Arch. Comput. Meth. Engng.
Vol. **10, 4,** 307-384 (2003)

Archives of Computational Methods in Engineering
State of the art reviews

Overview and Recent Advances in Natural Neighbour Galerkin Methods

E. Cueto
Aragón Institute for Engineering Research (I3A)
University of Zaragoza
María de Luna, 3. E-50018 Zaragoza. Spain

N. Sukumar
Department of Civil and Environmental Engineering
University of California, One Shields Avenue
Davis, CA 95616 USA

B. Calvo, M.A. Martínez, J. Cegoñino and M. Doblaré
Aragón Institute for Engineering Research (I3A)
University of Zaragoza
María de Luna, 3. E-50018 Zaragoza. Spain
mdoblare@posta.unizar.es

Summary

In this paper, a survey of the most relevant advances in natural neighbour Galerkin methods is presented. In these methods (also known as natural element methods, NEM), the Sibson and the Laplace (non-Sibsonian) interpolation schemes are used as trial and test functions in a Galerkin procedure. Natural neighbour-based methods have certain unique features among the wide family of so-called meshless methods: a well-defined and robust approximation with no user-defined parameters on non-uniform grids, and the ability to exactly impose essential (Dirichlet) boundary conditions are particularly noteworthy.
A comprehensive review of the method is conducted, including a description of the Sibson and the Laplace interpolants in two- and three-dimensions. Application of the NEM to linear and non-linear problems in solid as well as fluid mechanics is studied. Other issues that are pertinent to the vast majority of meshless methods, such as numerical quadrature, imposing essential boundary conditions, and the handling of secondary variables are also addressed. The paper is concluded with some benchmark computations that demonstrate the accuracy and the key advantages of this numerical method.

1 INTRODUCTION

The finite element method (FEM), which was conceived in the 1950s, has now become the most widely-used numerical method for computer simulation in academic research as well as in engineering practice. Finite elements provide the possibility of handling arbitrary geometries with complex boundary conditions and nonlinear material behaviour with relative ease—the rich mathematical framework to demonstrate convergence and adaptive refinement and error estimation strategies have paved the way for its present success in modeling and simulation. A practical drawback, however, is the need for regeneration of the mesh in moving boundary and large deformation problems. This is often done by the analyst, and is considered to be one of the most time-consuming tasks in a finite element analysis.

To overcome the difficulty associated with remeshing, the past decade has seen a tremendous surge in the development of a family of Galerkin and collocation-based numerical methods—these are known as particle, gridless, meshfree, or meshless methods. For instance, smooth particle hydrodynamics [87], diffuse element method [91], element-free Galerkin method (EFGM) [19], material point method (MPM) [112], reproducing kernel particle method (RKPM) [79], finite point method [92], partition of unity method [85, 11], *h*-*p* clouds [45], natural neighbour Galerkin methods, or natural element meth-

ISSN: 1134–3060 Received: November 2002

ods (NEM) [24, 111], boundary node method (BNM) [89], meshless local Petrov-Galerkin method (MLPG) [6], method of finite spheres [40], and finite cloud method [2] are some of the members of this family. We refer the interested reader to the review articles by Belytschko and co-workers [17] and Li and Liu [77] for a detailed discussion and comparison of different meshless and particle methods.

Meshless methods share the common characteristic of there being no explicit connectivity information between nodes; the approximant is built in a process that is transparent to the user. Most of these meshless methods emerged as a consequence of using existing interpolation or data approximation techniques, such as moving least squares (MLS) approximation [73], as trial and test functions in a Galerkin procedure. Meshless approximants such as MLS do not interpolate nodal values and as a consequence, do not exactly reproduce essential (Dirichlet) boundary conditions. This aspect as well as the topic of numerical integration in meshless methods has attracted significant research interest in recent years.

The natural element method (NEM) is a Galerkin-based meshless method that is built upon the notion of *natural neighbour* interpolation. This interpolation scheme has very striking properties, such as its strictly interpolating character, ability to exactly interpolate piece-wise linear boundary conditions, and a well-defined and robust approximation with no user-defined parameter on non-uniform grids.

In this paper, we review the most notable aspects of the NEM with emphasis on the recent advances achieved by the authors in its application to solid as well as fluid mechanics. The outline of the paper is as follows: in Section 2, some well-known computational geometric constructs, and natural neighbour-based (Sibson and Laplace) interpolation schemes are described in detail. The imposition of essential boundary conditions is analysed in Section 3, and issues concerning errors in numerical integration and the description of a stabilized nodal quadrature scheme are presented in Section 4. A mixed approximation with application to incompressible and nearly incompressible elasticity is discussed in Section 5. In Section 6 the ability of the method to handle fluid mechanics problems with free surfaces (e.g., mould filling) is addressed. In Section 7, the Laplace interpolant is used to construct a finite difference scheme on unstructured grids. Applications in fluid mechanics and biomechanics are pursued in Section 8, and finally in Section 9, the main conclusions that are drawn from this study are mentioned.

2 NATURAL NEIGHBOUR-BASED INTERPOLANTS

Sibson [103, 104] introduced the notion of natural neighbour interpolation for data approximation and smoothing, and further investigations on the construction and properties of the natural neighbour (or Sibson) interpolant were carried out by Farin [50] and Piper [95]. Recently, renewed interest has sparked in the computational geometry community on the Sibson interpolant and generalization of Voronoi-based interpolation schemes that are based on natural neighbours [59, 60, 61, 20].

2.1 Voronoi Diagram and Delaunay Triangulation

Consider a bounded domain Ω in d-dimensions described by a set $\mathbf{N}$ of M scattered nodes: $\mathbf{N} = \{n_1, n_2, \ldots, n_M\}$. The Voronoi diagram [9] $\mathcal{V}(\mathbf{N})$ of the set $\mathbf{N}$ is a sub-division of the domain into regions $V(n_I)$, such that any point in $V(n_I)$ is closer to node n_I than to any other node $n_J \in N$ $(J \neq I)$. The region $V(n_I)$ (first-order Voronoi cell) for a node n_I within the convex hull is a convex polygon (polyhedron) in $\mathbb{R}^2$ $(\mathbb{R}^3)$:

$$V(n_I) = \{\mathbf{x} \in \mathbb{R}^d \, : \, d(\mathbf{x}, \mathbf{x}_I) < d(\mathbf{x}, \mathbf{x}_J) \; \forall J \neq I\}, \tag{1}$$

where $d(\mathbf{x}_I, \mathbf{x}_J)$ is an appropriate distance function (usually the standard Euclidean distance is used) between $\mathbf{x}_I$ and $\mathbf{x}_J$.

The dual of the Voronoi diagram, the Delaunay tessellation, is constructed by connecting nodes that have a common (d–1)-dimensional Voronoi facet. Given any nodal set $\mathbf{N}$, the Voronoi diagram is unique, whereas the Delaunay tessellation is not—a simple example is the triangulation of a square where choosing either diagonal leads to two valid Delaunay triangulations. In Figure 1a, the Voronoi diagram and the Delaunay triangulation are shown for a nodal set consisting of seven nodes ($M = 7$). A Voronoi vertex and an edge are also indicated in Figure 1a. An important property of Delaunay triangles is the *empty circumcircle criterion* [75]—if $\mathcal{DT}(n_J, n_K, n_L)$ is any Delaunay triangle of the nodal set $\mathbf{N}$, then the circumcircle of $\mathcal{DT}$ contains no other nodes of $\mathbf{N}$. In Figure 1b, the Delaunay circumcircles for three triangles are shown.

Consider now the introduction of a point p with coordinates $\mathbf{x} \in \mathbb{R}^2$ into the domain Ω (Figure 1b). The Voronoi diagram $\mathcal{V}(n_1, n_2, \ldots, n_M, p)$ or equivalently the Delaunay triangulation $\mathcal{DT}(n_1, n_2, \ldots, n_M, p)$ for the M nodes and the point p is constructed. Now, if the Voronoi cell for p and n_I have a common facet (segment in $\mathbb{R}^2$ and polygon in $\mathbb{R}^3$), then the node n_I is said to be a *natural neighbour* of the point p [103]. The Voronoi cells for the point p and its natural neighbours are shown in Figure 1c, together with the convex hull of the set of points.

2.2 Sibson Interpolant

The natural neighbour (Sibson) interpolant was introduced by Sibson [103]. For simplicity and ease of exposition, we restrict our attention to 2-dimensions, although every concept is easily extended to $d > 2$. The definition of the Voronoi diagram (1*st*-order) appears in Eq. (1). By a similar extension, one can construct higher order (k-order, $k > 1$) Voronoi diagrams in the plane. Of particular interest in the present context is the case $k = 2$, which is the second-order Voronoi diagram. The second-order Voronoi diagram of the set of nodes $\mathbf{N}$ is a sub-division of the plane into cells V_{IJ}, such that V_{IJ} is the locus of all points that have n_I as the nearest neighbour, and n_J as the second nearest neighbour. The second-order Voronoi cell V_{IJ} ($I \neq J$) is defined as [103]

$$V_{IJ} = \{\mathbf{x} \in \mathbb{R}^2 \, : \, d(\mathbf{x}, \mathbf{x}_I) < d(\mathbf{x}, \mathbf{x}_J) < d(\mathbf{x}, \mathbf{x}_K) \; \forall K \neq I, \, J\}. \tag{2}$$

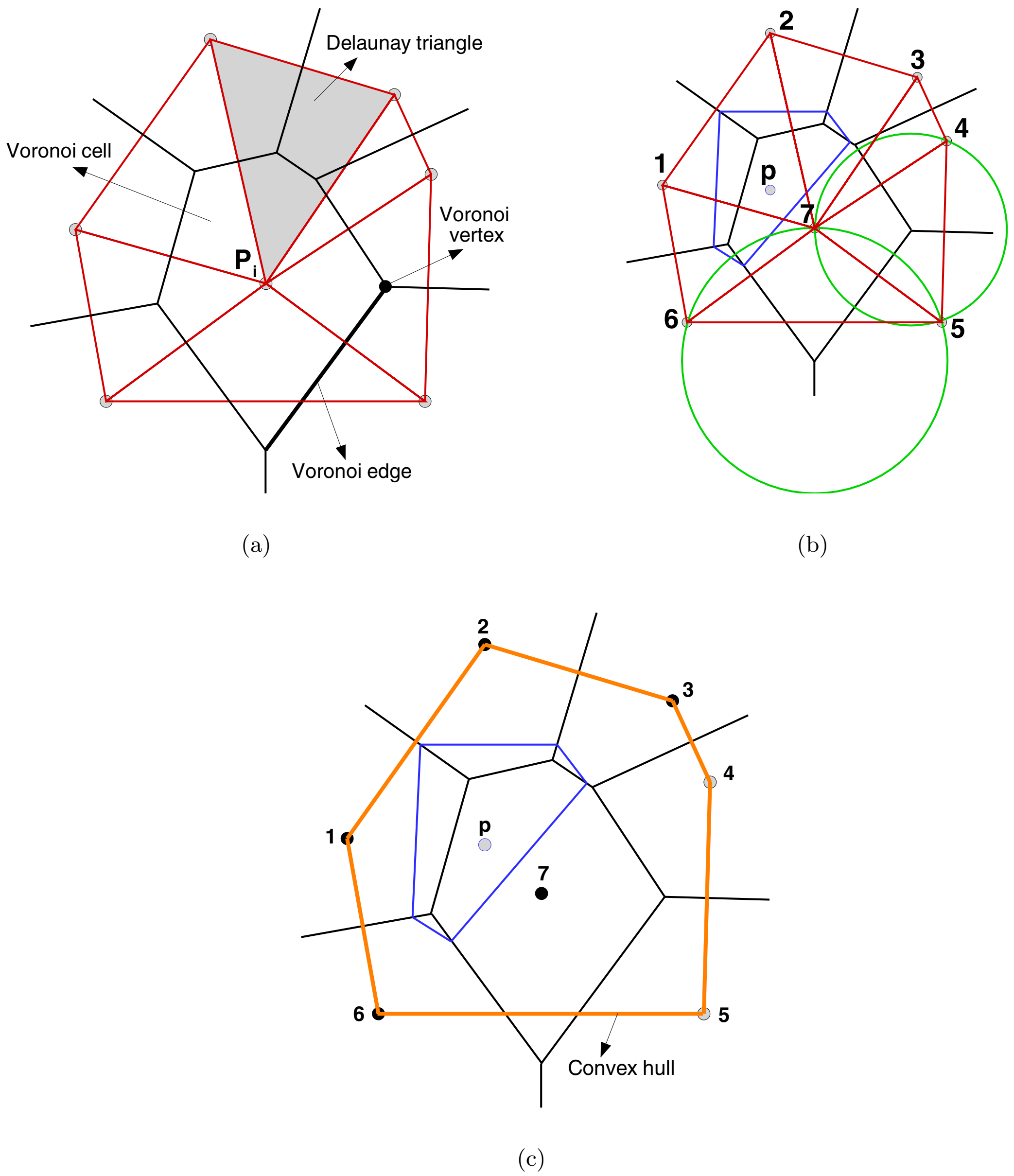
In order to quantify the neighbours for a point p with coordinate $\mathbf{x} = (x_1, x_2)$ that is inserted into the tessellation, Sibson [103] used the concept of second-order Voronoi cells, and thereby introduced natural neighbours and natural neighbour coordinates. Natural neighbour coordinates (shape functions) are used as the interpolating functions in natural neighbour (Sibson) interpolation, and as trial and test functions in a Galerkin implementation for the solution of partial differential equations (PDEs) [106]. Consider Figure 2a, where a point p is inserting into a tessellation. The natural neighbour shape function of p with respect to a natural neighbour I is defined as the ratio of the area of the second-order Voronoi cell (A_I) to the total area of the first-order Voronoi cell of p (A):

$$\phi_I(\mathbf{x}) = \frac{A_I(\mathbf{x})}{A(\mathbf{x})}, \quad A(\mathbf{x}) = \sum_{J=1}^{n} A_J(\mathbf{x}), \tag{3}$$

where $n = 5$ for the point p in Figure 2a. In 3-d, the Sibson shape function is defined as the ratio of polyhedral volumes.

