CMES, vol.66, no.3, pp.249-284, 2010

The Reproducing Kernel DMS-FEM: 3D Shape Functions and Applications to Linear Solid Mechanics

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

Abstract: We propose a family of 3D versions of a smooth finite element method (Sunilkumar and Roy 2010), wherein the globally smooth shape functions are derivable through the condition of polynomial reproduction with the tetrahedral B-splines (DMS-splines) or tensor-product forms of triangular B-splines and 1D NURBS bases acting as the kernel functions. While the domain decomposition is accomplished through tetrahedral or triangular prism elements, an additional requirement here is an appropriate generation of knotclouds around the element vertices or corners. The possibility of sensitive dependence of numerical solutions to the placements of knotclouds is largely arrested by enforcing the condition of polynomial reproduction whilst deriving the shape functions. Nevertheless, given the higher complexity in forming the knotclouds for tetrahedral elements especially when higher demand is placed on the order of continuity of the shape functions across inter-element boundaries, we presently emphasize an exploration of the triangular prism based formulation in the context of several benchmark problems of interest in linear solid mechanics. In the absence of a more rigorous study on the convergence analyses, the numerical exercise, reported herein, helps establish the method as one of remarkable accuracy and robust performance against numerical ill-conditioning (such as locking of different kinds) vis-á-vis the conventional FEM.

Keywords: DMS-splines; tetrahedral and triangular prism elements; knotcloud generation; polynomial reproduction; 3D elasticity equations.

1 Introduction

The finite element method (FEM), the most popular amongst numerical schemes for solving boundary value problems arising in solid mechanics, employs an element-based domain decomposition and piecewise smooth polynomial shape functions.

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

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

The discontinuities in the derivatives of the shape functions generally lead to jump terms in the weak forms of the governing equations and also discontinuous distributions of the approximated stress/strain fields. In the conventional FEM, such jump terms in the weak form, involving only the first order derivatives of the field variables, are ignored. On the other hand, for bending-dominated problems, wherein the weak forms often contain second order derivatives of the displacement fields, one generally looks for shape functions with uniform C^1 continuity in the domain interior. Arriving at shape functions with such higher order continuity has however proved to be formidable in 2D and beyond, and one often attempts to partially circumvent this problem via a family of so-called non-conformal shape functions, which are often complicated and problem-specific. While for most well-posed elliptic problems, the above inconsistencies do not lead to a loss of coercivity of the discretized equations, this may not be true for systems that may not be well posed (for instance, saddle point systems) or are prone to locking. In the context of solid mechanics, different mechanisms of locking, such as volumetric, shear or thickness locking, are possible. Thus, as the material approaches the incompressibility limit (*i.e.* as Poisson's ratio $\rightarrow 0.5$, or, equivalently the bulk modulus tends to infinity) numerical results via finite elements (especially, lower order ones) could behave spuriously (volumetric locking). Similarly, the presence of transverse shear stresses, especially in thin structures undergoing deformation due to bending, may stiffen the discretized system equations and thus cause erroneous displacements (Kui *et al.* 1985, Jose *et al.* 2002). Use of higher order elements (p-refinement), assumed strain method, etc., are extensively used as remedies for locking.

While an h-refinement reduces the error due to discontinuity of derivatives at the inter-element boundaries in an irreducible formulation, smooth stress/strain fields may be achieved by a mixed approach (Zienkiewicz, *et al.*, 2000). While mixed FE methods have been extensively researched, they involve a significant augmentation of the degrees-of-freedom (DOF-s), thereby increasing the dimensions of the discretized system matrices. Moreover, each mixed method with both displacements and their derivatives as DOF-s has to grapple with certain stability issues, i.e., such methods are required to satisfy the LBB (Ladyzenskaya-Babuska-Brezzi) condition (Brezzi and Fortin, 1991). Stabilization techniques are extensively reported in the literature (Hughes 1995; Hughes *et al.* 2004; Onate 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 as well as the dimension of the polynomial space used to construct the shape functions (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.

In the context of solid mechanics, considerable research has also gone in weakly enforcing the global smoothness of derivatives through the space-discontinuous Galerkin (SDG) method (Engel *et al.* 2002). Here an important issue is that of correctly obtaining the stabilizing terms (along with the stabilization parameters), which essentially penalize the derivative jumps across inter-element boundaries in the weak form. This is a crucial, yet largely unresolved, issue with the SDG method especially for nonlinear problems.

Mesh-less methods offer yet another alternative approach for obtaining such smooth solutions by doing away with the element-based functional discretization and representing the domain, instead, through a set of nodes (or particles). Prominent amongst these methods are the smooth particle hydrodynamics (SPH), (Gingold and Monaghan 1977), the diffuse element method (DEM) (Nayrole *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) (Babuska and Melenk 1997, Melenk and Babuska 1996), the *h-p* Clouds (Duarte and Oden 1997), the mesh-less local Petrov–Galerkin method (MLPG) (Atluri *et al.* 1999), error reproducing kernel method (ERKM) (Shaw and Roy 2007) and several others. However, these methods often do not yield stiffness matrices with low bandwidth, which is typical with the FEM. In addition, weak solutions with mesh-less methods demand a background integration mesh (Liu *et al.* 1995a, 1995b, Shaw *et al.* 2008) and, in this sense, many of these methods are not strictly mesh-less after all. Misalignment (non-conformability) of the background mesh with the local support as well as irregular distribution of particles leads to poor convergence and erroneous results (Dolbow and Belytschko, 1999). This difficulty is generally overcome via a substantial increase in the order of quadrature in many mesh-less methods, like in the EFG (Dolbow and Belytschko, 1999). The moving least square Petrov-Galerkin (MLPG) method employs a local weak form and thus bypasses the non-conformability issue (Atluri and Zhu 1998, Atluri *et al.* 1999, Atluri *et al.* 2000, Aturi and Zhu 2000, Atluri *et al.* 2004, Atluri *et al.* 2006). A further limitation of most mesh-less 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 leads to excessive smoothness of the approximation functions as well as increased bandwidth of the matrices. Unfortunately, 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, mesh-free shape functions are generally non-interpolating and hence may not strictly qualify as valid test functions as they do not vanish uniformly over the essential part of the domain boundary.

The NURBS-based parametric method (Shaw and Roy 2008) addresses some of the above difficulties. More specifically, this method automatically ensures the inclusion of only the minimum number of particles within the support of the NURBS kernel. This in turn is possible through a bijection (called a geometric map) between a rectangular (or cuboidal) parametric domain and the physical domain. Moreover, use of the NURBS cells as the integration cells are themselves often aids in achieving conformality in the numerical integration of the weak form of the governing equations. However, for most cases of practical interest (e.g. for non-simply connected domains), the geometric map may not exist. By way of overcoming this limitation, Shaw *et al.* (2008) have proposed a NURBS-based parametric method, which is claimed to bridge the mesh-free and FE formulations. In this, the physical domain is decomposed into a finite set of sub-domains (comparable to elements in the FEM) such that a geometric map can be established for each element sub-domain in this set (comparable to parametric FE methods). In any case, the geometric map in the class of parametric methods described above could precipitate ill-conditioning of the discretized equations. Moreover, the integration of the weak form is not necessarily conformal irrespective of the degrees of the reproduced polynomials.

Continuing with our focus on methods that, in some sense, bridge the FEM and mesh-free methods, it is obvious that discretization of complex domain is best handled via triangulation (or tetrahedralization) and that a scheme based on globally smooth shape functions defined using such discretization would work without a geometric map. Towards this, a reproducing kernel DMS-spline based approach for constructing globally smooth shape functions over a Delaunay triangulation of a bounded 2D domain has recently been proposed by Sunilkumar and Roy (2010). DMS is an acronym for Dahmen, Micchelli and Seidel, the authors who introduced the spline (Dahmen *et al.* 1992). A key aspect of this construction is the use of non-degenerate knotclouds that help achieve C^{n-1} global continuity of the shape functions obtained through n^{th} degree DMS-spline kernels. Numerical experiments with this scheme have been indicative of its superior numerical accuracy as well as stability (e.g. against locking) vis-á-vis most mesh-free methods and the conventional FEM.

The practically important problem of extending the concept of the reproducing kernel DMS-FEM to 3D domains is quite non-trivial and forms the subject of this work. As a direct extension of bivariate DMS-splines (in $\mathbb{R}^2$, defined of a Delaunay triangulation), reproducing kernel shape functions via trivariate n^{th} degree DMS-splines (constructed over a Delaunay tetrahedralization of the given 3D domain) are first derived. Knots are placed near each of the vertices of the tetrahedrons, which also act as background cells for numerical integration of the weak form equations. The following issues, nevertheless, need to be specially addressed in 3D domains: (1) discretization of a regular/irregular solid with regular tetrahedrons, whilst maintaining acceptably good aspect ratio, is not straightforward; (2) placement of knots around the tetrahedral vertices (whilst satisfying the condition that no four knots of a knot vector should fall on a plane) is non-trivial and may sensitively affect the numerical solution; and (3) tetrahedrons would be a poor choice to model very thin geometries (e.g. thin plates, shells or membranes and space-time blocks with very small time step). In view of these difficulties, an alternative element formulation, based on triangular prisms, is developed within the 3D version of DMS-FEM. In this case, the kernel function is a tensor product of bivariate DMS-splines and one dimensional NURBS. The polynomial reproducing shape functions so derived have a global C^{n-1} continuity, where n is the degree of bivariate DMS-splines as well as the one-dimensional NURBS. For numerical integration of the weak form, the triangular prisms also serve as nearly conformable background cells. The remarkably superior performance of the new 3D DMS-FEM over the FEM is brought forth through a host of numerical illustrations.

The paper is organised as follows. In section 2, multivariate DMS-splines and one dimensional NURBS are briefly described. Section 3 focuses on the construction of shape functions in 3D domains descretized through tetrahedrons and on the numerical integration issues. A few numerical illustrations of the DMS-FEM, employing tetrahedral elements, are provided in Section 4. The development of DMS-FEM shape functions over triangular prisms and the related implementation issues are considered in Section 5. This is followed, in Section 6, by the numerical implementation of the triangular prism elements in the context of some linear elastostatic problems. Finally, conclusions are drawn in Section 7.

2 Kernel functions

In the present papers, trivariate DMS-splines are the kernel functions (mollifiers) for the approximation scheme employing tetrahedrons for the discretization of 3D domains. On the other hand, tensor product of bivariate DMS-splines and one dimensional NURBS are proposed to serve as kernels for shape functions constructed over triangular prisms, which in turn discretize the 3D domain. For completeness, therefore, brief overviews of DMS-splines, multivariate DMS-splines and 1D NURBS are provided in what follows (see Franssen 1995 and Piegl and Tiller 1995 for a more detailed exposition).

