 CMES, vol.1655, no.1, pp.1-47, 2010

A Smooth Finite Element Method Based on Reproducing Kernel DMS-Splines

Sunilkumar N[1] and D Roy[1,2]

Abstract: The element-based piecewise smooth functional approximation in the conventional finite element method (FEM) results in discontinuous first and higher order derivatives across element boundaries. Despite the significant advantages of the FEM in modelling complicated geometries, a motivation in developing mesh-free methods has been the ease with which higher order globally smooth shape functions can be derived via the reproduction of polynomials. There is thus a case for combining these advantages in a so-called hybrid scheme or a 'smooth FEM' that, whilst retaining the popular mesh-based discretization, obtains shape functions with uniform C^p ($p \geq 1$) continuity. One such recent attempt, a NURBS based parametric bridging method (Shaw *et al.* 2008b), uses polynomial reproducing, tensor-product non-uniform rational B-splines (NURBS) over a typical FE mesh and relies upon a (possibly piecewise) bijective geometric map between the physical domain and a rectangular (cuboidal) parametric domain. The present work aims at a significant extension and improvement of this concept by replacing NURBS with DMS-splines (say, of degree $n > 0$) that are defined over triangles and provide C^{n-1} continuity across the triangle edges. This relieves the need for a geometric map that could precipitate ill-conditioning of the discretized equations. Delaunay triangulation is used to discretize the physical domain and shape functions are constructed via the polynomial reproduction condition, which quite remarkably relieves the solution of its sensitive dependence on the selected knotsets. Derivatives of shape functions are also constructed based on the principle of reproduction of derivatives of polynomials (Shaw and Roy 2008a). Within the present scheme, the triangles also serve as background integration cells in weak formulations thereby overcoming non-conformability issues. Numerical examples involving the evaluation of derivatives of targeted functions up to the fourth order and applications of the method to a few boundary value problems of general interest in solid mechanics over (non-simply connected) bounded domains in 2D are presented towards the

[1] Structures Lab, Department of Civil Engineering, Indian Institute of Science, Bangalore 560012, India

[2] Communicating author; Email: royd@civil.iisc.ernet.in

end of the paper.

Keywords: DMS-splines, Delaunay triangulation; globally smooth shape functions; polynomial reproduction; boundary value problems.

1 Introduction

Numerical solutions of models of complex engineering structures often pose challenges that include appropriate treatment of nonlinearity of various forms, the complicated domain geometry and the boundary. The most popular approximation, the finite element method (FEM), employs an element-based discretization of the spatial domain, which is a key feature as element-wise approximations of field variables not only provide a relief from the search of globally admissible functions, but also introduces versatility in approximating complex geometries with the accuracy of approximation generally increasing with decreasing element sizes. The governing equations are often solved through an element-wise application of the variational or Galerkin method (symmetric, unsymmetric or discontinuous), where the interpolating trial and test functions are piecewise polynomials over elements, thereby attaining at best C^0 continuity. Achieving C^1 or higher order global continuity uniformly in the domain interior is however a non-trivial problem, especially in 2D or still higher dimensional domains, and an efficient solution to this problem remains mostly elusive.

Use of a conventional element-based discretization has its other pitfalls as well. For instance, repeated interactions with the CAD during mesh refinement are a costly procedure. Then, in large deformation problems, solutions may get affected due to element distortions. Moreover, as the continuum is assumed to be connected, it is difficult to model a possible fracture of the material body into a number of pieces. A way out of some of these drawbacks is possible with mesh-free methods, wherein the domain is discretized by a set of nodes (also called particles). Over the last two decades, researchers have shown keen interest in developing and expanding the realm of applications of mesh free methods. Some of these methods include the smooth particle hydrodynamics (SPH), (Lucy 1977; Gingold and Monaghan 1977), the diffuse element method (DEM) (Nayroles *et al*. 1992), the element free Galerkin method (EFG) (Belytschko *et al*. 1994), the reproducing kernel particle method (RKPM) (Liu *et al*. 1995a, 1995b), Moving least square reproducing kernel (MLSRK) method (Liu *et al.* 1997), the partition of unity method (PUM) (Babuška and Melenk 1997), the *h*-*p* Clouds (Duarte and Oden 1997), the mesh-free local boundary integral equation method (LBIE) (Zhu *et al*. 1998), the mesh-less local Petrov–Galerkin method (MLPG) (Atluri *et al*. 1999), error reproducing kernel method (ERKM) (Shaw and Roy 2007a), error reproducing and interpolating kernel

method (ERIKM) (Shaw *et al.* 2008c) and several others. However, these methods do not possess the versatility of element-based domain discretization.

Incidentally, most mesh-free methods are not strictly 'mesh-free', especially when they are implemented using the weak formulation, wherein a set of background cells are used for integrating the weak form. The so-called conformability of integration cells vis-à-vis the distribution of particles and supports of shape functions determine the accuracy of integration and convergence of solutions thereof. However, the MLPG method eminently bypasses the non-conformability issue (Atluri and Zhu 1998, Atluri *et al.* 1999, Atluri *et al.* 2000, Aturi and Zhu 2000). Yet another limitation of most mesh-free methods is the sensitive dependence of solutions on the supports of window functions. The size of the support is only constrained by the minimum number of particles that it must contain to ensure the invertibility of the moment matrix (Han and Meng 2001). While a not-too-small support size prevents the moment matrix from being singular, a very large size might lead to excessive smoothness of the approximation. In the absence of a strictly quantitative criterion to arrive at the optimal support size, one typically resorts to costly numerical experiments to choose the right size. Moreover, most mesh-free shape functions are non-interpolating and hence may not strictly qualify as test functions as they do not vanish over the essential part of the domain boundary.

While mixed FE methods, which are capable of obtaining smooth stress or strain fields, have been extensively researched, they involve a significant augmentation of the degrees-of-freedom (DOF-s). Moreover, each mixed method with both displacements and their derivatives as DOF-s has to grapple with certain stability issues (Zienkiewicz, *et al.* 1967). Stabilization techniques are extensively reported in the literature (Hughes 1995; Hughes *et al.* 2004; Onate *et al.* 2006). Despite considerable research in developing and understanding the stability of mixed methods, the (unconditional or parameter-independent) coercivity of the bilinear form (especially following linearizations of nonlinear PDE-s) is not guaranteed (see, for instance, Auricchio *et al.* 2005) and may be extremely sensitive to element aspect ratios (Ainsworth and Coggins 2000). In fact, for linear systems, an analysis of the bilinear form often yields parameter bounds that are not sharp and this is yet another source of difficulty. In these techniques, accordingly, the parameters in the stabilizing terms in the weak form are generally arrived at through rigorous numerical experiments. Attempts have been made at developing mixed mesh-free methods that promise improved numerical behaviour against locking (for instance, the mixed MLPG method; Atluri *et al.* 2004, 2006a, 2006b; Soric and Jarak 2010). Since they admit an increased number of unknowns (displacements and strains and/or stresses) in the formulation, they can also handle the singularity issues which might arise due to ill-behaved derivatives of MLS shape functions. In particular, the mixed finite

volume MLPG method (Atluri *et al.* 2004) interpolates displacements and strains; uses Heaviside's step function as the test function and hence bypass domain integration. The reduced support size speeds up the computation, thus compensating for the increased number of unknowns. However, to the authors' knowledge, no attempts on obtaining the sharp bounds on the coercivity constant have so far been made. Other methods to arrive at such smooth solutions include Trefftz methods, the boundary integral method, and the discontinuous Galerkin methods. The Trefftz method (Gamallo and Astley 2007) share some similarities with the boundary element as well as penalty methods and thus requires a known solution to the homogeneous problem, which may not always be available for linear systems and/or locally linearized forms of many (possibly most) nonlinear systems. Here, as in the discontinuous Galerkin method (Engel *et al.* 2002), the higher order (typically C^1) continuity is only weakly enforced by penalizing the jump in the first order normal derivative across the inter-element boundary. Unfortunately, enforcing C^2 or still higher order global continuity in this way could be quite formidable. Moreover, boundary integral techniques (like the boundary element and Trefftz methods; Kita and Kamiya 1995), whilst bypassing domain integration, result in thickly populated stiffness matrices that demand special solvers and typically yield spurious solutions near the domain boundary.

The NURBS-based parametric method (Shaw and Roy 2008a) provides smooth solutions for the derivatives by combining the FE-based domain discretization with the global smoothness polynomial reproducing shape functions. Here a (bijective) geometric map, constructed through NURBS, is defined between the physical domain and a rectangular (cuboidal) parametric domain. The shape functions and their derivatives are obtained over the parametric domain (with trial functions constructed through tensor-product NURBS) so that polynomial reproduction and interpolation properties get satisfied over the physical domain. The selection of support size here is automatic and the integration cells are the NURBS cells themselves. But for most practical cases (e.g. for non-simply connected domains), a geometric map may not exist. To overcome this limitation, Shaw *et al.* (2008b) have proposed a NURBS-based parametric bridging method, wherein the physical domain is decomposed into a finite number of sub-domains such that a geometric map can be established for each of them. NURBS-based basis functions, constructed over each sub-domain, are appropriately blended across these sub-domains.

Use of the geometric map in the parametric methods above could cause ill-conditioning of the discretized equations and numerical pollution. Moreover, owing to the dual use of knots as particles, the integration of the weak form is not necessarily conformal. Since discretization of complex domains (say, in 2D) is best handled via triangulation and a scheme based on globally smooth shape functions constructed

using such triangles would work without a geometric map, we presently address the question on whether such a scheme can be worked out via triangular B-splines or triangular NURBS replacing the tensor-product NURBS in the parametric method. Specifically, we employ DMS splines (DMS being an acronym for Dahmen, Micchelli and Seidel, authors who introduced the spline; Dahmen *et al*. 1992) as the window (kernel) functions. The DMS-splines are defined as weighted sums of simplex splines over triangles. A key element of this construction is the knotclouds that help achieve C^{n-1} continuity of nth degree DMS-splines across inter-triangular boundaries. Presently, the physical domain in $\mathbb{R}^2$ is descritized into triangles using Delaunay triangulation. This provides this scheme a ready interface with the FEM wherein a similar discretization is often made use of. Unlike the FEM, however, the shape functions, derived based on the condition of reproduction of polynomials, possess inter-element continuity higher than C^0. Here the particles are located at the vertices as well as on the sides and interior of a triangle. The number of particles depends on the degree of DMS-splines. Depending on the choice of knots, the smooth shape functions, so derived, are supported within a close neighbourhood of the corresponding triangle. A procedure to generate the knotclouds whilst ensuring non-singularity is also outlined. Integration for the weak form equations is done over each triangle so that there is no misalignment of integration cells with the arrangement of nodes or the support of the globally smooth shape functions.

The paper is organised as follows. In Section 2, following a brief account of Delaunay triangulation, we provide the details of the construction of DMS-splines, generation of knotclouds and the evaluation procedure for such splines. In Section 3, a procedure for obtaining the globally smooth shape functions and their derivatives with DMS-splines as kernel functions is described. Numerical results of example problems are discussed in Section 4 followed by concluding remarks in Section 5.

2 DMS-Splines and Their Evaluation Schemes

Evaluation routines of DMS-splines have been developed by Fong and Seidel (1992), Pfeifle (1994) and Franssen (1995). The last author explains the evaluation scheme for s-variate DMS-splines of degree n. An essential element for constructing DMS-splines is a triangulation of the domain. In particular, we employ Delaunay triangulation $D_T(\mathbf{X})$, which is a triangulation for a set $\mathbf{X}$ of points in a plane such that no point in $\mathbf{X}$ is inside the circumcircle of any triangle in $D_T(\mathbf{X})$. Delaunay triangulation maximizes the minimum of all the angles of the triangles in the triangulation. By definition, the circumcircle of a triangle formed by three points from the original point set is empty if it does not contain vertices other than the three that define it. The Delaunay condition states that a triangle net is a Delaunay triangulation if

the circumcircles of all the triangles in the net are empty. This definition can be extended to 3-D domains by using a circumscribed sphere in place of the circumcircle.

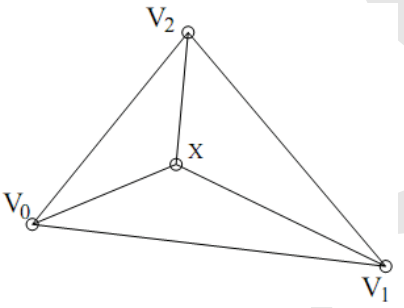


Figure 1: Illustration of barycentric co-ordinates of **x** with respect to a triangle v_0, v_1, v_2; they are the ratios between the areas of the separate sub-triangles to the area of the entire triangle

DMS-splines (also called as triangular B-spline), developed by Dahmen, Micchelli and Seidel (1992), are essentially weighted sums of simplex splines. They combine the overall global smoothness of simplex splines with the local control properties of B-patches (Franssen 1995). For completeness and a ready reference, DMS-splines in 2-D are briefly touched upon. Also outlined are the method of generating knotclouds and evaluation procedures of simplex and DMS-splines.

The j^{th} barycentric co-ordinate of a point **x** in $\mathbb{R}^2$, with respect to a triangle with vertices, v_0, v_1 and v_2 for $0 \leq j \leq 2$ is given by:

$$\lambda_j(\mathbf{x}|v_0v_1v_2) = \frac{Vol\left((v_0v_1v_2)\,(v_j{:=}x)\right)}{Vol\,(v_0v_1v_2)} \tag{1a}$$

Thus one may write

$$\mathbf{x} = \int_{j=0}^{2} \lambda_j(\mathbf{x}|v_0v_1v_2)\, v_j \tag{1b}$$

The half-open convex hull of a triangle V, denoted as $[V)$, is a subset of the convex hull of a triangle, such that for every point **x** in a triangulation one can determine exactly one triangle to which **x** belongs. Thus, for **x** lying on an edge shared by two triangles, it still only belongs to only one of the half-open convex hulls of those triangles. But, if **x** lies on the boundary of the discretized domain, it might not

belong to any triangle, although it does belong to the convex hull of the polygon. An exposition on how to arrive at the half-open convex hull of a triangle is available in Franssen (1995).

2.1 Simplex Splines

The simplex spline is a multivariate generalization of the well-known univariate B-splines. A degree n simplex spline is a smooth, degree n piecewise polynomial function defined over a set of $n+N_{dim}+1$ points $\mathbf{x} \in \mathbb{R}^{N_{dim}}$ called knots and the set of knots, knotset. If the knotset does not contain a collinear subset of (3 or more) knots then the simplex spline has overall C^{n-1} continuity. A detailed discussion of the theory of simplex splines is available in Micchelli (1995). We presently focus on bivariate simplex splines. A simplex spline defined over a knotset V is denoted as $M(.|V)$ and its value at $\mathbf{x} \in \mathbb{R}^2$ is denoted as $M(\mathbf{x}|V)$.

A constant simplex spline, defined over 3 knots and knotset $V = \{v_0, v_1, v_2\}$, is given by:

$$M(\mathbf{x}|V) = \begin{cases} \frac{1}{|det(V)|} & \text{if } \mathbf{x} \in [V) \\ 0 & \text{if } \mathbf{x} \notin [V) \end{cases} \tag{2}$$

A higher order simplex spline of degree n with knotset V is defined recursively as a weighted sum of three simplex splines, each of degree $n-1$. The number of vertices of the polygon over which nth degree simplex spline is defined is $m = n+2+1$. So the cardinality of V is $n+3$. The knotsets for the three $n-1$ degree simplex splines are chosen from V, leaving out one of the selected knots from V at a time as shown in Fig.2 in which the construction of a quadratic simplex spline is explained. The selected knots are marked by double circles. The support of the simplex spline is the half-open convex hull of V. The quadratic simplex spline is a weighted sum of three linear simplex splines, the domains of which are shown in Fig.2. Barycentric co-ordinates of $\mathbf{x}$ with respect to the selected triangle formed by the circled knots are used as the weights for the degree $n-1$ simplex splines when evaluated at $\mathbf{x}$. The recursive formula for the evaluation of degree n simplex splines is thus given by:

$$M(\mathbf{x}|V) = \int_{j=0}^{2} \lambda_j(\mathbf{x}|W) M\left(\mathbf{x}|V \setminus \{w_j\}\right) \tag{3}$$

where $W = \{w_0, w_1, w_2\} \subset V$ is the selected (non-degenerate) triangle. Any such W from the knotset V is sufficient to generate the simplex spline of degree n at $\mathbf{x}$. While the constant simplex spline is discontinuous at its domain boundary, the linear simplex spline is C^0 and the degree-n simplex spline is C^{n-1} everywhere (Franssen 1995).