The derivatives of the Sibson shape functions are obtained by differentiating Eq. (3):

$$\phi_{I,j}(\mathbf{x}) = \frac{A_{I,j}(\mathbf{x}) - \phi_I(\mathbf{x}) A_{,j}(\mathbf{x})}{A(\mathbf{x})} \quad (j = 1, \, 2), \tag{4}$$



(a)

(b)

(c)

Figure 1. (a) Voronoi diagram and Delaunay triangulation; (b) Delaunay circumcircles and (c) Natural neighbours (filled circles) of inserted point p

If the point $\mathbf{x} \to \mathbf{x}_I$, then $\phi_I(\mathbf{x}) = 1$ and all other shape functions are zero. Therefore, the properties of positivity, interpolation, and partition of unity are straightforward [110]:

$$0 \le \phi_I \le 1, \quad \phi_I(\mathbf{x}_J) = \delta_{IJ}, \quad \sum_{I=1}^{n} \phi_I(\mathbf{x}) = 1. \tag{5}$$

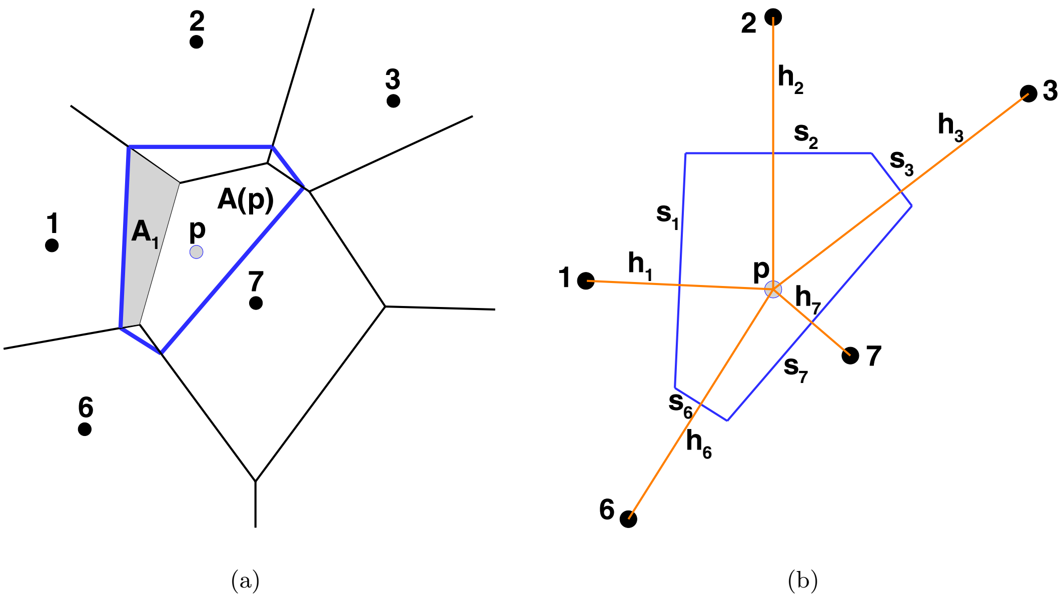


Figure 2. Natural neighbour-based interpolants. (a) Sibson interpolant; and (b) Laplace interpolant

Natural neighbour shape functions also satisfy the local coordinate property [103], namely

$$\mathbf{x} = \sum_{I=1}^{n} \phi_I(\mathbf{x})\mathbf{x}_I, \tag{6}$$

and hence in conjunction with the partition of unity property, it is readily derived that the Sibson interpolant can exactly represent any linear field which is known as linear completeness in the finite element literature [18].

2.3 Laplace Interpolant

We trace the roots of the Laplace interpolant with a view to summarize and unify some of the previous developments. As first noted in [107], Belikov and co-workers [14, 15, 16] as well as Hiyoshi and Sugihara [105, 59, 60] independently proposed a natural neighbour-based interpolant that was different from the Sibson interpolant. The former scientists who are in the field of data approximation and PDEs referred to the new interpolant as the non-Sibsonian interpolant, whereas Hiyoshi and Sugihara (computational geometers) coined it as the Laplace interpolant. Both groups recognized its connection to the Laplace equation, and used very similar approaches in delineating many of its properties. In this paper, we choose the name of Laplace interpolant. In [14, 15, 16], tools from vector calculus and data approximation theory are used to investigate the Laplace interpolant; Hiyoshi and Sugihara [105, 60] use the Minkowski theorem for convex polytopes [56] (Gauss's divergence theorem) and geometric-based theorems to study the Sibson and Laplace interpolants. They view the Laplace and Sibson interpolants as particular instances of Voronoi-based interpolants, and present a framework for generating a hierarchy of natural-neighbour based interpolants with increasing order of continuity at non-nodal locations. The Sibson interpolant can be obtained by the Voronoi-based integration of the Laplace interpolant and therein lies a reason why the Sibson interpolant is smoother than the Laplace interpolant [60]. By proceeding further, one can show that a limiting argument in the definition of the Laplace interpolant leads to the three-node Delaunay interpolant (constant strain finite elements).

The above results regarding to the Laplace interpolant are, however, preceded by an earlier work [33] on the Laplace weight that escaped the attention of the aforementioned researchers. In a series of papers, Christ and co-workers [32, 31, 33] investigated the possibility of carrying out quantum field theory computations in a discrete setting. Regular lattices violate some of the fundamental invariance postulates in quantum theory, and hence they considered the replacement of the space-time continuum by a random lattice. A discrete description of the continuum was obtained by using the Delaunay tessellation of all the sites. They associated the Laplace weight [33] to the Delaunay edges (link that connects two sites) in the tessellation. In [33], the divergence theorem is used to prove Eq. (6) as well as other identities on the lattice that are obeyed by the Laplace weights.

Let $\mathbf{N}$ denote a nodal set with the associated Voronoi cell for node I given in Eq. (1). Let t_{IJ} be the (d–1)-dimensional facet (segment in 2-d and polygon in 3-d) common to V_I and V_J, and $m(t_{IJ})$ the Lebesgue measure of t_{IJ}, i.e., a length in 2-d and an area in 3-d. If I and J do not have common facet, then $m(t_{IJ}) = 0$. Now, consider the introduction of a point p with coordinate $\mathbf{x} \in \mathbb{R}^d$ into the tessellation. If the point p has n natural neighbours, then the Laplace shape function for node I is defined as [33, 14]:

$$\phi_I(\mathbf{x}) = \frac{\alpha_I(\mathbf{x})}{\sum\limits_{J=1}^{n} \alpha_J(\mathbf{x})}, \quad \alpha_J(\mathbf{x}) = \frac{m(t_J(\mathbf{x}))}{h_J(\mathbf{x})}, \quad \mathbf{x} \in \mathbb{R}^d. \tag{7}$$

In 2-d, the above equation takes the form:

$$\phi_I(\mathbf{x}) = \frac{\alpha_I(\mathbf{x})}{\sum\limits_{J=1}^{n} \alpha_J(\mathbf{x})}, \quad \alpha_J(\mathbf{x}) = \frac{s_J(\mathbf{x})}{h_J(\mathbf{x})}, \quad \mathbf{x} \in \mathbb{R}^2, \tag{8}$$

where $\alpha_J(\mathbf{x})$ is the Laplace weight function, $s_I(\mathbf{x})$ is the length of the Voronoi edge associated with p and node I, and $h_I(\mathbf{x})$ is the Euclidean distance between p and node I (Figure 2b). The derivatives of the Laplace shape functions are obtained by differentiating Eq. (8):

$$\phi_{I,j}(\mathbf{x}) = \frac{\alpha_{I,j}(\mathbf{x}) - \phi_I(\mathbf{x})\alpha_{,j}(\mathbf{x})}{\alpha(\mathbf{x})} \quad (j = 1,\, 2), \tag{9}$$

where $\alpha(\mathbf{x}) = \sum_J \alpha_J(\mathbf{x})$. In 2-d, the Laplace shape function involves ratio of length measures whereas the Sibson shape function (see Eq. (3)) is based on the ratio of areas. Hence, the computational cost favors the Laplace interpolant in 2-d, with this advantage becoming even more significant in 3-d.

2.4 Trial and Test Functions

Let p be a point that belongs to $\Omega \subset \mathbb{R}^d$. The coordinate of p is denoted by $\mathbf{x}$. We can write the natural neighbour-based interpolation scheme for a scalar-valued function $u(\mathbf{x})$: $\Omega \to \mathbb{R}$ in the form:

$$u^h(\mathbf{x}) = \sum_{I=1}^{n} \phi_I(\mathbf{x}) u_I, \tag{10}$$

where u_I $(I = 1, 2, \ldots, n)$ are the unknowns at the n natural neighbours of p, and $\phi_I(\mathbf{x})$ is the Sibson or Laplace shape function associated with node I at p. The derivative of this interpolant is:

$$\frac{\partial u^h}{\partial x_j}(\mathbf{x}) = \sum_{I=1}^{n} \phi_{I,j}(\mathbf{x})\mathbf{u}_I \quad (j = 1, 2), \tag{11}$$

where the derivative of the Sibson and Laplace shape functions have been defined in Eq. (4) and Eq. (9), respectively. In [110] and [111], the Sibson and Laplace interpolants are used in a Galerkin method for applications in 2-d elasticity.

2.5 Properties

The properties of the Sibson shape functions, such as data interpolation, smoothness, and domain of support have been outlined by Sibson [103] and Farin [50]. The Laplace shape functions are also strictly positive, interpolate nodal data and form a partition of unity (see Eq. (5)). The support (region in which $\phi_I > 0$) of the Laplace shape function ϕ_I is the union of Delaunay circumcircles about node I (Figure 3), and they also form a linearly complete approximation [33]. In these respects, the Laplace and the Sibson shape functions share the same properties.

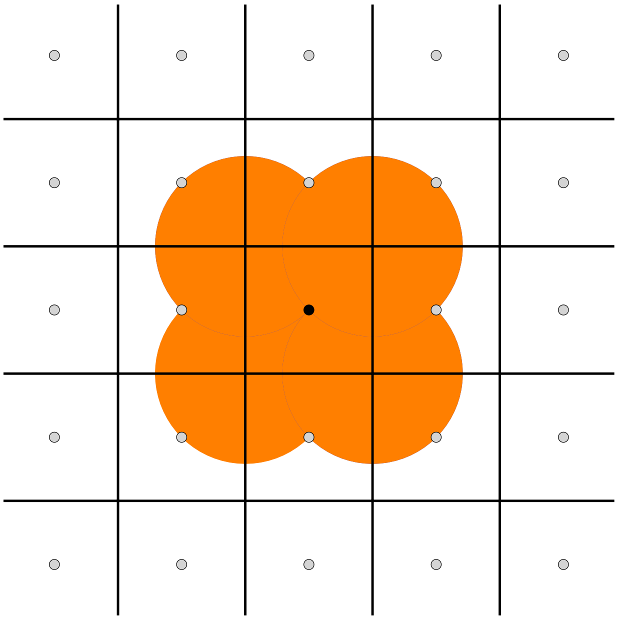


Figure 3. Support of natural neighbour-based shape functions

We present now the derivation for the linear completeness of the Laplace interpolant. For a domain Ω with boundary $\partial\Omega$, we can write

$$\int_{\Omega} \boldsymbol{\nabla} f \, d\Omega = \int_{\partial\Omega} f \mathbf{n} \, d\Gamma \tag{12}$$

by virtue of Gauss's theorem. On setting $f = 1$, we have

$$\int_{\partial\Omega} \mathbf{n}\, d\Gamma = \mathbf{0}, \tag{13}$$

which is also known as the Minkowski theorem [56] for convex polytopes in computational geometry. Now, on discretizing the above integral over the Voronoi cell of point p, we obtain [33, 105]

$$\sum_{I=1}^{n} \frac{\mathbf{x}_I - \mathbf{x}}{h_I(\mathbf{x})} s_I(\mathbf{x}) = \mathbf{0}, \tag{14}$$

and therefore

$$\mathbf{x} = \sum_{I=1}^{n} \phi_I(\mathbf{x})\mathbf{x}_I, \quad \alpha_I(\mathbf{x}) = \frac{s_I(\mathbf{x})}{h_I(\mathbf{x})}, \quad \phi_I(\mathbf{x}) = \frac{\alpha_I(\mathbf{x})}{\sum_J \alpha_J(\mathbf{x})}, \tag{15}$$

which is the linear reproducing condition in Eq. (6).