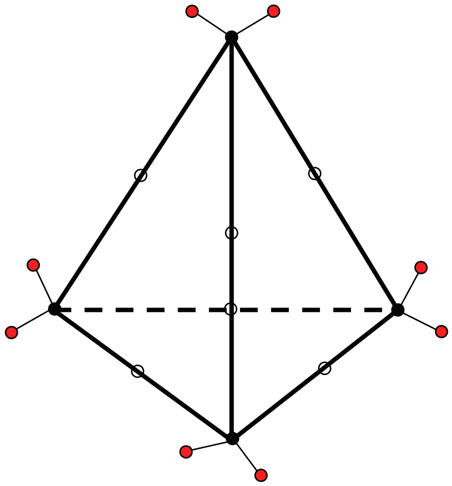


Figure 1: A tetrahedron with knotclouds generated for quadratic trivariate DMS-splines; the black circles represent vertices of the tetrahedron, red circles represent additional knots added and black and white circles stand for the control points which serve as particles in the present method

The s-variate DMS-spline at point $\mathbf{x} \in \Omega \subset \mathbb{R}^s$ is given by

$$F(\mathbf{x}) = \int_{I \in \mathscr{I}} \int_{|\beta|=n} \left| det\left(\check{V}^I_\beta\right)\right| M\left(\mathbf{x}|\mathrm{V}^I_\beta\right) c^I_\beta \tag{1}$$

$\check{V}^I_\beta$ is a triangle that consists of the last knots of the heads of the knotclouds contained in V^I_β, i.e. $\check{\mathrm{V}}^I_\beta = \left\{ v^I_{0_{\beta_0}}, v^I_{1_{\beta_1}}, v^I_{2_{\beta_2}} \right\}$ and knotclouds of all the vertices are given by (see Fig.1):

$$\mathrm{V}^I_\beta = \left\{ v^I_{0_0}, \ldots, v^I_{0_{\beta_0}}, v^I_{1_0}, \ldots, v^I_{1_{\beta_1}}, v^I_{2_0}, \ldots, v^I_{2_{\beta_2}} \right\}.$$

The knotcloud for a vertex $v_{i0} = v_i$ is the set $\{v_{i0}, \ldots, v_{in}\}$, which consists of the vertex itself and n additional knots (points) added to the vertex v_i for an n^{th} degree DMS-spline. $M\left(\mathbf{x}|\mathrm{V}^I_\beta\right)$ is an s-variate simplex spline of degree n and c^I_β are the control points and $\mathscr{I}$ represents a proper polyhedralization of the domain such that two polyhedra in $\mathscr{I}$ are either disjoint or share exactly a primitive of dimension lower than s spanned by their common vertices. The set of control points c^I_β is defined for every polyhedron $I \in \mathscr{I}$, where $\beta = (\beta_0, \dots, \beta_s)$ is an $s+1$ index with $|\beta| = \beta_0 + \cdots + \beta_s = n$ (Franssen, 1995).

The β^{th} s-variate DMS-spline basis functions are defined from Eq. (1) as:

$$\Phi^I_\beta = \left| det\left(\breve{V}^I_\beta\right)\right| M\left(\mathbf{x}|\mathrm{V}^I_\beta\right) \tag{2}$$

An s-variate simplex spline of degree n is defined recursively as follows:

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

where $V \subset \mathbb{R}^s$ is the knotset which again is a subset of V^I_β, $W = \{w_0, \dots, w_s\} \subset V$ is any affinely independent subset of V and $\lambda_j(\mathbf{x}|W)$ is the j^{th} barycentric co-ordinate which is the ratio of the volumes $W_x\,(w_j{:=}\mathbf{x})$ and W, given by:

$$\lambda_j(\mathbf{x}|W) = \frac{det\begin{pmatrix} 1 & & 1 & 1 & 1 & & 1 \\ w_0 & \dots & w_{j-1} & \mathbf{x} & w_{j+1} & \dots & w_s \end{pmatrix}}{det\begin{pmatrix} 1 & & 1 \\ w_0 & \dots & w_s \end{pmatrix}} \tag{4}$$

The i^{th} degree-n NURBS (Non-Uniform Rational B-Spline) basis functions in one dimension, useful for developing the triangular prism-based DMS-FEM, is defined as:

$$R^n_i(\zeta) = \frac{N_{i,n}(\zeta)}{\sum_{k=1}^{m} N_{k,n}(\zeta)} \tag{5}$$

Here $N_{i,n}(\zeta)$ is the i^{th} B-spline basis function of degree n (defined below) with $\zeta \in \mathbb{R}$ representing a non-decreasing set of real numbers called the knot vector in a parametric interval [0, 1]. The weights associated with the control points are assumed as unity. m represents the total number of B-spline basis functions. The i^{th} B-spline basis function of degree n is defined as:

$$N_{i,n}(\zeta) = \frac{\zeta - \zeta_i}{\zeta_{i+1} - \zeta_i} N_{i,n-1}(\zeta) + \frac{\zeta_{i+n+1} - \zeta}{\zeta_{i+n+1} - \zeta_{i+1}} N_{i+1,n-1}(\zeta), \quad i = 1, 2, \dots, m+n+1 \tag{6a}$$

with

$$N_{i,0}(\zeta) = \begin{cases} 1 & \text{if } \zeta_i \le \zeta \le \zeta_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{6b}$$

A detailed account of NURBS is available in Piegl and Tiller (1995).

The kernel functions employed in our tetrahedron-based DMS-FEM are the trivariate DMS-splines. As will be observed later, a typical shape functions in this scheme will be supported nearly over a tetrahedron or a set of adjacent tetrahedrons depending on the location of the point $\mathbf{x} = \{x_1, x_2, x_3\} \in \mathbb{R}^3$ where the shape function is evaluated. As with the NURBS-based parametric method, the selection of such supports is also automated in this method.

For the DMS-FEM employing triangular prism elements, a tensor product of bivariate DMS-splines and 1D NURBS serves as the kernel function whilst deriving the shape functions. The construct of this tensor product is as follows. Let m be the number of n^{th} degree NURBS basis functions. The number of n^{th} degree bivariate DMS-spline basis functions is $(n+1)(n+2)/2$. Now, the kernel function is defined as:

$$\widetilde{\Phi_{ij}}(\mathbf{x}) = \Phi_i(x_1, x_2) R_j^n(x_3) \tag{7}$$

where, $\Phi_j(x_1, x_2)$ is the j^{th} bivariate DMS-spline basis function given by Eq.(2), $R_k^n(x_3)$ is the 1D NURBS basis function given by Eq. (5), $i = 1, 2, \ldots, (n+1)(n+2)/2$ and $j = 1, 2, \ldots, m$.

Note that linear span of either of the compactly supported kernel functions described above defines a subspace Đ of distributions within $C^\infty(\mathbb{R}^3)$ and that these kernels constitute a locally finite C^∞ partition of unity. Also note that, in the construction of the tensor-product kernel, the 1D NURBS may be replaced by the 1D B-splines.

3 Tetrahedral elements: Shape functions and their derivatives

Given a regular tetrahedralization of a given, bounded 3D domain Ω, the derivations of shape functions and their derivatives proceed as follows.

3.1 Construction of Shape Functions

As a more straightforward analogue of the 2D version of DMS-FEM (Sunilkumar and Roy 2010), the 3D solid domain $\overline{\Omega} = \Omega \cup \partial\Omega$ can be discretized with tetrahedrons with trivariate DMS-splines acting as the kernel functions for deriving the polynomial reproducing shape functions. Let $\mathbf{x} \in \overline{\Omega} \subset \mathbb{R}^3$ be a point under consideration; it can be a quadrature point in the numerical integration of the weak form or, for that matter, any other point of interest where the value of a functional of the field variables and/or their derivatives is sought. Let a real-valued field variable

$u(\mathbf{x})$ be expressed through the following well-known integral identity:

$$u(\mathbf{x}) = \int_{\Omega} \delta(\mathbf{x}-\mathbf{y})u(\mathbf{y})\,d\Omega \tag{8}$$

where $\mathbf{y} \in \Omega$ and $\delta(.)$ is the Dirac delta function (Dirac distribution, which may be looked upon as a linear functional on the subspace Đ). Approximating $\delta(.)$ with a mollifier Φ, we may write

$$u(\mathbf{x}) \cong u^a(\mathbf{x}) = \int_{\Omega} \Phi(\mathbf{x}-\mathbf{y})u(\mathbf{y})\,d\Omega \tag{9}$$

A descritized version of Eq. (9), using numerical quadrature and such that $u^a(\mathbf{x})$ exactly reproduces polynomials up to a certain degree, is given by (Liu, *et al.* 1995b):

$$u^a(\mathbf{x}) = \int_{i=1}^{N_{nd}} \mathbf{H}^{\mathrm{T}}(\mathbf{x}-\mathbf{x_i})\,\mathbf{b}(\mathbf{x})\,\Phi(\mathbf{x}-\mathbf{x_i})\,u(\mathbf{x_i})\,\Delta V_i d\Omega \tag{10}$$

when $\overline{\Omega}$ is represented (discretized) by N_{nd} particles (nodes). Here ΔV_i is the nodal volume associated with node i, $\mathbf{H}(\mathbf{x})$ is a set of monomials defined as $\{x_1{}^{\alpha_1}x_2{}^{\alpha_2}x_3{}^{\alpha_3}\}_{|\alpha_1+\alpha_2+\alpha_3|\le p}$, p is the reproduced degree polynomial of the highest degree $(p \le n)$ and $\mathbf{b}(\mathbf{x})$ is a set of unknown coefficients with the vector function $\mathbf{H}^{\mathrm{T}}(\mathbf{x}-\mathbf{x_i})\,\mathbf{b}(\mathbf{x})$ generally referred to as the correction function. Eq. (10) can be recast as:

$$u^a(\mathbf{x}) = \int_{i=1}^{N_{nd}} \Psi_i(\mathbf{x})u(\mathbf{x_i}) \tag{11}$$

where $\Psi_i(\mathbf{x})$ is the so-called i^{th} shape function. Note that the nodal volume, ΔV_i, can be absorbed into $\mathbf{b}(\mathbf{x})$ to yield a modified coefficient vector $\overline{\mathbf{b}}(\mathbf{x})$. The elements of the latter are obtained based on the following polynomial reproduction conditions:

$$\sum_{i=1}^{N_{nd}} \Psi_i(\mathbf{x})\,1 = 1 \tag{12}$$

$$\sum_{i=1}^{N_{nd}} \Psi_i(\mathbf{x})\left(x_{1}{}_i^{\alpha_1} x_{2}{}_i^{\alpha_2} x_{3}{}_i^{\alpha_3}\right) = x_1{}^{\alpha_1}x_2{}^{\alpha_2}x_3{}^{\alpha_3} \quad |\alpha_1+\alpha_2+\alpha_3| \le p \tag{13}$$

$$\sum_{i=1}^{N_{nd}} \Psi_i(\mathbf{x})\,(x_1-x_{1i})^{\alpha_1}(x_2-x_{2i})^{\alpha_2}(x_3-x_{3i})^{\alpha_3} = \delta_{|\alpha_1||\alpha_2||\alpha_3|,0} \quad |\alpha_1+\alpha_2+\alpha_3| \le p \tag{14}$$

i.e.

$$\sum_{i=1}^{N_{nd}} \mathbf{H}^T(\mathbf{x}-\mathbf{x_i})\,\overline{\mathbf{b}}(\mathbf{x})\,\Phi(\mathbf{x}-\mathbf{x_i})\,\mathbf{H}(\mathbf{x}-\mathbf{x_i}) = \mathbf{H}(0) \tag{15}$$

The above equation may be written as: $\mathbf{M}(\mathbf{x})\,\overline{\mathbf{b}}(\mathbf{x}) = \mathbf{H}(0)$. Here

$$\mathbf{M}(\mathbf{x}) = \sum_{i=1}^{N_{nd}} \mathbf{H}^T(\mathbf{x}-\mathbf{x_i})\,\Phi(\mathbf{x}-\mathbf{x_i})\,\mathbf{H}(\mathbf{x}-\mathbf{x_i}) \tag{16}$$

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

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

provided that the moment matrix is invertible. We will be considering the issue of invertibility shortly. The global shape functions in 3D are given by:

$$\Psi_i(\mathbf{x}) = \mathbf{H}^T(\mathbf{x}-\mathbf{x_i})\,\mathbf{M}^{-1}(\mathbf{x})\,\mathbf{H}(0)\,\Phi(\mathbf{x}-\mathbf{x_i}) \tag{17}$$

If the i^{th} particle $\mathbf{x_i}$ falls inside a tetrahedron, the support of the associated shape function $\Psi_i(\mathbf{x})$ is the polyhedron obtained through the elements of the knotcloud and this support is nearly coincident with the associated tetrahedron as we typically employ knot-lengths of a smaller order than that of the tetrahedral sides. Using the same arguments, if $\mathbf{x_i}$ is located on a face of a tetrahedron, then both the tetrahedra which share that face constitutes the support of $\Psi_i(\mathbf{x})$. On the other hand, if $\mathbf{x_i}$ lies along one of the edges or vertices, all the tetrahedra which share the edge or vertex (with their knotcloud neighbourhood) will form the support of $\Psi_i(\mathbf{x})$ (see Fig. 2).