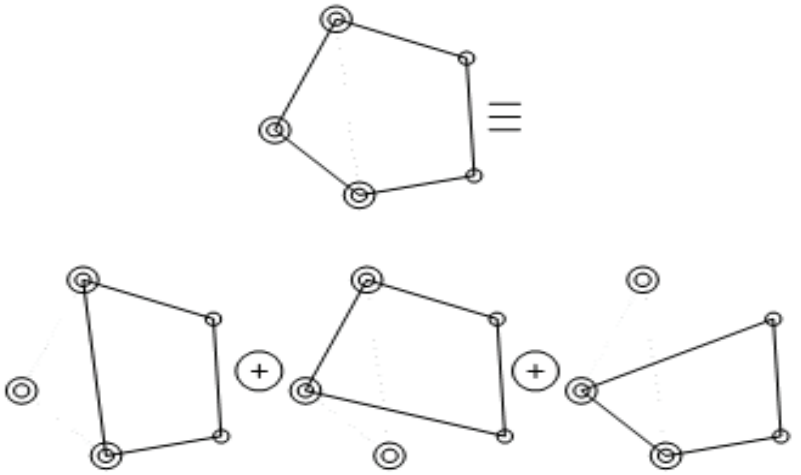


Figure 2: Selection of knotsets for 3 linear simplex splines to generate a quadratic simplex spline as their weighted sum: out of the three knots selected, one is left out, respectively, to form knotsets for the linear simplex splines

2.2 DMS-splines in $\mathbb{R}^2$

DMS-splines, which are weighted sum of simplex splines, are functions that combine the global smoothness of simplex splines with the desirable local control feature of B-patches (see Franssen 1995 for a detailed exposition). The domain of a DMS-spline surface is a proper triangulation $\mathscr{I} \subset \mathbb{R}^2$. In every vertex v_i, a knotcloud of $n+1$ knots, denoted as $\{v_{i0}, \ldots, v_{in}\}$ with $v_{i0} = v_i$, is defined. The knotsets are defined from these knotclouds. A set of control points in $\mathbb{R}^3$ are defined for each triangle $I \in \mathscr{I}$ for a degree n surface. The control points are denoted as c^I_β where β is a triple $(\beta_0, \beta_1, \beta_2)$ with $|\beta| = \beta_0 + \beta_1 + \beta_2 = n$. There are exactly $\frac{(n+1)(n+2)}{2}$ such β. The projections of the control points from $\mathbb{R}^3$ to $\mathbb{R}^2$ serve as particles in the generation of shape functions. A triangular domain, knotclouds and projection of control points on $\mathbb{R}^2$ (c^I_β) for constructing a quadratic DMS-spline in 2D is shown in Fig.3. The closer c^I_β lies to a vertex v_j of I, the more knots are taken from the corresponding knotcloud to form the knotset in the construction of simplex splines. Let $\check{\mathrm{V}}^I_\beta = \left\{ v^I_{0_{\beta_0}}, v^I_{1_{\beta_1}}, v^I_{2_{\beta_2}} \right\}$ be a triangle consisting of the last knots of the heads of knotclouds in V^I_β. The constant multiplier of $M\left(.|\mathrm{V}^I_\beta\right)$ in the calculation of a DMS-spline is $\left| det\left(\check{V}^I_\beta\right) \right|$. The DMS-spline basis functions at $\mathbf{x}$ corresponding to

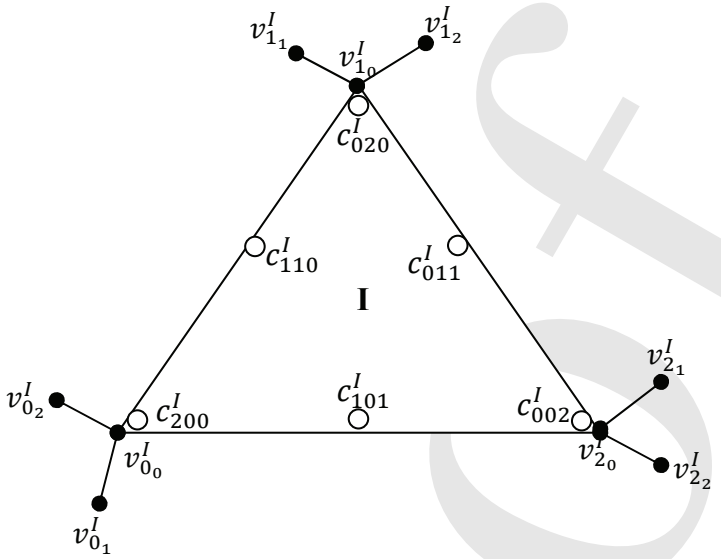


Figure 3: Parameters for a quadratic DMS-spline: the knotclouds of all vertices are the set $V^I_\beta = v_{(}0_0)^I, v_{(}0_1)^I, v_{(}0_2)^I, v_{(}1_0)^I, v_{(}1_1)^I, v_{(}1_2)^I, v_{(}2_0)^I, v_{(}2_1)^I, v_{(}2_2)^I$ and c^I_β are control points; the black circles represent knots and white circles, control points

c^I_β is $\left|det\left(\check{V}^I_\beta\right)\right| M\left(\mathbf{x}|\mathrm{V}^I_\beta\right)$. The point on a surface corresponding to $\mathbf{x}\in\mathbb{R}^2$ is given by

$$F(\mathbf{x}) = \int_{I\in\mathscr{I}}\int_{|\beta|=n}\left|det\left(\check{V}^I_\beta\right)\right| M\left(\mathbf{x}|\mathrm{V}^I_\beta\right) c^I_\beta \tag{4}$$

It is proved (Dahmen *et al.* 1992) that $\left|det\left(\check{V}^I_\beta\right)\right| M\left(\mathbf{x}|\mathrm{V}^I_\beta\right) \geq 0 \;\forall\; I \in \mathscr{I}$ and $\beta,\ |\beta| = n$ and $\int_{I\in\mathscr{I}}\int_{|\beta|=n}\left|det\left(\check{V}^I_\beta\right)\right| M\left(\mathbf{x}|\mathrm{V}^I_\beta\right) = 1$ (partition of unity). A triangle $I \in \mathscr{I}$ may also have non-zero contributions to points $\mathbf{x}$ in the domain that do not belong to the half-open convex hull of the triangle. This is a fundamental difference with the usual Bézier patch surface, whose values can be evaluated for each patch independently. This interference of triangles amongst themselves establishes the overall global smoothness of DMS-splines. The part of the surface corresponding to a specific triangle I is a DMS-patch, which is not only the contribution of this triangle, but the sum of contributions of all triangles to the point $\mathbf{x} \in I$.

2.3 Generation of Knotclouds

The knotclouds serve as a universal set for a triangle from which knotsets for simplex splines in the calculation of DMS-splines are derived. A major restriction on the placement of knotclouds is that no three knots in it can be collinear. This warrants extreme care to be exercised whilst generating the knotclouds over the (bounded) physical domain. In addition to collinearity, two additional restrictions are imposed to guarantee affine invariance (Franssen 1995):

1. If Ω^I is the interior of $\bigcap_{|\beta|\leq n}\left[\breve{V}_\beta^I\right]$, then $\Omega^I \neq \emptyset$, where $\emptyset$ is a subset with zero area.

2. In a triangle $I = \left\{v_p, v_q, v_r\right\}$, if one of the edges, say (v_p, v_q), is on the boundary, then the knotclouds for v_p and v_q must be placed on the opposite side of the boundary edge (v_p, v_q) with respect to that for v_r.

The first of the above is necessary for all DMS-splines. But the second one is important for surface reconstructions that do not include the whole of $\mathbb{R}^2$. Following this restriction, the knotclouds on the boundary of a domain may preferably be placed outside the domain. This will help avoid collinearity of knots and awkward polygonal shapes.

For the construction of degree n DMS-splines, n knots need to be added at each vertex of all the triangles in the triangulation. The following procedure is adopted for adding and placing knots at each vertex of a triangle:

- From the triangulation data available with Delaunay procedure, the boundary vertices of the physical problem domain are separated.
- For a vertex v_i, all the triangles which share it are located.
- The included angles $\left(\theta_{int}^j,\ j = 1, 2, \ldots, m\right)$ made by the triangles at v_i are calculated. Here θ_{int}^j is the j^{th} included angle for an internal vertex v_i (one that belongs to the domain interior) and m is the number of included angles (equal to number of triangles sharingv_i). The exterior angle (θ_{ext}) subtended by the boundary edges at the vertex v_i, when the latter is on the boundary, is also calculated.
- For an internal vertex v_i, elements of $\{\theta_{int}^j\}$ are sorted in the descending order of their magnitudes. The knot placement is done along lines originating from v_i and along directions obtained by dividing the angles as follows. If $m \geq n$, $\theta_{int}^j,\ j = 1, 2, \ldots, n$ are bisected to get the lines. If $m < n$ and $n = km + r$,

where r is the remainder of n/m, $\theta_{int}^{j},\ j = 1, 2, \ldots, r$ will be divided equally $(k+1)$ times and the rest $(m-r)$ angles, k times so that n lines are created.

- The distance of a knot from a vertex v_i, called knot-length, is chosen optimally (in some sense) based on the following observation. If it is too large or small, the knots of adjacent vertices may move either closer towards each other or to v_i, which may lead to irregular distribution of knots for the construction of simplex splines. The knot-length is arrived at as follows: The lengths of all edges of triangles meeting at v_i are calculated and the smallest among them is selected. Roughly 5 to 10% of this length is chosen as the knot-length for all knots to be placed near that vertex. This choice is found to work well for many problems, as numerically demonstrated later.

- A line is assumed along each division of the angle and a knot is placed on this line at the chosen knot-length from the associated v_i.

- For a boundary vertex v_i, the same procedure as that for the interior vertices is followed. The knots are then reflected (rotated by 1800) to the exterior of the domain.

The knotcloud generation for a bracket, which has internal and external boundaries, are shown in Fig.4(a) and (b) en route the construction of quadratic and cubic DMS-splines.

2.4 Recursive Evaluations of Simplex and DMS-splines in 2D

Simplex splines of degree n are weighted sums of three simplex splines of degree $n-1$. DMS-splines of degree n, on the other hand, are weighted sum of simplex splines of degree n.

2.4.1 Evaluation of simplex splines

As noted before (Eq. 2), a constant simplex spline, over a triangle V with vertices (knotset) $\mathrm{V} = \{v_0, v_1, v_2\}$, is evaluated as $M(\mathbf{x}|\mathrm{V}) = \begin{cases} \frac{1}{|det(V)|} & if\ \mathbf{x} \in [V) \\ 0 & if\ \mathbf{x} \notin [V) \end{cases}$. For the determinant $|det(V)|$ to be non-zero, $\mathbf{x}$ should lie in the half-open convex hull of V ($[V)$). The domain over which a linear simplex spline is defined in $\mathbb{R}^2$ is a quadrilateral (i.e., the polygon connecting the four knots is the quadrilateral). Let $\mathrm{V} = \{v_0, v_1, v_2, v_3\}$ be the knotset for a linear simplex spline. A degree n simplex spline is evaluated over an $(n+3)$-sided polygon as the weighted sum of three $(n-1)$ degree simplex splines. The knotset V contains $(n+3)$ knots, i.e.

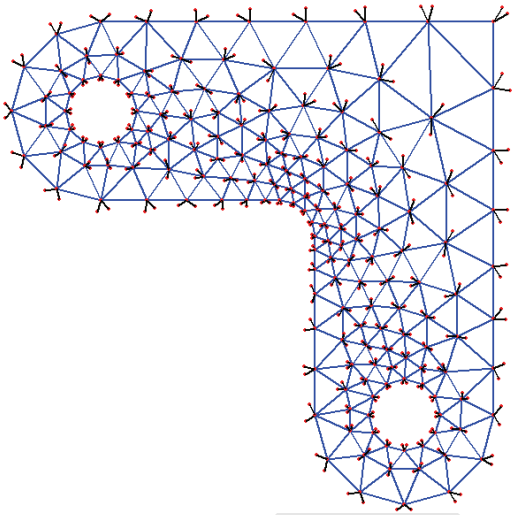


Fig 4(a)

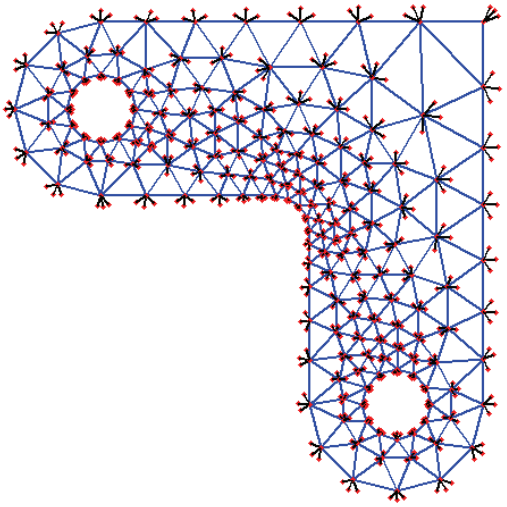


Fig 4(b)

Figure 4: Knot generation in a bracket after triangulation to construct (a) quadratic DMS-splines and (b) cubic DMS-splines

304 $\mathrm{V} = \{v_0, v_1, \ldots, v_{n+1}, v_{n+2}\}$. The recurrence formula is given as Eq. (3). The for-
305 mula involves the evaluation of 3^{n-1} linear simplex splines and 3^n constant sim-
306 plex splines in a naïve approach, as shown in Fig.5. But, by carefully identifying
307 the repetitions of simplex splines of various degree, number of evaluations can be
308 significantly reduced. The first three knots of knotsets of every simplex spline are
309 chosen as the set W (see Eq.3) in a straightforward approach. The number of in-
310 dependent linear and constant simplex splines to be evaluated will be $\frac{n(n+1)}{2}$ and
$\frac{(n+1)(n+2)}{2}$, respectively (Table 1).

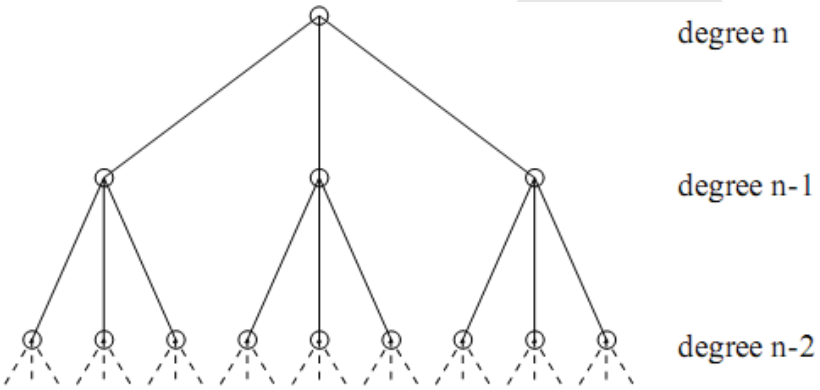


Figure 5: Each node in the tree represents a simplex spline to be evaluated to get the simplex spline represented by the root node; the number of simplex splines in each level = k^n, where k is the level of evaluation, starting from top

311

312 *2.4.2 Evaluation of DMS-splines*

313 A degree n DMS-spline basis function, evaluated at a point $\mathbf{x} \in \mathbb{R}^2$, is defined
314 over the control points c^I_β (Eq. 4) and is given by $\left|det\left(\check{V}^I_\beta\right)\right| M\left(\mathbf{x}|\mathrm{V}^I_\beta\right)$, where
315 $M\left(\mathbf{x}|\mathrm{V}^I_\beta\right)$ is a simplex splines of degree n and $\check{V}^I_\beta$, a triangle consisting of the
316 end knots of the knotclouds of the three vertices of the triangle I, corresponding
317 to a control point c^I_β, i.e. $\check{V}^I_\beta = \left\{v^I_{0_{\beta_0}}, v^I_{1_{\beta_1}}, v^I_{2_{\beta_2}}\right\}$. The number of control points is
318 equal to $\frac{(n+1)(n+2)}{2}$ in $\mathbb{R}^2$. For each triangle I in $\mathscr{I}$, the particles are located as the
319 projections of control points on the triangle, as shown in Figs.6(a) through (d).