The Sibson and Laplace shape functions are distinct in two key aspects: smoothness and symmetry. The Sibson shape function is $\mathcal{C}^1\backslash\mathbf{x}_I$ ($\mathcal{C}^0$ at nodal locations) [104], whereas the Laplace shape function is $\mathcal{C}^0$ at nodal locations as well as on the boundary of the support (see Figure 3) [60]. The Laplace weight function is symmetric ($\alpha_{IJ} = \alpha_{JI}$), but the Sibson weight is not.

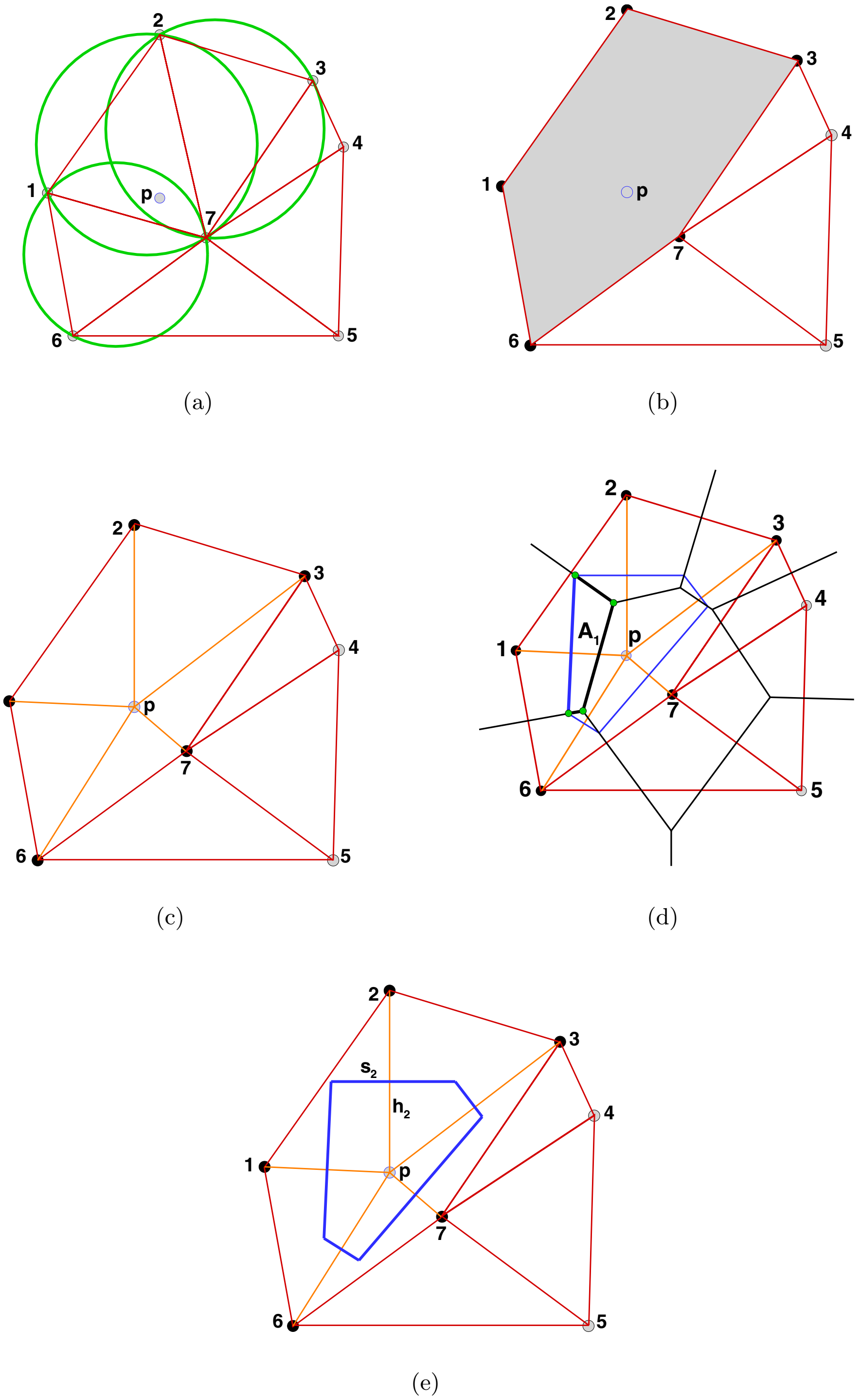
2.6 Computational Algorithm

In 2-d, a local implementation of the Bowyer-Watson algorithm [23, 119] can be readily adopted to compute the Sibson or Laplace interpolants. In some of the early studies on NEM [106, 110], Watson's algorithm [121, 120] was used, whereas [98] adopted Lasserre's algorithm [74]. Owens [93] and also Sambridge and co-workers [99] have developed a 3-dimensional implementation of natural neighbour interpolation. Watson's algorithm does not readily extend to 3-d and, in addition, the algorithm fails for points that lie on Delaunay edges. The Bowyer-Watson algorithm [23, 119] is the most efficient for evaluating Laplace shape functions in 2-d and 3-d, whereas Lasserre's algorithm [74] is suitable for computing the Sibson shape function in 3-dimensions.

We first describe the Bowyer-Watson algorithm, and then discuss Lasserre's algorithm. The sequence of illustrations in Figure 4 indicate the key steps in the evaluation of the Sibson and Laplace interpolants. Given a point p, the Delaunay circumcircles that violate the circumcircle criterion (p lies inside the circle) are found and the internal edges of the associated triangles are deleted (Figs 4a and 4b). Point p is connected to the facets on the outer boundary which defines the new triangulation (Figure 4c). Now, using simple geometric computations, the Sibson (Figure 4d) and the Laplace shape functions (Figure 4e) are evaluated. The Voronoi edge length (s_{IJ}) is the distance between adjacent Voronoi vertices. Simple algebraic formulas (see [110]) for the circumcenter of a triangle are used to evaluate the coordinate of the Voronoi vertices. For a triangle $t(A, B, C)$ with vertices $A(\mathbf{a})$, $B(\mathbf{b})$, and $C(\mathbf{c})$, the circumcenter (v_1, v_2) of t is:

$$v_1 = \frac{(a_1^2 - c_1^2 + a_2^2 - c_2^2)(b_2 - c_2) - (b_1^2 - c_1^2 + b_2^2 - c_2^2)(a_2 - c_2)}{D}, \tag{16a}$$

$$v_2 = \frac{(b_1^2 - c_1^2 + b_2^2 - c_2^2)(a_1 - c_1) - (a_1^2 - c_1^2 + a_2^2 - c_2^2)(b_1 - c_1)}{D}, \tag{16b}$$



(a)

(b)

(c)

(d)

(e)

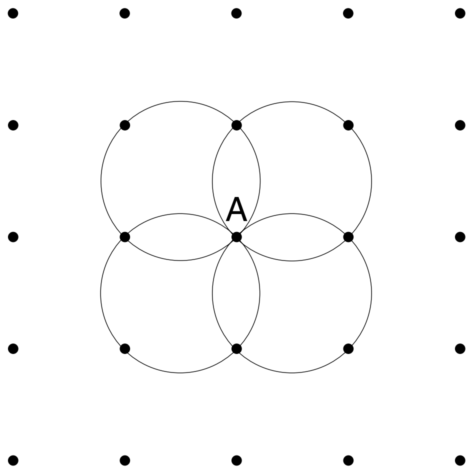
Figure 4. Computational algorithm for natural neighbour-based interpolants. (a) Delaunay triangulation and Delaunay circumcircles; (b) Deleted interior facets (edges) to form interior cavity; (c) Join boundary facets to point p to form new triangulation; (d) Sibson interpolant defined by overlapping areas of original Voronoi diagram and the Voronoi cell of p and (e) Laplace interpolant defined solely using the Voronoi cell of p

where D which is four times the area of triangle $t(A, B, C)$ is given by

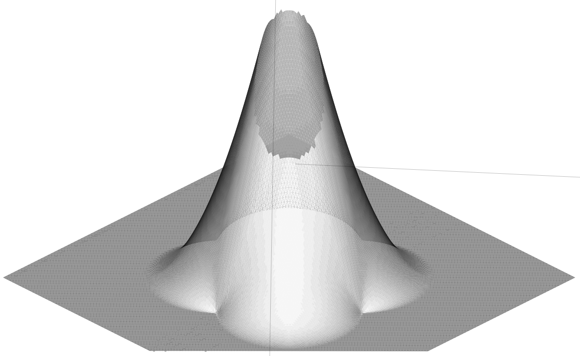
$$D = 2[(a_1 - c_1)(b_2 - c_2) - (b_1 - c_1)(a_2 - c_2)]. \tag{16c}$$

In the above equations, $\mathbf{a} = (a_1, a_2)$, $\mathbf{b} = (b_1, b_2)$, and $\mathbf{c} = (c_1, c_2)$ are the coordinates of the vertices of t.

In Figures 5b and 5c, the Sibson and Laplace shape functions are illustrated for a node located at the center of a regular grid (Figure 5a).



(a)



In this study, Lasserre's algorithm [74] has been adopted for the 3-d shape function computations. Lasserre's algorithm for the volume of convex polytopes in $\mathbb{R}^d$ is based on solving a constrained (bounded hyperplanes) linear programming problem. This approach is robust, and is applicable at any point in the domain for shape function computations. The starting point in the algorithm is the mathematical description of a convex polyhedron in terms of a set of algebraic inequalities [74]:

$$\{\mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{b}\}, \quad \mathbf{x} \in \mathbb{R}^d, \tag{17}$$

where $\mathbf{A}$ is a matrix of dimension (m, d), m is the number of restrictions that define the volume, and $\mathbf{b}$ is a column vector of dimension m. The volume enclosed by the polytope is denoted by $V(n, \mathbf{A}, \mathbf{b})$, where the i-th face of the polytope satisfies the following equation:

$$\{\mathbf{x} : (\mathbf{a}_i \cdot \mathbf{x}) = \mathbf{b}_i,\ \mathbf{x} \leq \mathbf{b}\}. \tag{18}$$

In Eq. (18), $\mathbf{a}_i$ represents the i-th column of $\mathbf{A}$, and its volume in $\mathbb{R}^{d-1}$ is $V_i(d{-}1, \mathbf{A}, \mathbf{b})$. A traditional approach to compute the volume of a polytope is given by the following relation:

$$V(n, \mathbf{A}, \mathbf{b}) = \frac{1}{d} \sum_{i=1}^{m'} d(\mathbf{p}, \mathbf{H}_i) V_i(n-1, \mathbf{A}, \mathbf{b}), \tag{19}$$

where $\mathbf{p}$ is a fixed point in space, m' is the minimum number of restrictions that define the polytope (redundant restrictions are not considered) and $d(\mathbf{p}, \mathbf{H}_i)$ is the distance from $\mathbf{p}$ to the hyperplane $\mathbf{H}_i$ given by the i-th restriction that defines the volume.

This algorithm can be constructed in a recursive way, such that the volume computation is performed within a binary tree, by starting with dimension d and leading to the computation of d longitudes in $\mathbb{R}$. The volume can be efficiently computed using the relation

$$V(n, \mathbf{A}, \mathbf{b}) = \frac{1}{d} \sum_{i=0}^{m} \frac{b_i}{|a_{it}|} V'_{it}(n{-}1, \overline{\mathbf{A}}_{i,t}, \overline{\mathbf{b}}_t). \tag{20}$$

In Eq. (20), $\overline{\mathbf{A}}_{i,t}$ represents the reduced matrix obtained from $\mathbf{A}$ by elimination of the t-th variable, by means of the equation $\mathbf{a}_i\mathbf{x} = b_i$; $\overline{\mathbf{b}}_t$ is the reduced vector after this elimination and a_{it} the t-th element of $\mathbf{a}_i$. V'_{it} represents the volume in dimension $d-1$ obtained with the reduced matrix $\overline{\mathbf{A}}_{i,t}$ and the reduced vector $\overline{\mathbf{b}}_t$. The value of t was chosen such that [24]

$$|a_{it}| = \max_j |a_{ij}|. \tag{21}$$

2.7 C^1 Natural Neighbour Interpolant

The construction of smooth interpolants is of interest since such classes of trial functions are necessary in a Galerkin formulation for the solution of higher-order PDEs. In a general framework, $\mathcal{C}^1$ continuity on the primary variable is required when second-order derivatives of the primary variable appear in the variational statement. For example, in the nucleation of a solid-solid phase transformation based on an energy functional that is dependent on the strain and strain gradients [35] or in strain-gradient damage models [5], $\mathcal{C}^1$-continuous trial functions are required in the Galerkin implementation.