3.2 Derivatives of Shape Functions

Since we will employ the DMS-FEM for obtaining weak solutions of (elliptic) PDE-s of interest in solid mechanics, all the derivatives considered below should be construed in the weak sense (Kesavan 2008). Derivatives of shape functions are constructed by a stable and numerically accurate scheme for computing derivatives of globally smooth shape functions proposed by Shaw and Roy (2008). 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:

$$\sum_{i=1}^{N_{nd}} \Psi_i^{(\gamma)}(\mathbf{x})\,\mathbf{H}(\mathbf{x}-\mathbf{x_i}) = (-1)^{|\gamma|}\mathbf{H}^{(\gamma)}(0), \quad \forall |\gamma| \leq p \tag{18}$$

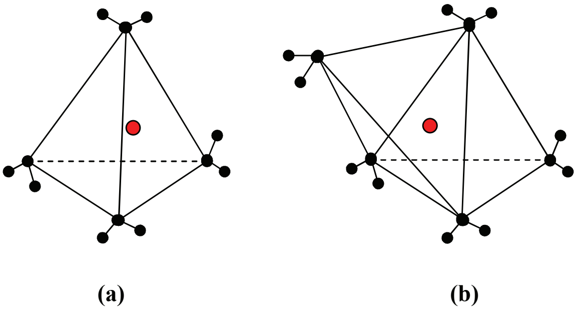


Figure 2: Support of a shape function; (a) a particle (the red circle) is inside a tetrahedron; hence the tetrahedron is (approximately) the support and (b) the particle is on the common face between two tetrahedrons; hence the two tetrahedrons (approximately) together is the support

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

$$\Psi_i^{(\gamma)}(\mathbf{x}) = \mathbf{H}^T(\mathbf{x}-\mathbf{x_i})\,\mathbf{b}^{\gamma}(\mathbf{x})\,\Phi(\mathbf{x}-\mathbf{x_i}) \tag{19}$$

Here $\mathbf{b}^{\gamma}(\mathbf{x})$ is the vector of unknown coefficients needed for derivative reproduction. The final form of $\Psi_i^{(\gamma)}(\mathbf{x})$, after $\mathbf{b}^{\gamma}(\mathbf{x})$ are determined from the canonical polynomial reproduction conditions, can be written as:

$$\Psi_i^{(\gamma)}(\mathbf{x}) = (-1)^{|\gamma|}\mathbf{H}^{(\gamma)}(0)\,\mathbf{M}^{-1}(\mathbf{x})\,\mathbf{H}^T(\mathbf{x}-\mathbf{x_i})\,\Phi(\mathbf{x}-\mathbf{x_i}) \tag{20}$$

3.3 Invertibility of the Moment Matrix

In many mesh-less methods, the supports of the shape functions (as determined through that of the weight or kernel function) need 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 (or, contained within the supports of) at least $\dim(P_p) = \frac{(p+N_{dim})!}{p!N_{dim}!}$ shape functions, where $\dim(P_p)$ is the cardinality of the basis of polynomial space of degree $\leq p$, and N_{dim} is the dimension of the domain Ω. So, if $\Omega \subset \mathbb{R}^3$ and given $\mathbf{x} \in \Omega$, the minimum number of shape functions required for ensuring invertibility of $\mathbf{M}(\mathbf{x})$ is $\frac{(p+3)!}{p!3!} = \frac{(p+3)(p+2)(p+1)}{6}$. The number of particles introduced in a tetrahedron (the support of the associated shape functions) is $\frac{(n+3)(n+2)(n+1)}{6}$ which will be equal to the number of trivariate DMS-spline basis functions. The DMS-splines are C^{n-1} continuous, everywhere, if the knots are generally admissible (i.e. no four knots are coplanar) (Dahmen *et al.* 1992). Therefore, if $p \leq n$, the invertibility condition given by Han and Meng (2001) will be satisfied. Now, if the point **x** belongs to one of the faces, edges or vertices of a tetrahedron $I \in \mathscr{I} \subset \mathbb{R}^3$, it may not always belong to the half open convex hull of all the tetrahedrons formed by subsets of the knotsets corresponding to the control points. This may lead to a reduction in the continuity of DMS-splines by one. It has also been verified numerically that this reduction in continuity is at most by one. In such a case, n has to be kept greater than p in order 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 (tetrahedron) to make the moment matrix invertible by choosing $n = p$ or $n = p + 1$ depending on whether the particle **x** is inside or on the boundary of the local support, respectively.

3.4 Numerical Integration

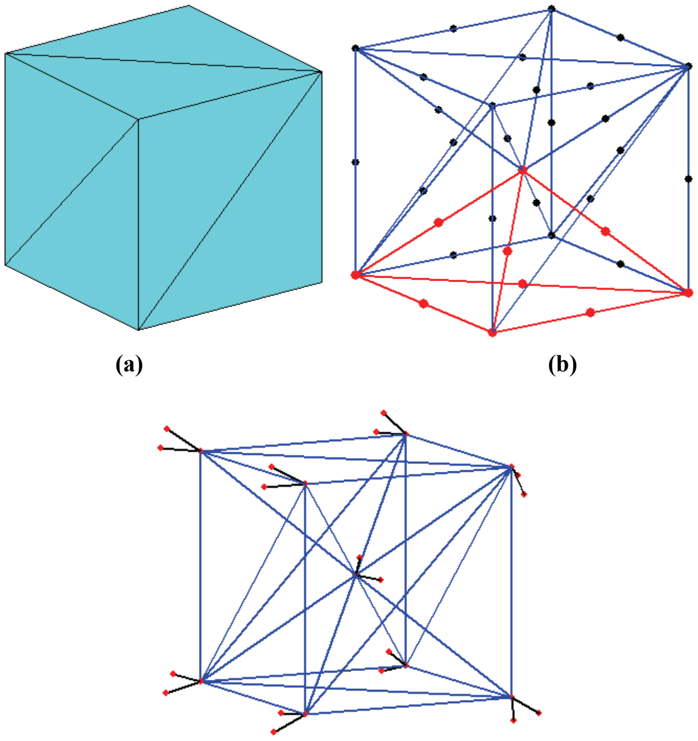
In constructing the weak form (as in a Galerkin projection) of a system of differential equations, a background mesh (similar to the mesh used in the FEM) is generally required in mesh-less methods for evaluating the integrals that arise. Integration is generally performed over each background cell by 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. This may lead to inaccurate integration leading to loss of accuracy and convergence of mesh-free methods (Dolbow and Belytschko 1999). In the context of many mesh-less methods, this difficulty is generally overcome via a substantial increase in the order of quadrature. In the conventional FEM, on the other hand, the elements themselves act as the background integration cells and so the integration scheme remains necessarily conformal. Similarly, in the present scheme, the tetrahedrons, which discretized the physical domain and enable construction of the DMS-spline basis functions, double up as the integration cells. Recall from Section 3.1 that the local support of the shape functions is a tetrahedron or tetrahedrons plus their associated knotcloud neighbourhood. However the volume of this neighbourhood is of a significantly smaller measure, thanks to the knot lengths (distance to an extra knot associated to a vertex from the vertex) being much smaller vis-á-vis the tetrahedron edges. Hence using the tetrahedrons themselves as the background integration cells should hardly introduce any meaningful numerical errors, a fact that is also verified through extensive numerical experiments. In the NURBS-based parametric bridge method (Shaw *et al.* 2008), since the mesh in the parametric space is used as integration cells, the number of such cells could 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), the quadrature must be made extremely high for achieving acceptable accuracy in the numerical integration. The present scheme is largely free of such misalignment issues because of the uniformity in the placement of knots and the extra knots not doubling up as particles (as in the NURBS based parametric methods).

4 DMS-FEM with Tetrahedral Elements: Numerical Results

To illustrate the proposed scheme, we first consider the errors in the approximations of polynomials, trigonometric functions and their derivatives over a cubical domain. We follow this up by applying the method for solving Laplace's and Poisson's equations.

4.1 Approximations of Certain Target Functions and their Derivatives

A unit cube is chosen as the domain for this example (Fig.3) and it is discretized with 12 tetrahedrons (Fig.3a) and 35 particles (Fig.3b). The approximations to the target functions and their derivatives are evaluated at the particles inside the cube with the new scheme using quadratic trivariate DMS-splines as kernel functions. The degree of polynomial reproduced is also kept as 2 $(i.e., \ p = 2)$. First, the crucial role played by the polynomial reproduction in the construction of shape functions is investigated. Thus, a polynomial function, $f(x,y,z) = (x+y+z)^2$, is approximately evaluated at certain locations in the unit cubical domain (the locations are shown in Table 1) with trivariate DMS-splines basis functions (without imposing the polynomial reproduction condition) and the proposed polynomial reproducing shape functions with trivariate DMS-splines as kernels. The results are reported in Fig.4 and Table 1. It can be observed that, in the former case (i.e., whilst approximating directly via trivariate DMS-splines), (absolute) errors in the approximation are higher (and, at some locations, enormously higher). This may be contrasted with the superior quality of approximation through the polynomial reproducing shape functions. This result is analogous to a similar contrast in the quality of the approximation, reported earlier in the context of the 2D version of the DMS-FEM (Sunilkumar and Roy 2010). Nevertheless, the contrast seems to get accentuated in the 3D DMS-FEM vis-á-vis its 2D counterpart. Pending a more rigorous justification, it appears that the explicit reproduction of polynomials help suppress the sensitive dependence of the numerical solution on the knotcloud placement and an additional source of error owing to the non-interpolating characteristics of DMS-splines whilst imposing Dirichlet boundary conditions. Indeed, with-



(a)

(b)

(c)

Figure 3: (a). A cube of dimensions 1×1×1 modelled with tetrahedrons, (b) particles generated in the cube for quadratic bivariate DMS-spline are shown as black and red dots; one of the tetrahedrons with its particles is shown in red, (c) knot-clouds generated for quadratic bivariate DMS-splines

out using the reproducing conditions, the orders of errors in the approximations of the derivatives of the field variables are even higher.

