the meshless methods, proposed by Lucy [1], was in the solution of astrophysical problems. The method, called smooth particle hydrodynamics (SPH), was further developed and studied in some research works by Monaghan [2, 3]. A fast growing attention on the subject can be seen in the 1990s when a number of methods were proposed. A good survey, up to the date, can be found in Reference [4].

The term 'meshless' has been used to convey the idea of using no grid of elements. Some early versions of the method uses background meshes to perform the integration required in the solution and thus are not 'truly meshless' methods. Considerable attempts to propose truly meshless methods were made during the past decade. Today, researches are convinced that the approach is useful for some class of problems, e.g. crack propagation, for which the finite element solution requires subsequent remeshing.

Meshless methods can generally be classified in two major categories. The first one is based on a weak integral form of the governing equations and thus requires a suitable integration scheme. Meshless methods based on collocation schemes can be included in the second category. For example the finite difference method on irregular grids falls within the latter category.

Among the methods using the weak formulation we find the diffuse element (DF) method proposed by Nayroles *et al.* [4]. In DF the unknowns are approximated through a moving least square (MLS) procedure and an integral form of the differential equations are satisfied in a Galerkin form of weighting.

Belytschko and co-workers followed a rather similar approach called element free Galerkin (EFG) method [5]. The main difference between DE and EFG lies in the computation of the derivatives of the functions. In the DE method, some terms of derivatives of the shape functions are ignored while in the EFG full derivatives of shape functions are used which leads to more accurate results. In both methods, numerical integration requires a mesh of quadrature points in the background of the domain and therefore they are not truly meshless methods.

In a series of papers written by Liu and co-workers, a new concept, referred to as reproducing kernel particle method (RKPM), for producing consistent base functions for the interpolation of unknowns was presented [6–10]. On the same line, the partition of unity (PU) method proposed by Babuška and Melenk should also be mentioned [11, 12]. Application of *hp*-cloud method, proposed by Duarte and Oden, on a grid of points is also considered as a mesh-less approach [13]. Other meshless methods, called meshless local Petrov–Galerkin (MLPG) and local boundary integral equation (LBIE), using moving least square approximations were proposed by Zhu *et al.* [14] and Atluri and Zhu [15, 16], respectively. On the same basis, De and Bathe presented the method of finite spheres [17].

In the context of collocation methods early research by Jensen in 1972, as a promotion of the finite difference method, can be categorized within the class of meshless methods [18].

In this work the derivatives of a function are approximated in terms of its nodal values by means of two-dimensional truncated Taylor series. This was an important step towards approximation of functions on irregular grids of nodes.

Another finite difference method on arbitrary grids was proposed by Perrone and Kao in 1975 [19]. Again, truncated Taylor series were used for approximating the unknown function. A specific procedure for selection of a set of neighbouring nodes in the cloud was introduced to guarantee the accuracy and stability of the results in the least square procedure.

One of the major drawbacks of the FD method on irregular grids is the instability of the results. Thus, the convergence of the solution is highly sensitive to the geometry of the cloud and to the selection of the nodes around the master node. Stabilization of the FD method has therefore been the goal of several research works. Demkowicz and co-workers published an article in which a new criteria was proposed for selection of neighbouring nodes [20]. It was proven that satisfying such criteria leads to convergent results when the FD method is applied on irregular grids. Another attempt for stabilizing the method was made by Liszka and co-workers, in 1996, who developed a new version of the FD method called *hp-Meshless cloud* method [21]. This method uses moving least squares (MLS) to fit a truncated Taylor series on a set of nodal values. To solve the instability problem, a new version of the moving least square approximation (Hermitian type of approximation) was developed in which the derivatives of the unknown function in the normal direction to the boundary were considered as additional variables for the boundary nodes. This technique improves the quality of the approximation in the vicinity of the boundaries.

The finite point method (FPM), proposed by Oñate *et al.* in 1996 for solution of fluid flow problems [22], uses a weighted least square (WLS) scheme for approximating the unknown function. The governing differential equations and the boundary conditions are satisfied using point collocation. In this research an upwinding approach was used as stabilization procedure. The application of the method to advective-diffusive fluid problems can be found in Reference [23]. A residual stabilization technique was later used as another alternative by the same authors [24]. The technique was a basis for development of a stabilized version of the FPM using Finite Calculus (FIC) proposed in 1998 [25]. The FIC method is not only useful for stabilization of results in FP, FD and FE methods but also has a wide range of applications in computational mechanics [26].

An overall view of the possibilities of the finite point method to incompressible flow problems was presented in Reference [27]. The reader is referred to a recent work performed by Löhner and co-workers for compressible flow problems as another attempt for achieving stabilized results using the finite point method [28].

For elasticity problems, Oñate *et al.* presented a finite point method using the FIC technique in order to overcome the instability of the results obtained from the conventional version of FPM [29, 30]. This stabilization procedure dramatically improves the convergence and accuracy of the method.

Although meshless methods based on weak form formulations are shown to be robust, they are very time consuming due to the integration required in the formulation. On the other hand, meshless methods based on point collocation, though not fully stable, are very fast and have received considerable attention. In this paper we shall focus on the finite point method and propose two simple remedies to overcome the instability problems.

The layout of the paper is the following. After a brief review of the least square procedures, we propose a mapping scheme to reduce the problem of ill-conditioning of the coefficient

matrix which sometimes occurs due to non-isotropic arrangement of the points. The procedure of polynomial fitting will be followed by a modified version of the method consistent with the mapping scheme and suitable for application of the Dirichlet boundary conditions.

In the section of solution methods, we shall propose a simple correction, inspired from the general form of the weighted residual approach. The method is in fact a reinterpretation of the finite calculus procedure using a special form of the characteristic length. In a detailed discussion we compare the similarities between the two methods. Several examples are given to demonstrate the performance of the proposed FP methods.

2. APPROXIMATION METHODS

In this section we shall give an overview to the approximation used to construct the base functions. The first step is choosing an appropriate polynomial.

2.1. The polynomial used

As mentioned in the previous section, the first stage for the numerical solution of differential equations is the approximation of the unknown function in terms of a set of nodal values. One of the simplest and oldest approximation methods, first used in finite difference methods on irregular grids, is based on truncated Taylor series.

Let the function $u(\mathbf{x})$ be continuously differentiable up to the required order. Its expansion in a two-dimensional space can be written as

$$u(\mathbf{x}) = u(\mathbf{0}) + x\frac{\partial u}{\partial x}(\mathbf{0}) + y\frac{\partial u}{\partial y}(\mathbf{0}) + \frac{1}{2}x^2\frac{\partial^2 u}{\partial x^2}(\mathbf{0}) + xy\frac{\partial^2 u}{\partial x \partial y}(\mathbf{0}) + \frac{1}{2}y^2\frac{\partial^2 u}{\partial y^2}(\mathbf{0}) + \cdots \quad (1)$$

Here x and y are suitable local co-ordinates. The function might be approximated by a finite number of terms of a Taylor series as follows

$$u(\mathbf{x}) \cong \hat{u}(\mathbf{x}) = u(\mathbf{0}) + x\frac{\partial u}{\partial x}(\mathbf{0}) + y\frac{\partial u}{\partial y}(\mathbf{0}) + \cdots + O\left(h^{p+1}\right) \quad (2)$$

where h is a measure of the average spacing between nodes and p is the order of the approximation. On view of (2) the unknown function may be approximated by a complete polynomial as

$$u(\mathbf{x}) \cong \hat{u}(\mathbf{x}) = \alpha_0 + \alpha_1 x + \alpha_2 y + \alpha_3 x^2 + \alpha_4 xy + \alpha_5 y^2 + \cdots \quad (3)$$

The coefficient values for a finite region around the ith master node are determined by imposing the following condition at a finite number of nodes

$$\hat{u}\left(\mathbf{x}_j\right) = u\left(\mathbf{x}_j\right) = \bar{u}_j; \quad j = 1, \ldots, n_i \quad (4)$$

where n_i is the number of nodes selected in the vicinity of the ith master node. Obviously, to obtain a square system of equations, n_i must, at least, be equal to the number of monomials in (3). Unfortunately, the application of the method on some grids may lead to a singular or ill-conditioned system of equations. Some of such arrangements are shown in Figure 1. To tackle the problem, a least square procedure with the number of nodal unknowns greater than
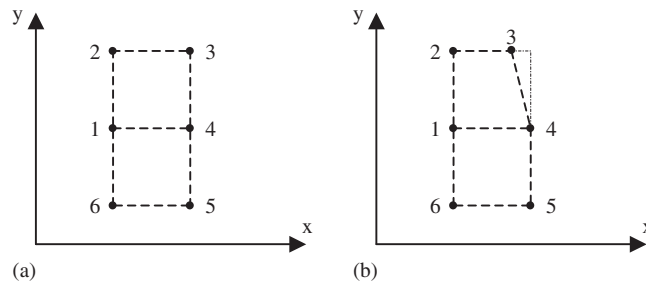
Figure 1. (a) Singular arrangement of nodes for fitting a quadratic polynomial;
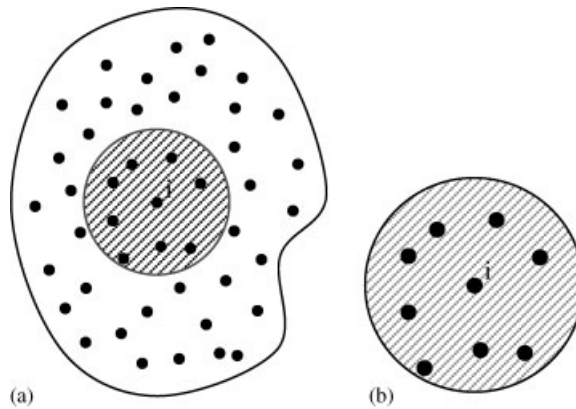and (b) ill-conditioned arrangement of nodes.

Figure 2. (a) Domain Ω with n nodes; and (b) sub-domain Ω_i with n_i nodes.

the number of monomials is usually employed. However even in such a least square fitting, there is not a guarantee to avoid an ill-conditioned matrix. A detailed description for this class of approximation methods is given in the next section.

2.2. Weighted least square method

Least square methods are useful procedures for approximating the unknown functions in mesh-less methods. The basis of the method are given next.

Assume that function $u(\mathbf{x})$ is to be approximated in domain Ω with n nodal points. The approximate function in sub-domain Ω_i in vicinity of the ith node with n_i neighbouring nodes (see Figure 2) may be written as Equation (3) or as

$$u(\mathbf{x}) \cong \hat{u}(\mathbf{x}) = \sum_{l=1}^{m} p_l(\mathbf{x})\alpha_l = \mathbf{p}^{\mathrm{T}}(\mathbf{x})\boldsymbol{\alpha} \tag{5}$$

Where $\mathbf{p}(\mathbf{x})$ is a vector of base monomials and $\boldsymbol{\alpha}$ is a vector of coefficients.