Some of the early developments in $\mathcal{C}^1$ finite element interpolants are due to Clough-Tocher [36], de Veubeke [43], and Irons [64]. Local $\mathcal{C}^k$ $(k > 0)$ interpolants on a regular finite element mesh are readily constructed using splines. For irregularly-spaced data, an attractive choice for constructing $\mathcal{C}^k$ functions is to use $\mathcal{C}^k$ weight functions that have compact support; for example Shepard's interpolant [101] or moving least squares approximants [73]. The latter has been used within the element-free Galerkin method [19] for Kirchhoff plate bending problems [72].

Higher-order natural neighbour (Sibson) interpolants have been presented by Sibson [104] and Farin [50]. Recently, Hiyoshi and Sugihara [61] have proposed a Voronoi-based integration approach to generate $\mathcal{C}^k$ interpolants using the Sibson and Laplace interpolants. Here, we summarize some of the previous developments on Farin's $\mathcal{C}^1$ interpolant, and its use within a Galerkin setting for fourth-order PDEs [106].

Farin [50] embedded Sibson's coordinate in the Bernstein-Bézier representation of a cubic simplex to realize a $\mathcal{C}^1$ interpolant. A review article on triangular Bernstein-Bézier surfaces can be found in [48], and a general treatment of multivariate polynomials over multi-dimensional simplices is given by de Boor [42]. In what follows, multi-index notation denoted by bold characters is used, so $\mathbf{i} = (i_1, i_2, \dots, i_n)$ denotes a multi-index with norm $|\mathbf{i}| = i_1 + i_2 + \dots + i_n$ [49]. Let $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_n)$, with the property $\sum_I \xi_I = 1$, be the barycentric coordinate of a simplex $\delta \in \mathbb{R}^{n-1}$. A Bernstein-Bézier surface of degree m over the simplex δ can be written in the form [42]

$$b(\boldsymbol{\xi}) = \sum_{|\mathbf{i}|=m} B^m_{\mathbf{i}}(\boldsymbol{\xi}) b_{\mathbf{i}}, \tag{22}$$

where $b_{\mathbf{i}}$ is known as the Bézier ordinate associated with the control point $\mathbf{i}/m$. The control net of $b(\boldsymbol{\xi})$ is the network of $(n+1)$-dimensional points $(\mathbf{i}/m, b_{\mathbf{i}})$. In Eq. (22), $B^m_{\mathbf{i}}(\boldsymbol{\xi})$ are m-variate Bernstein polynomials in n variables:

$$(\xi_1 + \xi_2 + \cdots + \xi_n)^m = \sum_{|\mathbf{i}|=m} B^m_{\mathbf{i}}(\boldsymbol{\xi}), \qquad B^m_{\mathbf{i}}(\boldsymbol{\xi}) = \binom{m}{\mathbf{i}} \xi_1^{i_1} \xi_2^{i_2} \dots \xi_n^{i_n}, \tag{23a}$$

where $\binom{m}{\mathbf{i}}$ is the multinomial coefficient defined as

$$\binom{m}{\mathbf{i}} = \frac{m!}{i_1!\, i_2! \dots i_n!}. \tag{23b}$$

In one-dimension with $n = 2$, we have $\xi_1 = 1 - x$ and $\xi_2 = x$ as the barycentric coordinates of a simplex $\delta \in \mathbb{R}$. The univariate linear Bernstein polynomials ($m = 1$) are $\{1 - x,\, x\}$; the quadratic polynomials ($m = 2$) are $\{(1-x)^2,\, 2(1-x)x,\, x^2\}$; and the cubic polynomials ($m = 3$) are $\{(1-x)^3,\, 3(1-x)^2 x,\, 3(1-x)x^2,\, x^3\}$, where $x \in [0, 1]$. Multivariate Bernstein polynomials have properties very much like their univariate counterparts—partition of unity, positivity, and cardinal interpolation. The control points (circles) and associated Bézier ordinate values ($b_{\mathbf{i}}$) for a cubic Bernstein-Bézier triangular patch are shown in Figure 6. The Bézier ordinates at the vertices are referred to as vertex ordinates, the ones along the edges are known as the tangent ordinates and the one in the center is the center ordinate. For further details on triangular Bernstein-Bézier patches, the interested reader can refer to [49].

Consider a point p with coordinate $\mathbf{x} = (x_1, x_2)$ in 2-d. Let the natural neighbour shape functions of p be $\boldsymbol{\Phi} : \Omega \to \mathbb{R}^n$, where $\boldsymbol{\Phi} = \big(\phi_1(\mathbf{x}),\, \phi_2(\mathbf{x}), \dots, \phi_n(\mathbf{x})\big)$. The generalization of Bézier surfaces over a convex polygonal domain was proposed by Loop and DeRose [80]. By using $\boldsymbol{\Phi}$ instead of $\boldsymbol{\xi}$ in Eq. (22), we can construct the surface [50]

$$w^m(\boldsymbol{\Phi}) = \sum_{|\mathbf{i}|=m} B^m_{\mathbf{i}}(\boldsymbol{\Phi}) b_{\mathbf{i}}. \tag{24}$$

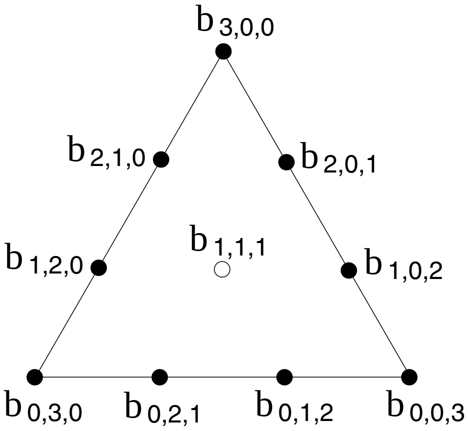


Figure 6. Cubic Bernstein-Bézier triangular patch

In the above equation, the Bézier ordinate $b_{\mathbf{i}}$ is associated with the control point $\mathbf{q}_{\mathbf{i}} \in \mathbb{R}^2$, where $\mathbf{q}_{\mathbf{i}}$ are the projection of the control points of the m-variate Bézier polynomial over the $(n-1)$-dimensional simplex onto the plane [50]:

$$\mathbf{q}_{\mathbf{i}} = \sum_{|\mathbf{j}|=1} B^1_{\mathbf{j}}(\mathbf{i}/m)\mathbf{x}_{\mathbf{j}}, \qquad |\mathbf{i}| = m. \tag{25}$$

The connectivity rule for Bézier simplexes states that the domain simplex has all its vertices connected to all other vertices. If $\mathbf{q}_{\mathbf{i}}$ and $\mathbf{q}_{\mathbf{j}}$ are two Bézier points in the n-gon simplex, then the rule indicates that there must exist integers r and s such that the multi-indices $\mathbf{i}$ and $\mathbf{j}$ satisfy

$$\mathbf{i} - \mathbf{e}_r = \mathbf{j} - \mathbf{e}_s, \tag{26}$$

where $\mathbf{e}_\alpha = (\delta_{1\alpha}, \delta_{2\alpha}, \ldots, \delta_{k\alpha}, \ldots, \delta_{n\alpha})$ denotes the multi-index having zero in all components except for the αth component, which is one.

If we choose $m = 1$ in Eq. (24) and let $w_I = w(\mathbf{x}_I)$ denote the nodal function value, we obtain

$$w^1(\mathbf{x}) = \sum_{I=1}^{n} \phi_1^{\delta_{1I}}(\mathbf{x})\phi_2^{\delta_{2I}}(\mathbf{x})\ldots\phi_n^{\delta_{nI}}(\mathbf{x}) b_{\mathbf{e}_I} = \sum_{I=1}^{n} \phi_I(\mathbf{x}) w_I \tag{27}$$

which is the original Sibson interpolant. Hence, Eq. (24) can be viewed as a generalized form of the Sibson interpolant. For $m = 3$, we arrive at the following surface representation [50]:

$$w^3(\mathbf{\Phi}) = \sum_{|\mathbf{i}|=3} B^3_{\mathbf{i}}(\mathbf{\Phi}) b_{\mathbf{i}}, \tag{28}$$

which was the $\mathcal{C}^1$ interpolant used by Sukumar *et al.* [109] in a Galerkin implementation for the solution of the biharmonic equation.

Farin [50] has outlined the properties of the $\mathcal{C}^1$ interpolant given in Eq. (28). By judicious choice of the center ordinate, quadratic completeness is realized, i.e., the interpolant can exactly reproduce a general quadratic function [50]. As opposed to the above, the $\mathcal{C}^1$

interpolant proposed by Sibson [104] can reproduce only spherical quadratics, i.e., functions of the form $a_0 + a_1 x + a_2 y + a_3(x^2 + y^2)$. By virtue of the quadratic completeness property, the $\mathcal{C}^1$ natural neighbour interpolant can exactly represent a state of constant curvature (second derivatives of the displacement for the thin plate problem) which is required in order to pass the patch test for the biharmonic equation [109].

2.7.1 Construction of $\mathcal{C}^1$ trial function

We use the interpolant presented in Eq. (28) as the $\mathcal{C}^1$ trial function. Let the point p in 2-d have n natural neighbours, with $\phi_I(\mathbf{x})$ the natural neighbour shape function of node I. Consider an interpolation scheme for a scalar-valued function $w(\mathbf{x})$: $\Omega \subset \mathbb{R}^2 \to \mathbb{R}$, in the form:

$$w^h(\mathbf{\Phi}) = \sum_{|\mathbf{i}|=3} B^3_{\mathbf{i}}(\mathbf{\Phi}) b_{\mathbf{i}}, \tag{29}$$

where the dependence on position $\mathbf{x}$ is implicit since the components of $\mathbf{\Phi}$ are the natural neighbour shape functions that are functions of $\mathbf{x}$. Referring to Eqs. (23a) and (24), we note that there are $n^2 + \binom{n}{3}$ Bézier ordinates in Eq. (29), and consequently $n^2 + \binom{n}{3}$ terms on the right-hand side of the above equation. In Figure 7, the vertex, tangent, and center Bézier ordinates with respect to node 1 are indicated for a pentagonal simplex ($n = 5$).

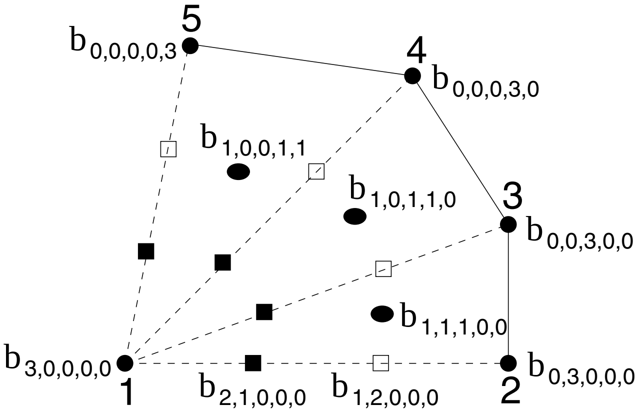


Figure 7. Bézier ordinates for a pentagonal simplex ($n = 5$)

The above form of the interpolant is suitable for applications in the context of data interpolation or surface approximation since nodal function values and sometimes even nodal gradient data are known *a priori*. For the numerical solution of PDEs by a Galerkin procedure, however, this is not the case since nodal function and nodal gradient values are unknowns, which are to be determined from the solution of the discrete system: $\mathbf{K}\,\mathbf{d} = \mathbf{f}$. To meet the desired goal, we use a transformation that renders the resulting interpolant amenable to numerical computations and implementation in the context of the numerical solution of PDEs. This is achieved by re-writing Eq. (29) in the following form:

$$w^h(\mathbf{\Phi}) = \{\mathbf{B}(\mathbf{\Phi})\}^T\{\mathbf{b}\} = \{\mathbf{B}(\mathbf{\Phi})\}^T[\mathbf{T}]\{\mathbf{w}\} = \{\mathbf{\Psi}(\mathbf{\Phi})\}^T\{\mathbf{w}\}, \tag{30a}$$

where

$$\{\mathbf{b}\} = [\mathbf{T}]\{\mathbf{w}\}, \quad \{\mathbf{\Psi}(\mathbf{\Phi})\}^T = \{\mathbf{B}(\mathbf{\Phi})\}^T[\mathbf{T}]. \tag{30b}$$