In the same vein as above, Table 2 goes on to establish a similar observation in terms of relative L^2 error norms for polynomial and non-polynomial target functions as well as their derivatives. Specifically, the relative L^2 error norm is defined as (with f^a representing the functional approximant for the targeted function f over a bounded domain Ω, which is again a unit cube):

$$f - f^{a}{}^{rel}_{L^2} = \frac{\left(\int_\Omega (f - f^a)^2 d\Omega\right)^{1/2}}{\left(\int_\Omega f^2 d\Omega\right)^{1/2}} \tag{21}$$

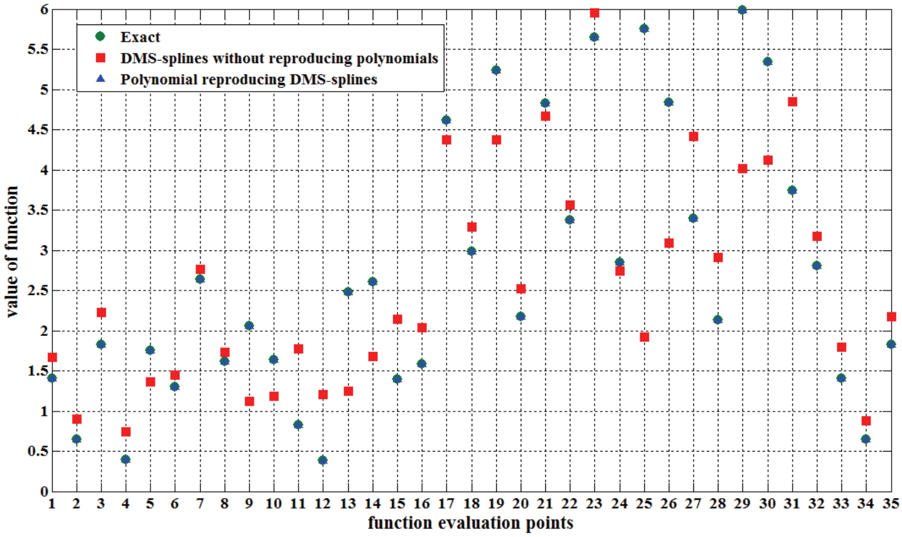


Figure 4: Plot of values of the polynomial function $f(x,y,z) = (x+y+z)^2$ against the evaluation points; the function values evaluated with polynomial reproducing DMS-splines (blue triangles) very closely match with the exact values (green circles) whereas those evaluated with DMS-spline basis functions without reproducing polynomials (red squares) are erroneous

It is observed that the relative L^2 error norms of the polynomial functions and their derivatives are very small. This happens as shape functions reproducing second degree polynomials are used. However, in the approximation of non-polynomial functions (as is the case with the trigonometric function) and their derivatives (third column of Table 2), the use of quadratic shape functions develop larger relative L^2 error norms, especially for the higher order derivatives.

4.2 Laplace's and Poisson's equations

The second order Laplace's and Poisson's equations in 3D, often useful as workhorse elliptic PDE-s for validating new schemes, are given respectively as:

$$\frac{\partial^2 f(x,y,z)}{\partial x^2} + \frac{\partial^2 f(x,y,z)}{\partial y^2} + \frac{\partial^2 f(x,y,z)}{\partial z^2} = 0 \tag{22}$$

and

$$\frac{\partial^2 f(x,y,z)}{\partial x^2} + \frac{\partial^2 f(x,y,z)}{\partial y^2} + \frac{\partial^2 f(x,y,z)}{\partial z^2} = g(x,y,z) \tag{23}$$

Table 1: Functional approximation at selected points in the unit cubical domain; exact and approximated values (along with the errors) via direct and polynomial reproducing DMS-splines are given

Co-ordinates of points			Exact value of function	DMS-splines basis functions without reproducing polynomials		Polynomial reproducing shape functions with DMS-splines kernels	
x	y	z		Approximate value	Error	Approximate value	Error
0	0	0	1.4005625065	1.6705969236	-0.2700344	1.4005625065	1.9984e-15
1	0	0	0.6522255014	0.9015320134	-0.2493065	0.6522255014	6.6613e-16
0	1	0	1.8309856496	2.2232289069	-0.3922433	1.8309856496	3.5527e-15
1	1	0	0.3964575260	0.7434046950	-0.3469472	0.3964575260	-6.6613e-16
0	0	1	1.7519680208	1.3581182820	0.3938497	1.7519680208	3.5527e-15
1	0	1	1.3022699721	1.4483980732	-0.1461281	1.3022699721	-6.6613e-16
0	1	1	2.6339704048	2.7626820999	-0.1287117	2.6339704048	-3.5527e-15
1	1	1	1.6173381964	1.7298742992	-0.1125361	1.6173381964	-6.6613e-16
0.5	0.5	0.5	2.0552281087	1.1225476071	0.9326805	2.0552281087	1.3323e-15
0.5	0	0	1.6411580563	1.1850092182	0.4561488	1.6411580563	8.6597e-15
0.5	0.5	0	0.8275920085	1.7784095273	-0.9508175	0.8275920085	-3.3085e-14
0.25	0.25	0.25	0.3795897800	1.2033674129	-0.823778	0.3795897800	-4.9960e-16
1	0.5	0	2.4767593524	1.2506888560	1.226070	2.4767593524	1.3323e-15
0.75	0.25	0.25	2.6070705946	1.6784813347	0.9285893	2.6070705946	-3.9080e-14
0.75	0.75	0.25	1.3912976220	2.1414664888	-0.7501689	1.3912976220	1.3323e-14
0	0.5	0	1.5830822552	2.0420488898	-0.4589666	1.5830822552	0
0.5	1	0	4.6192791814	4.3760834399	0.2431957	4.6192791814	-8.8818e-16
0.25	0.75	0.25	2.9797979461	3.2869710948	-0.3071731	2.9797979461	-1.7764e-15
0.5	0	1	5.2317435605	4.3762946625	0.8554489	5.2317435605	-8.8817e-16
0.5	0.5	1	2.1696291168	2.5202262434	-0.3505971	2.1696291168	8.8817e-16
0.25	0.25	0.75	4.8222962148	4.6665762207	0.1557200	4.8222962148	-1.7764e-15
1	0.5	1	3.3760302151	3.5620810337	-0.1860508	3.3760302151	4.8850e-15
0.75	0.25	0.75	5.6512596310	5.9561623065	-0.3049027	5.6512596310	-2.6645e-15
0.75	0.75	0.75	2.8459641342	2.7371531182	0.1088110	2.8459641342	3.9968e-15
0	0.5	1	5.7571390046	1.9196031580	3.837536	5.7571390046	-2.5757e-14
0.5	1	1	4.8385975826	3.0865681067	1.752029	4.8385975826	-5.3291e-15
0.25	0.75	0.75	3.3999013975	4.4157437788	-1.015842	3.3999013975	-1.0125e-13
0.5	0	0.5	2.1299240766	2.9068583827	-0.7769343	2.1299240766	1.1102e-14
1	0	0.5	5.9835312813	4.0164067530	1.967124	5.9835312813	1.7764e-15
0	0	0.5	5.3401192075	4.1198849460	1.220234	5.3401192075	2.8422e-14
0.5	1	0.5	3.7396577541	4.8435054220	-1.103848	3.7396577541	-6.4393e-14
1	1	0.5	2.8004630290	3.1755224801	-0.375059	2.8004630290	1.3323e-15
0	1	0.5	1.4005625065	1.7969957365	-0.396433	1.4005625065	-3.0198e-14
0	0.5	0.5	0.6522255014	0.8833565204	-0.2311310	0.6522255014	-2.3315e-15
1	0.5	0.5	1.8309856496	2.1711227868	-0.3401371	1.8309856496	-1.9318e-14

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

$f(x,y)$	$(x+y+z)$	$(x+y+z)^2$	$sin(x+y+z)$
n	2	2	2
N_{nd}	35	35	35
N_e	12	12	12
$f - f^{a\,rel}_{L^2}$	1.847×10^{-13}	3.802×10^{-13}	9.948×10^{-2}
$\frac{\partial f}{\partial x} - \left(\frac{\partial f}{\partial x}\right)^{a\,rel}_{L^2}$	5.241×10^{-13}	1.157×10^{-12}	6.084×10^{-1}
$\frac{\partial f}{\partial y} - \left(\frac{\partial f}{\partial y}\right)^{a\,rel}_{L^2}$	2.227×10^{-13}	1.041×10^{-12}	6.139×10^{-1}
$\frac{\partial f}{\partial z} - \left(\frac{\partial f}{\partial z}\right)^{a\,rel}_{L^2}$	4.518×10^{-13}	9.435×10^{-13}	6.193×10^{-1}
$\frac{\partial^2 f}{\partial x \partial y} - \left(\frac{\partial^2 f}{\partial x \partial y}\right)^{a\,rel}_{L^2}$	1.517×10^{-12}	3.588×10^{-12}	2.654×10^{0}
$\frac{\partial^2 f}{\partial x \partial z} - \left(\frac{\partial^2 f}{\partial x \partial z}\right)^{a\,rel}_{L^2}$	1.472×10^{-12}	3.402×10^{-12}	2.715×10^{0}
$\frac{\partial^2 f}{\partial y \partial z} - \left(\frac{\partial^2 f}{\partial y \partial z}\right)^{a\,rel}_{L^2}$	1.356×10^{-12}	2.882×10^{-12}	2.774×10^{0}
$\frac{\partial^2 f}{\partial x^2} - \left(\frac{\partial^2 f}{\partial x^2}\right)^{a\,rel}_{L^2}$	2.320×10^{-12}	4.692×10^{-12}	8.478×10^{0}
$\frac{\partial^2 f}{\partial y^2} - \left(\frac{\partial^2 f}{\partial y^2}\right)^{a\,rel}_{L^2}$	3.438×10^{-12}	7.645×10^{-12}	8.468×10^{0}
$\frac{\partial^2 f}{\partial z^2} - \left(\frac{\partial^2 f}{\partial z^2}\right)^{a\,rel}_{L^2}$	1.468×10^{-12}	3.117×10^{-12}	8.458×10^{0}

where $f(x,y,z)$ and $g(x,y,z)$ are L^2 functions in the domain $\Omega \subset \mathbb{R}^3$. Weak forms of the homogeneous Laplace's equation and inhomogeneous Poisson's equation under Diritchlet boundary conditions are obtained and solved via the present scheme over a unit cubical domain. Note that, by the Lax-Milgram theorem (Kesavan 2008), the $H^1(\Omega)$-ellipticity of bilinear forms of the above equations and hence the uniqueness of solutions of the weak forms are assured. Whilst deriving the DMS-FEM shape functions, only polynomials up to the second degree are presently reproduced. An advantage of using Laplace's and Poisson's equations is that a family of exact solutions, subject to suitable Dirichlet boundary conditions, may readily be constructed. For instance, an exact solution of Eq. (21) is provided by

$$f(x,y,z) = -x^3 - y^3 - z^3 + 3x^2y + 3y^2z + 3z^2x \tag{24}$$

with the Dirichlet boundary function obtained by taking the trace of the above function on the surface of the domain. Note that, by the Lax-Milgram theorem (Kesavan

2008), the exact solution above is also unique. Similarly, for Eq. (22), a unique exact solution is

$$f(x,y,z) = (x+y+z)^2 \tag{25}$$

so that $g(x,y,z) = 6$ and the Dirichlet boundary conditions imposed accordingly. The relative L^2 error norms are obtained as 9.876×10^{-2} and 1.201×10^{-14} respectively for Laplace's and Poisson's equations. The higher relative L^2 error norm in the former case is expected owing to the exact solution being a cubic polynomial with the shape functions enabled to reproduce only up to quadratic polynomials. Plots of the approximated function values against the exact values are given as Fig. 5 for the Laplace and Poisson cases.