As mentioned in the previous section, the values of α_l for the ith cloud are determined from the satisfaction of (4). Assuming that generally n_i is greater than m, then satisfaction of (4) requires a least square procedure, which leads to minimization of the following discrete norm:

$$J = \sum_{j=1}^{n_i} w_i(\mathbf{x}_j)\big[\bar{u}_j - \hat{u}(\mathbf{x}_j)\big]^2 = \sum_{j=1}^{n_i} w_i(\mathbf{x}_j)\big[\bar{u}_j - \mathbf{p}^{\mathrm{T}}(\mathbf{x}_j)\boldsymbol{\alpha}\big]^2 \tag{6}$$

where $w_i(\mathbf{x})$ is a suitable weighting function for the ith cloud.

Note that for each cloud a local co-ordinate system is defined, the origin of which is located at the master node of the cloud (node number i). The approximate function for the cloud is obtained on such a co-ordinate system.

The discrete norm in Equation (6) is minimized as

$$\frac{\partial J}{\partial \boldsymbol{\alpha}} = 0 \tag{7}$$

which yields the following system of equations:

$$\mathbf{A}\boldsymbol{\alpha} = \mathbf{B}\bar{\mathbf{u}} \tag{8}$$

where

$$\mathbf{A} = \sum_{j=1}^{n_i} w_i(\mathbf{x}_j)\mathbf{p}(\mathbf{x}_j)\mathbf{p}^{\mathrm{T}}(\mathbf{x}_j) \tag{9}$$

and

$$\mathbf{B} = \big[w_i(\mathbf{x}_1)\mathbf{p}(\mathbf{x}_1), \ldots, w_i(\mathbf{x}_{n_i})\mathbf{p}(\mathbf{x}_{n_i})\big] \tag{10}$$

Solution of (8) gives

$$\boldsymbol{\alpha} = \mathbf{A}^{-1}\mathbf{B}\bar{\mathbf{u}} \tag{11}$$

Depending on the arrangement of the nodes and the weighting function used in the cloud, matrix $\mathbf{A}$ may become singular (see Figure 1 for instance). Here we will assume that $\mathbf{A}$ is regular so that $\mathbf{A}^{-1}$ is available. In case that the node arrangement is directional, matrix $\mathbf{A}$ may become ill-conditioned but with the aid of appropriate cloud dimensions, as suggested in the forthcoming subsections, this effect can be reduced.

The approximation for the ith cloud is obtained by substituting Equation (11) in (5) as

$$\hat{u}(\mathbf{x}) = \mathbf{p}^{\mathrm{T}}(\mathbf{x})\mathbf{A}^{-1}\mathbf{B}\bar{\mathbf{u}} = \mathbf{N}(\mathbf{x})\bar{\mathbf{u}} \tag{12}$$

where $\mathbf{N}(\mathbf{x})$ is a matrix containing the shape functions for each cloud.

The term 'cloud' refers here to a circular or rectangular area, with the centre at the master node, containing at least the minimum number of points required for the weighted least square (WLS) procedure. The way we allocate an area to a master node will be described later.

A primary cloud is chosen first with some selected points. However, the primary cloud is just used for further process in order to find the final configuration of the main cloud over which the least square procedure is performed. The selected points in the final cloud might be slightly different (due to aspect ratio) from those selected in the primary cloud. The weighting function is defined over the final configuration of the cloud.
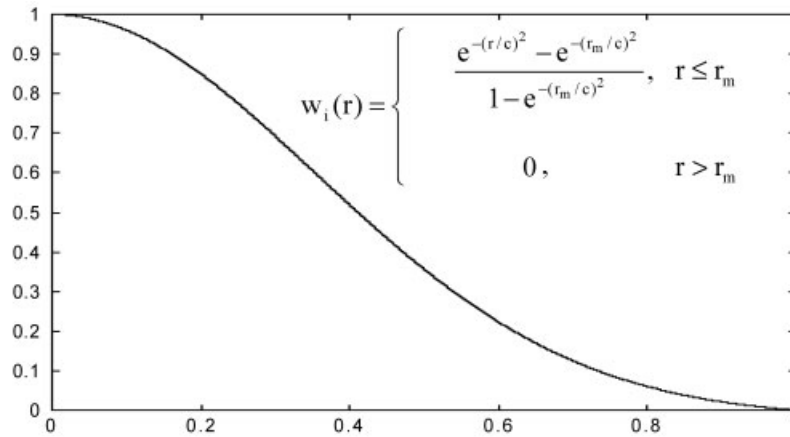
Figure 3. Normalized Gaussian weight function.

There is an important criterion for selecting the weighting functions in (6). The maximum value of the w_i should, preferably, occur at $\mathbf{x} = \mathbf{0}$ (the master node of the cloud) to capture the nodal value, or at least to obtain a sufficiently close value, to be used at the Dirichlet boundary condition. It might be argued about the necessity of assigning the maximum weight to the master node and some may prefer to let this happen at the vicinity of the master node. This assumption intuitively emerges from the fact that the approximate information at the master node is to be extracted, preferably, from the information available at the node, as a first priority, and from the rest of the points in the cloud with less priority as the weighting indicates. If the maximum weighting is assigned to other places, there will be no guarantee for such extraction of information when the point distribution around the master node is dense. However, what we shall explain in the forthcoming sections can be applied to such cases with no loose of generality.

In this paper we shall use a truncated Gaussian weighting function, as Figure 3 depicts, defined by

$$w(r) = \begin{cases} \dfrac{\exp(-r^2/c^2) - \exp(-r_m^2/c^2)}{1 - \exp(-r_m^2/c^2)}, & 0 \leqslant r \leqslant r_m \\ 0, & r > r_m \end{cases} \tag{13}$$

where r denotes the distance between the point and the master node. Also c and r_m denote two distances proportional to the distance between the master node and the most remote point in the cloud, i.e. $c = \lambda r_{\max}$ and $r_m = \rho r_{\max}$. The weighting function is, in fact, suitable for circular clouds. This weighting was suggested in Reference [5] and later used in many studies [15, 16, 22]. For rectangular clouds the product of two weighting functions written along the two principle directions may be used. It may be noticed that the weighting at the most distant node can be controlled by parameters c and ρ in the weight expression. This might be interpreted as choosing a larger fictitious area, containing the selected points (although it may cover more points than those selected). In this paper we have used $c = 0.25 r_{\max}$ and $r_m = 2 r_{\max}$.
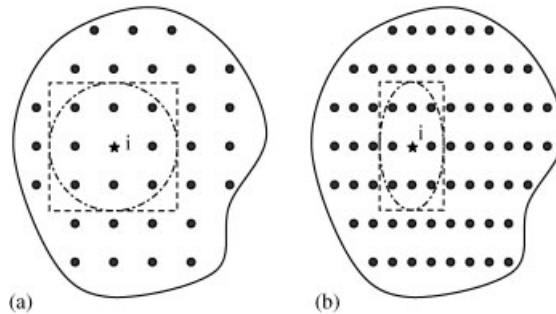
Figure 4. Cloud selection for a typical master node: (a) isotropic distribution of nodes; and (b) non-isotropic distribution of nodes.

It is obvious that since the approximation in the least square procedure is based on the minimization of an error norm, the approximate function does not have the selectivity property. In other words

$$\hat{u}(\mathbf{x}_j) \neq \bar{u}_j \tag{14}$$

This property creates an inconsistency to satisfy the Dirichlet boundary conditions and this is a source of instability when the meshless procedure is used with a weak formulation [6–10]. Although, for point collocation methods the deficiency has not been reported, we recommend to employ the modified form of the WLS method as defined in the next section.

2.3. Selection of clouds

Several cloud definitions may be given for a collection of nodes. In general, the geometry of the cloud should follow the distribution of the nodes. In other words, for isotropic distributions circular or square clouds are preferred, while for non-isotropic distributions the use of elliptic or rectangular clouds may give better results (Figure 4).

Here we shall briefly describe the cloud selection for both isotropic and non-isotropic distribution of points.

2.3.1. Isotropic point distribution. When the points are distributed with a uniform density a circular or square cloud may be used. The dimension of the cloud depends on the number of points required for polynomial fitting. For instance, if circular clouds are used, the procedure starts by choosing a minimum radius for each circle followed by a progressive increase of the radius until the required number of points, say twice as many as the monomial terms, is available. The dimension of the first circle and the rate of the enlargement are usually chosen by average spacing of the points.

A general adaptive solution, may start from an isotropic point distribution. As the point insertion goes on, a directional point distribution may be observed where the variation of the function is very high. Therefore, in order to generalize the application of any meshless method, it is essential to devise a suitable scheme for handling such arrangements of points.
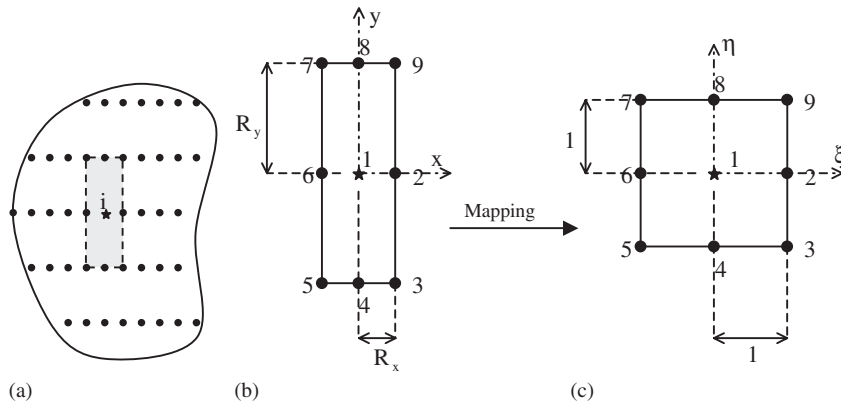
Figure 5. (a) Non-isotropic grid; (b) cloud of a typical node;
and (c) normalized cloud of the typical node.

2.3.2. Non-isotropic point distribution. When the point density is directional, the application of circular or square clouds may lead to an ill-conditioning or even singular coefficient matrix in Equation (9). Even in robust numerical solutions, such as the FEM, similar situations may happen when elements with high aspect ratio are used and the shape functions are defined in terms of global non-scaled co-ordinates. Nowadays, it is well understood that defining the shape functions in local normalized co-ordinates solves the problem. In this section, we propose a remedy for such situations in meshless methods, based on an analogy with the FEM.

Application of normalized clouds. As the anisotropy of the grid increases, matrix **A** in Equation (9) becomes ill-conditioned and the quality of the approximation deteriorates. In the classical least square procedure, i.e. with weights equal to unity, the elements in **A**, as is seen in (10), are proportional to the powers of the relative distances between the master node and the other nodes (recalling that the origin of the co-ordinates has been placed at the master node). In a WLS procedure, the weights in the expression of **A** may accelerate the dimension dependency of the solution. We note that when circular clouds are used this effect is worse than in rectangular clouds. The reason is that in circular clouds the weighting is defined in terms of the absolute maximum distance between nodes.

In order to prevent such undesirable effect, a mapping may be used to compute the values of the shape functions and their derivatives with a higher quality. A fictitious square domain is chosen as an intermediate cloud and all points within the main cloud are transferred to such fictitious isotropic sub-domain. The least square procedure is then performed on the normalized cloud. This may be interpreted as preconditioning the matrix **A** in a consistent manner.