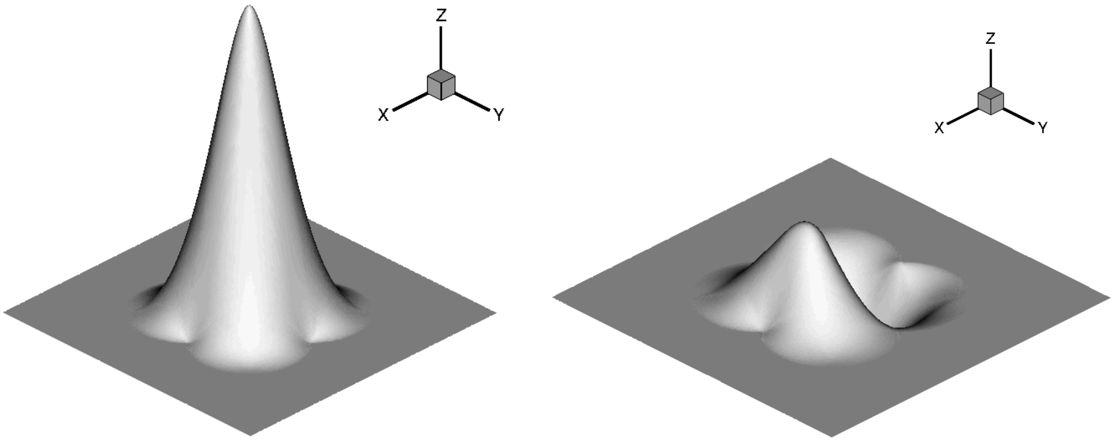
In Eq. (30), $\{\mathbf{B}\}$ and $\{\mathbf{b}\}$ are column vectors of dimension $n^2 + \binom{n}{3}$, and $[\mathbf{T}]$ is a transformation matrix of dimensions $(n^2 + \binom{n}{3}) \times 3n$. The transpose of the shape function vector $\{\mathbf{\Psi}(\mathbf{\Phi})\}^T = \{\psi_1(\mathbf{\Phi}),\, \psi_2(\mathbf{\Phi}),\, \psi_3(\mathbf{\Phi}), \dots ,\, \psi_{3n-2}(\mathbf{\Phi}),\, \psi_{3n-1}(\mathbf{\Phi}),\, \psi_{3n}(\mathbf{\Phi})\}$, and the transpose of the nodal vector $\{\mathbf{w}\}^T = \{w_1,\, \theta_{1x},\, \theta_{1y}, \dots ,\, w_n,\, \theta_{nx},\, \theta_{ny}\}$, where $w_I = w(\mathbf{x}_I)$ are the nodal function values, and $\theta_{Ix} = w_{,x}(\mathbf{x}_I)$ and $\theta_{Iy} = w_{,y}(\mathbf{x}_I)$ are the nodal gradient components. For the thin plate problem, w_I are the nodal displacements, and θ_{Ix} and θ_{Iy} are the nodal rotations. The matrix $[\mathbf{T}]$ is a transformation matrix that maps the nodal function and gradient values to the Bézier ordinates. The transformation from $\{\mathbf{B}(\mathbf{\Phi})\} \rightarrow \{\mathbf{\Psi}(\mathbf{\Phi})\}$ is based on a simple observation. In order to construct a $\mathcal{C}^1$ surface over an unstructured nodal grid, in general the nodal function values and nodal gradient values are required. In the Bernstein-Bézier surface representation given in Eq. (29), the vertex Bézier ordinates are identical to the nodal function values, and the tangent and center Bézier ordinates are related to the nodal function and nodal gradient data; see [109] for the relations.

The matrix $[\mathbf{T}]$ facilitates the representation of the interpolant in terms of nodal function values and nodal gradients, which renders it amenable to use in a PDE-setting; moreover, interpolation to both is achieved in [109]. We can hence view the $\mathcal{C}^1$ form of the Sibson interpolant as a bivariate generalization of one-dimensional Hermite cubic polynomials.

On constructing the transformation matrix $[\mathbf{T}]$ and carrying out the matrix-vector product indicated in Eq. (30b), we can express the $\mathcal{C}^1$ trial function as:

$$w^h(\mathbf{x}) = \sum_{j=1}^{3n} \psi_j(\mathbf{x}) w_j, \tag{31}$$

where $\psi_{3I-2}(\mathbf{x})$, $\psi_{3I-1}(\mathbf{x})$, and $\psi_{3I}(\mathbf{x})$ are the shape functions for node I that are associated with the nodal degrees of freedom w_I, θ_{Ix}, and θ_{Iy}, respectively. In Figure 8, the $\mathcal{C}^1$ shape functions $\psi_{3A-2}(\mathbf{x})$ and $\psi_{3A-1}(\mathbf{x})$ are illustrated for a node located at the center of a regular grid (see Figure 5a).



(a)

(b)

Figure 8. $\mathcal{C}^1(\Omega)$ Sibson shape function. (a) $\psi_{3A-2}(\mathbf{x})$ and (b) $\psi_{3A-1}(\mathbf{x})$

2.8 Numerical Applications

We discuss the numerical applications of the Sibson and Laplace interpolants. Braun and Sambridge [24] and Traversoni [118] adopted the Sibson interpolant, whereas Sukumar and co-workers [111] used the Laplace interpolant, as trial and test functions in a Galerkin implementation for the solution of partial differential equations. In [24], the numerical method was coined as the natural element method and in [111], it was referred to as natural neighbour Galerkin methods.

In [110, 111], both the Sibson and the Laplace interpolant were shown to reduce to a finite element interpolation for certain specific cases: in 1-d, these interpolants are identical to a linear finite element interpolation and in 2-d, if a point p has three neighbours, barycentric coordinates are obtained and for the case of four natural neighbours ($n = 4$) at the vertices of a rectangle, bilinear interpolation on the rectangle is obtained.

In [106, 109], the $\mathcal{C}^1$ form of the Sibson interpolant [50] was used in a Galerkin formulation for the biharmonic equation. In one-dimension, the $\mathcal{C}^1$ interpolant reduces to a cubic (Hermite) polynomial. If a point p in 2-d has two natural neighbours ($n = 2$), the interpolant is cubic between the two nodes; if $n = 3$, the approximation is a cubic surface over a triangular patch; and for a regular rectangular nodal grid ($n = 4$), the approximation is a bicubic patch [50].

Apart from applications of the Sibson and the Laplace interpolant in data approximation (e.g., see [122, 98, 68, 105]), these interpolants have been used as trial and test function in a Galerkin method (within the purview of *meshless methods*). Most of the initial applications focused on problems in solid mechanics, but recent explorations in other areas such as biomechanics [52] and non-Newtonian fluid mechanics [83] are noteworthy. In [108], the emergence of the Laplace weight in finite difference approximations on non-uniform grids is studied, and the link between Galerkin finite elements, finite volume and finite difference schemes on unstructured grids is also elucidated.

In quantum field theory, the replacement of the space-time continuum by a random lattice with an appropriate weight measure was explored in [33], and the properties of the Laplace weight were studied towards that goal. Lattice and spring-network models for fracture have received a lot of attention [13, 66], but previous studies have indicated the inherent difficulties associated with carrying out elastically homogeneous and grid-insensitive fracture simulations on random lattice networks. A partial resolution to the above shortcomings was proposed by Bolander and Saito [22], where the Laplace weight was used to successfully perform grid-insensitive crack propagation simulations on Voronoi grids, and recently the coupling of fracture and moisture transport in cement-based composites within the same framework has also been carried out [21]. In [57], the convergence properties of the non-symmetric random walk are studied using the Laplace weight. The diverse applications of the Laplace interpolants point to its promise as an attractive weight measure for irregular distribution of nodes in space, and in addition to its use in numerical computations on irregular grids (lattices).

3 IMPOSING ESSENTIAL BOUNDARY CONDITIONS IN NATURAL NEIGHBOUR GALERKIN METHODS

The imposition of essential boundary conditions is a key problem in most meshless methods, since the approximating functions are not interpolants and consequently nodal parameters in the Galerkin discrete system of equations are not the nodal displacements. Several techniques have been employed in an attempt to overcome this difficulty. These include the use of Lagrange multipliers [19], penalty formulations [53] or coupling with a single strip of finite elements [17], among others. While the last of these approaches leads to a loss of the meshless character of the method, none of the other approaches lead to true interpolation along the boundary of the domain. In fact, node-wise imposition of essential boundary conditions does not suffice for a proper interpolation along the boundary, since interior nodes have, in general, non-negligible influence on the boundary.

These problems are common to methods that use radial basis functions, or functions built upon tensorial product with square- or rectangular-support. Natural neighbour interpolation has different approximation characteristics in regard to imposition of boundary conditions, which is explained in detail in the following sub-section.

3.1 Approximation along Convex Boundaries

Natural neighbour interpolation leads to linear interpolation along the boundary of convex domains. This fact can be proved as follows: Consider a Delaunay triangle with two nodes on a convex essential boundary Γ_u (Figure 9), and a local coordinate system ξ defined between nodes 1 and 2. Assume for simplicity that ξ has only three natural neighbours, namely 1, 2 and 3, the shape function can be computed as follows:

$$\phi_I(\xi) = \frac{A_I(\xi)}{A(\xi)} \ (I = 1, 2, 3) \tag{32}$$

where $A(\xi) = \sum_{J=1}^{3} A_J(\xi)$. From Figure 9, the areas are computed as

$$A_1(\xi) = \lim_{L\to\infty} L\frac{1-\xi}{2} + \delta_1, \ \ A_2(\xi) = \lim_{L\to\infty} L\frac{\xi}{2} + \delta_2, \ \ A_3 = \delta_3 \tag{33}$$

where δ_I represent finite areas. We now have

$$\begin{aligned}
\phi_1(\xi) &= \lim_{L\to\infty} \frac{L(1-\xi) + 2\delta_1}{L + 2\delta_1 + 2\delta_2 + 2\delta_3}, \\
\phi_2(\xi) &= \lim_{L\to\infty} \frac{L\xi + 2\delta_2}{L + 2\delta_1 + 2\delta_2 + 2\delta_3}, \\
\phi_3(\xi) &= \lim_{L\to\infty} \frac{2\delta_3}{L + 2\delta_1 + 2\delta_2 + 2\delta_3},
\end{aligned} \tag{34}$$

and on taking the limits, we obtain

$$\phi_1(\xi) = 1 - \xi, \ \ \phi_2(\xi) = \xi, \ \ \phi_3(\xi) = 0. \tag{35}$$

It can be seen that, due to the unbounded area of Voronoi cells associated with nodes 1 and 2 on the convex boundary, the contribution of interior node 3 vanishes along Γ_u. This inference is, however, not valid for non-convex domains, where the contribution of interior nodes on the boundary is finite; in [106] errors of around 2% are reported. Thus, one can conclude that a sampling criterion is required to achieve true interpolation along non-convex boundaries. To accomplish this there are two choices—a CAD-description of the domain can be used, or secondly, one can use the notion of α-shapes to obtain a discrete representation of the domain.

3.2 Approximation along Non-Convex Boundaries

3.2.1 α-shapes based natural element method

In [39], a modification was introduced in the NEM. This modification is based on the concept of α-shapes. This is a generalization of the concept of *convex hull* of a cloud of points and are widely used in the field of scientific visualization and computational geometry to *extract* the shape of a cloud of points (nodes). This concept was first developed by Edelsbrunner

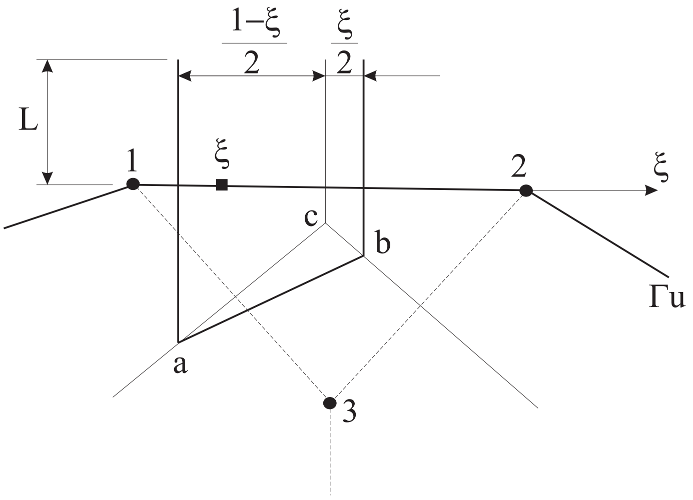


Figure 9. Linear interpolation along a convex boundary Γ_u

[46, 47]. Thus there is no need for explicit definition of the boundary of the domain. Instead, it is dynamically extracted from the cloud of nodes by invoking the notion of α-shape, which results in true interpolation over any boundary.

In essence, an α-shape is a polytope that is not necessarily convex nor connected. It is triangulated by a subset of the Delaunay triangulation of the nodes, and hence the empty circumcircle criterion holds. Let N be a finite set of points in $\mathbb{R}^3$ and α a real number, with $0 \leq \alpha < \infty$. A k-simplex σ_T with $0 \leq k \leq 3$ is defined as the convex hull of a subset $T \subseteq N$ of size $\mid T \mid= k+1$. Let b be an α-ball, i.e., an open ball of radius α. A k-simplex σ_T is said to be α-exposed if there exist an empty α-ball b with $T = \partial b \bigcap N$ where $\partial \Omega$ refers to the boundary of the ball. In other words, a k-simplex is said to be α-exposed if an α-ball that passes through its defining points contains no other point of the set N.