Such illustrations of the DMS-FEM, developed so far, could continue beyond these elliptic problems with applications in 3D solid mechanics. However, in doing so, a few issues related to the geometry modelling and knotcloud generation are of some concern with the current approximation scheme. First, unlike the discretization via triangulation of an irregular area in 2D, discretizing an irregular-shaped solid body into regular tetrahedrons is not as easy. Indeed, when it comes to modelling solid bodies with uniform thickness, regular brick or prism elements are preferred to tetrahedrons in the FE meshing with a view to better convergence and numerical accuracy. For modelling very thin geometries with 3D elements (a requirement that also arises in space-time FE formulations over 2D spatial domains with very thin time blocks), tetrahedrons are not quite useful, since, in doing so, they would be highly ill-shaped. Triangular prism elements would be a better choice here too.

Even more importantly, whilst constructing trivariate DMS-splines in the context of the DMS-FEM, knot placement near each vertex of a tetrahedron must be consistent with the restriction that no four knots in a knot vector lie in a plane. A small sphere having the radius of the knot length (typically taken to be a small fraction of the characteristic length of the tetrahedral edges meeting at the vertex) is pictured near each vertex and knots are placed carefully on its circumference so that the restriction is satisfied. But, the procedure becomes much more involved when the degree of trivariate DMS-splines is greater than 2. So, a p-refinement of the present strategy is not straightforward.

In view of the above limitations, we propose an alternative scheme wherein the solid geometry can be modelled with triangular prisms so that a tensor product of bivariate DMS-splines and 1D NURBS could be used as the kernel function. A clear advantage in such a case is that the knot placement is limited to the 2D domain. The method is explained in detail in the next section.

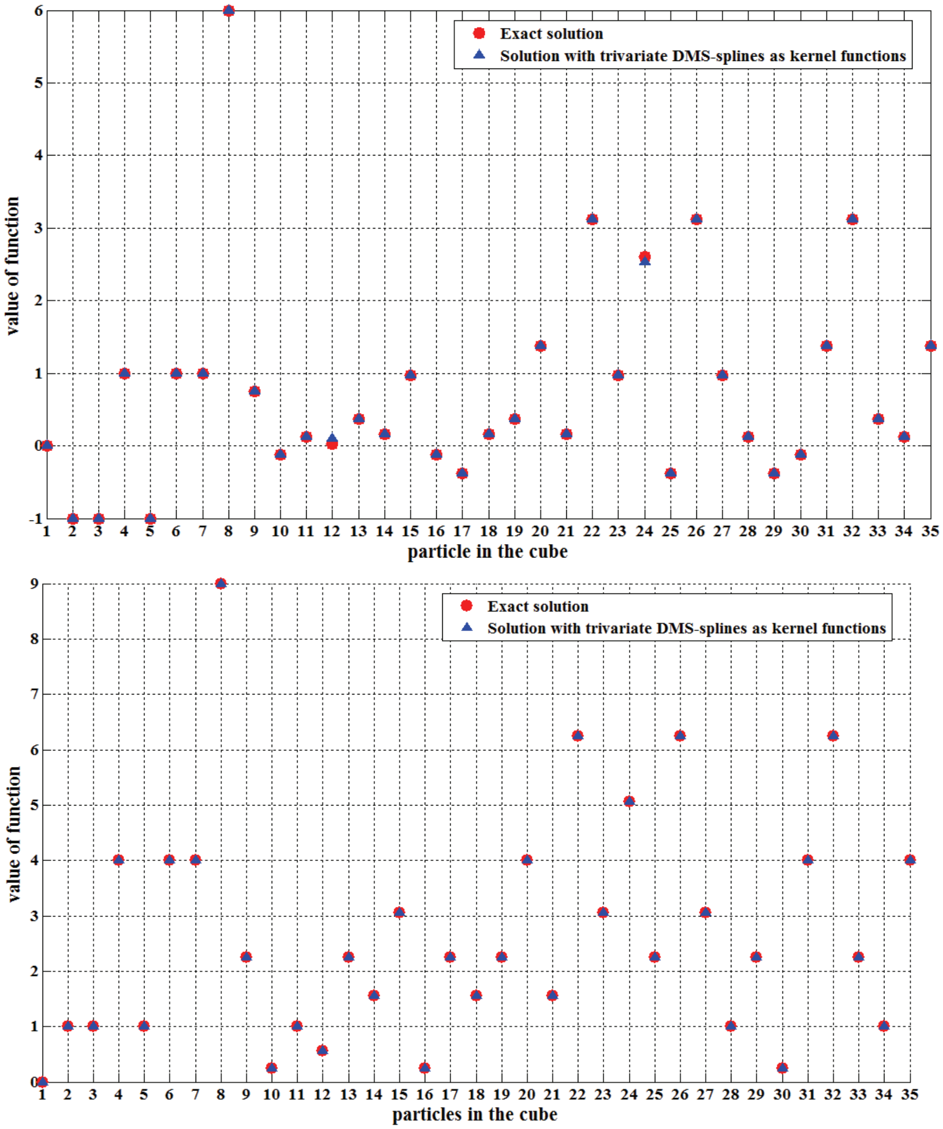


Figure 5: Plots showing computed values of functions against exact values at all particles for solution of Laplace's equation (top) and Poisson's equation (bottom)

5 DMS-FEM with Triangular Prisms

Complex irregular geometries in 2D are best modelled by triangulation. Delaunay triangulation is one among the best schemes of its kind in the sense that it generates well-shaped triangles whilst making sure that the vertices of no other triangles fall in the circumcircle of one triangle. An extruded triangle becomes a triangular prism (3D element) (Fig.6) and, in addition to tetrahedrons, such triangular prisms may also be employed to model (a large class of) solid domains. Specifically in the context of the DMS-FEM, the triangular prism element should enjoy certain advantages over tetrahedral elements, as envisioned in the last section. Towards constructing the associated shape functions, it is thus intended to use a tensor product of bivariate DMS-splines defined over triangles and 1D NURBS, defined over a parametric line segment (in the interval [0,1]) is adopted as the kernel function (Eq.7). Subject to small knot lengths, the local support of such a kernel function will roughly be the triangular prism itself when the particle of interest is inside the prism or a set of triangular prisms when the particle lies on a common face, edge or vertex. A more detailed construction of the shape functions and their derivatives is explained in the next subsection.

5.1 Construction of Shape Functions and their Derivatives

Let $\Omega \subset \mathbb{R}^3$ be a bounded domain, which is regular enough to be adequately discretized through triangular prisms. As before, let $\mathbf{x}$ be a point of interest in this domain at which a real-valued function u is represented as:

$$u(\mathbf{x}) = \int_{\Omega} \delta(\mathbf{x}-\mathbf{y}) u(\mathbf{y})\, d\Omega \tag{8}$$

where $\mathbf{y} \in \Omega$. Making use of the kernel function to approximately represent the Dirac delta function $\delta(.)$ and after applying polynomial reproduction conditions (Liu *et al.* 1995b), the following descritized form of the approximant can be arrived at:

$$u^a(\mathbf{x}) = \sum_{i=1}^{N_{nd}} \mathbf{H}^{\mathrm{T}}(\mathbf{x}-\mathbf{x_i})\,\overline{\mathbf{b}}(\mathbf{x})\,\hat{\Phi}(\mathbf{x}-\mathbf{x_i})\,u(\mathbf{x_i})\,d\Omega \tag{26}$$

where $\hat{\Phi}(\mathbf{x}-\mathbf{x_i})$ is the kernel function, $\mathbf{x_i}$ is the ith particle in the local support, N_{nd} is the total number of nodes or particles and $\overline{\mathbf{b}}(\mathbf{x})$ represents a vector of unknown coefficients in which nodal volume (which arises due to discretized representation of the integral via quadrature) is absorbed (see, for instance, Eq.10). Now, Eq. 25 can be recast as:

$$u^a(\mathbf{x}) = \sum_{i=1}^{N_{nd}} \hat{\Psi}_i(\mathbf{x}) u(\mathbf{x_i}) \tag{27}$$

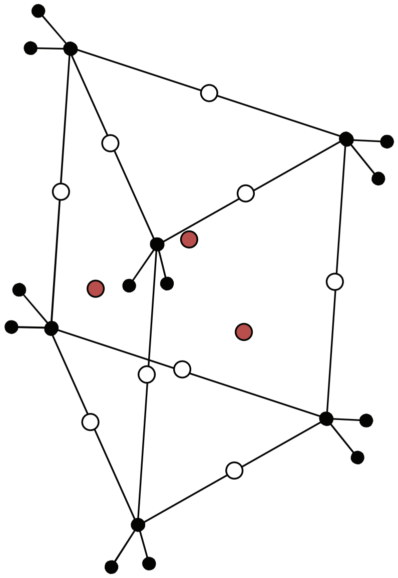


Figure 6: A triangular prism domain of the kernel function resulting from tensor product of bivariate DMS-splines (over triangular domain) and 1D NURBS (over a line); degree of DMS-splines and NURBS in this illustration is 2; black circles represent knotclouds including triangle vertices, white circles represent the particles corresponding to control points on edges other than the vertices and red circles represent particles (control points) on the lateral faces

The coefficients $\overline{\mathbf{b}}(\mathbf{x})$ are determined from polynomial reproducing conditions, as given in Eqs. 12 to 14, with $\Psi_i(\mathbf{x})$ replaced by $\hat{\Psi}_i(\mathbf{x})$, the ith shape function corresponding to discretization through triangular prism elements. The moment matrix may presently be written as:

$$\mathbf{M}(\mathbf{x}) = \sum_{i=1}^{N_{nd}} \mathbf{H}^T(\mathbf{x}-\mathbf{x_i})\,\hat{\Phi}(\mathbf{x}-\mathbf{x_i})\,\mathbf{H}(\mathbf{x}-\mathbf{x_i}) \tag{28}$$

Thus the shape function $\hat{\Psi}_i(\mathbf{x})$ is given by:

$$\hat{\Psi}_i(\mathbf{x}) = \mathbf{H}^T(\mathbf{x}-\mathbf{x_i})\,\mathbf{M}^{-1}(\mathbf{x})\,\mathbf{H}(0)\,\hat{\Phi}(\mathbf{x}-\mathbf{x_i}) \tag{29}$$

The derivatives of shape functions are constructed following the same procedure given in Section 3.2. The final form of the γ^{th} derivative, $\hat{\Psi}_i^{(\gamma)}(\mathbf{x})$, of the shape function is derived as:

$$\hat{\Psi}_i^{(\gamma)}(\mathbf{x}) = (-1)^{|\gamma|}\mathbf{H}^{(\gamma)}(0)\,\mathbf{M}^{-1}(\mathbf{x})\,\mathbf{H}^T(\mathbf{x}-\mathbf{x_i})\,\hat{\Phi}(\mathbf{x}-\mathbf{x_i}) \tag{30}$$

5.2 Invertibility of the Moment Matrix

The construction of global shape function is contingent upon the invertibility of moment matrix $\mathbf{M}(\mathbf{x})$ (as seen from Eq. 28). Towards this, the number of basis functions, whose supports must cover the point $\mathbf{x}$ is $(p+1)(p+2)(p+3)/6$ (Han and Meng 2001; also see Section 3.3). Let us examine how this condition is satisfied in the presently conceived scheme. Consider the case when a particle $\mathbf{x}$ is inside a triangular prism (often a Gauss point for numerical integration of the weak form), which is obviously adopted as the local support of the associated shape function. The number of bivariate DMS-splines basis functions over a triangle is $(n+1)(n+2)/2$ and the minimum number of one-dimensional NURBS basis functions over a line segment with the use of open knot vector is $(n+1)$ (Piegl and Tiller 1995). Hence the number of shape functions that are non-zero at the particle location (or, equivalently, the number of particles introduced in a triangular prism) is

$$(n+1)(n+2)/2 \times (n+1).$$

Now, if $p = n$, we have

$$(n+1)(n+2)/2 \times (n+1) > (p+1)(p+2)(p+3)/6 \qquad \forall\, p = n > 0$$

Hence the requirement for the invertibility of the moment matrix is met in this case. When $\mathbf{x}$ lies on one of the boundaries (edges, faces or vertices) of the triangular prisms, the degree of the kernel function is kept as $(p+1)$ to account for a possible loss of continuity of bivariate DMS-splines (Sunilkumar and Roy 2010). In this case, the local support of the shape functions includes all the triangular prisms which share the boundary to which $\mathbf{x}$ belongs. In this case too, the invertibility requirement of the moment matrix is automatically satisfied because of this increase in the size of the support.