Table 1: Number of evaluations of simplex splines required at each level

Level of evaluation (k)	Degree of simplex spline	No of evaluations in a naïve scheme	No of independent evaluations done
1	n	1	1
2	$n-1$	3	3
3	$n-2$	9	6
...	...	...	...
k	$n-k+1$	3^{k-1}	$k(k+1)/2$
...	...	...	...
n	1	3^{n-1}	$n(n+1)/2$
$n+1$	0	3^n	$(n+1)(n+2)/2$

3 Shape Functions and their Derivatives

Generation of globally smooth shape functions in 2D domain with DMS-splines as weight functions will be discussed in this section. The DMS-spline in $\mathbb{R}^2$ is denoted as $\Phi(x,y)$. A DMS-spline is supported over a triangle and its knotcloud neighbourhood defined as the polygon formed by connecting the knots, distributed following the restrictions mentioned in Section 2.3 and located close to the vertices of the triangle. DMS-spline is constructed corresponding to a given nodal point $\mathbf{x}$ in a physical domain in $\mathbb{R}^2$, over the triangle to the half-open convex hull of which $\mathbf{x}$ belongs. In general, DMS-splines satisfy the partition of unity and their derivatives (including the splines themselves) are globally smooth. But, a direct functional approximation based on these functions and their derivatives may sensitively depend on the placement of knots around the vertices of triangles. Thus, when the knots are far away from or very close to the vertices, the DMS-splines may numerically deviate from the partition of unity property and the total volume under their derivatives may not be close to zero, especially for $\mathbf{x}$ close to or on the boundary of the triangle. So, use of DMS-splines and their derivatives directly as shape functions and derivatives of shape functions (respectively) may lead to considerable errors in the approximation of a variable (results from a numerical example to illustrate this is given in Figs.11 and 12). We propose to study if an explicit reproduction of polynomials could help overcome this difficulty. Thus the shape functions are constructed by the reproduction of all the elements in a complete set of polynomials (a set containing all the terms in the Pascal's triangle) of degree $p \leq n$, with DMS-splines as weight functions. It is hoped that a numerical robust imposition of the partition of unity property and global smoothness for the shape functions might be possi-

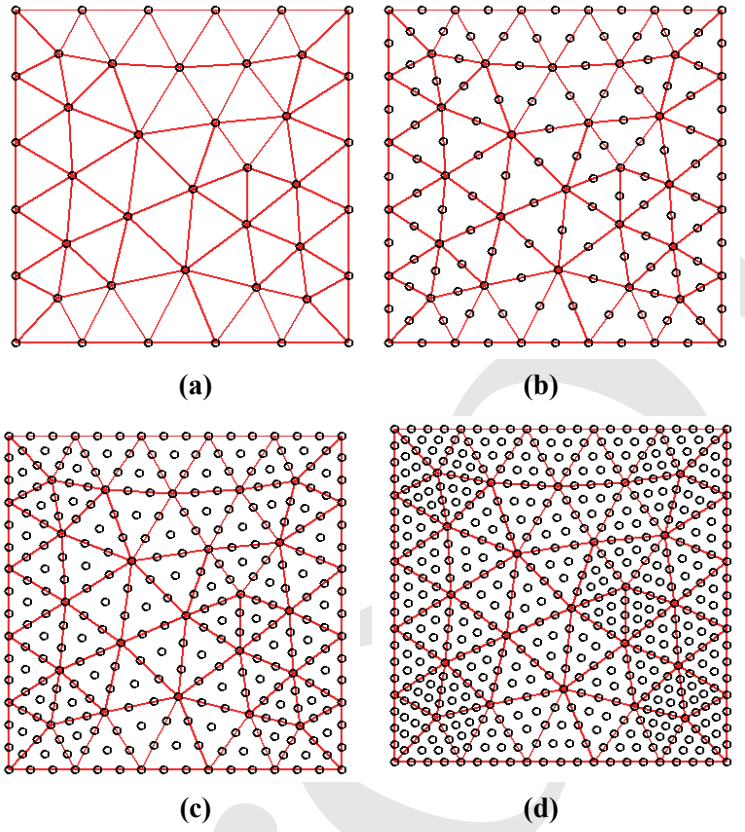


Figure 6: Distribution of particles (projections of control points) for DMS-splines of various degrees (n) in a triangulation; the black circles represent particles; (a) n=1, three particles in a triangle, (b) n=2, six particles in a triangle, (c) n=3, ten particles in a triangle, (d) n=4, fifteen particles in a triangle

ble through this route. Accordingly, we also propose to construct the derivatives of shape functions by reproducing the corresponding derivatives of elements of a complete polynomial set with DMS-splines as weight functions, following Shaw and Roy (2008a).

3.1 Generation of Shape Functions

Consider a bounded domain $\Omega \subset \mathbb{R}^2$ and a sufficiently smooth function $u(x,y)$, which is required to be approximated over the domain. Consistent with the FEM or a mesh-free method, an approximant $u^a(x,y)$ to the targeted function is assumed

to be of the form:

$$u^a(x,y) = \int_{i=1}^{N_{nd}} \Psi_i(x,y) u_i \tag{5}$$

where N_{nd} represents the number of nodes in the domain, $u_i = u(x_i, y_i)$ is the value of the targeted function at particle i and $\Psi_i(x,y)$ are the globally smooth shape functions. Following the practice in many mesh-free methods (e.g. the reproducing kernel methods (Liu, *et al*. 1995a, Shaw, *et al*. 2008a, etc), the latter can be written as:

$$\Psi_i(x,y) = \mathbf{H}^T(x-x_i, y-y_i)\, b(x,y) \Phi(x-x_i, y-y_i) \tag{6}$$

where $\mathbf{H}^T(x,y)$ is a set of polynomials defined as $\left\{x^\alpha y^\beta\right\}_{|\alpha+\beta|\le p}$, p is degree of polynomial $(p \le n)$, $b(x,y)$ are coefficients of the polynomials in $\mathbf{H}$ and $\Phi(x-x_i, y-y_i)$ is the DMS-spline based at (x_i, y_i) acting as the weight function. The coefficients $b(x,y)$ are obtained based on the following polynomial reproduction conditions:

$$\int_{i=1}^{N_{nd}} \Psi_i(x,y) 1 = 1 \tag{7}$$

$$\int_{i=1}^{N_{nd}} \Psi_i(x,y) \left(x_i^\alpha y_i^\beta\right) = x^\alpha y^\beta \quad |\alpha+\beta| \le p \tag{8}$$

$$\int_{i=1}^{N_{nd}} \Psi_i(x,y) \left((x-x_i)^\alpha (y-y_i)^\beta\right) = \delta_{|\alpha||\beta|,0} \quad |\alpha+\beta| \le p \tag{9}$$

$$\int_{i=1}^{N_{nd}} \mathbf{H}^T(x-x_i, y-y_i)\, b(x,y) \Phi(x-x_i, y-y_i)\, \mathbf{H}(x-x_i, y-y_i) = \mathbf{H}(0)$$

$$\mathbf{M}(x,y) b(x,y) = \mathbf{H}(0)$$

Here

$$\mathbf{M}(x,y) = \int_{i=1}^{N_{nd}} \mathbf{H}^T(x-x_i, y-y_i)\, \Phi(x-x_i, y-y_i)\, \mathbf{H}(x-x_i, y-y_i) \tag{10}$$

is the so-called moment matrix. So the coefficient vector is given by:

$$b(x,y) = \mathbf{M}^{-1}(x,y) \mathbf{H}(0)$$

provided that the moment matrix is invertible. We will be considering this issue shortly. Presently, the global shape functions in two-dimensions are given by:

$$\Psi_i(x,y) = \mathbf{H}^T(x-x_i, y-y_i)\, \mathbf{M}^{-1}(x,y) \mathbf{H}(0)\, \Phi(x-x_i, y-y_i) \tag{11}$$

If a nodal point **x** is inside a triangle, the support of the shape function is defined by a polygon that contains the triangle and is formed by the knotcloud associated with the vertices of that triangle as shown in Fig. 7a. If **x** falls on an edge shared by two triangles, the polygonal support of the shape function will include both the triangles sharing that edge as shown in Fig. 7b. As a third alternative, if **x** coincides with a common vertex shared by several triangles, then the support for the shape function will be in the form of a polygon containing all these triangles which share that vertex (Fig. 7c).

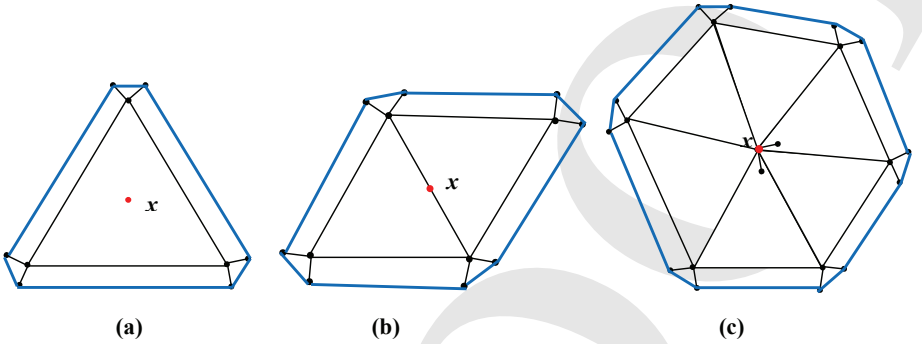


Figure 7: Supports (outer polygons) of shape functions when the nodal point x (red dot) is (a) inside a triangle, (b) on an edge shared by two triangles and (c) on one of the vertices of triangles; the knotcloud shown as black dots is for quadratic DMS-splines

3.2 Derivatives of Shape Functions

A stable and numerically accurate scheme for computing derivatives of globally smooth shape functions has been proposed by Shaw and Roy (2008a). It is based on the premise that γ^{th} derivatives of such shape functions reproduce γ^{th} derivatives of any arbitrary element of the space P_p of polynomials of degree $p \geq |\gamma|$. Using this principle, consistency relations for the derivatives may be written as:

$$\int_{i=1}^{N_{nd}} \Psi_i^{(\gamma)}(x,y)\mathbf{H}(x-x_i, y-y_i) = (-1)^{|\gamma|}\mathbf{H}^{(\gamma)}(0), \quad \forall\ |\gamma| \leq p \tag{12}$$

where $\Psi_i^{(\gamma)}(x,y) \triangleq D^{\gamma}\Psi_i(x,y)$ is the γ-th derivative that exactly reproduces γ^{th} derivatives of elements in the space P_p for $p \geq |\gamma|$. Now, $\Psi_i^{(\gamma)}(x,y)$ may be written as:

$$\Psi_i^{(\gamma)}(x,y) = H^T(x-x_i, y-y_i)\, b^{\gamma}(x,y)\Phi(x-x_i, y-y_i) \tag{13}$$

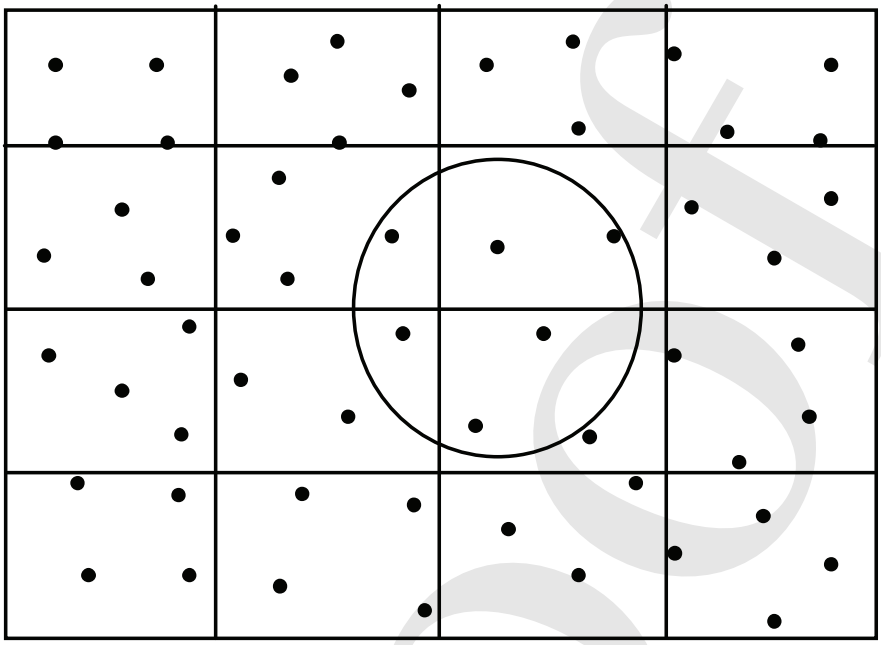


Figure 8: Misalignment of local support domain (circle) and background integration cells (rectangular) in a mesh-free method (particles are black dots)

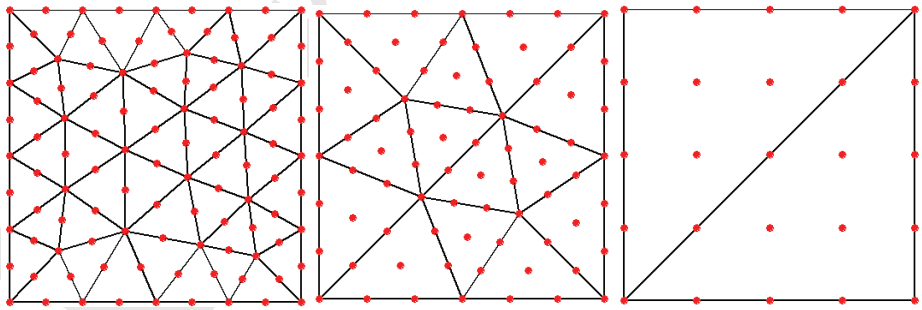


Figure 9: Aligned local domain (triangles) of DMS-spline basis functions and integration cells (triangles) in the present scheme (particles are red dots); fine to coarse triangulations

Here $b^{\gamma}(x,y)$ is the vector of unknown coefficients for derivative reproduction. The final form of $\Psi_i^{(\gamma)}(x,y)$ can be written as:

$$\Psi_i^{(\gamma)}(x,y) = (-1)^{|\gamma|}\mathbf{H}^{(\gamma)}(0)\,\mathbf{M}^{-1}(x,y)H^T(x-x_i,y-y_i)\,\Phi(x-x_i,y-y_i) \tag{14}$$

3.3 Invertibility of the Moment Matrix

In most of the mesh-free methods, the support of the shape functions (as determined through that of the weight or kernel function) needs to be user-specified subject to such considerations like ensuring invertibility of the moment matrix, adequacy of smoothness of shape functions and limiting computation time. Following Proposition 3.5 in Han and Meng (2001), the necessary condition for the moment matrix $\mathbf{M}(\mathbf{x})$ at a point $\mathbf{x} \in \Omega$ to be invertible is that $\mathbf{x}$ must be covered by at least $\dim(P_p) = \frac{(p+N_{dim})!}{p!N_{dim}!}$ shape functions, where $\dim(P_p)$ is the cardinality of the polynomial space of degree $\leq p$, and N_{dim} is the dimension of the domain Ω. So, if $\Omega \subset \mathbb{R}^2$, the number of shape functions required for ensuring invertibility of $\mathbf{M}(x,y)$ is $\frac{(p+2)!}{p!2!} = \frac{(p+2)(p+1)}{2}$. The number of nodes or particles introduced in a triangle (local support domain of shape functions) is $\frac{(n+2)(n+1)}{2}$ which will be equal to the number of DMS-spline basis functions corresponding to the control points. The DMS-splines are C^{n-1} continuous, everywhere, if the knots are in general position (i.e. no three knots are collinear) (Dahmen *et al.* 1992). Therefore, if $p \leq n$, the invertibility condition given by Han and Meng will be satisfied, in general. Now, if $\mathbf{x}$ falls on one of the edges or vertices of a triangle $I \in \mathscr{I} \subset \mathbb{R}^2$, it may not always belong to the half open convex hull of all the sub-triangles formed by subsets of the knotsets corresponding to the control points. This may lead to reduction of continuity of DMS-splines by one. In such a case n has to be kept greater than p to satisfy the invertibility requirement. In the present method, it is ensured that the minimum number of particles (shape functions) is included in a local support (triangle) to make the moment matrix invertible by choosing $n = p$ or $n = p+1$ as the nodal point $\mathbf{x}$ is inside or on the boundary of the local support, respectively.