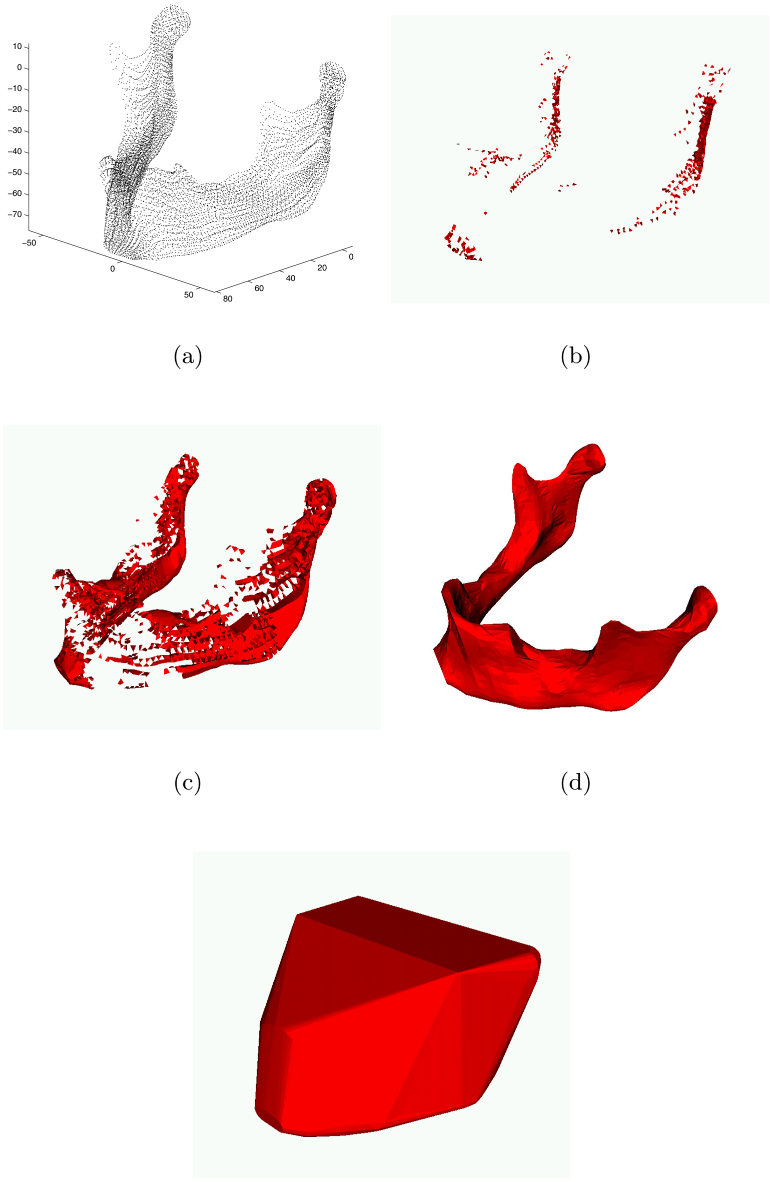
We can now define the family of sets $F_{k,\alpha}$ as the sets of α-exposed k-simplices for the given set N. This allows us to define an α-shape of the set N as the polytope whose boundary consists of the triangles in $F_{2,\alpha}$, the edges in $F_{1,\alpha}$ and the vertices or nodes in $F_{0,\alpha}$. As remarked before, an α-shape is a polytope that can be triangulated by a subset of the Delaunay triangulation or tetrahedralization, i.e., by an α-complex. An example of three-dimensional family of α-shapes is shown in Figure 10.

This definition constitutes the basis for what we have called the α-NEM. If the natural neighbourhood is limited to the case in which two nodes belong to the same triangle (tetrahedron) in a certain α-complex, the linear interpolation property over convex boundaries is also extended to the non-convex case. Hence the Voronoi cells are no longer the basis for the computation of the shape function. Instead, we consider a cell

$$T_I = \{\boldsymbol{x} \in \mathbb{R}^3 : d(\boldsymbol{x}, n_I) < d(\boldsymbol{x}, n_J) \ \ \forall J \neq I \wedge \sigma_T \in \mathcal{C}_\alpha(N)\}, \tag{36}$$

where $\mathcal{C}_\alpha(N)$ refers to an appropriate α-complex, and σ_T is the k-simplex that form n_I, n_J and any of the other node in the set N.

Consider a two-dimensional (extension to 3-d is straightforward) regular gridded set N of nodes along a non-convex boundary Γ_u (Figure 11). Let h be the nodal spacing, and the minimum value α to achieve a proper reproduction of the geometry is $\alpha = \frac{\sqrt{2}}{2}h$. Consider a point $\boldsymbol{x}$ that belongs to Γ_u; for a proper imposition of essential boundary conditions, this point must have an unbounded associated cell. For this to occur, the node A, that in a



(a)

(b)

(c)

(d)

(e)

Figure 10. Evolution of the family of α-shapes of a cloud of points representing a mandible. Shapes $\mathcal{S}_0$ (a), $\mathcal{S}_{1.0}$ (b), $\mathcal{S}_{1.5}$ (c), $\mathcal{S}_{3.5}$ (d) and $\mathcal{S}_\infty$ (e) are depicted

Delaunay triangulation will make the second order cell associated with $\boldsymbol{x}$ be bounded, must not be a natural neighbour of $\boldsymbol{x}$. The worst case occurs when $\boldsymbol{x}$ tends to B. The α-ball that would make the points A, B and $\boldsymbol{x}$ pertain to $\mathcal{C}_\alpha(N \bigcup \boldsymbol{x})$ would have a radius $\alpha' = h$. This is valid whenever the angle formed by the segments $\overline{AB}$ and $\overline{BC}$ is less than 90°. As shown in Figure 12, the choice of an appropriate α-shape leads to a continuous shape function.

In other cases (such as cracks for example), additional information is required to fully

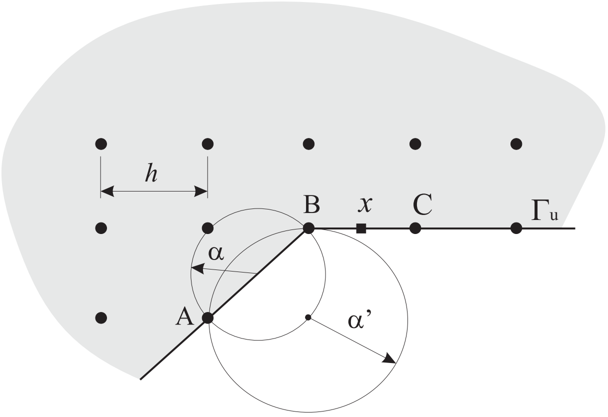


Figure 11. Neighbourhood in the context of α-complexes

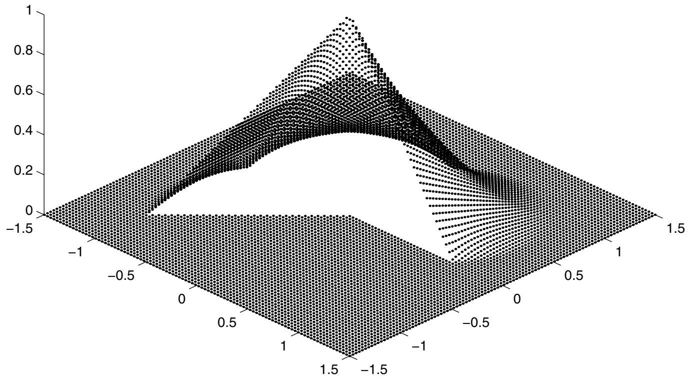
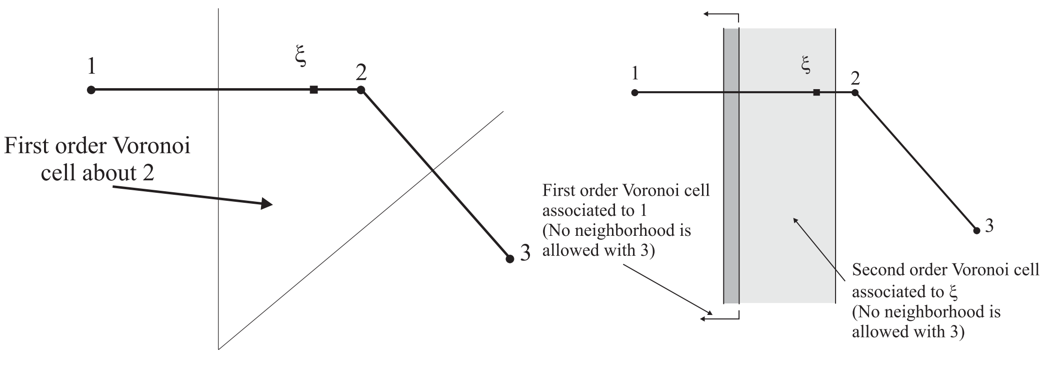


Figure 12. Shape function associated to node B in Figure 11

describe the model. In Sukumar [110], the definition of the domain is based on Planar Straight Line Graph (PSLG), which is a collection of points and segments that must be maintained in the final triangulation. In this case, it would be necessary to use a conforming Delaunay triangulation, and the proper imposition of essential boundary conditions is not guaranteed.

The proposed modification of the shape function computation breaks the duality (only outside the domain) between the Voronoi cells and the triangulation. However, it has been demonstrated that it leads to more accurate results [39], with comparable levels of accuracy on both convex and non-convex domains. For further insight on this topic, consider a situation similar to that in Figure 11. In the Figure 13a, the Voronoi cell about node 2 is depicted. Assume that the triangle 123 has been subtracted from the Delaunay triangulation in order to meet the α-criterion. Since no neighbourhood is allowed between points placed along segment 1–2 and node 3, the new Voronoi cell becomes unbounded for all points ξ along 1–2; the shape function is shown in Figure 12.

It is also well known that an α-shape does not accurately reproduce the domain of interest if non-uniform nodal (point) distributions are used. Since such distributions are usually needed when modeling high gradients of the primary variable of the problem, the use of density-scaled α-shapes [114] has been proposed. In simple terms, a density-scaled α-shape is an α-shape whose circumcircle radius is scaled according to a local measure of


1
ξ
2
First order Voronoi cell about 2
3
1
ξ
2
3
First order Voronoi cell associated to 1 (No neighborhood is allowed with 3)
Second order Voronoi cell associated to ξ (No neighborhood is allowed with 3)

(a) Standard Voronoi cell

(b) Modified Voronoi cell

Figure 13. Effect of the choice of neighbourhood in the Voronoi cell computation

the nodal (point) density. In this work, such a measure is established as

$$\delta(n_I) = \sum_{n_J \in N} 1 - \frac{d(n_I, n_J)}{\lambda} \quad \forall n_J \text{ such that } d(n_I, n_J) < \lambda, \tag{37}$$

where the radius λ (a constant) is a measure of the local neighbourhood, and $d(\cdot,\cdot)$ is the Euclidean distance. If in a certain k-simplex

$$\delta(\sigma_T) > \sum_{n_J \in N} \frac{\delta(n_J)}{|N|} = \mu \tag{38}$$

holds, or equivalently the k-simplex has a local density larger than the mean, the radius of the α-ball is changed. The new value of α for σ_T, namely α', is defined as:

$$\alpha' = \frac{\alpha}{\delta(\sigma_T)^{\beta}}, \tag{39}$$

where β is a user-defined parameter that should be adjusted depending on the nodal distribution and $\delta(\sigma_T)$ is a measure of the nodal density in the k-simplex being considered.

As indicated previously, the approach pursued in the α-NEM is distinct from the traditional NEM. The α-NEM constructs the model entirely in terms of nodes, and no explicit information (i.e., via a CAD-description) is required for the domain. If the nodes are judiciously placed, which means no more than they can actually reproduce the *shape* of the domain in the α-shape sense, it has been demonstrated [39] that the shape function is strictly an interpolant along any essential boundary. On the other hand, if NEM is based on a PSLG in 2-d, to preserve the Delaunay property, a conforming triangulation is required for non-convex domains which involves the introduction of additional (Steiner) points that are not user-controlled. Even in this case, the interpolating character of the shape function along the boundary is not guaranteed, and to the authors' knowledge the extension of the method to non-convex domains in 3-d has not yet been addressed.

3.2.2 Tracking an explicit definition of the boundary of the domain

The construction of a natural neighbour interpolant over an α-shape of the cloud of nodes is frequently a very restrictive constraint if the boundary of the domain is explicitly known in an analytical form, such as through a CAD-description. The need to build the interpolant over an α-shape—or even over a subset of an α-shape—forces the cloud of nodes to have a regularity not always desirable while only a sampling criterion in the neighbourhood of the boundary is strictly necessary if the boundary of the domain is known.