Let us consider a cloud with directional point density. A local co-ordinate system ξ, η is chosen with origin at the master node where (see Figure 5)

$$\xi = \frac{x}{R_x}, \quad \eta = \frac{y}{R_y} \tag{15}$$

Here R_x and R_y denote maximum distances along x and y measured from the master node and the exterior nodes in the cloud. In Equations (9) and (10), **A** and **B** have now the

following forms in terms of the local co-ordinates:

$$\mathbf{A} = \sum_{j=1}^{n_i} w_i(\xi_j)\mathbf{p}(\xi_j)\mathbf{p}^{\mathrm{T}}(\xi_j), \quad \mathbf{B} = \left[w_i(\xi_1)\mathbf{p}(\xi_1), \ldots, w_i(\xi_{n_i})\mathbf{p}(\xi_{n_i}) \right]$$

where the weighting function is rewritten as

$$w(\xi) = \frac{\exp\left(-(\xi^2 + \eta^2)/c^2\right) - \exp\left(-\rho^2/c^2\right)}{1 - \exp\left(-\rho^2/c^2\right)}$$

we have used $c = 0.25$, $\rho = 2$ and as usual $-1 \leqslant \xi \leqslant 1$, $-1 \leqslant \eta \leqslant 1$. Note that we are using a circular weighting function on a square cloud.

The coefficient matrix $\mathbf{A}$ is not longer dependent on the cloud dimensions. The approximate function is also expressed in terms of the local co-ordinates as

$$\hat{u}(\xi) = \mathbf{p}^{\mathrm{T}}(\xi)\mathbf{A}^{-1}\mathbf{B}\bar{\mathbf{u}} = \mathbf{N}(\xi)\bar{\mathbf{u}} \tag{16}$$

provided again that $\mathbf{A}^{-1}$ exists, which means that the points should be properly distributed in the new square domain. If $\mathbf{A}$ is still singular, the dimensions of the main cloud are successively altered, and the procedure is repeated with new selected points and repetition of the mapping, until a non-singular coefficient matrix is obtained. If such strategy does not lead to a regular matrix, a point selection criteria, as used in Reference [19] or [20], might be employed within the normalized cloud. According to our experience, usually there is no need to use any point selection criteria.

Having found the base functions, the derivatives with respect to the global co-ordinates may be computed by the chain rule

$$\frac{\partial}{\partial x} = \frac{1}{R_x}\frac{\partial}{\partial \xi}, \quad \frac{\partial}{\partial y} = \frac{1}{R_y}\frac{\partial}{\partial \eta} \tag{17a}$$

or

$$\frac{\partial^2}{\partial x^2} = \frac{1}{R_x^2}\frac{\partial^2}{\partial \xi^2}, \quad \frac{\partial^2}{\partial x \partial y} = \frac{1}{R_x R_y}\frac{\partial^2}{\partial \xi \partial \eta}, \quad \frac{\partial^2}{\partial y^2} = \frac{1}{R_y^2}\frac{\partial^2}{\partial \eta^2} \tag{17b}$$

and so forth. The effect of such a mapping is illustrated in the following sample problem.

Sample problem. A rectangular cloud with 9 points, similar to Figure 5, is considered for the polynomial fitting. A second order complete polynomial with six unknown coefficients is to be fitted on nine values at the points. The least square procedure is performed using both non-normalized and normalized co-ordinates. It is desirable to recapture the unit value for the base function at the central node, as this will reproduce the nodal value after the fitting process. As an indication for the deterioration of the fitting process due to the aspect ratio dependency of matrix $\mathbf{A}$, in Table I the value obtained for the central base function of node 1 at the origin of the co-ordinates, i.e. at $x = y = 0$ is monitored for different aspect ratios. It can be seen that the quality of the base function deteriorates when the aspect ratio increases and a non-mapped cloud is used. On the other hand, when using a mapped cloud the quality remains unchanged. It is noteworthy that in this sample problem we have used a weighting function with circular support on a rectangular cloud. The circular support function, i.e. a circle with radius of $r_{\max}$, passes over the corners of the cloud.

Table I. Values of shape function at a master node with non-mapped and mapped clouds.

R_x/R_y	N(0, 0) (Nonmapped cloud)	N(0, 0) (Mapped cloud)
1.0	0.9677	0.9677
0.5	0.7546	0.9677
0.25	0.5516	0.9677

Fast deterioration of the fitting process quality is mainly due to the low values of the weights of points 3, 4, 5, 7, 8 and 9, in comparison with the weight of points 2 and 6, while R_x/R_y decreases. For instance for $R_x/R_y = 0.25$ the maximum weight for the six points at the upper and lower edges amounts to 2.9×10^{-7} while for points 2 and 6 the weight values amount to 0.39 which results in a ratio with logarithmic order of six. For a square cloud, the weight values at points 2, 4, 6 and 8 are equal to $w = 3.35 \times 10^{-4}$, which in comparison with the weight value at points 3, 5, 7 and 9 the ratio is of logarithmic order of three. In fact, when R_x/R_y decreases the procedure resembles to fitting a second order polynomial using information from just three points. This is a physical interpretation of what happens within the weighting least square procedure.

Looking into the sample problem in more detail, one may write Equation (9) in partitioned form as

$$\begin{bmatrix} A_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} \begin{Bmatrix} \alpha_0 \\ \boldsymbol{\alpha}_1 \end{Bmatrix} = \begin{Bmatrix} R_0 \\ \mathbf{R}_1 \end{Bmatrix}$$

where the first unknown α_0, corresponding to the approximation at master node, in Equation (3) is split from the remaining part of $\boldsymbol{\alpha}$. Now if the polynomial coefficients are determined as

$$\alpha_0 = \frac{1}{\left(A_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21}\right)} \left(R_0 - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{R}_1\right), \quad \boldsymbol{\alpha}_1 = \mathbf{A}_{22}^{-1}\left(\mathbf{R}_1 - \mathbf{A}_{21}\alpha_0\right)$$

it can be concluded that the accuracy and value of the approximation at the master node is dependent on the regularity of $\mathbf{A}_{22}$. Denoting the weight values at points 2 to 9 in Figure 5 as $w_2 = w_6 = a$, $w_4 = w_8 = b$ and $w_3 = w_5 = w_7 = w_9 = c$, a parametric form of matrix $\mathbf{A}_{22}$ for the above sample problem is

$$\mathbf{A}_{22} = \begin{bmatrix} 2(a+b+c)R_y^2h^2 & 2(a-b)R_y^2h & 0 & 0 & 0 \\ 2(a-b)R_y^2h & 2(a+b+c)R_y^2 & 0 & 0 & 0 \\ 0 & 0 & 2(a+b+c)R_y^4h^4 & 2(a-b)R_y^4h^3 & 2(a+b)R_y^4h^2 \\ 0 & 0 & 2(a-b)R_y^4h^3 & 2(a+b)R_y^2h^2 & 2(a-b)R_y^4h \\ 0 & 0 & 2(a+b)R_y^4h^2 & 2(a-b)R_y^4h & 2(a+b+c)R_y^4 \end{bmatrix}$$

where $h = R_x/R_y$. It can be seen that when h decreases matrix $\mathbf{A}_{22}$ becomes ill-conditioned, irrespective of the weight values (note that $a<1$, $b<1$ and $c<1$).

Also, even if the aspect ratio is not so large, the weight values can accelerate the ill-conditioning of $\mathbf{A}_{22}$ particularly when one of the weights, a or b, becomes much larger that the other two, i.e. $a \gg b \gg c$. In that case the third and fifth rows of above matrix tend to become a multiple of each other. This is the case, for instance, when $R_x/R_y = 0.25$ in which

a is of six logarithmic order larger than the other two weights. Of course, this is not the case for a square cloud in which both *a* and *b* are equal and are of just three logarithmic orders larger than *c*.

Clearly, the ill-conditioning may also happen in a square cloud if *c* (the weight value at the corner points in Figure 5) becomes very small. This means that we are fitting a second order polynomial on just five points. We note that the user can control the weight values at the corner points by allocating different values to the parameters in the expression of the weights given previously.

It can also be concluded that the effect of the aspect ratio on decreasing the quality of the least square procedure increases when higher order polynomials are used.

In above example we have just focused on values obtained at a master node as an indicator. The reader may notice that the 3×3 non-zero matrix in $\mathbf{A}_{22}$ includes coefficients of second order monomials, and therefore, when ill-conditioning happens all the corresponding coefficients are of poor accuracy. This leads to inaccurate results in problems with second order differential equations for instance, letting alone the inaccuracy of the solution at the Dirichlet boundary.

We also note that even if a modified version of WLS is employed so that the recovered value at the central point becomes unity, as recommended in the next subsection, the problem of ill-conditioning of the coefficient matrix remains unless a normalized cloud is used.

Before explaining the way the main cloud is selected, it is worthwhile to complete the mapping by a simple rotation in local normalized axes.

In case the point distribution is not along the global axes, the governing direction may be sought by finding the principal axes of the point pattern. This requires calculating the moments of inertia of the points assuming equal mass/weight for the points as

$$m_x = \sum (x - x_i)^2, \quad m_y = \sum (y - y_i)^2, \quad m_{xy} = \sum (x - x_i)(y - y_i) \tag{18}$$

The corresponding angle is then obtained as

$$\theta = \frac{1}{2} \tan^{-1}\left(\frac{2m_{xy}}{m_x - m_y}\right) \tag{19}$$

The first derivatives, for instance, in global co-ordinates may be evaluated by

$$\frac{\partial}{\partial x} = \frac{\cos\theta}{R_{x0}}\frac{\partial}{\partial\xi} - \frac{\sin\theta}{R_{y0}}\frac{\partial}{\partial\eta}, \quad \frac{\partial}{\partial y} = \frac{\sin\theta}{R_{x0}}\frac{\partial}{\partial\xi} + \frac{\cos\theta}{R_{y0}}\frac{\partial}{\partial\eta} \tag{20}$$

with $2R_{x0}$ and $2R_{y0}$ being the cloud dimensions in the new co-ordinates.

Now the question of the appropriate cloud selection must be addressed. For a very irregular grid of nodes, the cloud selection and mapping process may be summarized as follows (see Figure 6 for details):

- A number of the nearest points to the master node, say the minimum number required for the final least square procedure, are selected. The distance between the master node and the most remote point is calculated. A circular area is considered of radius equal to the calculated length. The area may cover more points than those selected at the beginning of the step. The circular area may be viewed as a primary cloud. It should be noted that the cloud chosen in this step is not the one on which the least square is performed. In fact, such a primary cloud is only intended for evaluating the local governing directions of the point distribution.
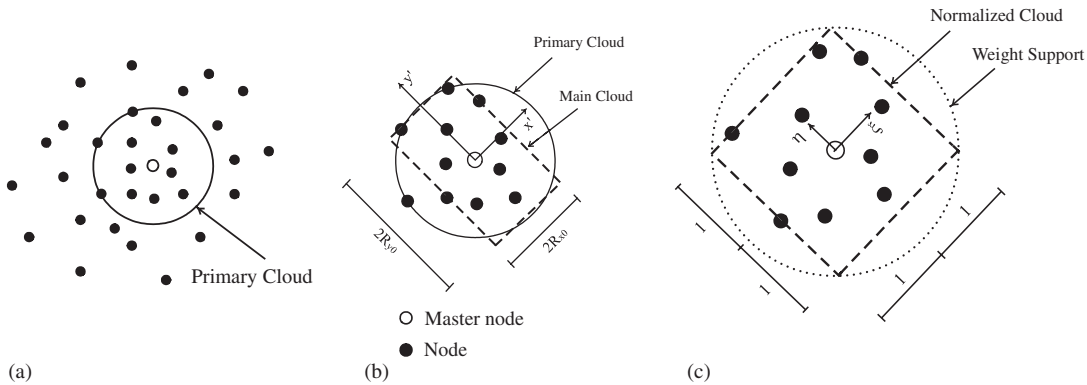
Figure 6. (a) A primary circular cloud with n_i points; (b) main cloud is built through finding the principal axes and proper aspect ratio; and (c) normalized cloud with circular weight support.