3.4 Numerical Integration

In solving the weak form (as in a Galerkin projection) of a system of differential equations, a background mesh (similar to the mesh used in the FEM) is generally required in mesh-free methods for evaluating the integrals that arise. Integration is generally performed over each background cell by a quadrature rule (e.g. Gauss quadrature). Thus a meshing scheme like that in the FEM is anyway required. But, in doing so, the supports of shape functions may not often align (i.e., be identical) with the integration cells (Fig. 8). This may lead to inaccurate integration leading to loss of accuracy and convergence of mesh-free methods (Dolbow and

Belytschko 1999). This difficulty is generally overcome via a substantial increase in the order of quadrature in many mesh-free methods. In the present scheme, the physical domain is initially represented by a triangulation that enables construction of the DMS-spline basis functions. Recall from Section 3.1 that the local support of the shape functions is a triangle or triangles and associated knotcloud neighbourhood. But, since the knot lengths (distance to an extra knot associated to a vertex from the vertex) are chosen to be very small compared to the triangle edges, the local support may be considered as the triangle or triangles themselves for the purpose of numerical integration. So, the triangulation itself serves as integration mesh in the present scheme. Roughly speaking, a coarse triangulation with higher order quadrature or a fine triangulation with lower order quadrature should generally give good results. In the NURBS-based parametric bridge method (Shaw, *et al.* 2008b), since the mesh in the parametric space is used as integration cells, the number of such cells will be in excess of what might have been sufficient. Moreover, the alignment of integration cells and the supports of shape functions is usually not available for $n > 2$. In other methods like the element free Galerkin (EFG), higher order quadrature is used for more accurate integration. The present scheme is free of any such misalignment issues because of the uniformity in the placement of knots and the extra knots being not used as particles (nodes). Numerical experiments with the present method show that just a 3-point Gauss quadrature with quadratic DMS-splines (along with a rather fine triangulation; Fig 9a) or a 7-point Gauss quadrature with cubic or quartic DMS-splines (with a coarse triangulation; Fig 9b and 9c) is adequate to get good accuracy.

3.5 Imposition of essential boundary conditions

The shape functions in the present scheme, like many other mesh free shape functions, do not satisfy the Kronecker delta property. This makes them non-interpolating and hence a direct imposition of Dirichlet boundary conditions is not straightforward. Several solutions to this problem have been reported in the literature (Sonia and Antonio 2004, Cai and Zhu 2004, Shuyao and Atluri 2002, Zhu and Atluri 1998, etc.). The penalty method is adopted in this work to impose the Dirichlet (essential) boundary conditions. Thus, consider the boundary value problem given by:

$$\begin{aligned} &\triangle u = f \quad \text{in } \Omega \\ &u = u_d \quad \text{on } \Gamma_{\mathrm{d}}, \\ &\nabla u \cdot \mathbf{s} = g_s \quad \text{on } \Gamma_s \end{aligned} \tag{15}$$

where $\Gamma_d \cup \Gamma_s = \partial\Omega$ and $\mathbf{s}$ is the outward normal unit vector on $\partial\Omega$. If the shape functions are interpolating (so that the test functions are identically zero on the

Dirichlet boundary) the weak form associated with Eq. (15) is:

$$\int_\Omega \nabla v \cdot \nabla u \mathrm{d}\Omega - \int_{\Gamma_s} v \nabla u \cdot \mathbf{s} \mathrm{d}\Gamma = \int_\Omega v f \mathrm{d}\Omega \tag{16}$$

where v and u are the test and trial functions respectively. With the use of the penalizer α, the weak form can be rewritten as:

$$\int_\Omega \nabla v \cdot \nabla u \mathrm{d}\Omega - \int_{\Gamma_s} v g_s \mathrm{d}\Gamma = \int_\Omega v f \mathrm{d}\Omega + \int_{\Gamma_d} \alpha (u - u_d) v \mathrm{d}\Gamma \tag{17}$$

A proper choice of the penalizer α (usually chosen to be a large positive number) should lead to an accurate imposition of the Dirichlet boundary conditions. Indeed, as $\alpha \to \infty$, one can show that the solution u corresponding to the weak form (17) satisfies the Dirichlet condition. However, quite unlike the NURBS-based parametric bridge method (Shaw *et al*. 2008b), shape functions (especially those corresponding to the triangle vertices) via the present scheme are 'nearly interpolating'. This can be observed from the fact that the typical knot-length is presently smaller than the characteristic triangle size by at least an order or more. Hence, referring to Fig. 7(a) for instance, the triangle and the support of the shape function nearly overlap. In other words, using the partition of unity property and the fact that shape functions must be zero on the support boundary, it follows that the shape function for a nodal point nearly attains the value of unity at that node (especially if the node is a vertex).

3.6 Sparseness of the stiffness matrix

The smoothness in the functional approximation in most meshless methods typically require that the 'band' of interacting nodes is larger than that in the FEM. This leads to a larger bandwidth of the stiffness matrix and in turn an increased computational time for the inversion of the discretized equations. While the present method shares common FE-based domain decomposition (via Delaunay triangulation), the shape functions are nevertheless generated by the reproducing kernel particle technique. In this context, we recall that the compact support of the proposed shape function (as well as its derivatives) is nearly the triangle itself (with respect to which it is constructed) and that the compact support contains only the minimum number of particles required for the inversion of the moment matrix. This observation, combined with the adoption of the triangles themselves as background integration cells, naturally leads to the computed system stiffness matrix being sparse (i.e., with the smallest bandwidth permissible within the polynomial reproducing framework) and hence the sparse equation solvers, often used with commercially available FEM codes, can be employed in the present method too.

4 Numerical Results

First, we consider polynomial and non-polynomial (trigonometric and exponential) functions, and their derivatives up to fourth order and approximate them over a square domain with DMS-spline based global shape functions. The need for polynomial reproduction in deriving the shape functions is brought forth in the first example. Solutions of Laplace's and Poisson's equations and comparisons with exact solutions are presented next. Solutions of a few second order boundary value problems, involving plane stress and plane strain cases, are then attempted with the present method and comparisons provided with a few other available methods, e.g. the parametric mesh-free method, RKPM and FEM with Q4 (4-noded quadrilateral) as well as T6 (6-noded triangle) elements. Problems involving non-simply connected domains are also solved to demonstrate the advantages of the present method over the NURBS-based parametric bridge method. N_e denotes the number of triangles in the triangulation, in the following tables and figures.

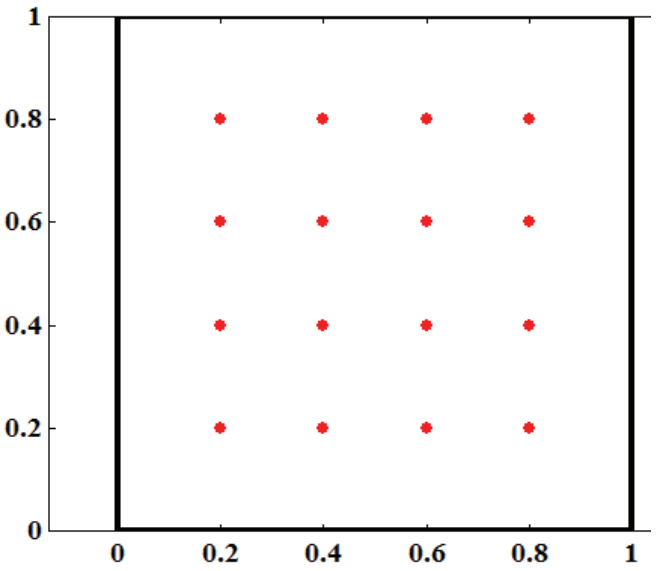


Figure 10: A square domain of dimension 1×1; the function values and their derivatives are calculated at points marked as red dots

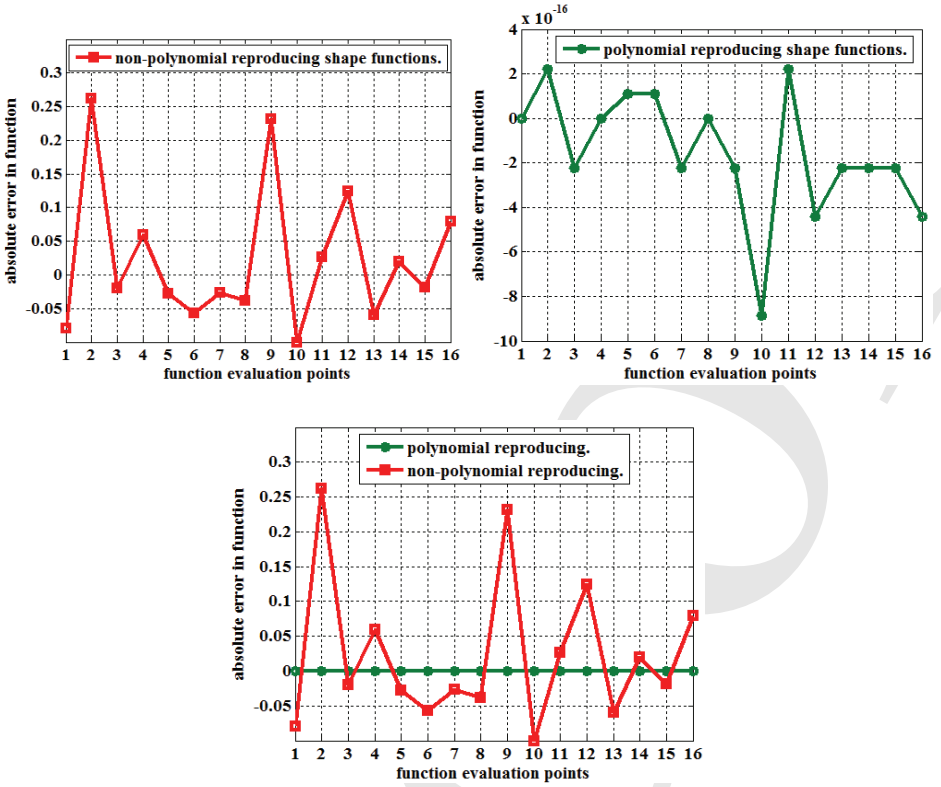


Figure 11: Plots of absolute error magnitudes for $f(x,y) = x+y$ at the evaluation points; (a) using standard DMS shape functions; (b) using polynomial reproducing DMS shape functions and (c) a direct comparison of errors in both cases

4.1 Approximating a Few Target Functions and their Derivatives

4.1.1 Polynomial functions

In order to illustrate the need for polynomial reproduction whilst generating the shape functions, a linear polynomial function, $f(x,y) = x+y$ and its first derivative with respect to x are evaluated at 16 points (represented by the red dots in Fig. 10) in a square domain of dimension 1×1 (unit2) with standard DMS-spline based shape functions (without polynomial reproduction) and polynomial reproducing DMS-spline based shape functions. In both cases, linear DMS-splines are used with the domain being discretized by 16 particles and 18 triangles. The absolute error magnitudes in both cases vis-à-vis the exact values are plotted against the function evaluation points in Figs. 11 and 12. Remarkably, large errors in the approximated

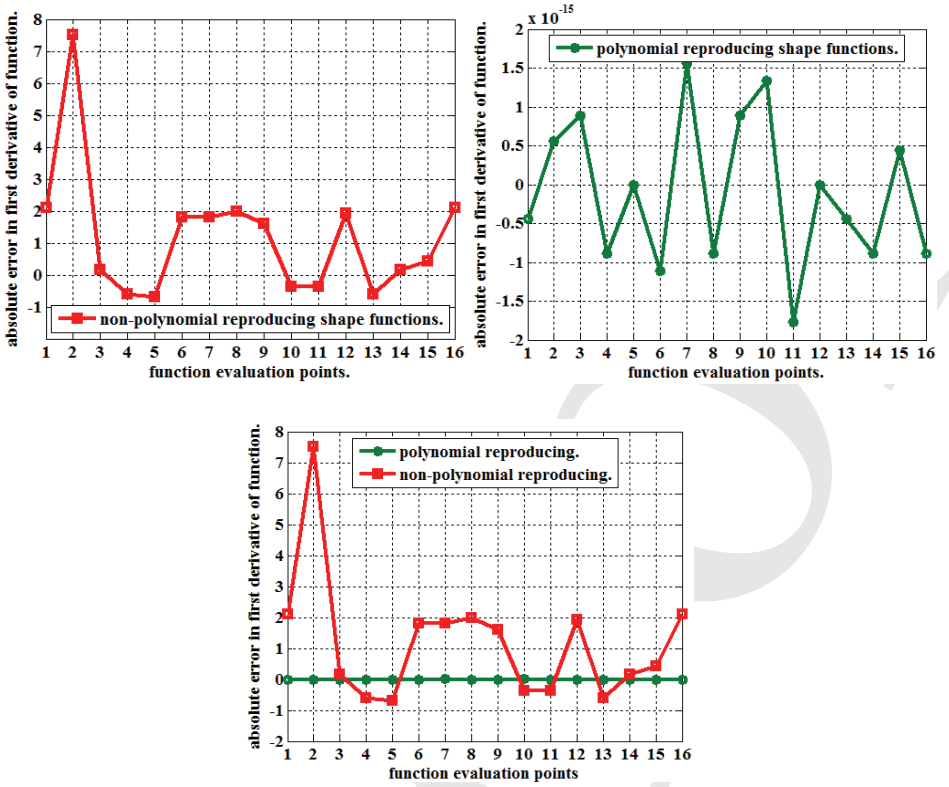


Figure 12: Plot of absolute errors in the first derivative of function $f(x,y) = x+y$ against the points at which the function is evaluated, (a) using non-polynomial reproducing shape functions (b) using polynomial reproducing shape functions and (c) comparison of errors in both cases

function and its derivative are observed if the explicit condition of polynomial reproduction is dropped in the derivation of the shape functions. Moreover, it is also observed from Table 2 that, while an increase in the degree of DMS-splines does not decrease the error substantially, an increase in the number of triangles reduces the error, to an extent, in the case of standard DMS shape functions. Similar results, shown in Table 3 through the polynomial reproducing shape functions, are once more indicative of the crucial role played by the polynomial reproduction step in obtaining an accurate functional approximation. Indeed, as verified via Table 3, relative L^2 error norms for polynomial functions and their derivatives up to the fourth order via the proposed shape functions are low even when there is only minimum number of triangles (two) in the triangulation of the domain. In these two

tables, the relative L^2 error norm is defined as (f^a represents the approximant for the targeted function f over a bounded domain Ω):

$$f - f^{a\ rel}_{L^2} = \frac{\left(\int_\Omega (f-f^a)^2 d\Omega\right)^{1/2}}{\left(\int_\Omega f^2 d\Omega\right)^{1/2}} \tag{18}$$

The construction of the standard DMS-spline basis functions is done over a triangulated domain with additional knots in the neighbourhood of the triangle vertices, as explained in Section 2. For a point $\mathbf{x}$ well within the domain (away from the triangle edges), the basis functions follow partition of unity and total volume under the derivative functions is zero (modulo very small approximation errors). But when $\mathbf{x}$ is close to the triangle edges and the knots are too close or far away from the vertices, errors creep in computing the basis functions and their derivatives, which deviate from the above properties. This high sensitivity of the basis functions to the knot placement is consistently observed across the numerical experiments. Shape functions via polynomial reproduction are thus consistently adopted to overcome this difficulty.