In this section, we review a condition first proposed in [38] to ensure linear interpolation along any kind of explicitly described boundary. This condition, as noted in previous sections, must be related to the desired level of detail for each region of the solid, and must affect only nodes neighbouring the boundary. It is clear from the α-shape reconstruction of a volume that regions to be represented at a finer level of detail need to be sampled finer. This is also true if the interpolating character of the natural neighbour shape function must hold along non-convex boundaries.

This new condition will be based on the so-called *medial axis* of the domain of interest. As opposed to the radius of curvature which is not properly defined in 3-d, the medial axis and the distance of a point to it is uniquely defined. The medial axis [3] of a d–1 dimensional, twice-differentiable, surface $\Gamma = \partial\Omega$ in $\mathbb{R}^d$ is the closure of the set of points which have two or more closest points in Γ. An example of the medial axis of a curve is shown in Figure 14.

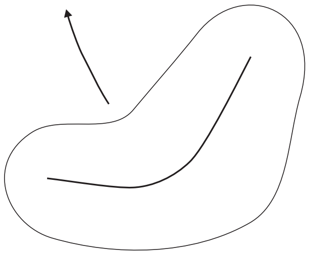


Figure 14. Medial axis of a two-dimensional curve

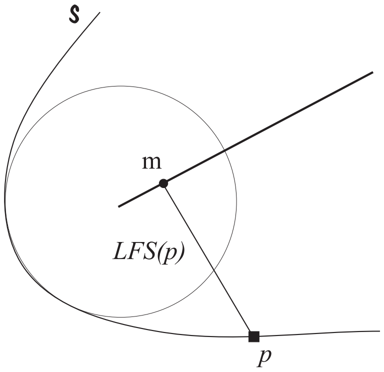


Figure 15. Computation of the LFS at a point p

The *local feature size* (Amenta *et al.* [3]), $LFS(p)$, of a point $p \in \Gamma$ is defined as the Euclidean distance from p to the closest point m on the medial axis. The LFS of a point quantifies the "level of detail" up to which the domain is represented at a point. In Figure 15 the computation of the LFS at a point is shown. We seek to define a measure of the sampling density of the curve or surface.

The surface Γ is said to be ε-sampled by a subset $\{n_I\}_{I=0}^{m}$ of the set N if every point $p \in \Gamma$ is within a distance $\varepsilon \cdot LFS(n_I)$ of a sample node $n_I \in \Gamma$. Since the behaviour of the natural neighbour interpolant along boundaries must not depend on nodes neighbouring these, we also define the concept of *curve Voronoi disk* [3]. A curve Voronoi disk (sphere in three dimensions) is a maximal disk, empty of nodes, centered at a point of the curve Γ:

Theorem 3.1. *A curve Voronoi disk on an ε-sampled surface Γ, with $\varepsilon \leq 1$, intersects Γ in a $d-1$ topological disk (that is, in a bicontinuous transformation of a $d-1$ disk).*

Proof. The proof we present follows Amenta *et al.* [4]. First we demonstrate that any Euclidean disk b intersecting the boundary either intersects it in a d–1 topological disk or contains a point of the medial axis. Consider the case in which $b \cap \Gamma$ is not a d–1 topological disk, for otherwise the proof is trivial. Now, there are two possible situations: first case is when $b \cap \Gamma$ contains a closed loop (a Jordan curve), then a part of the medial axis has to be entirely inside b; otherwise $b \cap \Gamma$ is formed by two or more components of Γ. In the latter case, consider the center of the disk c, and its nearest point in Γ, p. If p is not unique, then c begins to the medial axis, and the proof follows. Otherwise, consider the point q nearest c to be in some other connected component $\Gamma_q \neq \Gamma_p$. Any point $\mathbf{x}$ on the line segment cq is closer to q than to any other point outside b, so the closest point to $\mathbf{x}$ on Γ is always some point that belongs to $b \cap \Gamma$. Since at one end of the segment the nearest component is Γ_p and at the other is Γ_q, there must be a change in the nearest component at some point of the segment. A point with nearest points on two distinct components of $b \cap \Gamma$ exists, so this point belongs to the medial axis. This ends the first part of the proof.

For the complete proof, consider the contrapositive of the theorem: let b be a disk that does not intersect Γ in a topological disk. Then by the previous proof it contains a point of the medial axis m. The nearest point to the center c of b is at distance $LFS(c)$, so $d(c, m) \geq LFS(c)$. Since b contains the segment (c, m), its diameter is greater than $LFS(c)$. □

Corolary 3.2. *If the curve Voronoi disk is empty of nodes then the segment between the two nodes that define it can not be neighbour of any other portion of the boundary of the domain. This leads to the desired interpolating character of the natural neighbour interpolant (Sibson or Laplace) along similar lines used in the α-shape criterion, with no restriction imposed on the placement of strictly interior nodes.*

If the surface Γ is not twice-differentiable, as is the case for cracks or sharp corners, then the medial axis touches the curve and the radius of curvature vanishes. Thus, an infinite sampling density would be necessary to satisfy the previously presented criterion. This situation is similar to the case of using α-shapes, where an ideal α-value of zero would be necessary in the limit. In this case, it is possible to enrich the natural interpolation by adding to the basis, functions that are discontinuous along the crack or corner through the notion of partition of unity [85].

3.2.3 Imposition of interface conditions in piece-wise homogeneous domains

A problem closely related to the ones presented so far is the issue of imposing interface conditions in regions with two or more materials (see Figure 16). Kinematically admissible displacements must be $\mathcal{C}^0$ along the interface between materials, since gradient discontinuities must be allowed. Most meshless methods lead to a $\mathcal{C}^\infty$ approximation, and the introductions of such a discontinuity must be done explicitly, usually by adding new degrees of freedom that are associated with functions that have a jump in their derivatives; for example, see [17].

The approach followed in natural neighbour computations is essentially different. It has been shown that these interpolants can reproduce linear fields along any boundary, and hence it suffices if the set of nodes belonging to one material region are disallowed

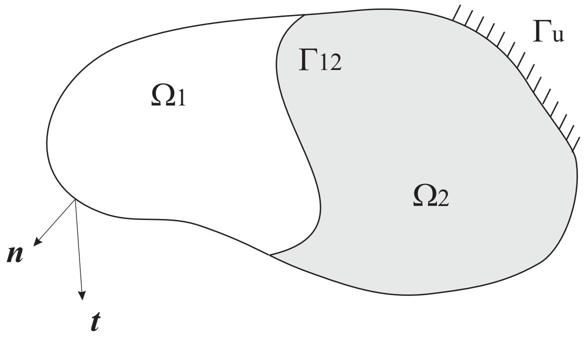


Figure 16. Bimaterial boundary-value problem

to be natural neighbours of those belonging to the other(s). This is straightforward in two-dimensional computations, but the three-dimensional case is a bit more involved.

In the two-dimensional case, there exist various approaches to the problem of triangulating a domain which is decomposed into two or more regions and whose interface is known and explicitly described by a set of segments that can must be maintained in the final triangulation. This set is known as a PSLG (see Section 3.2.1). One of the approaches is to construct a *conforming* triangulation, in which the empty circumcircle criterion is maintained by introducing additional points, which are known as *Steiner* points. A different approach is based on a *constrained* triangulation, i.e., a triangulation in which some triangles lose their Delaunay condition in order to enforce the constraints posed by the interface segments. This approach is not useful within the context of natural neighbor interpolation since the Delaunay property and its duality to the Voronoi diagram is lost. In the finite element method, material properties are associated with the elements and the edges of the finite element coincide with the material boundary. Hence the satisfaction of displacement and traction (weakly) continuity on the material boundary is trivial.

In the α-NEM method, a PSLG is not used. Two or more sets of nodes belonging to different material regions of the model are considered, and material properties are assigned to the nodes. We then construct the shapes of the different sub-domains, starting with the lowest α valued α-shape and storing each of them separately. The first α-shape is constructed in the usual way, no matter if there exist one or more material regions in the model. The second α-shape is constructed by taking into account not only its defining nodes, but also the nodes placed on the material boundary that belongs to the first α-shape. If the value for α in this second α-shape is greater than in the first one, and assuming similar nodal spacing parameters h on both sides of the boundary, the interface between materials will not be properly reproduced due to the lower value of α in the first α-shape. In this region, a number of triangles that overlap the first α-shape will appear. If the nodal spacing parameter h was appropriately chosen so as to accurately reproduce the boundary, no loss of generality is made if we assume that it is strictly lower than the minimum radius of curvature of the boundary at this zone and consequently, strictly lower than α. It is easy to conclude that the overlapping triangles will be those that have the three nodes pertaining to the first α-shape. A simple loop over the generated triangle list will suffice to find those triangles that must be eliminated.

This simple algorithm has been used to construct the α-shape of the head of a femur with a prosthesis that is shown in Figure 17. Since the two triangle lists are stored separately (they represent domains of different materials) it is very easy to search for natural neighbours of a point. This is necessary to ensure $\mathcal{C}^0$ continuity on the boundary. This approach greatly simplifies the search for natural neighbours, while maintaining the Delaunay character of the resulting triangulation without the need to add any Steiner points.

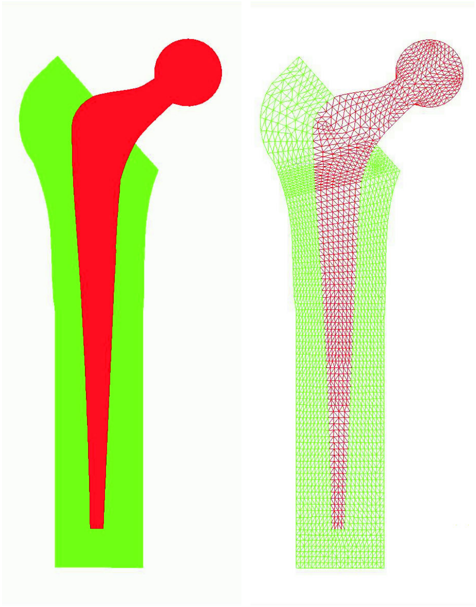


Figure 17. α-shape of a femur with a prosthesis

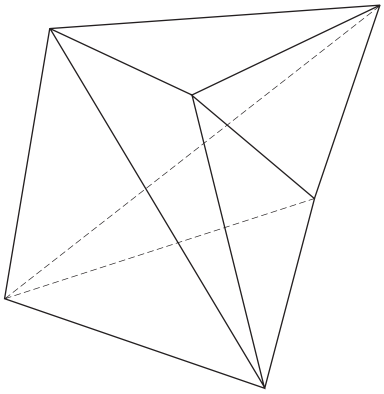


Figure 18. Schönhardt's polyhedron

However, this algorithm is not directly extendable to 3-d. This is due to a variety of reasons. First, there does not exist a decomposition of a general, non-convex, three-dimensional polyhedron into tetrahedra without the addition of Steiner points. This fact is true even for non-Delaunay triangulations. The simplest of these non-convex tetrahedra is the so-called Schönhardt polyhedron [100] which is shown in Figure 18.

If, in addition, we need to develop a Delaunay triangulation constrained to a set of triangles that define the boundary of the different regions of the model (usually known as a *Piece-wise Linear Complex* (PLC) [102]), we will need to add Steiner points that are not user-controlled. An algorithm to do this can be found in the work of Chazelle and Palios [27], but it does not perform a Delaunay tetrahedralization. The authors do not know of any implementation of this algorithm. In addition, the problem of determining when to introduce a Steiner point is an NP-hard problem [97], with complexity $\mathcal{O}(n^2)$ in the worst case, if there are n triangular facets limiting the domain. In [37] this problem was approached using the concept of α-shapes. Instead of introducing up to n^2 Steiner points, information provided by the α-shapes is used to build a Delaunay tetrahedralization of a topologically equivalent set of nodes. In the algorithm, first the lower α-valued α-complex is constructed. After this, those tetrahedra that contain any interface node are mirrored towards the second α-shape (Figure 19).