- The angle of the principal axes of the selected points is found through relations (18) and (19).
- The average spacing of the points is determined along the two new axes. This can be performed by sorting out the node numbers with respect to their new co-ordinates so that $x'_1 = x'_{min}$ and $x'_n = x'_{max}$ for the x' direction and once more, $y'_1 = y'_{min}$ and $y'_n = y'_{max}$ for the direction of y'. Now distinct locations of point concentrations, along x' and y', must be found. In our work we have used a very simple approach for finding the point concentration locations, although more sophisticated (and expensive) methods may be applied. All sequentially ordered co-ordinates having a difference less than a specified tolerance, e.g. $\Delta x' < \left(x'_{max} - x'_{min} \right) / (2n_i)$ and $\Delta y' < \left(y'_{max} - y'_{min} \right) / (2n_i)$ with n_i being the number of points, are considered as a point concentration and the average of the co-ordinates is taken as their representative. Assuming that $M_{x'}$ and $M_{y'}$ are the number of points concentrated along the two axes, clearly $M_{x'} \leqslant n_i$ and $M_{y'} \leqslant n_i$, the average point spacing is be found as

$$\bar{D}_{(x')} = \frac{1}{M_{x'}} \sum_{k=1}^{M_{x'}-1} \left| \bar{x}'_{k+1} - \bar{x}'_k \right|, \quad \bar{D}_{(y')} = \frac{1}{M_{y'}} \sum_{k=1}^{M_{y'}-1} \left| \bar{y}'_{k+1} - \bar{y}'_k \right|$$

where $\bar{x}'_k$ and $\bar{y}'_k$ denote the co-ordinates of the points concentrated along the rotated axes x' and y', respectively.
- A primary rectangular domain with aspect ratio of $\bar{D}_{(y')}/\bar{D}_{(x')}$ is built on the master node (with centre at the master node and edges parallel to the rotated axes). The dimension of the rectangle is increased sequentially, maintaining the aspect ratio, until a minimum number of points fall into the cloud. The sequence length may be chosen as one of the average spacing calculated. The selected points, at this stage, are not necessarily the same as those selected at the first step.
- The dimensions of the cloud along the two axes, to be used in the normalization procedure, are calculated as $R_{x0} = \text{Max}\left(\left| x'_{max} \right|, \left| x'_{min} \right| \right)$ and $R_{y0} = \text{Max}\left(\left| y'_{max} \right|, \left| y'_{min} \right| \right)$.

- The polynomial fitting is performed using the new positions of the points in the normalized co-ordinates. If the coefficient matrix $\mathbf{A}$ is singular, the rectangular cloud selected in the previous step, i.e. the last one with aspect ratio of $\bar{D}_{(y')}/\bar{D}_{(x')}$, is further enlarged and some new points are added to the first set. The procedure is repeated until a non-singular coefficient matrix is attained. Generally, there is no need to repeat the first step and change the dimensions of the primary cloud, as the information required for the approximation at the master node is taken from the closest points whose distribution has been already calculated. However, if the number of points added is much larger than that of the first set, say three times as large, it might be preferable to repeat the cloud selection process with a larger value for the minimum number of points, or a sequence of enlargements with a sequence length of $\bar{D} = \sqrt{\bar{D}_{(x')}^2 + \bar{D}_{(y')}^2}$.

Once the approximate functions over the cloud at each master node, are found the shape function derivatives with respect to the global co-ordinates are evaluated through Equations (20).

In Section 4 we present an example to show that without using the above procedure, especially when the arrangement of the points is directional, wrong answers may be obtained.

2.4. Modified weighted least square method

We have shown in previous section that using normalized local co-ordinates not only reduces the instability induced by the fitting processes, but also improves the accuracy of the procedure near the master node so that the selectivity property of Equation (14) is achieved approximately. In this section a modified version of the weighted least square method is recommended, in which the deviation from unity is recovered and the approximate function passes exactly by the central node value of each cloud. A similar method was presented in Reference [21] for direct calculation of derivatives of the approximate function in global co-ordinates. The method is revisited here for evaluation of the base functions in the normalized cloud to be differentiated later with respect to the global axes through relations (20).

Assume that for the ith node a cloud of n_i neighbouring nodes is defined. The local number of the master node is set to be unity. The approximate function, in the normalized co-ordinates, may be written as

$$\hat{u}(\xi) = \bar{u}_1 + \alpha_1 \xi + \alpha_2 \eta + \alpha_3 \xi^2 + \alpha_4 \xi\eta + \alpha_5 \eta^2 + \cdots \tag{21}$$

In other words

$$\hat{u}(\xi) = \bar{u}_1 + \sum_{l=1}^{m-1} p_l(\xi)\alpha_l = \bar{u}_1 + \mathbf{p}^{\mathrm{T}}(\xi)\boldsymbol{\alpha} \tag{22}$$

An example of $\mathbf{p}(\xi)$ in above equation is

$$\mathbf{p}(\xi) = \left\langle \xi, \eta, \xi^2, \xi\eta, \eta^2 \right\rangle^{\mathrm{T}}; \quad m = 6 \text{ (for quadratic approximation)} \tag{23}$$

In order to determine the α_l values, we substitute Equation (22) into (6). Hence

$$J = \sum_{j=2}^{n_i} w_i(\xi_j)\left[(\bar{u}_j - \bar{u}_1) - \mathbf{p}^{\mathrm{T}}(\xi_j)\boldsymbol{\alpha}\right]^2 \tag{24}$$

As the error for node 1 is equal to zero, it does not enter in the above norm.

The discrete norm of Equation (24) is minimized as follows:

$$\frac{\partial J}{\partial \alpha_l} = 0; \quad l = 1, \ldots, m - 1 \tag{25}$$

which as before leads to

$$\mathbf{A}\boldsymbol{\alpha} = \mathbf{B}\bar{\mathbf{u}}^* \tag{26}$$

where

$$\mathbf{A} = \sum_{j=2}^{n_i} w_i(\xi_j)\mathbf{p}(\xi_j)\mathbf{p}^{\mathrm{T}}(\xi_j), \quad \mathbf{B} = \left[w_i(\xi_2)\mathbf{p}(\xi_2), \ldots, w_i(\xi_{n_i})\mathbf{p}(\xi_{n_i}) \right] \tag{27}$$

and

$$\bar{\mathbf{u}}^* = \left\langle \bar{u}_2 - \bar{u}_1, \ldots, \bar{u}_{n_i} - \bar{u}_1 \right\rangle^{\mathrm{T}} \tag{28}$$

One can use the following equation to replace $\bar{\mathbf{u}}^*$ by $\bar{\mathbf{u}}$:

$$\bar{\mathbf{u}}^* = \mathbf{T}_1 \bar{\mathbf{u}} \tag{29}$$

with $\mathbf{T}_1$ being the following transformation matrix:

$$T_1[i, j] = \begin{cases} -1; & j = 1 \\ 1; & j = i + 1; \quad (i = 1, \ldots, n_i - 1), \ (j = 1, \ldots, n_i) \\ 0; & \text{otherwise} \end{cases} \tag{30}$$

Equation (26) may be rewritten as

$$\mathbf{A}\boldsymbol{\alpha} = \mathbf{B}\mathbf{T}_1\bar{\mathbf{u}} \tag{31}$$

which gives

$$\boldsymbol{\alpha} = \mathbf{A}^{-1}\mathbf{B}\mathbf{T}_1\bar{\mathbf{u}} \tag{32}$$

Once again we have assumed that the point distribution is such that $\mathbf{A}^{-1}$ is available. Substitution of (32) into (22) gives

$$\hat{u}(\xi) = \bar{u}_1 + \mathbf{p}^{\mathrm{T}}(\xi)\mathbf{A}^{-1}\mathbf{B}\mathbf{T}_1\bar{\mathbf{u}} \tag{33}$$

On the other hand we have

$$\bar{u}_1 = \mathbf{T}_2\bar{\mathbf{u}} \tag{34}$$

where

$$\mathbf{T}_2 = \langle 1, 0, \ldots, 0 \rangle_{1 \times n_i} \tag{35}$$

Finally we have

$$\hat{u}(\xi) = \left(\mathbf{T}_2 + \mathbf{p}^{\mathrm{T}}(\xi)\mathbf{A}^{-1}\mathbf{B}\mathbf{T}_1 \right) \bar{\mathbf{u}} = \mathbf{N}(\xi)\bar{\mathbf{u}} \tag{36}$$

where $\mathbf{N}(\xi)$ is the shape function matrix for the ith cloud.

The procedure gives slightly different results for the derivatives of the functions when it is used in the collocation formulation. The reader will notice that the method differs from a simple shifting of the polynomial fitted in (12) since the number of variables in the least square procedure is less.

3. SOLUTION OF BOUNDARY VALUE PROBLEMS

Having found appropriate base functions for interpolation of the unknowns, a suitable solution method should be chosen. The mathematical model is usually summarized as a set of differential equations

$$\mathscr{A}(\mathbf{u}) = \mathbf{0} \quad \text{in } \Omega \tag{37}$$

which is to be satisfied with the following boundary conditions:

$$\mathscr{B}(\mathbf{u}) = \mathbf{0} \quad \text{on } \Gamma_t \tag{38a}$$

and

$$\mathbf{u} - \mathbf{u}_0 = \mathbf{0} \quad \text{on } \Gamma_u \tag{38b}$$

where $\mathscr{A}$ and $\mathscr{B}$ are differential operators for the governing equation in the domain and the Neumann boundary conditions, respectively. The domain Ω is a subset of R^n in which Equation (37) is to be satisfied and $\Gamma = \Gamma_u \cup \Gamma_t$ is the boundary of the domain on which Equations (38a) and (38b) are to be satisfied.