4.1.2 Non-polynomial functions

Trigonometric and exponential functions, and their derivatives up to the fourth order are now computed over the same square domain and at the same points as in the case of polynomial functions and relative L^2 error norms are tabulated in Table 4.

As expected, we observe from Table 4 that as the functional complexity (say, in terms of its departure from polynomials as measured by the number of monomial bases needed for approximation over an interval) increases, so do the relative L^2 error norms. To have an understanding of the order of errors involved, plots of the exact and approximated fourth order derivatives of the function $f(x,y) = sin(\pi x)\, cos(\pi y)$ are given in Fig. 13.

4.2 Laplace's and Poisson's Equations

The second order Laplace's and Poisson's equations in 2D, often used as workhorse examples for validating new schemes in computational mechanics, are given respectively as:

$$\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} = 0 \tag{19}$$

and

$$\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} = g(x,y) \tag{20}$$

Table 2: Relative $\mathbf{L}^2$ error norms of polynomial functions and their derivatives with non-polynomial reproducing (standard) DMS-spline shape functions

$f(x,y)$	$(x+y)$	$(x+y)^2$	$(x+y)^3$	$(x+y)^4$	$(x+y)^5$
n	1	2	3	4	5
N_{nd}	4	9	16	25	36
N_e	2	2	2	2	2
$f-f^{a\,rel}_{L^2}$	1.79×10^{0}	3.79×10^{0}	6.08×10^{0}	9.96×10^{0}	1.50×10^{1}
$\frac{\partial f}{\partial x}-\left(\frac{\partial f}{\partial x}\right)^{a\,rel}_{L^2}$	1.76×10^{1}	2.61×10^{1}	3.57×10^{1}	6.75×10^{1}	9.46×10^{1}
$\frac{\partial f}{\partial y}-\left(\frac{\partial f}{\partial y}\right)^{a\,rel}_{L^2}$	1.90×10^{1}	2.93×10^{1}	3.28×10^{1}	7.13×10^{1}	9.04×10^{1}
n	2	3	4	5	6
particles	9	16	25	36	49
triangles	2	2	2	2	2
$f-f^{a\,rel}_{L^2}$	1.83×10^{0}	3.43×10^{0}	5.97×10^{0}	9.25×10^{0}	1.44×10^{1}
$\frac{\partial f}{\partial x}-\left(\frac{\partial f}{\partial x}\right)^{a\,rel}_{L^2}$	1.19×10^{1}	1.86×10^{1}	3.71×10^{1}	5.26×10^{1}	9.40×10^{1}
$\frac{\partial f}{\partial y}-\left(\frac{\partial f}{\partial y}\right)^{a\,rel}_{L^2}$	1.33×10^{1}	1.64×10^{1}	3.75×10^{1}	4.65×10^{1}	9.97×10^{1}
n	1	2	3	4	5
particles	16	49	100	169	256
triangles	18	18	18	18	18
$f-f^{a\,rel}_{L^2}$	4.18×10^{-1}	6.10×10^{-1}	9.37×10^{0}	1.35×10^{0}	2.40×10^{0}
$\frac{\partial f}{\partial x}-\left(\frac{\partial f}{\partial x}\right)^{a\,rel}_{L^2}$	9.17×10^{0}	1.35×10^{1}	2.23×10^{1}	3.89×10^{1}	7.36×10^{1}
$\frac{\partial f}{\partial y}-\left(\frac{\partial f}{\partial y}\right)^{a\,rel}_{L^2}$	9.22×10^{0}	1.50×10^{1}	2.38×10^{1}	3.58×10^{1}	5.43×10^{1}

where, $f(x,y)$ and $g(x,y)$ are functions in $\mathbb{R}^2$. Weak forms of the homogeneous Laplace's equation and inhomogeneous Poisson's equation under Diritchlet boundary conditions are solved via the present scheme over square and triangular domains.

Square domain

The square domain (of size 1×1) is the same as that used in the previous examples. The relative L^2 and H^1 (Sobolev) error norms are computed and tabulated (Table 5). The relative H^1 error norm is defined as:

$$f-f^{a\,rel}_{H^1}=\frac{\left(\int_\Omega\left[(f-f^a)^2+\left(f_{,x}-f^a_{,x}\right)^2+\left(f_{,y}-f^a_{,y}\right)^2\right]d\Omega\right)^{1/2}}{\left(\int_\Omega\left[f^2+{f_{,x}}^2+{f_{,y}}^2\right]d\Omega\right)^{1/2}} \tag{21}$$

Table 3: Relative $\mathbf{L}^2$ error norms of polynomial functions and their derivatives with polynomial reproducing DMS-spline based global shape functions

$f(x,y)$	$(x+y)$	$(x+y)^2$	$(x+y)^3$	$(x+y)^4$	$(x+y)^5$
n	2	3	4	5	6
p	1	2	3	4	5
N_{nd}	9	16	25	36	49
N_e	2	2	2	2	2
$\left\|f-f^{a\,rel}\right\|_{L^2}$	2.44×10^{-15}	2.47×10^{-13}	4.78×10^{-12}	9.74×10^{-10}	1.23×10^{-7}
$\left\|\frac{\partial f}{\partial x}-\left(\frac{\partial f}{\partial x}\right)^{a\,rel}\right\|_{L^2}$	4.13×10^{-15}	1.35×10^{-12}	9.05×10^{-11}	1.14×10^{-8}	5.96×10^{-7}
$\left\|\frac{\partial f}{\partial y}-\left(\frac{\partial f}{\partial y}\right)^{a\,rel}\right\|_{L^2}$	2.75×10^{-15}	4.68×10^{-13}	2.23×10^{-11}	3.37×10^{-9}	5.95×10^{-7}
$\left\|\frac{\partial^2 f}{\partial x^2}-\left(\frac{\partial^2 f}{\partial x^2}\right)^{a\,rel}\right\|_{L^2}$	-	2.31×10^{-12}	4.26×10^{-10}	2.86×10^{-8}	4.68×10^{-6}
$\left\|\frac{\partial^2 f}{\partial y^2}-\left(\frac{\partial^2 f}{\partial y^2}\right)^{a\,rel}\right\|_{L^2}$	-	3.09×10^{-13}	6.21×10^{-11}	9.71×10^{-9}	3.87×10^{-6}
$\left\|\frac{\partial^2 f}{\partial x\partial y}-\left(\frac{\partial^2 f}{\partial x\partial y}\right)^{a\,rel}\right\|_{L^2}$	-	6.45×10^{-13}	2.00×10^{-10}	1.93×10^{-8}	2.97×10^{-7}
$\left\|\frac{\partial^3 f}{\partial x^3}-\left(\frac{\partial^3 f}{\partial x^3}\right)^{a\,rel}\right\|_{L^2}$	-	-	5.42×10^{-10}	1.29×10^{-7}	6.05×10^{-6}
$\left\|\frac{\partial^3 f}{\partial y^3}-\left(\frac{\partial^3 f}{\partial y^3}\right)^{a\,rel}\right\|_{L^2}$	-	-	1.68×10^{-10}	2.17×10^{-8}	4.76×10^{-6}
$\left\|\frac{\partial^4 f}{\partial x^4}-\left(\frac{\partial^4 f}{\partial x^4}\right)^{a\,rel}\right\|_{L^2}$	-	-	-	2.94×10^{-7}	1.14×10^{-5}
$\left\|\frac{\partial^4 f}{\partial y^4}-\left(\frac{\partial^4 f}{\partial y^4}\right)^{a\,rel}\right\|_{L^2}$	-	-	-	1.99×10^{-8}	1.70×10^{-5}
$\left\|\frac{\partial^4 f}{\partial x^2\partial y^2}-\left(\frac{\partial^4 f}{\partial x^2\partial y^2}\right)^{a\,rel}\right\|_{L^2}$	-	-	-	9.09×10^{-8}	1.32×10^{-5}

where ',' stands for partial differentiation. Presently, the exact solution for Laplace's equation is given by:

$$f(x,y) = -x^3 - y^3 + 3x^2y + 3xy^2 \tag{22}$$

with the trace of the above function on the domain boundary providing the Dirichlet boundary condition. 3- and 7-point Gauss quadrature rules are used for numerical integration with $p = 2$ and $p = 3,\ 4$, respectively. Since derivatives involved in the relative H^1 error norm are not the primary variables, they are computed at the nodes with polynomial degree $p-1$. It is clear from the table that, while using $p = 3$ and 4 considerably reduces the relative L^2 error norm, relative H^1 error norm reduces substantially with $p = 4$. This is clearly due to the fact that the targeted polynomial function within the domain is of degree 3. For two triangles and 9 nodes, relative L^2 error norm corresponding to $p = 2$ appears very small owing to 8 out of the 9

Table 4: Relative $\mathbf{L}^2$ error norms of trigonometric and exponential functions and their derivatives

$f(x,y)$	$sin\,(xy)$	$sin\,(\pi x)\,cos\,(\pi y)$	$e^{(x+y)}$	$e^{(xy)}$
n	6	6	6	6
p	5	5	5	5
N_{nd}	361	361	361	361
N_e	18	18	18	18
$f - f^{a\,rel}_{L^2}$	1.65×10^{-7}	4.16×10^{-6}	1.03×10^{-7}	2.97×10^{-7}
$\frac{\partial f}{\partial x} - \left(\frac{\partial f}{\partial x}\right)^{a\,rel}_{L^2}$	9.15×10^{-6}	2.79×10^{-4}	4.51×10^{-6}	1.60×10^{-5}
$\frac{\partial f}{\partial y} - \left(\frac{\partial f}{\partial y}\right)^{a\,rel}_{L^2}$	5.17×10^{-6}	2.07×10^{-4}	1.65×10^{-6}	8.43×10^{-6}
$\frac{\partial^2 f}{\partial x^2} - \left(\frac{\partial^2 f}{\partial x^2}\right)^{a\,rel}_{L^2}$	3.19×10^{-4}	6.89×10^{-3}	5.48×10^{-5}	5.47×10^{-4}
$\frac{\partial^2 f}{\partial y^2} - \left(\frac{\partial^2 f}{\partial y^2}\right)^{a\,rel}_{L^2}$	1.15×10^{-4}	5.95×10^{-3}	3.05×10^{-5}	1.45×10^{-4}
$\frac{\partial^2 f}{\partial x \partial y} - \left(\frac{\partial^2 f}{\partial x \partial y}\right)^{a\,rel}_{L^2}$	2.04×10^{-4}	3.88×10^{-3}	1.89×10^{-5}	2.72×10^{-4}
$\frac{\partial^3 f}{\partial x^3} - \left(\frac{\partial^3 f}{\partial x^3}\right)^{a\,rel}_{L^2}$	6.61×10^{-3}	3.89×10^{-1}	2.56×10^{-3}	1.61×10^{-2}
$\frac{\partial^3 f}{\partial y^3} - \left(\frac{\partial^3 f}{\partial y^3}\right)^{a\,rel}_{L^2}$	3.00×10^{-3}	3.05×10^{-1}	1.20×10^{-3}	7.72×10^{-3}
$\frac{\partial^4 f}{\partial x^4} - \left(\frac{\partial^4 f}{\partial x^4}\right)^{a\,rel}_{L^2}$	8.14×10^{-2}	$1.51 \times 10^{+1}$	9.59×10^{-2}	2.50×10^{-1}
$\frac{\partial^4 f}{\partial y^4} - \left(\frac{\partial^4 f}{\partial y^4}\right)^{a\,rel}_{L^2}$	8.29×10^{-2}	8.08×10^{0}	5.13×10^{-2}	2.05×10^{-1}
$\frac{\partial^4 f}{\partial x^2 \partial y^2} - \left(\frac{\partial^4 f}{\partial x^2 \partial y^2}\right)^{a\,rel}_{L^2}$	9.63×10^{-2}	7.46×10^{0}	5.03×10^{-2}	1.83×10^{-1}

nodes being on the Dirichlet boundary. The plots of error norms are given in Figs. 14 and 15.

Similar results for Poisson's equation on the square domain are given in Table 6 and Fig. 16. If one chooses the solution for Poisson's equation as: $f(x,y) = (x+y)^2$, the forcing function $g(x,y)$ in equation (20) becomes equal to 4. The relative L^2 norms for $n = 2$ and 3 and relative H^1 norm for $n = 3$ are of the order 10^{-11} even with relatively fewer triangles in the triangulation.

Triangular domain

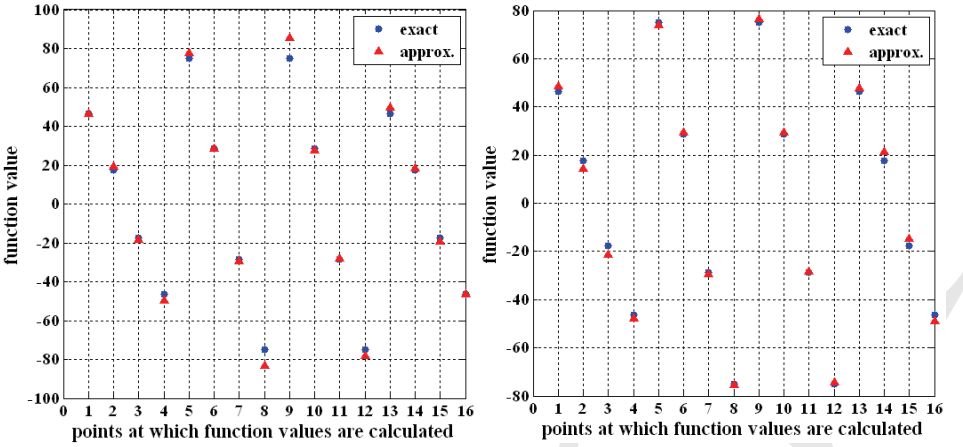
We consider an isosceles triangular domain having base and height equal to unity (Fig.17). The error norms (Tables 7 and 8 and Figs. 18 and 19) follow almost the same orders as those over the previously adopted square domain. DMS-splines with degree 3 gave good results in terms of relative L^2 error norms for solution

Table 5: Relative $\mathbf{L}^2$ and $\mathbf{H}^1$ error norms for different values of n and p for the solution of Laplace's equation on a 1×1 square domain; exact solution is $\mathbf{f}(\mathbf{x},\mathbf{y}) = -\mathbf{x}^3 - \mathbf{y}^3 + 3\mathbf{x}^2\mathbf{y} + 3\mathbf{x}\mathbf{y}^2$

n	p	N_{nd}	N_ε	relative L^2 norm	relative H^1 norm
2	2	9	2	2.6770e-015	8.7596e-001
2	2	49	18	5.4383e-004	3.3199e-001
2	2	85	34	2.3485e-004	2.3802e-001
2	2	105	44	8.8498e-005	1.9596e-001
2	2	133	56	1.0864e-004	1.8392e-001
2	2	149	64	5.8411e-005	1.5327e-001
2	2	189	82	5.8926e-005	1.3130e-001
2	2	253	112	3.3418e-005	1.2021e-001
2	2	321	144	2.0978e-005	1.1105e-001
2	2	513	236	6.6280e-006	8.2411e-002

Table 6: Relative $\mathbf{L}^2$ and $\mathbf{H}^1$ error norms for different values of n and p for the solution of Poisson's equation on a 1×1 square domain; exact solution is $\mathbf{f}(\mathbf{x},\mathbf{y}) = (\mathbf{x}+\mathbf{y})^2$

n	p	N_{nd}	N_ε	relative L^2 norm	relative H^1 norm
2	2	9	2	1.1085e-015	5.7094e-001
2	2	49	18	2.8673e-015	1.3520e-001
2	2	85	34	2.2191e-015	9.9984e-002
2	2	133	56	3.7470e-015	7.1038e-002
2	2	241	106	4.5526e-015	4.9335e-002
3	3	16	2	1.7116e-012	8.4630e-012
3	3	100	18	1.2748e-012	8.6641e-012
3	3	178	34	9.8971e-013	1.0475e-011
3	3	283	56	1.3155e-012	1.3720e-011
3	3	520	106	1.1802e-012	1.5523e-011