5.3 Numerical Integration

Integrals in the weak form of the governing equations are numerically evaluated with the triangular prisms acting as integration cells. The numerical integration is carried out with Gauss quadrature rule as follows. Let the integral to be evaluated be denoted by

$$\mathfrak{I} = \int_a^b f(\mathbf{x})\, d\Omega \tag{31}$$

Let the number of Gauss points on the base triangle of the triangular prism cell be n_{G2} and that along the thickness dimension be n_{G1}. Gauss points and weights over the triangle and the thickness dimension are respectively represented by the sets $\{x_{G2}\}$ and $\{w_2\}$ (each containing n_{G2} elements) and $\{x_{G1}\}$ and $\{w_1\}$ (each containing n_{G1} elements). Now the integral in Eq. 30 can be written using the quadrature rule as:

$$\mathfrak{I}^a = \sum_{i=1}^{n_{G1}} \sum_{j=1}^{n_{G2}} w_{1i} w_{2j} f\left(x_{G1i}, x_{G2j}\right) J \tag{32}$$

where, J is the jacobian of transformation of the standard element to the domain integration cell. The standard element is a triangular prism having right angled triangular base with co-ordinates of vertices as (0,0), (1,0), (0,1) and unit thickness.

As we have already discussed, a non-alignment of supports of shape functions with the integration cells may lead to non-conformal, and hence inaccurate, integration in several mesh-free methods. This difficulty is generally overcome via a substantial increase in the order of quadrature. In the present scheme, just as in the case with tetrahedrons, the supports of the shape functions are automatically with the triangular prisms so that the issue of non-conformality does not arise.

6 DMS-FEM via Triangular Prisms: Numerical Results

In addition to the examples considered in Section 4 for validating the tetrahedral element formulations, we also consider several linear elastostatic problems of interest in solid mechanics.

6.1 Approximations of Target Functions and their Derivatives

Certain polynomial, trigonometric and exponential functions in 3D and their derivatives are first approximated using the Triangular Prism DMS-FEM (TP-DMS-FEM) at selected points within a unit cube and their relative L^2 error norms (defined in Eq. 20) are reported in Table 2. 54 triangular prisms in three layers are used to discretize the domain (Fig.7). More specifically, the top face of the cube is triangulated with the Delaunay procedure and all of those triangles are extruded by 1/3rd of the height of the cube so as to form the first layer of triangular prisms. Three such layers cover the whole cube. The number of particles in the domain depends upon the degree of kernel function used, as already explained in Section 5.2. The points at which the functions and their derivatives are approximately evaluated are chosen as the centroids of triangular prisms.

It is evident from Table 3 that the polynomial functions and their derivatives are evaluated nearly exactly by the shape functions when the degree of kernel functions, highest degree of polynomial reproduced and degree of the approximately evaluated polynomial functions are the same. But, in the case of trigonometric and

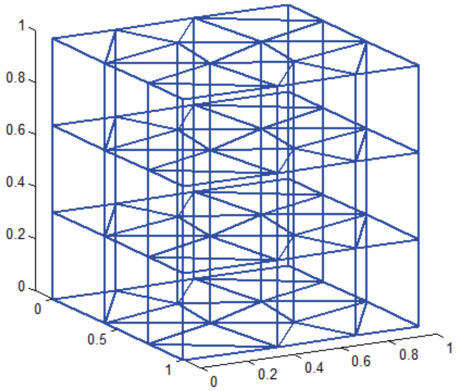


Figure 7: A unit cube discretized with three layers of triangular prisms, each layer containing 18 triangular prisms

exponential functions (last two columns of Table 3), which are at best approximated within finite dimensional polynomial spaces, the relative L^2 error norms are higher even though they are substantially lower than in the tetrahedral DMS-FEM. This is especially possible with TP-DMS-FEM as substantially higher-degree kernel functions can be used in the construction of shape functions to attain better accuracy. Recall that an attempt to use such higher-degree kernels in the tetrahedral DMS-FEM has to grapple with the problematic issue of non-degenerate placement of higher number of knots in the 3D. It has been observed that when kernel functions of degree 2, 3 and 4 are employed (in the TP-DMS-FEM), the relative L^2 error norms of the trigonometric and exponential functions vary from 10^{-4} to 10^{-7} and those of their derivatives (up to the second order), from 10^{-2} to 10^{-4}.

6.2 Laplace's and Poisson's Equations

The second order Laplace's and Poisson's equations and their exact solutions are given in Eqs. 21, 22, 23 and 24. A unit cube is again adopted as the problem domain and Dirichlet boundary conditions are imposed as in Section 4. Stiffness matrices are generated, as usual, with numerical integration of Galerkin weak forms (Eqs. 21 and 22). The results, corresponding to linear kernel and shape functions, are given in Fig. 8. The approximated function values at the interior particles (which are unknowns in the discretized system equations) are considered for plotting. The approximated values are very close to the exact ones and the relative L^2 error norms in both the cases are of the order of 10^{-2}.

Table 3: Relative $\mathbf{L}^2$ error norms of some functions and their derivatives via DMS-FEM with triangular prisms

$f(x,y)$	$(x+y+z)^2$	$(x+y+z)^3$	$sin(x+y+z)$	$exp(x+y+z)$
n	2	3	5	5
N_{nd}	343	1000	4096	4096
N_e	54	54	54	54
$f-f^{a\,rel}_{L^2}$	7.720×10^{-16}	3.806×10^{-15}	3.054×10^{-8}	2.637×10^{-8}
$\frac{\partial f}{\partial x}-\left(\frac{\partial f}{\partial x}\right)^{a\,rel}_{L^2}$	2.600×10^{-15}	2.914×10^{-14}	1.082×10^{-6}	4.603×10^{-7}
$\frac{\partial f}{\partial y}-\left(\frac{\partial f}{\partial y}\right)^{a\,rel}_{L^2}$	3.180×10^{-15}	2.464×10^{-14}	7.958×10^{-7}	3.527×10^{-7}
$\frac{\partial f}{\partial z}-\left(\frac{\partial f}{\partial z}\right)^{a\,rel}_{L^2}$	5.153×10^{-16}	2.855×10^{-15}	4.965×10^{-7}	2.360×10^{-7}
$\frac{\partial^2 f}{\partial x\partial y}-\left(\frac{\partial^2 f}{\partial x\partial y}\right)^{a\,rel}_{L^2}$	9.321×10^{-14}	4.506×10^{-13}	1.224×10^{-5}	1.064×10^{-5}
$\frac{\partial^2 f}{\partial x\partial z}-\left(\frac{\partial^2 f}{\partial x\partial z}\right)^{a\,rel}_{L^2}$	6.680×10^{-15}	3.555×10^{-15}	1.238×10^{-5}	1.113×10^{-5}
$\frac{\partial^2 f}{\partial y\partial z}-\left(\frac{\partial^2 f}{\partial y\partial z}\right)^{a\,rel}_{L^2}$	9.044×10^{-15}	5.847×10^{-15}	1.566×10^{-5}	1.353×10^{-5}
$\frac{\partial^2 f}{\partial x^2}-\left(\frac{\partial^2 f}{\partial x^2}\right)^{a\,rel}_{L^2}$	1.348×10^{-13}	4.641×10^{-13}	6.260×10^{-6}	6.446×10^{-6}
$\frac{\partial^2 f}{\partial y^2}-\left(\frac{\partial^2 f}{\partial y^2}\right)^{a\,rel}_{L^2}$	1.053×10^{-13}	6.030×10^{-13}	8.924×10^{-6}	7.943×10^{-6}
$\frac{\partial^2 f}{\partial z^2}-\left(\frac{\partial^2 f}{\partial z^2}\right)^{a\,rel}_{L^2}$	4.147×10^{-14}	3.550×10^{-14}	2.074×10^{-5}	1.809×10^{-5}

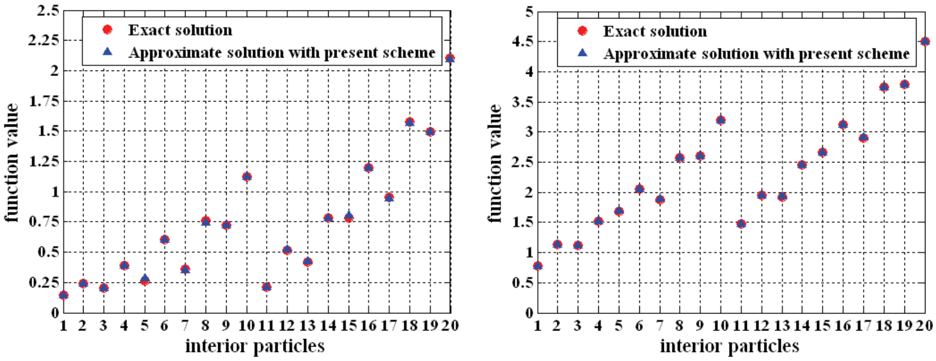


Figure 8: Comparison of exact and approximate solutions of Laplace's equation (left) and Poisson's equation (right)