To solve the problem numerically, we start from the general weighted residual expression. Thus

$$\int_\Omega \mathbf{W}(\mathbf{x})\mathscr{A}(\hat{\mathbf{u}}) \, \mathrm{d}\Omega + \int_{\Gamma_t} \bar{\mathbf{W}}_t(\mathbf{x})\mathscr{B}(\hat{\mathbf{u}}) \, \mathrm{d}\Gamma + \int_{\Gamma_u} \bar{\mathbf{W}}_u(\mathbf{x})(\hat{\mathbf{u}} - \mathbf{u}_0) \, \mathrm{d}\Gamma = \mathbf{0} \tag{39}$$

in which $\mathbf{W}$, $\bar{\mathbf{W}}_t$ and $\bar{\mathbf{W}}_u$ are suitable weighting functions arranged in diagonal matrices. If the equation holds for any arbitrary weighting functions, then it follows that $\hat{\mathbf{u}} = \mathbf{u}_{\text{exact}}$. But as this is not generally possible, depending on the choice for the set of weighting functions, $\hat{\mathbf{u}}$ will be an approximate of $\mathbf{u}_{\text{exact}}$. The essential boundary conditions are usually satisfied exactly (see Section 2.4) and thus the main integral equation is

$$\int_\Omega \mathbf{W}(\mathbf{x})\mathscr{A}(\hat{\mathbf{u}}) \, \mathrm{d}\Omega + \int_{\Gamma_t} \bar{\mathbf{W}}_t(\mathbf{x})\mathscr{B}(\hat{\mathbf{u}}) \, \mathrm{d}\Gamma = \mathbf{0} \tag{40}$$

Now the weighting functions are to be chosen so that the number of equations and nodal values be in balance to obtain a system of equations as

$$\mathbf{K}\bar{\mathbf{u}} = \mathbf{f} \tag{41}$$

Depending on the choices for $\mathbf{W}$ and $\bar{\mathbf{W}}_t$ different numerical solutions will result. To explain our interpretation of the weighted form of (40), the weak formulation will be written next.

3.1. Weak formulation

The weak formulation is obtained when an integral by parts is performed for the integral containing the operator $\mathscr{A}$ giving

$$-\int_{\Omega} \mathscr{A}_1[\mathbf{W}(\mathbf{x})]\mathscr{A}_2[\hat{\mathbf{u}}]\,\mathrm{d}\Omega + \int_{\Gamma} \mathbf{W}(\mathbf{x})\mathscr{C}[\hat{\mathbf{u}}]\,\mathrm{d}\Gamma + \int_{\Gamma_t} \bar{\mathbf{W}}_t(\mathbf{x})\mathscr{B}(\hat{\mathbf{u}})\,\mathrm{d}\Gamma = \mathbf{0} \qquad (42)$$

In boundary value problems, usually, operators $\mathscr{B}$ and $\mathscr{C}$ contain similar terms so that they can be eliminated when the weighting functions are selected by choosing

$$\bar{\mathbf{W}}_t = -[\mathbf{W}]_{\Gamma_t} \qquad (43)$$

which leads to elimination of the natural boundary conditions. Now, if the Galerkin method is employed, i.e. $\mathbf{W} = \mathbf{N}$, the approximation is said to be optimal (for elliptic problems). Indeed, it can be shown that in such a case the solution is equivalent to minimization of a suitable norm of the approximate solution error with respect to nodal variables [32].

   The reader will notice that if we substitute Equation (43) into (40) gives

$$\int_{\Omega} \mathbf{W}\mathscr{A}(\hat{\mathbf{u}})\,\mathrm{d}\Omega - \int_{\Gamma_t} \mathbf{W}\mathscr{B}(\hat{\mathbf{u}})\,\mathrm{d}\Gamma = \mathbf{0} \qquad (44)$$

It should be noted that as the base functions and the weighting functions are defined on small sub-domains few of them have edges at the boundaries. Therefore

$$\int_{\Omega_s} \mathbf{W}_s\mathscr{A}(\hat{\mathbf{u}})\,\mathrm{d}\Omega = \mathbf{0} \quad \text{if } \Omega_s \cap \Gamma = \emptyset \qquad (45)$$

and

$$\int_{\Omega_s} \mathbf{W}_s\mathscr{A}(\hat{\mathbf{u}})\,\mathrm{d}\Omega - \int_{\Gamma_t} \mathbf{W}_s\mathscr{B}(\hat{\mathbf{u}})\,\mathrm{d}\Gamma = \mathbf{0} \quad \text{if } \Omega_s \cap \Gamma \neq \emptyset \qquad (46)$$

3.2. Collocation methods

Now if we assume that Ω_s is small enough so that we can choose $\mathbf{W}_s$ such that

$$\mathbf{W}_s = \mathbf{I}, \quad \Omega_s \to 0 \qquad (47)$$

Then

$$\int_{\Omega_s} \mathbf{W}_s\mathscr{A}(\hat{\mathbf{u}})\,\mathrm{d}\Omega = \bar{\mathbf{R}}_s\Omega_s, \quad \bar{\mathbf{R}}_s = \bar{\mathscr{A}}_s(\hat{\mathbf{u}}) \qquad (48a)$$

and

$$\int_{\Gamma_s} \mathbf{W}_s\mathscr{B}(\hat{\mathbf{u}})\,\mathrm{d}\Gamma = \bar{\mathbf{R}}_t\Gamma_s, \quad \bar{\mathbf{R}}_t = \bar{\mathscr{B}}(\hat{\mathbf{u}}) \qquad (48b)$$

In above, a 'bar' is used to indicate evaluation of the average of the function over Ω_s and Γ_s is the projection of Ω_s on the boundary. Equations (45) and (46) can be rewritten as

$$\bar{\mathscr{A}}_s(\hat{\mathbf{u}})\Omega_s = \mathbf{0} \quad \text{if } \Gamma_s \cap \Gamma = \emptyset \qquad (49a)$$

and

$$\bar{\mathscr{A}}_s(\hat{\mathbf{u}})\Omega_s - \bar{\mathscr{B}}_s(\hat{\mathbf{u}})\Gamma_s = \mathbf{0} \quad \text{if } \Gamma_s \cap \Gamma \neq \emptyset \qquad (49b)$$

Since Ω_s is assumed to be very small, the average values on Ω_s may be taken as proportional to the corresponding values at the master node and thus

$$\bar{\mathscr{A}}_s(\hat{\mathbf{u}}) \cong \alpha_i \big[\mathscr{A}(\hat{\mathbf{u}})\big]_i, \quad \bar{\bar{\mathscr{B}}}(\hat{\mathbf{u}}) \cong \beta_i \big[\bar{\mathscr{B}}(\hat{\mathbf{u}})\big]_i \tag{50}$$

Here i denotes the number of the master node and α and β are appropriate coefficients. Assuming that Ω_s is so small that the average values and the master node values are of the same sign, then Equations (49) can be written as

$$\alpha_i \big[\mathscr{A}(\hat{\mathbf{u}})\big]_i \Omega_s = \mathbf{0} \quad \text{if } \Gamma_s \cap \Gamma = \emptyset \tag{51a}$$

and

$$\alpha_i \big[\mathscr{A}(\hat{\mathbf{u}})\big]_i \Omega_s - \beta_i \big[\mathscr{B}(\hat{\mathbf{u}})\big]_i \Gamma_s = \mathbf{0} \quad \text{if } \Gamma_s \cap \Gamma \neq \emptyset \tag{51b}$$

Above equation is equivalent to using a collocation method considering the area and the edge length influenced by the node as the weights.

We notice that in Equation (51b) the equilibrium equations and the Neumann boundary condition appear with opposite signs in a single equation. It may be also observed that in Equation (51b) the ratio of the sub-domain to its projection to the boundary, i.e. a characteristic length, is of more importance than the absolute values. Therefore, to solve the problem using a finite number of points, the smallest area around a point is chosen in order to evaluate an appropriate characteristic length. The reader will notice that such a characteristic length is only used at the boundary nodes.

In summary, the following equations must be solved for nodes at inside or at the boundary of the domain:

$$\big[\mathscr{A}(\hat{\mathbf{u}})\big]_i = \mathbf{0} \quad \text{for inside nodes} \tag{52a}$$

$$\gamma (A_1 + A_2) \big[\mathscr{A}(\hat{\mathbf{u}})\big]_i - (L_1 + L_2) \big[\mathscr{B}(\hat{\mathbf{u}})\big]_i = \mathbf{0} \quad \text{for boundary nodes} \tag{52b}$$

In above, A_1 and A_2 are the areas associated to the node and the nearest neighbouring points as shown in Figure 7, and L_1 and L_2 are their projections on the boundary. Also γ is a suitable factor. Our studies show that $\gamma = 1$ and 2 are the best choices for elasticity and Poisson's equations, respectively.

3.3. Analogy with finite calculus (FIC)

A similar set of stabilized discretized equations can be obtained using the finite calculus (FIC) method proposed by Oñate [25]. The FIC method has been used for stabilization of the finite point method [29, 30] using a generalization of the method originally proposed for convection–diffusion and fluid flow problems [25]. To illustrate the analogy between the two methods, in this section we shall overview some features of FIC method.

The idea of the FIC method is to apply the basic thermodynamic balance laws not to an infinitesimal domain but to a domain with finite dimensions. This leads to a modified system of governing equations which contain additional terms than the standard differential equations of the infinitesimal theory.

Let $\mathscr{A}(\mathbf{u}) = 0$ represent the standard differential equations at a point in two dimensional space. The modified governing equations obtained with the FIC method can be written as [25, 26]

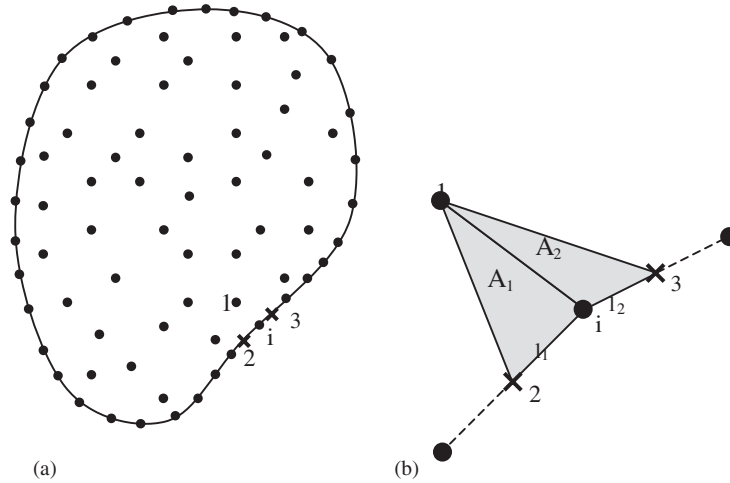$$\mathscr{A}(\mathbf{u}) - \tfrac{1}{2} \mathbf{h}^{\mathrm{T}} \nabla \mathscr{A}(\mathbf{u}) = 0 \tag{53}$$

Figure 7. Evaluation of α_i and β_i values for Neumann boundary nodes: point 1 is the closest inside point to master node i, l_1 and l_2 are the projections of constructed sub-domains A_1 and A_2 to the boundary of the domain.

where $\mathbf{h}$ is a characteristic length vector whose direction is to be specified. Several stabilization methods may be interpreted by choosing appropriate magnitude and direction for $\mathbf{h}$. For example, in fluid flow problems a streamline upwind Petrov–Galerkin (SUPG) may be recovered if $\mathbf{h}$ is taken in direction of the velocity $\mathbf{u}$, i.e.

$$\mathscr{A}(\mathbf{u}) - \frac{1}{2}\frac{h}{|\mathbf{u}|}\mathbf{u}^{\mathrm{T}}\nabla\mathscr{A}(\mathbf{u}) = 0 \tag{54}$$

Other possibilities are discussed in details in References [25, 26].