(a) Values of $\frac{\partial^4 f}{\partial x^4}$ at points 1 to 16. **(b) Values of $\frac{\partial^4 f}{\partial y^4}$ at points 1 to 16.**

Figure 13: Deviation of approximate values from exact: fourth derivatives of function $f(x,y) = \sin(\pi x)\cos(\pi y)$

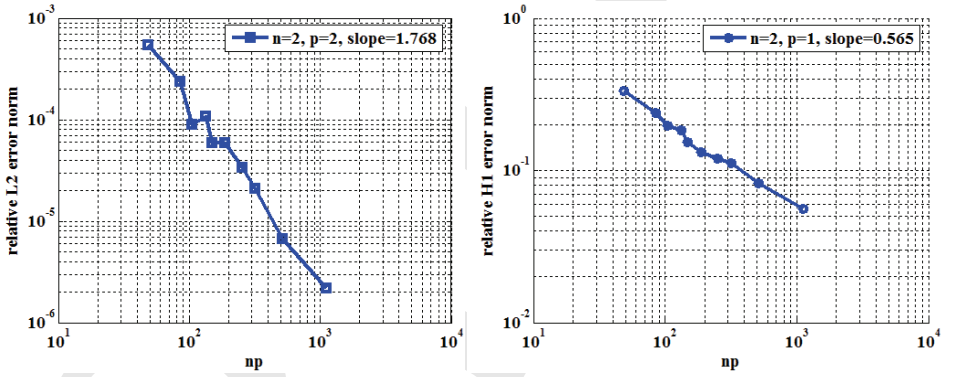


Figure 14: Error plots for solution of Laplace's equation on a square domain with degree of DMS-splines (n) = 2 on log-log scale

of Laplace's equation with a degree 3 polynomial as the exact solution over both square and triangular domains. H^1 error norms are remarkably low with DMS-splines of degree 4 (square domain). For Poisson's equation with a second degree polynomial as the exact solution, DMS-splines with degree 2 in the approximation scheme gave excellent results, as expected, in terms of relative L^2 error norms for both domains; however in terms of relative H^1 error norms, degree 3 DMS-splines

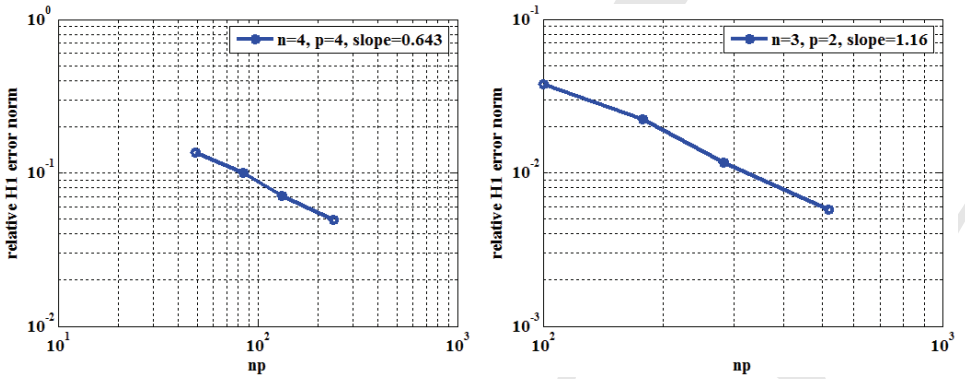


Figure 15: H^1 error plots for solution of Laplace's equation on a square domain with degree of DMS-splines (n) = 3 and 4 on log-log scale

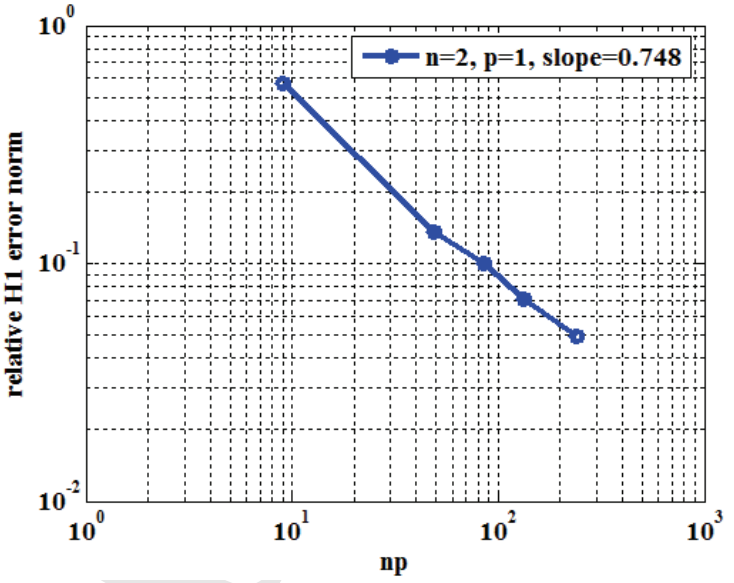


Figure 16: H^1 error plot for solution of Poisson's equation on a square domain with degree of DMS-splines (n) = 2 on log-log scale

Table 7: Relative $\mathbf{L}^2$ and $\mathbf{H}^1$ error norms for different values of n and p for the solution of Laplace's equation over a triangular domain; exact solution is $\mathbf{f}(\mathbf{x},\mathbf{y}) = -\mathbf{x}^3 - \mathbf{y}^3 + 3\mathbf{x}^2\mathbf{y} + 3\mathbf{x}\mathbf{y}^2$

n	p	N_{nd}	N_ε	relative L^2 norm	relative H^1 norm
2	2	6	1	0.0000e+000	8.5495e-001
2	2	21	6	9.6275e-004	4.8196e-001
2	2	28	9	1.8471e-004	3.3962e-001
2	2	45	16	6.7271e-005	2.7053e-001
2	2	84	33	7.0390e-005	1.8635e-001
2	2	146	61	3.6987e-005	1.4596e-001
2	2	295	130	1.0125e-005	9.3762e-002
2	2	1059	496	1.1654e-006	4.9086e-002
3	3	10	1	7.7432e-013	4.1871e-001
3	3	55	9	1.5094e-012	4.2113e-002
3	3	91	16	1.5026e-012	2.4358e-002
3	3	121	22	1.7792e-012	2.1451e-002
3	3	166	31	2.1267e-012	1.6027e-002
3	3	175	33	1.7908e-012	1.5216e-002
3	3	310	61	1.8506e-012	6.6388e-003
3	3	637	130	1.8947e-012	3.6726e-003

in the approximation scheme perform better in both cases.

The motivation for choosing a triangular domain is that it does not admit a bijective geometric map with a square parametric domain. Hence, if NURBS-based parametric bridge method (Shaw *et al.* 2008b) is made use of for the same problem, at least three sub-domains, each of which is geometrically bijective with the square parametric domain, must be defined on the triangle and assembly performed to arrive at the solution. This difficulty is not at all encountered in the DMS-based approach, which is thus eminently more suited to irregular domain geometries. This point is further illustrated in some of the examples involving plane stress and plain strain problems considered in the next section.

4.3 Plane Stress and Plane Strain Problems

First, we consider linear isotropic cases of plane stress and plane strain problems. Here, we also aim at comparing some of the results with a few other mesh-free methods and the FEM.

Table 8: Relative $\mathbf{L}^2$ and $\mathbf{H}^1$ error norms for different values of n and p for the solution of Poisson's equation on a triangular domain. Exact solution is $\mathbf{f}(\mathbf{x},\mathbf{y}) = (\mathbf{x}+\mathbf{y})^2$.

n	p	N_{nd}	N_ε	relative L^2 norm	relative H^1 norm
2	2	6	1	0.0000e+000	6.9300e-001
2	2	28	9	1.4187e-015	1.7598e-001
2	2	45	16	1.4928e-015	1.2513e-001
2	2	59	22	2.2053e-015	9.0950e-002
2	2	84	33	1.9395e-015	8.1210e-002
2	2	146	61	1.9862e-015	6.2223e-002
2	2	295	130	1.8111e-015	3.9158e-002
3	3	10	1	3.7988e-013	3.3324e-012
3	3	55	9	1.0115e-012	5.7882e-012
3	3	91	16	1.0443e-012	8.3737e-012
3	3	121	22	1.0979e-012	8.3986e-012
3	3	175	33	1.2309e-012	1.0973e-011
3	3	310	61	1.1232e-012	1.4606e-011

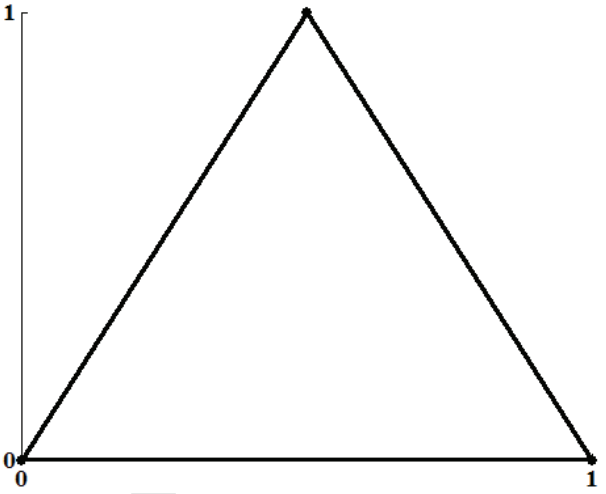


Figure 17: Triangular domain with three particles at its vertices

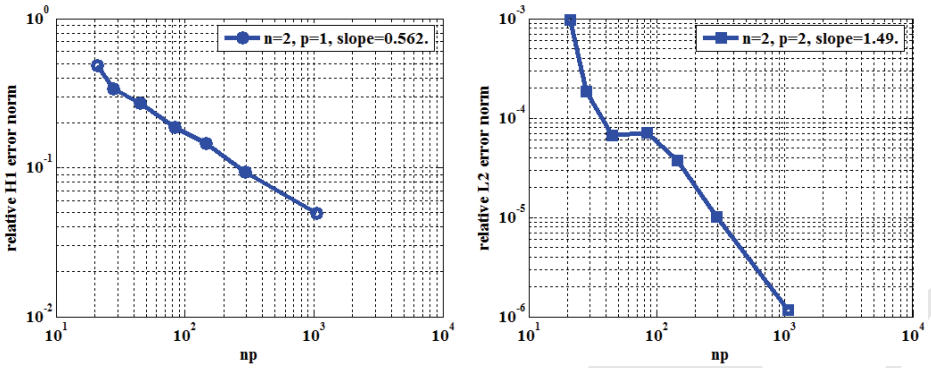


Figure 18: Error plots for solutions of Laplace's equation over a triangular domain with degree of DMS-splines (n) = 2 on log-log scale

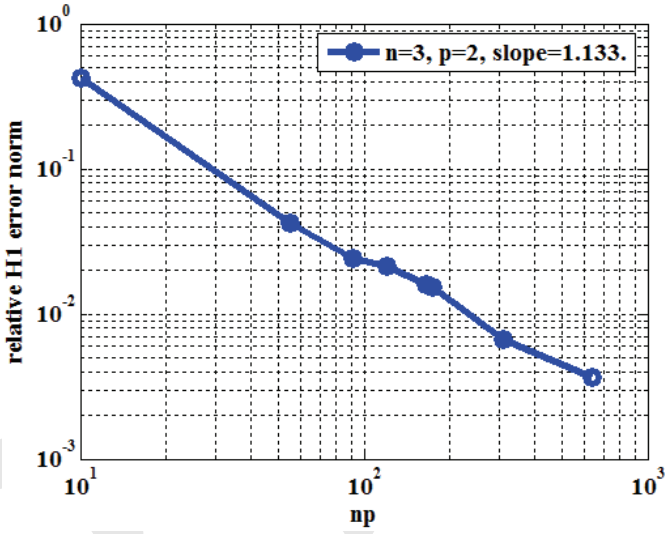


Figure 19: H^1 error plot for solution of Laplace's equation over a triangular domain with degree of DMS-splines (n) = 3 on log-log scale

4.3.1 Cook's membrane

Cook's membrane is one of the benchmark problems in two-dimensional elasticity, especially in assessing the performance and robustness of a scheme against possible ill-conditioning of the discretized equations. Thus, we consider a skewed plate-like structure (of unit thickness) with the planar dimensions as indicated in Fig.20. It

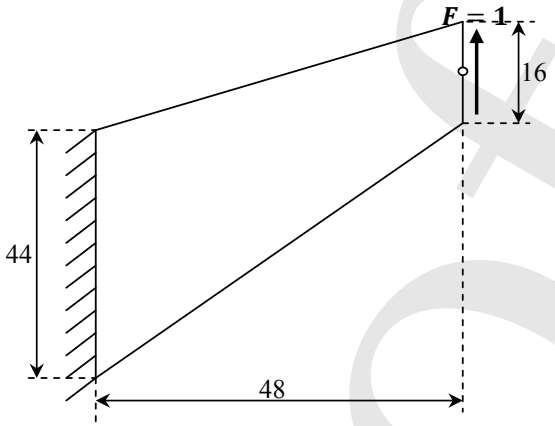


Figure 20: Cook's membrane: the small white circle on its right edge represents the point at which its vertical displacement is computed

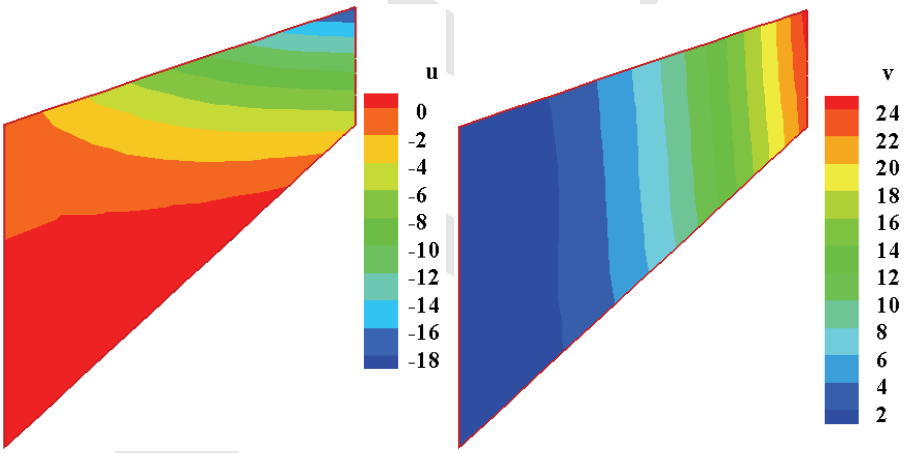


Figure 21: The Cook's membrane with displacement contours; (a) x-displacement (u), (b) y-displacement (v)

is clamped at one end and a unit shear force is applied at the other. In the plane stress case, Young's modulus (E) is assumed as unity and Poisson's ratio $\nu = 0.33$. Vertical deflection of the centre point of its free edge has been reported by many authors (such as Simo *et al*. 1989) to be 23.96 (units) as an FEM-based converged solution. The deflection contours of the Cook's membrane are shown in Fig.21. Comparisons of convergence to the reported solution with those via the RKPM, parametric mesh free method, FEM Q4 (4-noded quadrilateral elements), FEM T6 (6-noded triangular elements) and, finally, the present scheme are reported in Fig.22.