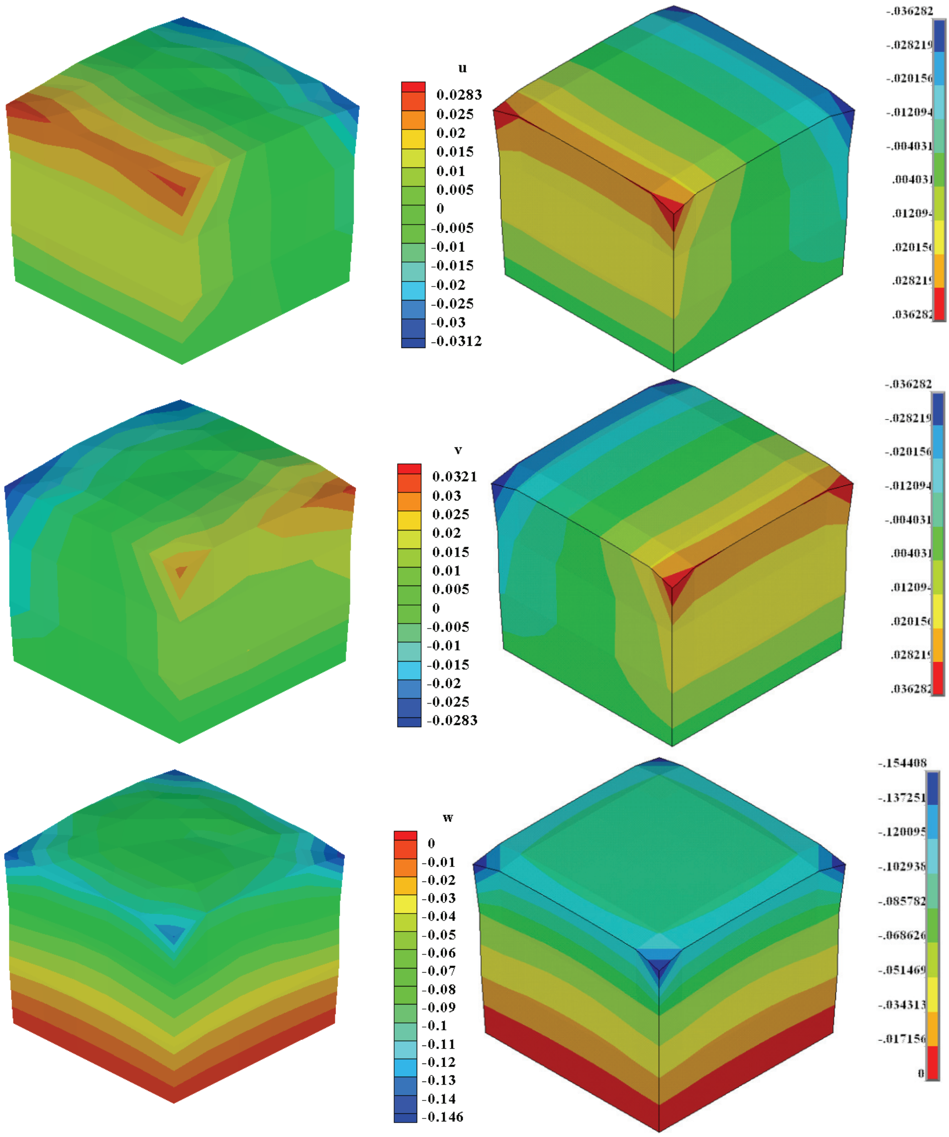


Figure 9: Displacement contours of the compressed cube; u, v and w are the displacements along x, y and z directions; results with TP-DMS-FEM is shown to the left and FEM results (using ANSYS®) on the right; comparable number of elements and particles (nodes) are used in both the methods

6.3 Linear Elastostatic Problems

Three linear elastostatic problems are solved with the TP-DMS-FEM and the results are compared with corresponding ones with the FEM (ANSYS®software).

6.3.1 Compression of a unit cube clamped at the base

To start with, a unit cube, clamped at its bottom face, is considered. Assuming the material to be linear, elastic and isotropic, the following properties are chosen: Young's modulus $E = 1$ and Poisson's ratio $\nu = 0.3$. The cube is decomposed into three layers of triangular prisms as shown in Fig.7. A total load of 0.1 unit, divided equally among all the particles on the top face of the cube, is applied. Quadratic kernels are employed to construct the shape functions. The deflected profiles of the cube are plotted in Fig.9 along with the FEM solutions for comparison. The solution obtained with 8000 20-noded brick elements is adopted as converged result. The absolute maximum values of u, v and w displacement fields, corresponding to the converged solution, are 0.031011, 0.031011 and 0.142592 respectively. The corresponding values obtained with the present scheme when the cube is modelled with 192 triangular prisms and 1043 particles, are 0.031187, 0.032084 and 0.145964. Finite element solution with 192 20-noded brick elements and 1143 nodes gives these values as 0.036282, 0.036282 and 0.154408 respectively. It can thus be concluded that the solution via TP-DMS-FEM yields substantially faster convergence than the FEM.

In order to assess the performance of the TP-DMS-FEM against volumetric locking, we now consider the deformation under compression of a unit cube whilst varying Poisson's ratio (ν) upwards towards 0.5 until the material approaches the incompressibility limit. As $\nu \to 0.5$ (i.e., the bulk modulus tends to $+\infty$), the associated bilinear form, even whilst remaining continuous, almost loses its ellipticity (coercivity). Hence the inversion of the discretized equations poses a challenge to the numerical scheme. Maximum displacement of the cube along the loading direction (z) is plotted against ν in Fig. 10(a). Remarkable numerical stability is exhibited by the method as $\nu \to 0.5$ and this is evident from Figs. 10(a) and (b). In Fig. 10(b), K and G represent the bulk and shear moduli respectively.

Fig.11 shows the displacement contours of the compressed cube via TP-DMS-FEM for $\nu = 0.49999999999999$. The extraordinary numerical stability against locking of the method, as $\nu \to 0.5$, is again demonstrated in this figure. It is of relevance to note that, for this problem, solutions via the FEM are far more prone to spurious numerical behaviour. Indeed, the ANSYS® software accepts a maximum value of $\nu = 0.4999$.

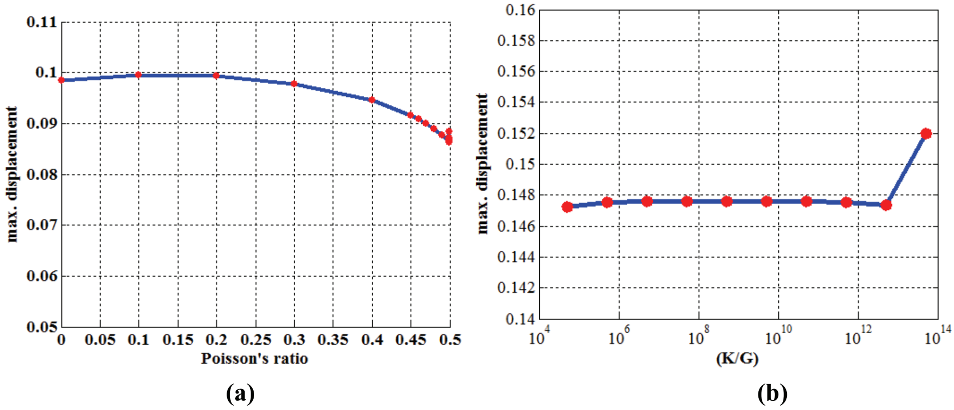


Figure 10: Numerical stability of the TP-DMS-FEM as the incompressibility limit is approached (i.e. $\nu \rightarrow 0.5$); maximum displacement of the cube in the z-direction is plotted against (a) Poisson's ratio, (b) K/G on a log scale (last 10 data points in (a) are plotted)

6.3.2 *Bending of a cantilever beam*

A cantilever beam of dimensions $1000 \times 100 \times 100$ unit^3 (Fig.12) is considered in this example. The left end of the cantilever is clamped and a total load of 2×10^5 units is applied vertically upwards at its free end. Linear isotropic elastic material properties are assumed with $E = 2 \times 10^5$ and $\nu = 0.3$. The exact solution, based on the theory of bending, corresponds to a maximum vertical displacement (v) = 20 units at the free end. The beam is presently solved through the TP-DMS-FEM using linear 3D elasticity theory and the domain discretization through the triangular prisms is shown in Fig. 12. The contours of computed displacements along x, y and z directions (u, v, w respectively) are plotted in Fig. 13 along with the FEM results (obtained via ANSYS®). The maximum displacement along the y direction obtained with TP-DMS-FEM using 258 triangular prisms and 1519 particles is 19.97887 units. The solutions given by the FEM with a comparable number of nodes and 20-noded brick elements are shown in Fig. 13 and the associated maximum displacement along the $y-$direction is 19.975 units.

6.3.3 *Bending of a thin plate clamped at two edges*

A thin plate with dimensions $600 \times 385 \times 3$ m^3 is clamped uniformly along its longer edges and is allowed to bend only by self weight. Once more, it is intended to solve the problem using 3D linear elasticity theory. The material properties adopted are: E $= 2 \times 10^{11}$ Pa, $\nu = 0.3$, density, $\rho = 7800$ kg/m^3. The plate is

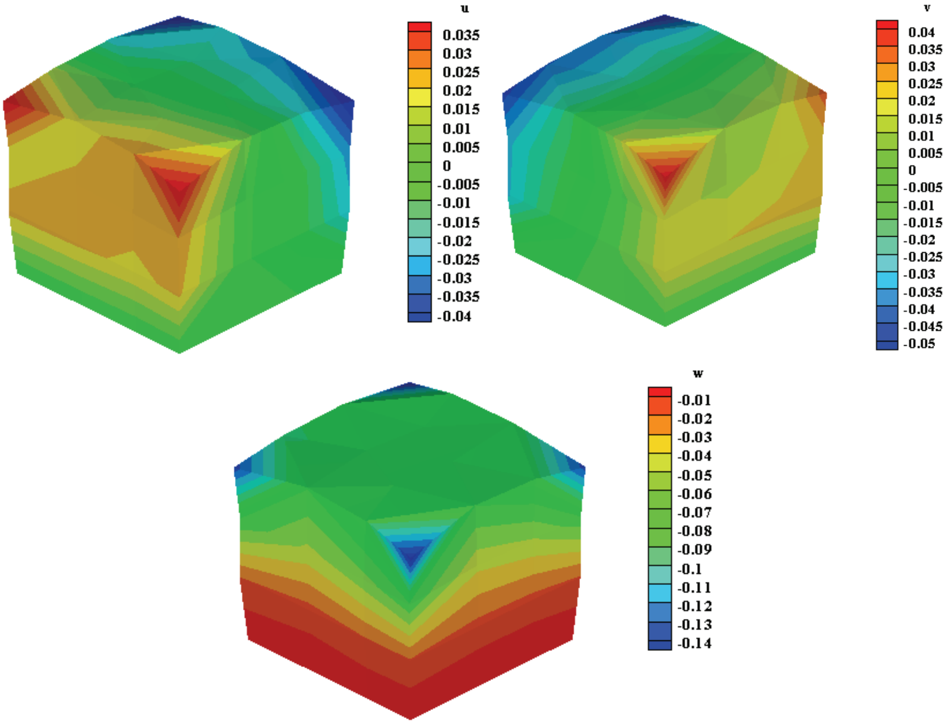


Figure 11: Displacement contours of the compressed cube for ν=0.49999999999999; u, v and w are the displacement fields along x, y and z directions respectively