The FIC approach if applied at the boundaries with the Neumann condition results in the following modified equation [25]:

$$\mathscr{B}(\mathbf{u}) - \tfrac{1}{2}\mathbf{h}^{\mathrm{T}}\mathbf{n}\mathscr{A}(\mathbf{u}) = \mathbf{0} \tag{55}$$

where $\mathbf{n}$ is the unit normal to the boundary. This equation may be compared with Equation (51b). Here we note that depending on the direction of $\mathbf{h}$ the stabilized boundary conditions can be written in two forms:

$$\mathscr{B}(\mathbf{u}) - \tfrac{1}{2}h_n\mathscr{A}(\mathbf{u}) = \mathbf{0} \quad \text{if } \mathbf{h}^{\mathrm{T}}u > 0 \tag{56a}$$

and

$$\mathscr{B}(\mathbf{u}) + \tfrac{1}{2}h_n\mathscr{A}(\mathbf{u}) = \mathbf{0} \quad \text{if } \mathbf{h}^{\mathrm{T}}u < 0 \tag{56b}$$

where $h_n = \left|\mathbf{h}^{\mathrm{T}}\mathbf{n}\right|$.

When the stabilization is generalized to elasticity problems the choice of the sign in the equation remains unsolved.

Note that the form of Equations (56) is analogous to that of Equation (51b) deduced from the weighted residual formulation. Also, as discussed in the previous subsection, the choice of (56b) in which both the equilibrium equation and boundary condition residuals appear with similar signs is not consistent with the optimal solutions. Hence the form of Equation (56a) should be used in practice.

Incidentally in References [29, 30] the correct choice of signs in Equation (56a) was selected and it was shown that the resulting FPM gave stable results for a variety of problems. It is noteworthy that our formulation shows that there is no need to add any additional terms for the residual equations at the inside nodes. This coincides with the observation made in References [29, 30] in which adding a stabilization term to the equations within the analysis domain, similar to Equation (53), has been recognized to be not effective.

Considering the choice of the sign, another difference with the FIC stabilization procedure presented in References [29, 30] is the way we select the characteristic length parameter via Equation (52b).

4. NUMERICAL RESULTS

In this section we present some numerical examples to show the reliability of the method proposed in this paper. A quadratic interpolation has been used in all problems.

4.1. Simply supported beam under uniform load

A two dimensional simply supported beam subjected to uniform load has been solved under plane stress condition. In order to demonstrate the effect of using normalized and rotated co-ordinates the point distribution is selected first as the pattern shown in Figure 8(b). The
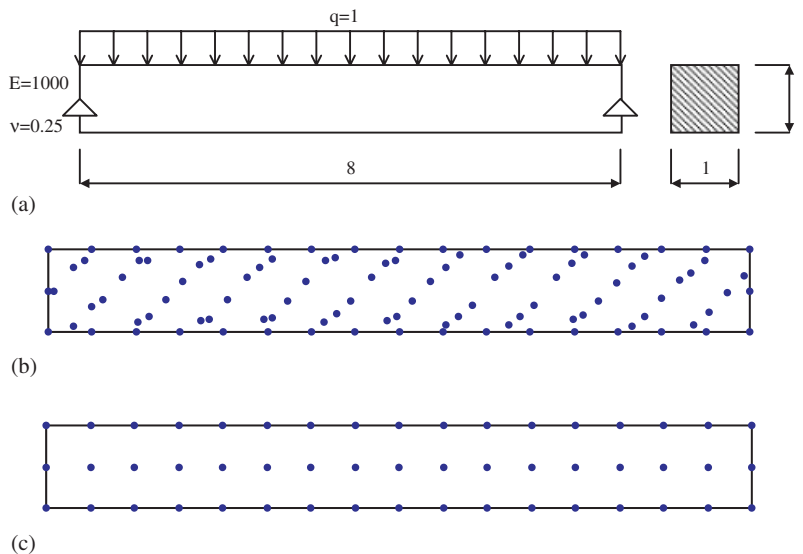

Figure 8. (a) Simply supported beam under uniform load; (b) Skew grid; and (c) Cartesian grid.
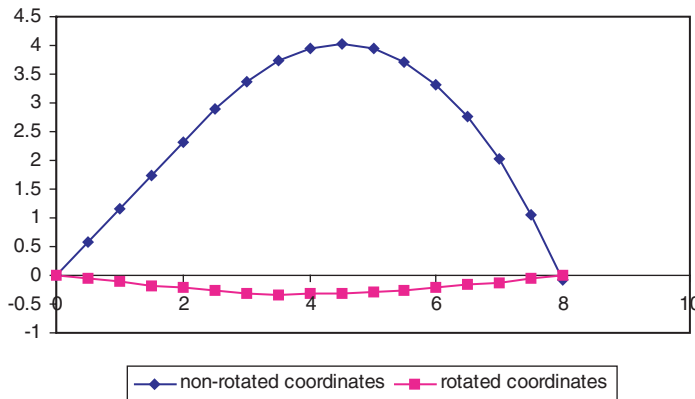
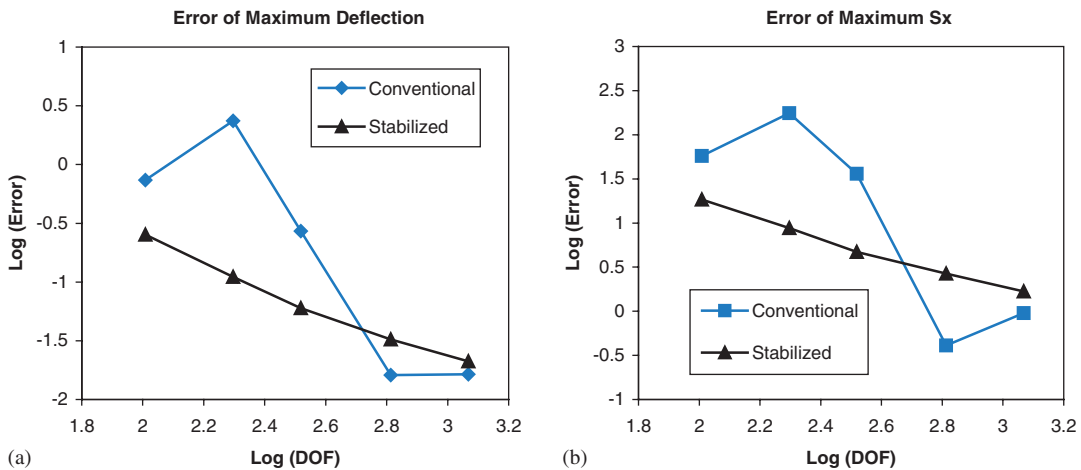Figure 9. Vertical displacement at the top fibre of simply supported beam.

Figure 10. Solution of simply supported beam under uniform load using conventional and stabilized versions of FPM: (a) error of maximum vertical displacement; and (b) error of maximum tensile stress.

displacements obtained using the proposed approach are given in Figure 9. It can clearly be seen that without using the normalized and rotated co-ordinates some wrong answers are obtained. From this example on we shall use the mapping algorithm to minimize the effect of instability due to polynomial fitting.

The problem is also used to study the convergence rate of solution when the proposed stabilized Finite Point Method is employed. The simply supported beam has been solved using a series of regular point distributions one of which is depicted in Figure 8(c). Although not shown in the figure, to prevent singularity effect at the supports, two parabolic distributions of shear stresses equilibrated with the distributed loads, are considered. Only one node is restrained at each end to eliminate rigid body motion. The beam is modeled as the one given in Reference [33]. Maximum deflection and flexural stress are used to evaluate the accuracy of the solutions. Figures 10(a) and (b) show the results of the errors for both the conventional
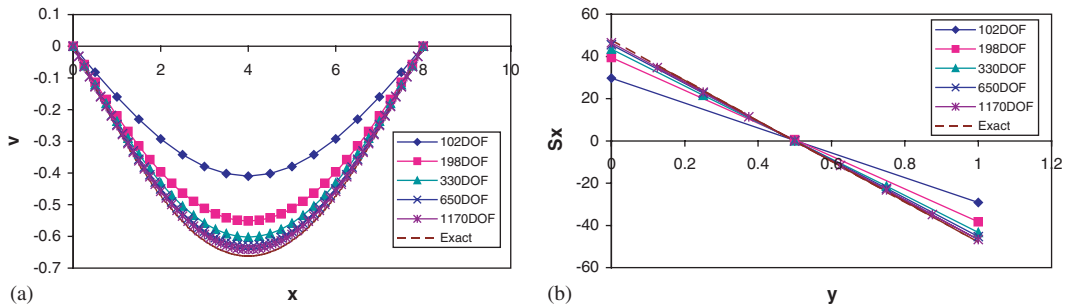
Figure 11. Simply supported beam under uniform load: (a) distribution of vertical displacement along the neutral axis of the beam; and (b) distribution of normal stress along the central cross-section of the beam ($x = 4$).
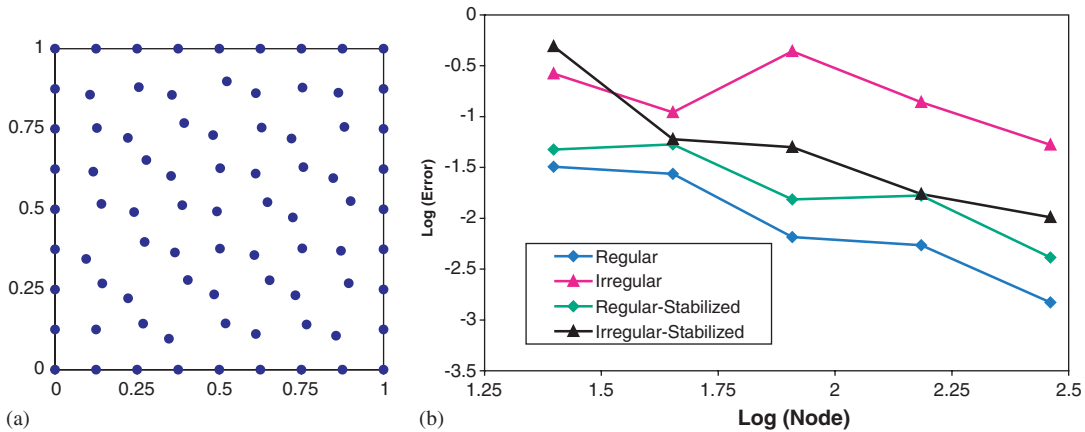

Figure 12. Laplace equation in a square domain: (a) the domain; and (b) solutions by conventional and stabilized versions of FPM.

and the stabilized versions of the Finite Point Method. It can be seen that the monotonic convergence is lost when the solution is performed with no stabilization terms. However, when the boundary described in Section 3 is used, i.e. when the equilibrium residual terms are added to the traction residuals, monotonic convergence is achieved.

The deflection and shear stress diagrams for various numbers of degrees of freedom are given in Figures 11(a) and (b).