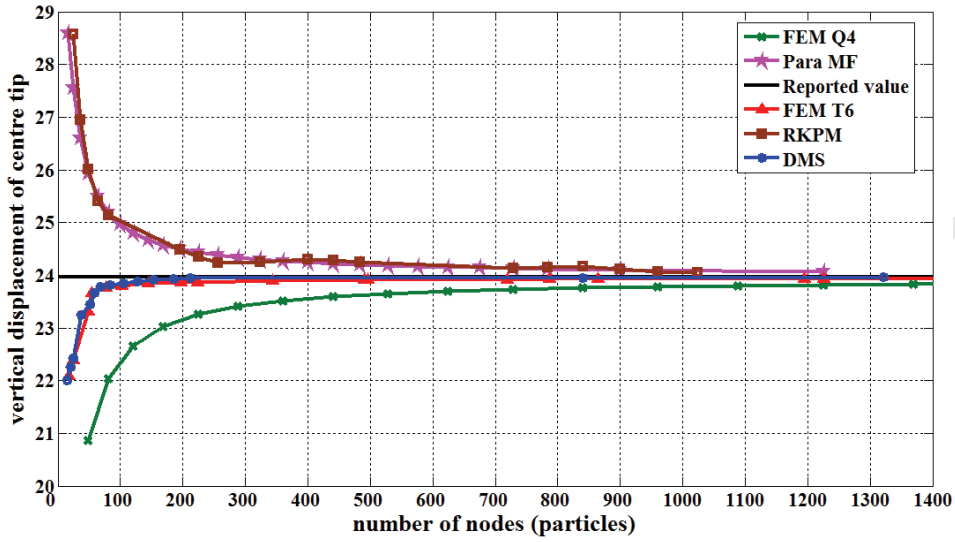
As observed, the present scheme approaches the reported value of the vertical displacement at the centre tip faster than the others. Whereas FEM with Q4 elements, NURBS-based parametric mesh-free method and RKPM take more than 1000 particles to reach adequately close to the reported value, the present scheme takes just about 100 particles to do so. It can be seen from the magnified views (Fig.22(b)) that the present scheme is even better than FEM T6 in terms of convergence to the true solution.

A plane strain case is also discussed where the same material and geometric data are used except that Poisson's ratio is kept as a variable. The numerical stability of the methods as the material approaches the incompressibility limit ($\nu \longrightarrow 0.5$) is under focus and the results are shown in Fig. 23. Comparable number of particles or nodal points (as applicable; about 100 of them) is used with the FEM as well as the present scheme. It is seen that the numerical behaviour of FEM T6 and the current approach is considerably more stable vis-à-vis FEM Q4 (Figs. 23 and 24). Moreover, as seen from the magnified views in Fig.23(b), the present scheme with degree 3 DMS-splines behaves better than that with degree 2 DMS-splines.

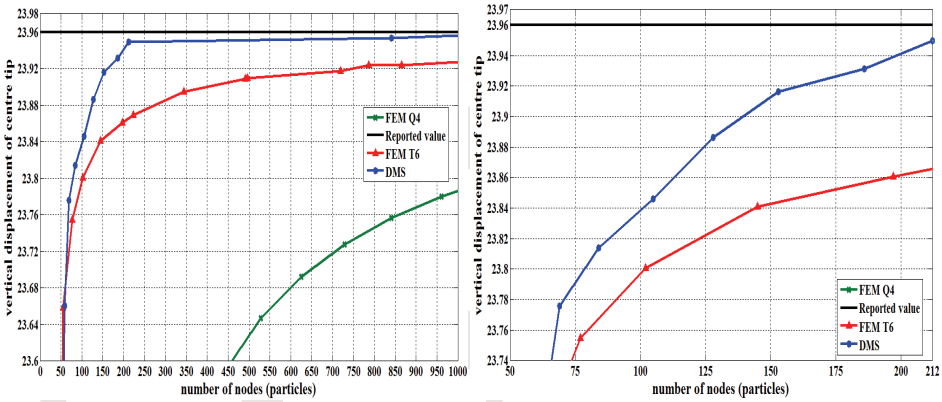
4.3.2 An infinite plate with a circular hole

Consider an infinite plate with a hole of unit radius, subjected to uniform stretching along the (horizontal) x-direction. A square portion of the plate with the hole at its centre and having dimensions 20 times the radius of the hole is considered with the assumption that the effect of stress concentration due to the central hole completely dies out at the domain boundary. It is known (Timoshenko and Goodier 1934) that the stress concentration factor ($\frac{\sigma_x}{\sigma_b}$, where σ_x is the normal stress at a point along a cross-section of the plate and σ_b is the applied stretching stress at the boundary) is 3 at the circumference of the hole (at point A) and reduces quickly towards the boundary along the line AB. The following material properties are considered: $E = 10000,\ \nu = 0.3$. The thickness of the plate is assumed to be unity. Quadratic DMS-splines are used as kernel functions.

Taking advantage of the symmetry of the plate, only one-quarter of it is modelled

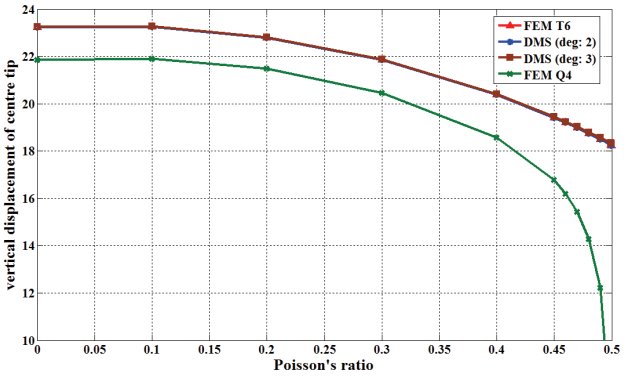


(a)

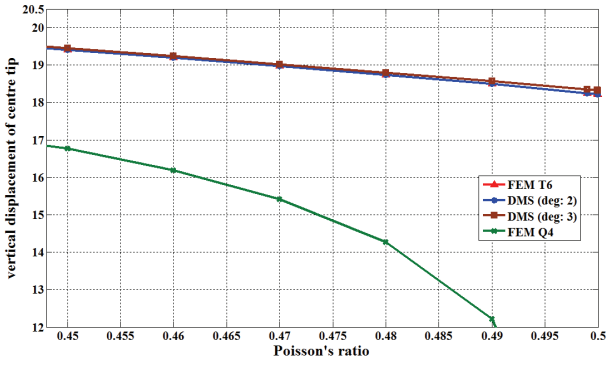


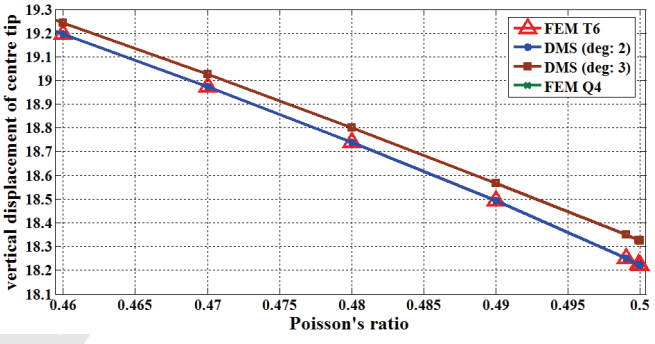
(b)

Figure 22: (a) Comparisons of the performance of different mesh-based and mesh free methods in determining vertical displacement of the centre tip of Cook's membrane (plane stress problem); (b) Cook's membrane (plane stress): magnified views depicting the performance of the present scheme and the FEM in determining vertical displacement of the centre tip; faster convergence of the present scheme vis-Ã -vis FEM with T6 elements is clearer; solution via FEM with Q4 elements is far worse



(a)





(b)

Figure 23: (a) Comparison of the numerical stability of FEM Q4, FEM T6 and DMS-spline global scheme in determining vertical displacement of the centre tip of Cook's membrane as $\nu \to 0.5$ (plane strain problem); (b) Comparison of the numerical stability of FEM Q4, FEM T6 and DMS-spline global scheme in determining vertical displacement of the centre tip of Cook's membrane as $\nu \to 0.5$ (plane strain problem) - magnified views: degree 3 DMS-splines perform better than second degree DMS-splines and FEM T6 elements

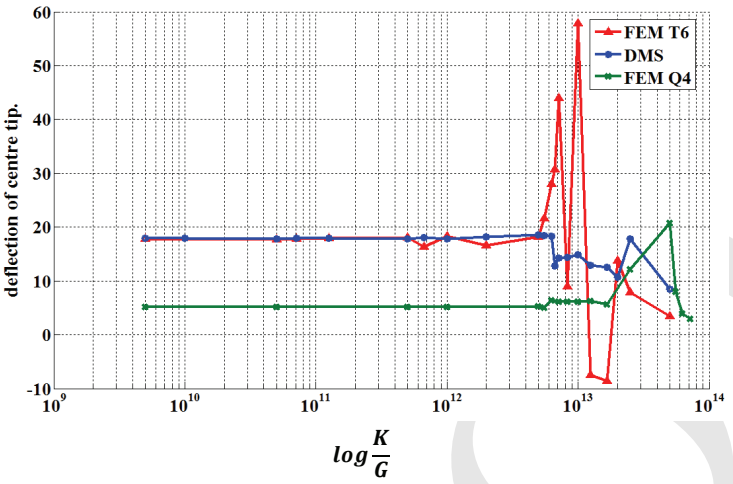


Figure 24: Behaviour of FEM T6, FEM Q4 and DMS-splines based global scheme as $\nu \to 0.5$ plotted on a semi-log graph; K = bulk modulus and G = Shear modulus

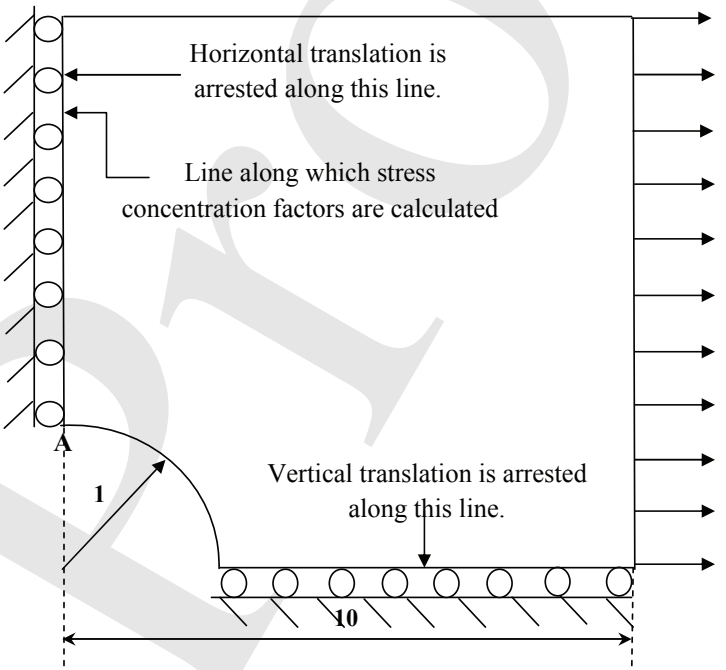


Figure 25: One-quarter of plate with circular hole; Dimensions, boundary conditions and applied stretching force are shown

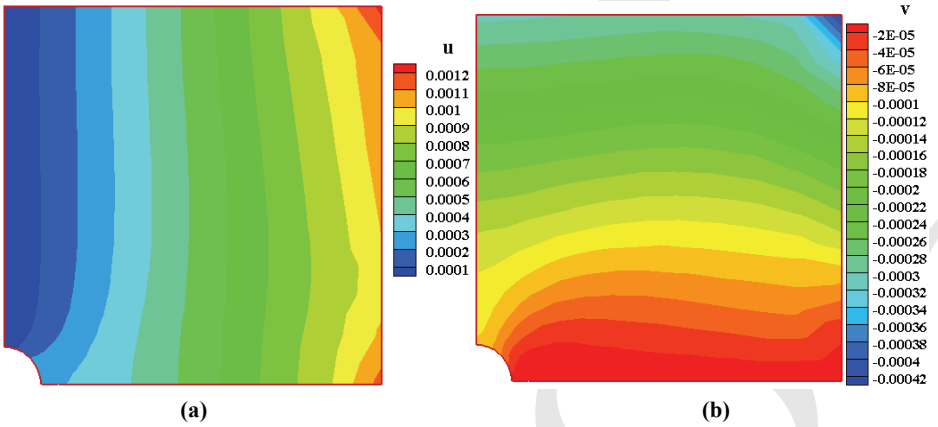


Figure 26: Displacement contours of the plate with circular hole; (a) *x*-displacement (u), (b) *y*-displacement (v)

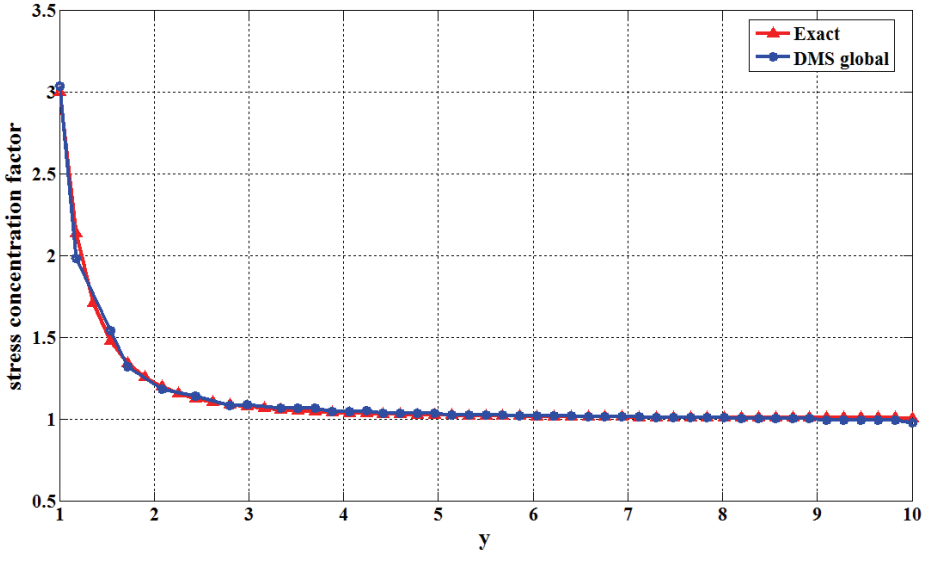


Figure 27: Stress concentration factors plotted along a cross-section (y direction) of the plate

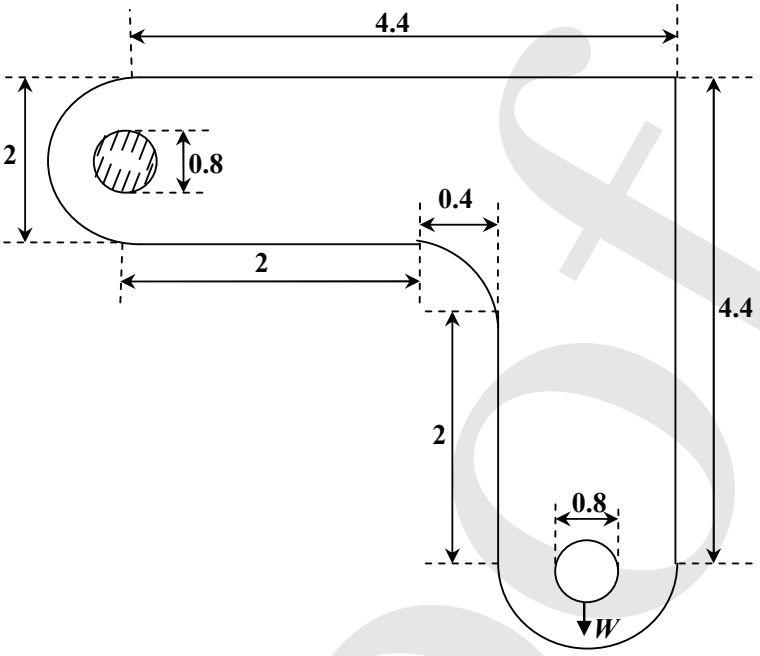


Figure 28: A bracket with dimensions, boundary conditions and concentrated load W

for numerical work (Fig.25). The displacement contours of the plate is given in Fig.26 and plot of stress concentration factors along a cross-section of the plate (line AB in Fig.25) via the present scheme is shown in Fig.27 along with the exact solution. Fig.27 shows that the stress concentration factors determined with the present approximation scheme are in close agreement with the exact values.