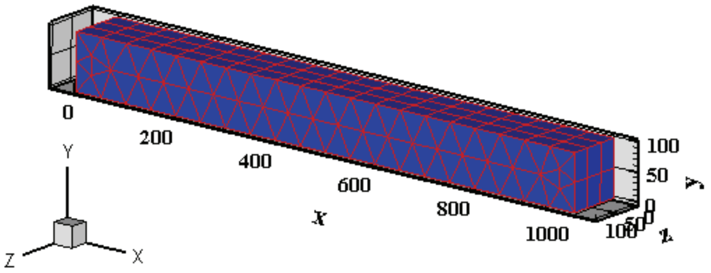
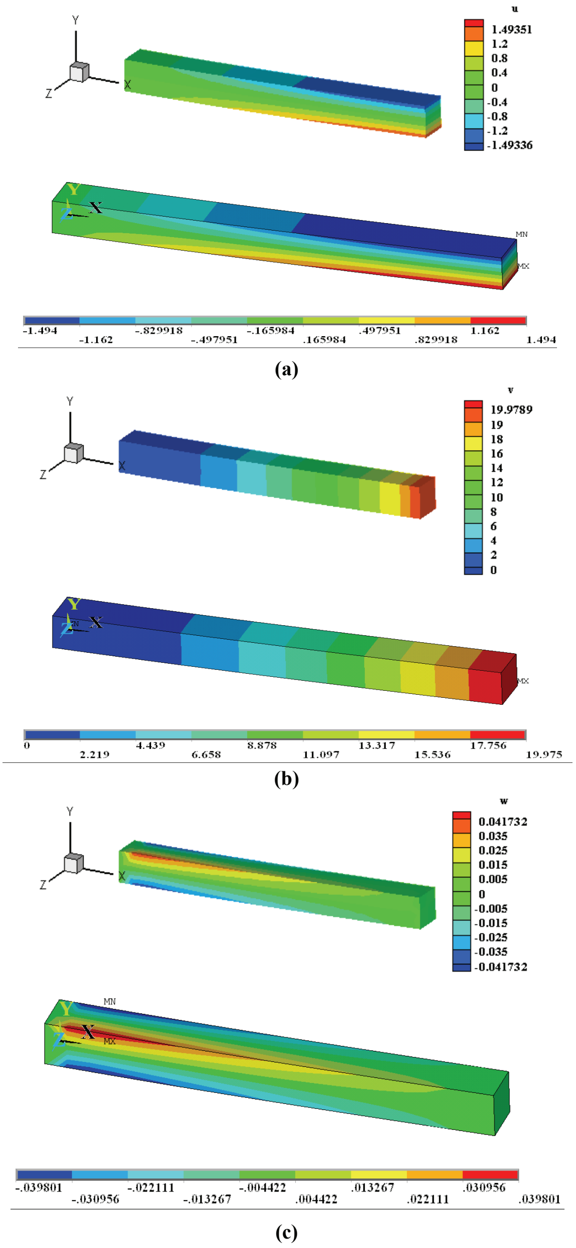


Figure 12: Cantilever: domain discretization via triangular prisms



(a)

(b)

(c)

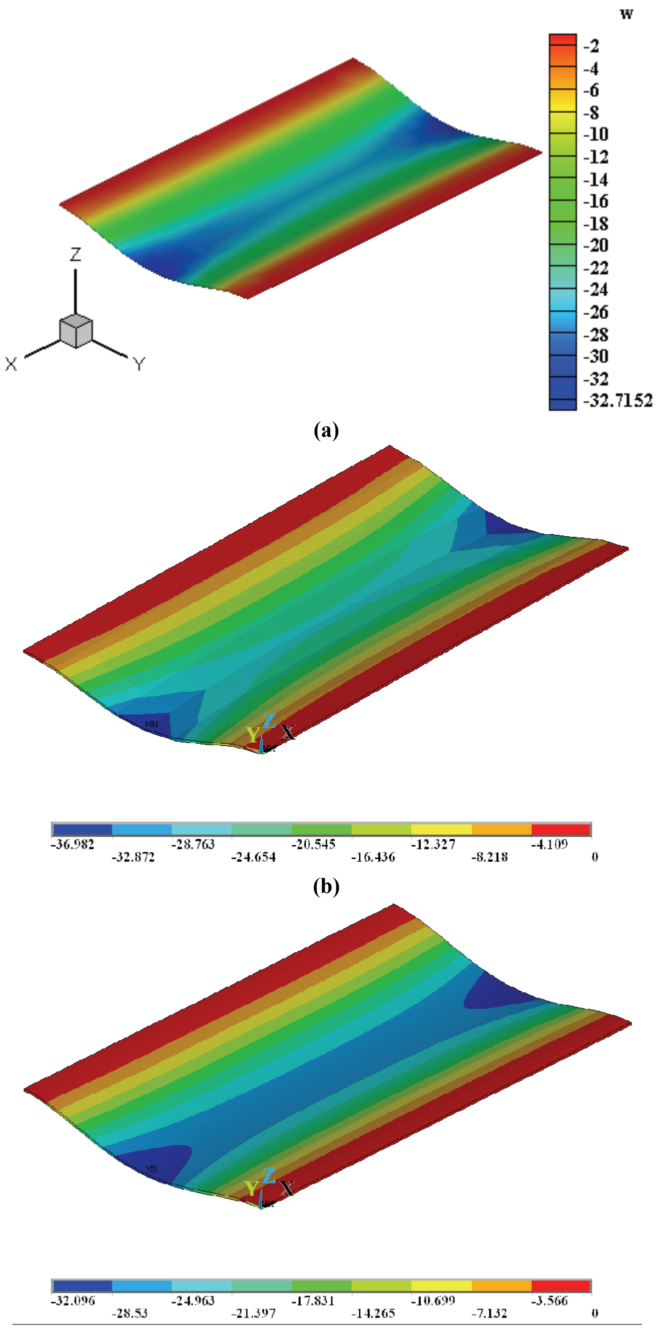
Figure 13: Displacement contours of the cantilever beam; (a) x - displacement (u), (b) y - displacement (v), (c) z - displacement (w); results with present approximation scheme on top and those with FEM 20 noded solid elements, below; similar number of elements and particles are used in both the methods

modelled with 286 triangular prisms in the TP-DMS-FEM. The transverse (w) displacements, so found, are compared with those obtained with 288 solid shell finite elements (ANSYS®software). The results are shown in Figs. 14(a) and (b). The ANSYS®solution using 7020 solid shell elements is considered as the converged one (Fig. 14c). It can be observed from Fig. 14 that the solution with the present scheme is closer to the converged one than the FEM solution with comparable number of elements. For instance, the absolute maximum value of w obtained with 286 triangular prisms in the present scheme is 32.7152 m, which is much closer to the converged solution (32.096) than that obtained with 288 solid shell finite elements (36.982).

7 Concluding Remarks

The present work on the 3D DMS-FEM constitutes a significant extension of its 2D counterpart, the smooth FEM based on the reproducing kernel DMS-splines over Delaunay triangulations of 2D domains (Sunilkumar and Roy 2010). In particular, we have presently developed a couple of different element formulations in 3D. The first corresponds to domain decomposition by tetrahedrons, wherein the polynomial reproducing shape functions are derived with DMS-splines over the tetrahedrons acting as the kernel functions. The second approach, on the other hand, uses a tensor-product form of the kernel function, which is composed of bivariate DMS-splines over 2D triangles and 1D NURBS. The kernel functions are in turn fed to the construction of the shape functions based on polynomial reproduction, as in the first case. The latter formalism requires that the 3D domain be decomposed through triangular prism elements. Both the formulations thus share certain commonalities with the conventional FEM and a class of mesh-less methods. However, unlike several mesh-less methods, the numerical integration of the weak form (of the governing equations) remains nearly conformal in view of the local supports of the shape functions almost coinciding with the tetrahedral or triangular prism elements.

Knot placement in the generation of trivariate DMS-splines (and hence the smooth shape functions over the tetrahedrons) is a highly involved task especially as the degree of the DMS-spline increases. For instance, this is numerically verified in the context of a few second order elliptic equations (Laplace's and Poisson's equations), wherein numerical solutios tend to become erroneous as the degree of the DMS-spline (and, consequently, that of polynomial reproduction) exceeds 2. Additionally, modelling very thin geometries (within the 3D framework) is cumbersome with tetrahedrons, where the aspect ratios may quickly deteriorate and the knot placement becomes even more problematic. These difficulties provide the motivation for the second approach based on triangular prism elements. Indeed, numerical



(a)

(b)

(c)

Figure 14: Displacement contours of a thin plate with (a) present approximation scheme modelling the plate with 286 triangular prisms, (b) FEM using 288 solid shell elements (ANSYS) and (c) FEM using 7020 solid shell elements (ANSYS)

experiments with the latter approximation scheme reveal that it works remarkably well for problems involving nearly incompressible material and for very thin plate-like structures modelled through 3D elasticity approach. This is in sharp contrast to the conventional FEM (without explicit stabilization terms in the weak form), wherein numerical pollution due to locking occurs. Even in terms of the rate of convergence, the DMS-FEM fares much better than the FEM.

One can anticipate, given the faster convergence and nearly locking-free behaviour of the new DMS-FEM, that this method should be particularly suitable for arresting the propagation of errors in problems whose solutions necessitate iterative computations (e.g., repeated inversions of discretized system matrices). In addition to a large class of geometrically and materially nonlinear problems wherein considerable numerical pollution is possible owing to discretization errors and uncertainty in the prescription of system parameters as well as boundary/initial conditions, this method should also be of great use in the discretization of forward PDE-s in the solution of inverse problems, most of which are characteristically ill-posed. Moreover, rigorous convergence estimates, leading to a-priori and a-posteriori error estimates and hence a multi-grid version of the method, are also of considerable interest as elements of future research.

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.

Amrouche C.; Philippe G.; Ciarlet L. G.; Srinivasan K. (2006): On the characterizations of matrix fields as linearized strain tensor fields, *Journal of Mathematiques*, Vol.86, pp.116-132.

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.; Liu H. T.; Han Z. D. (2006): 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.; 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.

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.

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

Brezzi F.; Fortin M. (1991): *Mixed and hybrid finite element methods*, Springer-Verlag, New York.

Dahmen W.; Micchelli C. A.; Seidel H. P. (1992): Blossoming begets B-splines built better by B-patches, *Mathematics of Computation*, Vol.59, number 199, pp.97 – 115.

Dolbow J.; Belytschko T. (1999): Numerical integration of the Galerkin weak form in mesh free methods, *Computational Mechanics*, Vol.23, pp.219 – 230.

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

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

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

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

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

Hughes T. J. R. (1995): Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, *Computer Methods in Applied Mechanics and Engineering*, 127, pp.387 – 401.

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

Jose M. A.; Cesar de Sa; Renato M.; Natal J.; Robertt A.; Fontes V.; Pedro M.; Almeida A. (2002): Development of shear locking-free shell elements using an enhanced assumed strain formulation, *International Journal for Numerical Methods in Engineering*, Vol.53, pp.1721-1750.

Kesavan, S. (2008): *Functional analysis and applications*, New Age Publishers, New Delhi.

Kui L. X.; Liu G. Q.; Zienkiewicz (1985): A generalized displacement method for the finite element analysis of thin shells, *International Journal for Numerical Methods in Engineering*, Vol.21, pp.2145-2155.

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

Liu W. K.; Jun S.; Li S.; Adee J.; Belytschko T. (1995a): Reproducing kernel 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.

Melenk J. M.; Babuska I. (1996): The partition of unity method, *International Journal for Numerical Methods in Engineering*, Vol.40, pp.727-758.

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.

Piegl L.; Tiller W. (1995): *The NURBS. Book*, Springer, Berlin.

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

Shaw A.; Roy D. (2008): NURBS-based Parametric Mesh-free Methods, *Computer Methods in Applied Mechanics and Engineering*, Vol. 197, pp. 1541-1567.

Shaw A.; Banerjee B.; Roy D. (2008): A NURBS-based method parametric bridging mesh free and finite element formulations, *CMES: Computer Modelling in En-*

gineering and Sciences, Vol. 26(1), pp.31-35.

Sunilkumar N.; Roy D. (2010): A Smooth Finite Element Method Based on Reproducing Kernel DMS-Splines, *CMES: Computer Modelling in Engineering and Sciences*, in the press.

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*, 21, pp.211-222.

Zienkiewicz O. C.; Taylor R. L.; Zhu, J. Z. (2000): *The finite element method: Its basis and fundamentals*, McGraw-Hill Publishers.