4.2. Laplace equation in a unit square

To demonstrate the performance of the standard and stabilized versions of FPM for problems with scalar differential equations, solution of a Laplace equation as

$$\nabla^2 u = 0, \quad 0 \leqslant x \leqslant 1, \quad 0 \leqslant y \leqslant 1$$

on a square domain shown in Figure 12(a) is considered as another benchmark problem. The Dirichlet boundary conditions, at $x = 0$ and $x = 1$, and Neumann boundary conditions, at

$y = 0$ and 1, are defined in accordance with the following exact solution:

$$u(x, y) = -x^3 - y^3 + 3xy^2 + 3x^2 y$$

Two sets of regular and irregular point distributions have been chosen for the numerical solutions. In all cases the mapping algorithm presented in Section 2.3.2 has been used and thus the main difference between the versions lies in the application of the stabilization method via the correction of the boundary residuals.

To demonstrate the deviation of the results from the exact values in all points, the following error norm is defined

$$\varepsilon = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(u_i^{\text{exact}} - u_i^{\text{approx}} \right)^2}$$

where u_i^{exact} and u_i^{approx} are the exact and approximate values of u_i and N is the total number of discrete values.

The results obtained using the standard FPM (Figure 12(b)) clearly demonstrate the lack of monotonic convergence specially for irregular patterns. It can be seen that when the stabilization presented in Section 3 is applied, the convergence is improved as Figure 12(b) depicts.

It is worth to compare the overall rate of convergence for regular grids of nodes in Figures 12(b). It is observed that the convergence rate of the stabilized version is slightly less than that of the conventional FPM. The reason may lie in the fact that the stabilized version is much closer to an optimal solution than the original version. In fact, in a nearly optimal solution, the error density distribution is expected to be uniform over the domain. This means that the error at the boundaries propagates within the domain in the improved version and thus the overall accuracy appears to be deteriorated (the reader will note that the ratio of interior node number to the total number of nodes increases when the mesh becomes finer). For irregular grids the same discussion seems to be valid if the first non-monotonic part of the curves are put aside and the convergence rates at the last three points are compared. In any case, an important consequence is the elimination of the sharp upward jumps appearing in the conventional (non-stabilized) FPM solutions.

4.3. A cantilever beam with an end load

A two dimensional cantilever beam, shown in Figure 13(a) is solved under plane stress condition. The exact solution can be found in literature [33]. The main difference between this beam example and the previous one lies in the nature of the tractions. The end load is the resultant of a parabolic distribution of the shear stress. The results are obtained for regular and irregular point distributions. Here again to prevent singularity at the fixed end, a parabolic distribution of shear stresses is considered and just one point is restrained in order to eliminate the rigid body motion in the vertical direction. All horizontal displacements are restrained at the fixed end and the prescribed values are obtained from the exact solution.

Figures 13(b) and (c) show the results for a regular point distribution when no stabilization is used for the boundary points. The errors in these figures represent error norms, as defined in the previous problem. Similar to the previous example the convergence rate is not monotonic. However, when the stabilization described in Section 3 is used, a major improvement is observed and the rate of convergence becomes monotonic. This can clearly be seen in Figures 14(a) and (b).
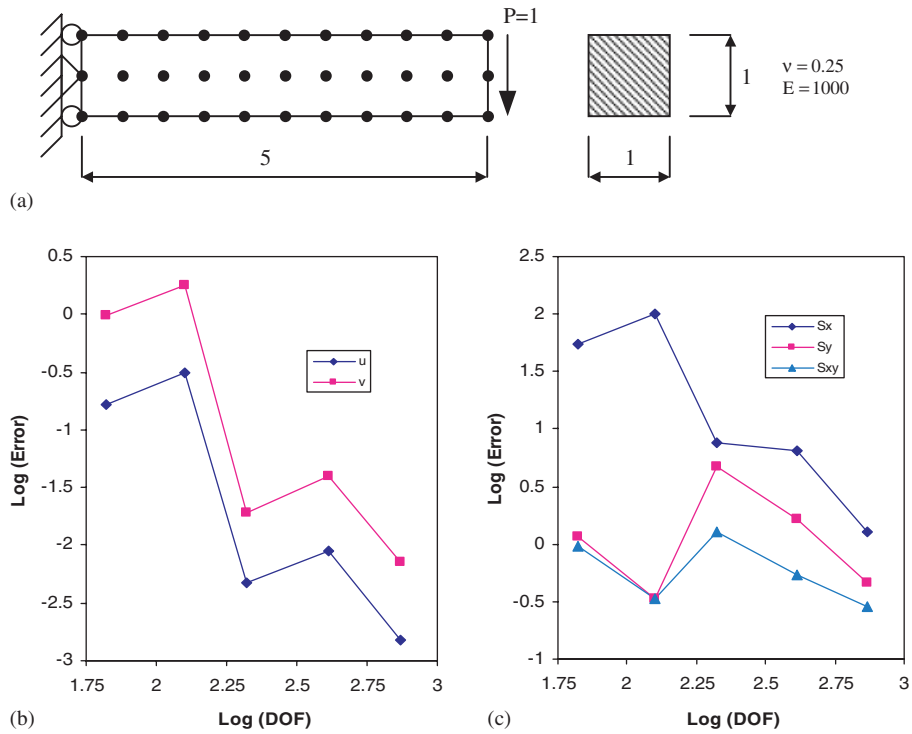
Figure 13. Cantilever beam under unit end load. Solution with conventional FPM: (a) geometry; (b) error norm values for displacement field; and (c) error norm values for stress field.
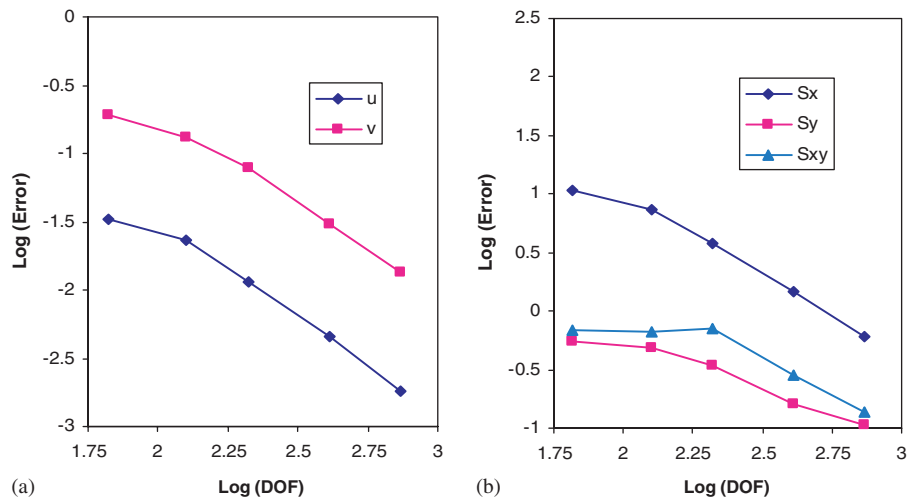

Figure 14. Cantilever beam under unit end load. Solution with stabilized version of FPM: (a) error norm values for displacement field; and (b) error norm values for stress field.
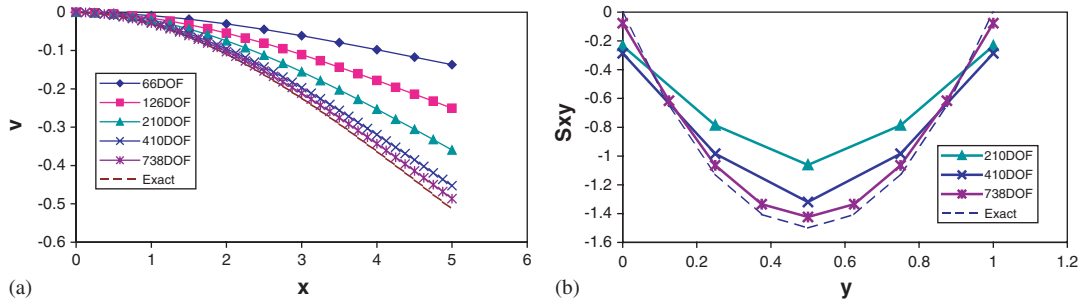
Figure 15. Cantilever beam under unit and load: (a) distribution of vertical displacement along the neutral axis fibre of the beam; and (b) distribution of shear stress along the central cross-section of the beam ($x = 2.5$).

Figure 15(a) depicts the beam deflection for various solutions. The stress distribution of the shear stress on a cross section at the mid span is given in Figure 15(b).

4.4. Perforated plate under uniform tensile tractions

The final example is a plane stress solution of a perforated plate under tension. A quarter of the plate is solved using the tractions evaluated from the exact stress field (Figure 16). The exact solution of the problem is given below:

$$\sigma_x = \bar{t}_x \left\{ 1 - \frac{a^2}{r^2} \left[\tfrac{3}{2} \cos(2\theta) + \cos(4\theta) \right] + \frac{3a^4}{2r^4} \cos(4\theta) \right\}$$

$$\sigma_y = -\bar{t}_x \left\{ \frac{a^2}{r^2} \left[\tfrac{1}{2} \cos(2\theta) - \cos(4\theta) \right] + \frac{3a^4}{2r^4} \cos(4\theta) \right\}$$

$$\sigma_{xy} = -\bar{t}_x \left\{ \frac{a^2}{r^2} \left[\tfrac{1}{2} \sin(2\theta) + \sin(4\theta) \right] - \frac{3a^4}{2r^4} \sin(4\theta) \right\}$$

$$u_r = \frac{\bar{t}_x}{4G} \left\{ r \left[\frac{(\kappa - 1)}{2} + \cos(2\theta) \right] + \frac{a^2}{r} \left[1 + (1 + \kappa) \cos(2\theta) \right] - \frac{a^4}{r^3} \cos(2\theta) \right\}$$

$$u_\theta = \frac{\bar{t}_x}{4G} \left[(1 - \kappa) \frac{a^2}{r} - r - \frac{a^4}{r^3} \right] \sin(2\theta)$$

where

$$G = \frac{E}{2(1 + v)}, \quad \kappa = \frac{3 - v}{1 + v}$$

In above equations, a is the radius of the hole and (r, θ) are the polar co-ordinates of the points, with origin at centre of the hole.

Figure 17 shows some of the results obtained from solution of the problem by application of the stabilized version of FPM presented in this paper.
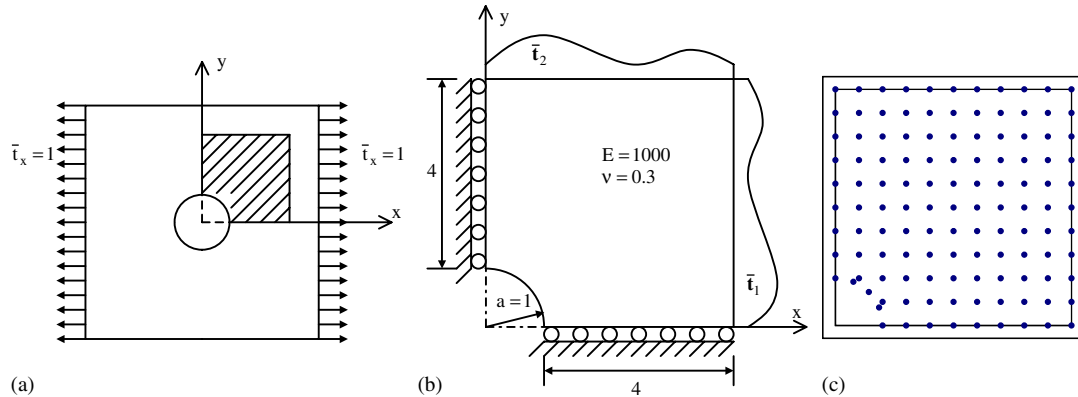
Figure 16. Square plate under tension: (a) infinite domain under tension;
(b) a quarter of the finite domain under tractions; and (c) a grid of
nodes for solution of the problem.