4.3.3 A bracket subjected to a concentrated force

A two-dimensional bracket as shown in Fig.28 is considered as the final example. The dimensions, boundary conditions and application of a concentrated force are as shown in the figure. Thickness of the bracket is again taken as unity and the material properties are: $E = 200000, \quad \nu = 0.3$. A concentrated load $W = 100$ is applied as shown in Fig.28. Quadratic DMS-splines are used as kernel functions. This example aims at highlighting the advantage (e.g. the algorithmic simplicity) of the proposed scheme over the NURBS-based parametric bridge method (Shaw *et al.*, 2008b). In the reported results via the latter scheme, the same bracket had to be divided into several sub-domains so as to establish a family of bijective geometric maps between each sub-domain and the parametric domain. The deflection

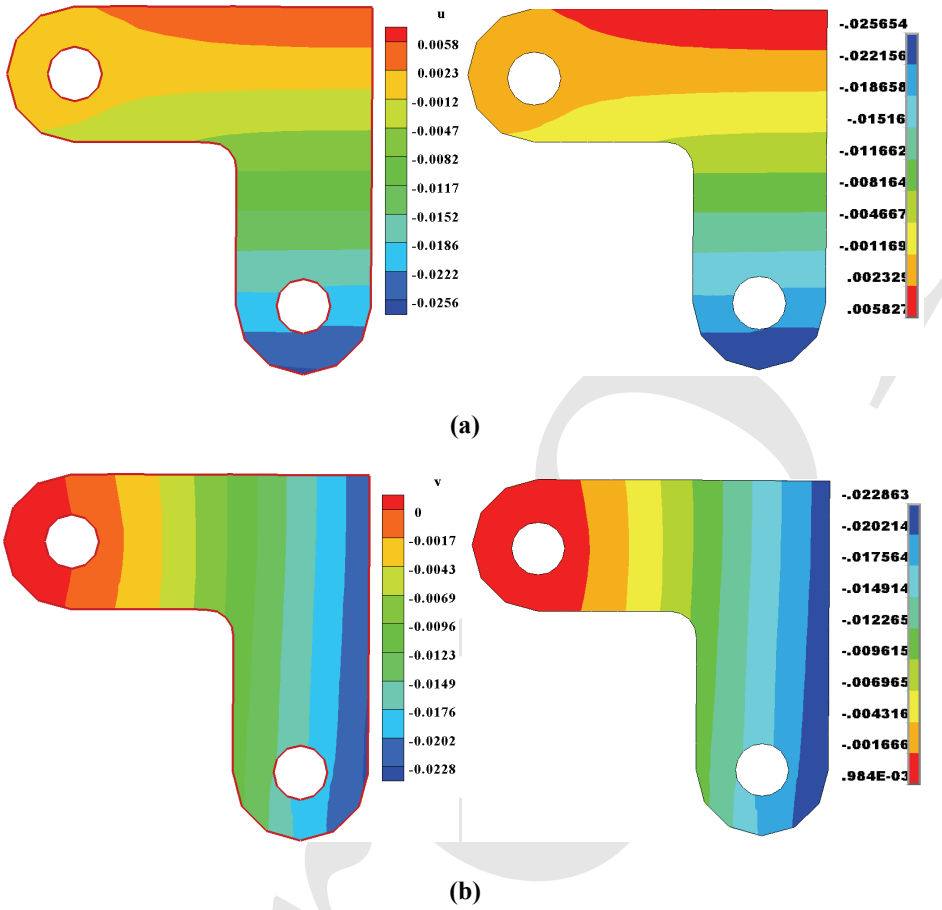


Figure 29: Displacement contours of bracket; results of present method is shown to the left and FEM results (using ANSYS), to the right; (a) x-displacement (u), (b) y-displacement (v)

588 contours of the bracket through the present method and FEM (using ANSYS) are
589 shown in Fig.29. The displacements are in good agreement with FEM solutions.

590 5 Concluding Remarks

591 This work constitutes a remarkable improvement over an earlier effort to bridge
592 the FEM and mesh-free methods based on tensor-product NURBS (Shaw *et al.*
593 2008a). The improvement has been arrived at by replacing tensor-product NURBS
594 with triangular B-splines (or DMS-splines), constructed over a Delaunay triangu-

lation of the domain. This has enabled establishing globally smooth functional and derivative approximations for general 2D domains (including non-simply connected domains) whilst bypassing a geometric map, which can, potentially, precipitate ill-conditioning of the discretized system equations and numerical pollution of solutions. Thus, while remaining strictly within the conventional domain discretization of the FEM, one obtains numerically robust and smooth approximations to the target function and its derivatives across element boundaries – a characteristic feature of mesh-free methods. The numerical robustness of the present functional approximation is however dependent crucially upon an admissible placement of knots around the triangle vertices and using a polynomial reproduction step while generating the shape functions. While the generation of knotclouds near the vertices of triangles is carefully done so as to strictly satisfy the non-collinearity of any three knots fall, the latter ploy involving polynomial reproduction is, in particular, shown to remarkably reduce the sensitive dependence of the approximant to small (and admissible) variations in the knot locations. The polynomials reproduced are of degree $p \leq n$, where n is the degree of DMS-splines. The shape functions so generated are C^{n-1} across triangle boundaries. Unlike the NURBS-based parametric bridge method, which was a precursor to this development, the nodes presently do not double up as knots and this prevents possible misalignment of background cells (i.e., the triangles themselves) with the supports of shape functions (irrespective of the degree of the employed DMS-splines) while applying the scheme to the weak form of the governing equations. Some of the advantages of the proposed scheme are brought out through a number of appropriately chosen numerical examples that include, among others, the Cook's membrane problem and a few boundary value problems defined over non-simply connected domains in 2D.

DMS-splines are being increasingly used for solid modelling in the computer graphics literature (Gang Xu, *et al.*, 2008, Yunhao *et al.*, 2007). Given the importance of solid modelling in the pre-processor of every commercial finite element code and the potential of DMS splines in this respect, the present scheme assumes a sense of timeliness as it offers a seamless interface between solid modelling and basic FE-based computing. This could be particularly helpful if the need arises for repeated re-meshing during a possible *h-p* refinement.

Further investigations on the method are currently under progress. These include a-priori and a-posteriori error estimates, extension to 3D domains and applications to problems involving material and geometric non-linearity. Of specific interest in the last category are the problems of simulating ultra-thin membranes and plates developing shear bands. Some of these results would soon be reported elsewhere.

References

Ainsworth M.; Coggins P. (2000): The stability of mixed *hp* finite element methods for Stokes flow on high aspect ratio elements, *SIAM. Journal of Numerical Analysis*, Vol. 38, pp.1721-1761.

Atluri S. N.; Zhu T. (1998): A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics, *Computational Mechanics*, Vol. 22, pp.117-127.

Atluri S. N.; Zhu T. (2000): The meshless local Petrov-Galerkin (MLPG) approach for solving problems in elasto-statics, *Computational Mechanics*, Vol. 25, pp.169-179.

Atluri S. N.; Han Z. D.; Rajendran A. M. (2004): A new implementation of the meshless finite volume method, through MLPG mixed approach, *CMES: Computer Modelling in Engineering and Sciences*, Vol. 6, no.6, pp.491-513.

Atluri S. N.; Kim H. G.; Cho J. Y. (1999): A critical assessment of the truly Meshless local Petrov–Galerkin (MLPG) methods, *Computational Mechanics*, Vol. 24, pp. 348-372.

Atluri S. N.; Sladek J.; Sladek V.; Zhu T. (2000): The local boundary integral equation (LBIE) and it's meshless implementation for linear elasticity, *Computational Mechanics*, Vol. 25, pp.180-198.

Atluri, S. N.; Liu H. T.; Han Z. D. (2006a): Meshless local Petrov-Galerkin (MLPG) mixed collocation method for elasticity problems, *CMES: Computer Modelling in Engineering and Sciences*, Vol. 14, no.3, pp.141-152.

Atluri, S. N.; Liu H. T.; Han Z. D. (2006b): Meshless local Petrov-Galerkin (MLPG) mixed finite difference method for solid mechanics, *CMES: Computer Modelling in Engineering and Sciences*, Vol. 15, no.1, pp.1-16.

Auricchio F.; Beirao da Veiga L.; Lovadina C.; Reali A. (2005): A stability study of some mixed finite elements for large deformation elasticity problems, *Computer Methods in Applied Mechanics and Engineering*, Vol. 194, pp.1075-1092.

Babuška I.; Melenk J. M. (1997): The partition of unity method, *International Journal of Numerical Methods in Engineering*, Vol. 40, pp.727-758.

Belytschko T.; Lu Y. Y.; Gu L. (1994): Element-free Galerkin methods, *International Journal of Numerical Methods in Engineering*, Vol. 37, pp.229-256.

Cai Y. C.; Zhu H. H. (2004): Direct imposition of essential boundary conditions and treatment of material discontinuities in the EFG. method, *Computational Mechanics*, Vol. 34, pp.330-338.

Dahmen W.; Micchelli C. A.; Seidel H. P. (1992): Blossoming begets B-splines

668 built better by B-patches, *Mathematics of computation*, Vol. 59, no. 199, pp.97-
669 115.

670 **Duarte C. A.; Oden J. T.** (1997): An h-p adaptive method using clouds, *Computer*
671 *Methods in Applied Mechanics and Engineering*, Vol. 139, pp.237-262.

672 **Engel G.; Garikipati K.; Hughes T. J. R.; Larson M. G.; Mazzei L.; Taylor**
673 **R. L.** (2002): Continuous/discontinuous finite element approximations of fourth
674 order elliptic problems in structural and continuum mechanics with applications to
675 thin beams and plates, and strain-gradient elasticity, *Computer Methods in Applied*
676 *Mechanics and Engineering*, Vol. 191, pp.3669-3750.

677 **Fong P.; Seidel H.** (1993): An implementation of triangular B-spline surfaces over
678 arbitrary triangulations, *Computer Aided Geometric Design*, Vol. 10, pp.267-275.

679 **Franssen M. G. J.** (1995): *Evaluation of DMS-splines*, Masters thesis, Department
680 of Mathematics and Computing science, Eindhoven university of technology.

681 **Gamallo P.; Astley R. J.** (2007): A comparison of two Trefftz-type methods: the
682 ultraweak variational formulation and the least square method, for solving short-
683 wave 2-D Helmholtz problems, *International Journal for Numerical Methods in*
684 *Engineering*, Vol. 71, pp.406-432.

685 **Gang Xu; Guo-Zhao Wang; Xiao-Diao Chen** (2008): Free-form deformations
686 with rational DMS-spline volumes, *Journal of Computer Science and Technology*,
687 Vol. 23(5), pp.862-873.

688 **Gingold R. A.; Monaghan J. J.** (1977): Smoothed particle hydrodynamics: theory
689 and application to non-spherical stars, *Monthly Notices of the Royal Astronomical*
690 *Society*, Vol. 181, pp.275-389.

691 **Han W.; Meng X.** (2001): Error analysis of the reproducing kernel particle method,
692 *Computer Methods in Applied Mechanics in Engineering*, Vol. 190, pp.6157-6181.

693 **Hughes T. J. R.** (1995): Multiscale phenomena: Green's functions, the Dirichlet-
694 to-Neumann formulation, subgrid scale models, bubbles and the origins of sta-
695 bilized methods, *Computer methods in applied mechanics and engineering*, 127,
696 pp.387-401.

697 **Hughes T. J. R.; Guglielmo Scovazzi; Leopoldo P. Franca** (2004): *Multiscale*
698 *and stabilized methods*, Encyclopaedia of computational mechanics, John Wiley &
699 sons, ltd.

700 **Kita Eisuke; Kamiya Norio** (1995): Trefftz method: an overview, *Advances in*
701 *Engineering Software*, 24, pp.3-12.

702 **Liu W. K.; Jun S.; Zhang Y. F.** (1995a): Reproducing kernel particle methods,
703 *International Journal for Numerical Methods in Fluids*, Vol. 20, pp.1081-1106.

704 **Liu W. K.; Jun S.; Li S.; Adee J.; Belytschko T.** (1995b): Reproducing ker-

nel particle methods for structural dynamics, *International Journal for Numerical Methods in Engineering*, Vol. 38, pp.1655-1679.

Liu W. K.; Li S.; Belytschko T. (1997): Moving least square reproducing kernel methods (I) methodology and convergence, *Computer Methods in Applied Mechanics and Engineering*, Vol. 143, pp.113-154.

Lucy L. B. (1977): A numerical approach to the testing of the fission hypothesis, *The Astronomical Journal*, Vol. 82, pp.1013-1024.

Micchelli C. A. (1995): Mathematical aspects of geometric modelling, In: *CBMS-NSF regional conference series in applied mathematics*, vol. 65. Philadelphia, PA: SIAM.

Nayroles B.; Touzot G.; Villon P. (1992): Generalizing the finite element method: diffuse approximation and diffuse elements, *Computational Mechanics*, Vol. 10, pp.307-318.

Onate E.; Valls A.; Garcia J. (2006): FIC/FEM formulation with matrix stabilizing terms for incompressible flows at low and high Reynolds numbers, *Comput. Mech.*, 38, pp.440-455.

Pfeifle; Seidel H. P. (1994): Faster evaluation of quadratic bivariate DMS-spline surfaces, *Canadian Human Computer Communications Society Graphics Interface*, pp.182-190.

Shaw A.; Roy D. (2007a): A NURBS-based Error Reproducing Kernel Method with Applications in Solid Mechanics, *Computational Mechanics*, Vol. 40, pp.127-148.

Shaw A.; Roy D. (2008a): NURBS-based Parametric Mesh-free Methods, *Computer methods in applied mechanics and engineering*, Vol. 197, pp.1541-1567.

Shaw A.; Banerjee, B.; Roy D. (2008b): A NURBS-based parametric method bridging mesh free and finite element formulations, *CMES: Computer Modelling in Engineering and Sciences*, Vol. 26(1), pp.31-35.

Shaw A.; Bendapudi S.; Roy D. (2008c): A Kriging-based Error Reproducing and Interpolating Kernel Method for Improved Mesh-free Approximations, *International Journal for Numerical Methods in Engineering*, Vol. 73, pp.1434-1467.

Shuyao Long; Atluri S. N. (2002): A. meshless local Petrov-Galerkin method for solving the bending problem of a thin plate, *CMES: Computer Modelling in Engineering and Sciences*, Vol. 3(1), pp.53-63.

Simo, J. C.; Fox D. D.; Rifai M. S. (1989): On a stress resultant geometrically exact shell model: Part II-The linear theory; computational aspects, *Computer methods in applied mechanics and engineering*, Vol. 73, pp.53-92.

Sonia Fernandez-Mendez; Antonio Huerta (2004): Imposing essential boundary

conditions in mesh-free methods, *Computer Methods in Applied Mechanics and Engineering*, Vol. 193, pp1257-1275.

Soric J.; Jarak T. (2010): Mixed meshless formulation for analysis of shell-like structures, *Computer Methods in Applied Mechanics and Engineering*, Vol. 199, pp. 1153-1164.

Timoshenko S. P.; Goodier, J. N. (1934): *Theory of elasticity*, McGrawhill international editions.

Yunhao Tan; Jing Hua; Ming Dong (2007): 3D reconstruction from 2D images with hierarchical continuous simplices, *The Visual Computer*, Vol. 23, pp.905-914.

Zhu T.; Atluri, S. N. (1998): A modified collocation method and a penalty formulation for enforcing the essential boundary conditions in the element free Galerkin method, *Computational Mechanics*, Vol. 21, pp.211-222.

Zhu T.; Zhang J.; Atluri S. N. (1998): A meshless local boundary integral equation (LBIE) method for solving nonlinear problems, *Computational Mechanics*, Vol. 22, pp.174-186.

Zienkiwicz, O. C.; Taylor, R. L.; Zhu J. Z. (1967): *The finite element method: Its basis and fundamentals*, McGraw-Hill Publishers.

759