5. CONCLUSIONS

The finite point method was revisited in this paper. As for other collocation methods, numerical instabilities may appear from application of the standard (non-stabilized) FPM. In this paper we have shown that polynomial fitting and treatment of the collocation equations near the boundaries are the two main sources of instability in the FPM.

Application of normalized clouds was proposed and examined through some examples. The method is similar to using mapping in the FEM and helps to overcome the ill-conditioning in the least square fitting specially when the point distribution in the cloud is directional. The study of a sample problem has shown that wrong answers might be obtained when the conventional form of polynomial fitting is used.

A stabilized version of the FPM using equilibrium residuals at the boundaries was presented. The formulation is consistent with the general formulation, which is usually used for weighted residual methods and leads to optimal solutions.

The analogy between the proposed stabilization method and the finite calculus (FIC) approach has been discussed. The conclusions can be summarized as

- Using the weighted residual approach shows that one of the remedies for instability effect in boundary value problems lies in the treatment of the Neumann boundary conditions.
- Stabilization based on an optimal solution via a weighted residual formulation has a unique form and the residuals of the equilibrium equations and the Neumann boundary conditions appear with opposite signs at the boundary equations where tractions are prescribed. Although the same form of stabilization may be obtained using the FIC method, the method proposed here helps to choose adequately the sign of the stabilization term at the Neumann boundary.
- A small area next to the boundary node, and its projection, may be used as the suitable weights in the proposed stabilization approach.
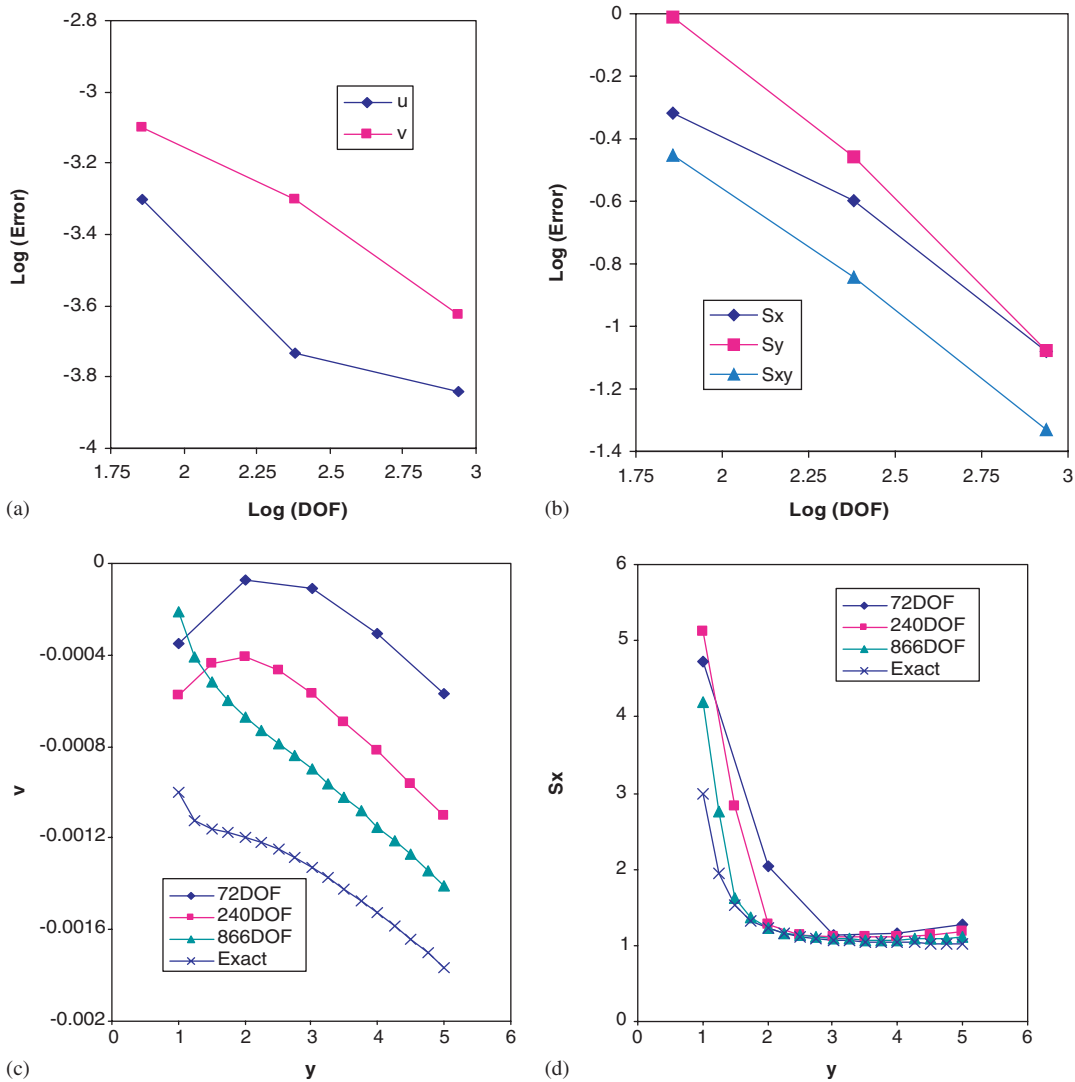
Figure 17. Stabilized FPM solution of square plate with hole: (a) error norm for displacement field; (b) error norm for stress field; (c) distribution of vertical displacement along line $x = 0$; and (d) distribution of horizontal stress along line $x = 0$.

Examples show that the problem of non-monotonic convergence, usually observed in the application of the standard (non-stabilized) FPM, is effectively reduced using the stabilized formulation. Further studies on the mathematical basis of the method are needed in order to ensure that the monotonic convergence is always achievable.

REFERENCES

1. Lucy LB. A numerical approach to the testing of the fission hypothesis. *Astronomical Journal* 1977; **82**(12):1013–1024.

2. Monaghan JJ. Why particle methods work. *SIAM Journal on Scientific and Statistical Computing* 1982; **3**(4):422–433.
3. Monaghan JJ. An introduction to SPH. *Computer Physics Communications* 1988; **48**:89–96.
4. Nayroles GB, Touzot, Villon P. Generalizing the finite element method: diffuse approximation and diffuse elements. *Computational Mechanics* 1992; **10**:307–318.
5. Belytschko T, Lu YY, Gu L. Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering* 1994; **37**:229–256.
6. Liu WK, Chen Y. Wavelet and multiple scale reproducing kernel methods. *International Journal for Numerical Methods in Engineering* 1995; **21**:901–931.
7. Liu WK, Chen Y, Chang CT, Belytschko T. Advances in multiple scale kernel particle methods. *Computational Mechanics* 1996; **18**:73–111.
8. Liu WK, Chen Y, Jun S, Chen JS, Belytschko T, Pan C, Uras RA, Chang CT. Overview and application of the reproducing kernel particle methods. *Archive of Computational Methods in Engineering*: *State of the Art Reviews* 1996; **3**:3–80.
9. Liu WK, Jun S, Li S, Adee J, Belytschko T. Reproducing kernel particle methods for structural dynamics. *International Journal for Numerical Methods in Engineering* 1995; **38**:1655–1679.
10. Liu WK, Jun S, Zhang YF. Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids* 1995; **20**:1081–1106.
11. Melenk JM, Babuška I. The partition of unity finite element method: basic theory and applications. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:289–314.
12. Babuška I, Melenk JM. The partition of unity method. *International Journal for Numerical Methods in Engineering* 1997; **40**:727–758.
13. Duarte CA, Oden JT. An *h–p* adaptive method using clouds. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:237–262.
14. Zhu T, Zhang J-D, Atluri SN. A local boundary integral equation (LBIE) method in computational mechanics, and a meshless discretization approach. *Computational Mechanics* 1998; **21**:223–235.
15. Atluri SN, Zhu T. A new Meshless Petrov–Galerkin (MLPG) approach in computational mechanics. *Computational Mechanics* 1998; **22**:117–127.
16. Atluri SN, Zhu T. The meshless local Petrov–Galerkin (MLPG) approach for solving problems in elasto-statics. *Computational Mechanics* 2000; **25**:169–179.
17. De S, Bathe KJ. The method of finite spheres. *Computational Mechanics* 2000; **25**:329–345.
18. Jensen PS. Finite difference techniques for variable grids. *Computers and Structures* 1972; **2**:17–29.
19. Perrone N, Kao R. A general finite difference method for arbitrary meshes. *Computers and Structures* 1975; **5**:45–58.
20. Demkowicz L, Karafiat A, Liszka T. On some convergence results for FDM with irregular mesh. *Computer Methods in Applied Mechanics and Engineering* 1984; **42**:343–355.
21. Liszka TJ, Duarte CAM, Tworzydlo WW. hp-Meshless cloud method. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:263–288.
22. Oñate E, Idelsohn S, Zienkiewicz OC, Taylor RL. A finite point method in computational mechanics—applications to convective transport and fluid flow. *International Journal for Numerical Methods in Engineering* 1996; **139**:3839–3866.
23. Oñate E, Idelsohn S. A mesh free finite point method for advective–diffusive transport and fluid flow problems. *Computational Mechanics* 1998; **21**:283–292.
24. Oñate E, Idelsohn S, Zienkiewicz OC, Taylor RL, Sacco C. A stabilized finite point method for analysis of fluid mechanics problems. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:315–346.
25. Oñate E. Derivation of stabilized equations for numerical solution of advective–diffusive transport and fluid flow problems. *Computer Methods in Applied Mechanics and Engineering* 1998; **151**:233–265.
26. Oñate E. Possibilities of finite calculus in computational mechanics. *International Journal for Numerical Methods in Engineering*, September 2002, submitted.
27. Oñate E, Sacco C, Idelsohn S. A finite point method for incompressible flow problems. *Computing and Visualization in Science* 2000; **2**:67–75.
28. Löhner R, Sacco C, Oñate E, Idelsohn S. A finite point method for compressible flow. *International Journal for Numerical Methods in Engineering* 2002; **53**:1765–1779.
29. Oñate E, Perazzo F, Miquel J. A finite point method for elasticity problems. *Computers and Structures* 2001; **79**:2151–2163.

30. Oñate E, Perazzo F, Miquel J. Advances in the stabilized finite point method for structural mechanics. International Center for Numerical Methods in Engineering (CIMNE), Universidad Politécnica de Cataluña (Tech. Rep. 164). *Presented in European Conference on Computational Mechanics (ECCM)*, München, Germany, 1999.
31. Aluru NR. A point collocation method based on reproducing kernel approximations. *International Journal for Numerical Methods in Engineering* 2000; **47**:1083–1121.
32. Zienkiewicz OC, Taylor RL. *The Finite Element Method* (5th edn). Butterworth-Heineman: Stoneham, MA, 2000.
33. Timoshenko SP, Goodier JN. *Theory of Elasticity* (3rd edn). McGraw-Hill, New York, 1970.