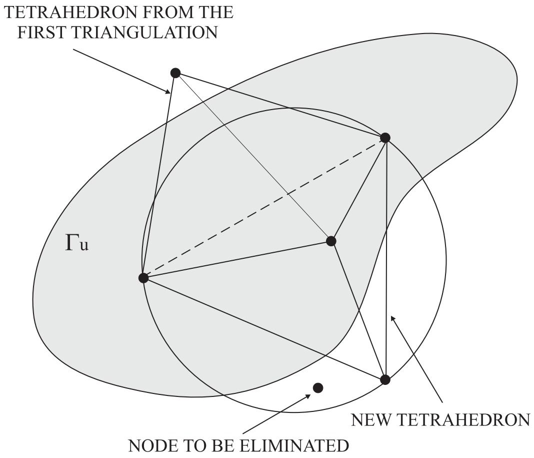
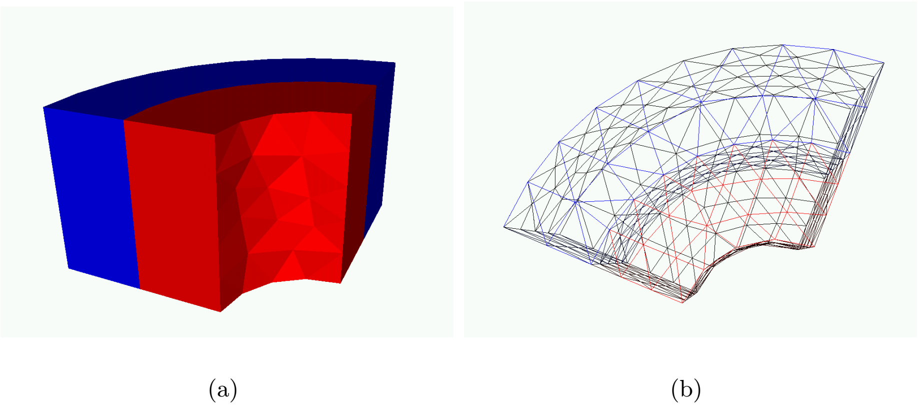


Figure 19. Mirroring of the interface tetrahedra to build a conforming 3D α-complex

This mirroring produces the addition of n new nodes in the model. All the nodes that fall within the α-ball of the mirrored tetrahedra are deleted in order to preserve the Delaunay character of the tetrahedralization. The final number of nodes in the model will be about the same and the algorithm runs in $\mathcal{O}(n)$ time (plus the standard $\mathcal{O}(n \log n)$ time for the α-shape). Like the two-dimensional case, after the completion of the second α-shape, some tetrahedra that overlap the first one must be deleted. Again, these tetrahedra correspond to those that have their four nodes in the first α-shape. An example (α-shape of two concentric cylinders) that illustrates this algorithm is shown in Figure 20.



(a) (b)

Figure 20. α-shape of two concentric cylinders

3.3 NEM–FEM Coupling

The coupling of meshless methods to finite elements has been an active research topic in recent years, since most meshless approximations fail to interpolate along essential boundaries which complicates the imposition of essential boundary conditions. Belytschko and co-workers [17, 71] proposed the coupling between FEM and EFGM to impose essential (Dirichlet) boundary conditions. A single string of finite elements is used to ensure the proper imposition of essential boundary conditions, but the meshfree character of the method is lost. Also, in [62] the coupling and the enrichment of traditional FE with the EFGM is proposed. Typically, in these approaches, a region exists in which both FE and EFG approximations have influence. In the case of the natural element method (with both Sibsonian and Laplace interpolants) coupling to FE is straightforward. Once the proper interpolation is achieved along the boundary, precluding nodes in the NEM region to be natural neighbours of those in the FEM region suffices to ensure a fully compatible weak form of the problem. Both continuity and consistency of the method are trivially met if linear finite elements are used.

3.4 Numerical Examples

The convergence of the NEM and the α–NEM for a benchmark problem in 2-d and 3-d elasticity, respectively, is studied. The weak form for elasticity with no body forces is: find $\mathbf{u} \in \mathcal{S}$ such that

$$\int_{\Omega} \delta\boldsymbol{\varepsilon} : \boldsymbol{\sigma}\, d\Omega = \int_{\Gamma_t} \partial\mathbf{u} \cdot \bar{\mathbf{t}}\, d\Gamma \ \forall \partial\mathbf{u} \in \mathcal{V}, \tag{40}$$

where $\mathbf{u}$ is the trial solution, and $\delta\mathbf{u}$ is the test or weighting function (δ is the first variation operator). A standard Galerkin procedure is used to obtain the discrete equations.

3.4.1 Convergence study in 2-d elasticity: infinite plate with a circular hole

An infinite plate with a traction free circular hole under unidirectional tension along x_1 is considered (Figure 21). The exact solution to this problem is given in [117] as well as [113]. The domain $ABCDE$ shown in Figure 21 is modeled with the exact tractions imposed along BC and CD. Due to symmetry, the essential boundary conditions are: $u_2 = 0$ along AB, and $u_1 = 0$ along DE.

In polar coordinates (r, θ), the exact stress distribution for $\sigma_0 = 1$ is given by

$$\sigma_{11}(r,\theta) = 1 - \frac{a^2}{r^2}\left(\frac{3}{2}\cos 2\theta + \cos 4\theta)\right) + \frac{3}{2}\frac{a^4}{r^4}\cos 4\theta, \tag{41a}$$

$$\sigma_{22}(r,\theta) = -\frac{a^2}{r^2}\left(\frac{1}{2}\cos 2\theta - \cos 4\theta)\right) - \frac{3}{2}\frac{a^4}{r^4}\cos 4\theta, \tag{41b}$$

$$\sigma_{12}(r,\theta) = -\frac{a^2}{r^2}\left(\frac{1}{2}\sin 2\theta + \sin 4\theta)\right) + \frac{3}{2}\frac{a^4}{r^4}\sin 4\theta, \tag{41c}$$

where a is the radius of the circular hole. The displacement components (rigid-body displacement and rotation set to zero) are:

$$u_1(r,\theta) = \frac{a}{8\mu}\left[\frac{r}{a}(\kappa+1)\cos\theta + 2\frac{a}{r}((1+\kappa)\cos\theta + \cos 3\theta) - 2\frac{a^3}{r^3}\cos 3\theta\right], \tag{42a}$$

$$u_2(r,\theta) = \frac{a}{8\mu}\left[\frac{r}{a}(\kappa-3)\sin\theta + 2\frac{a}{r}((1-\kappa)\sin\theta + \sin 3\theta) - 2\frac{a^3}{r^3}\sin 3\theta\right], \tag{42b}$$

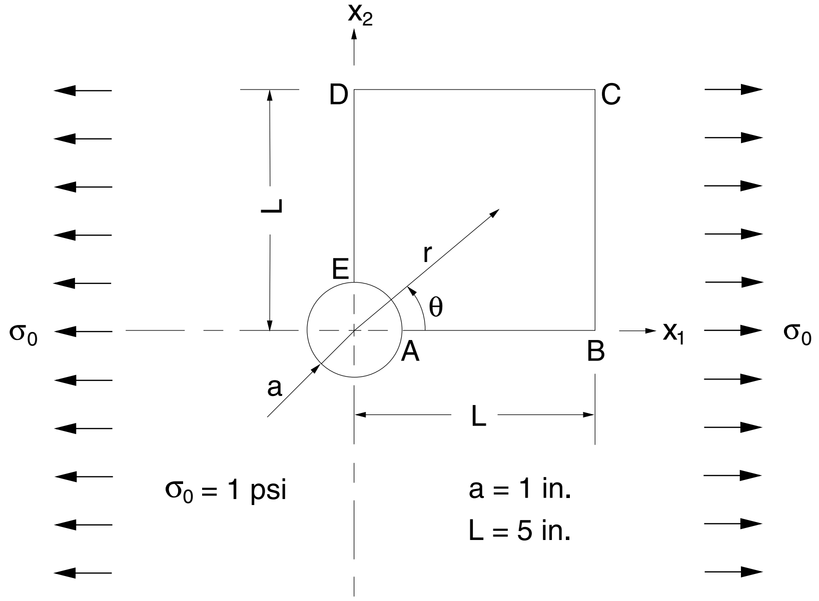


Figure 21. Plate with a circular hole under tension

where μ is the shear modulus and κ (Kolosov constant) is defined as

$$\kappa = \begin{cases} 3 - 4\nu & \text{(plane strain)}, \\ \dfrac{3-\nu}{1+\nu} & \text{(plane stress)}. \end{cases} \tag{43}$$

In the numerical computations, $a = 1$, $L = 5$, and plane strain conditions are assumed. The nodal discretizations used in the computations are shown in Figure 22. For the numerical integration of the weak form of the problem, three-point Hammer quadrature rule over the triangles was employed.

Sibson interpolation was chosen to approximate the trial (displacement) solution as well as to construct the test space. In Figure 24, the rates of convergence (R) in displacement and energy for NEM and constant strain finite elements are presented.

The theoretical convergence rates for the displacements and strains using finite elements (non-singular problems) are $R = 2$ and $R = 1$, respectively. It is observed from Figure 24 that the rates of convergence of NEM and FE are about the same, with NEM showing better absolute accuracy in displacements and strains. In Figure 24a, the stress concentration factor ($\sigma_{11}^{exact}/\sigma_0 = 3$) at point E is indicated within parentheses. It is seen that NEM is able to accurately capture the stress concentration at point E. In Figure 23, the numerical and exact normal stress σ_{11} are plotted along the edge ED (see Figure 21). The grid shown in Figure 22c is used, and 240 equi-spaced output points between $r = 1$ and $r = 5$ are chosen in the computations. Agreement between the NEM and the exact stress solution is excellent. The displacement along the edge is linear between two adjacent nodes, and hence one observes the jumps in the stress σ_{11} at the nodes.

3.4.2 Convergence study in 3-d elasticity

To test the performance of the α–NEM in 3-d, we consider the well-known benchmark problem of a hollow long cylinder under internal pressure. Problem dimensions and geometry are shown in Figure 25. The analytical solution of this problem can be found in most

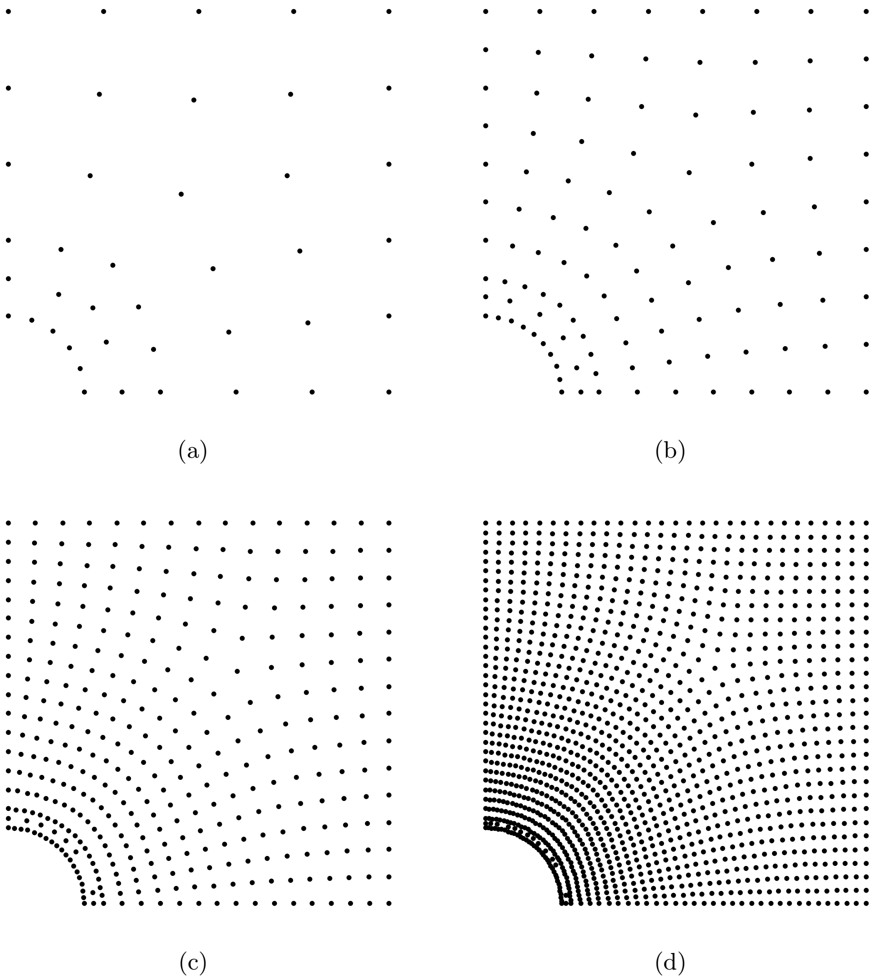


Figure 22. Nodal discretization for the plate with a hole problem. (a) 41 nodes, (b) 108 nodes, (c) 361 nodes, and (d) 1345 nodes

elasticity books, such as Timoshenko and Goodier [117]. The exact solution in cylindrical coordinates (ρ, θ, z) is:

$$\sigma_\rho = \frac{R_i^2 p}{R_e^2 - R_i^2}\Big(1 - \frac{R_e^2}{\rho^2}\Big) \tag{44a}$$

$$\sigma_\theta = \frac{R_i^2 p}{R_e^2 - R_i^2}\Big(1 + \frac{R_e^2}{\rho^2}\Big) \tag{44b}$$

$$\sigma_z = \nu(\sigma_\rho + \sigma_\theta) \tag{44c}$$

$$\varepsilon_\rho = \frac{1}{E}(\sigma_\rho - \nu\sigma_\theta - \nu\sigma_z) \tag{45a}$$

$$\varepsilon_\theta = \frac{1}{E}(\sigma_\theta - \nu\sigma_\rho - \nu\sigma_z) \tag{45b}$$

$$\varepsilon_z = 0 \tag{45c}$$

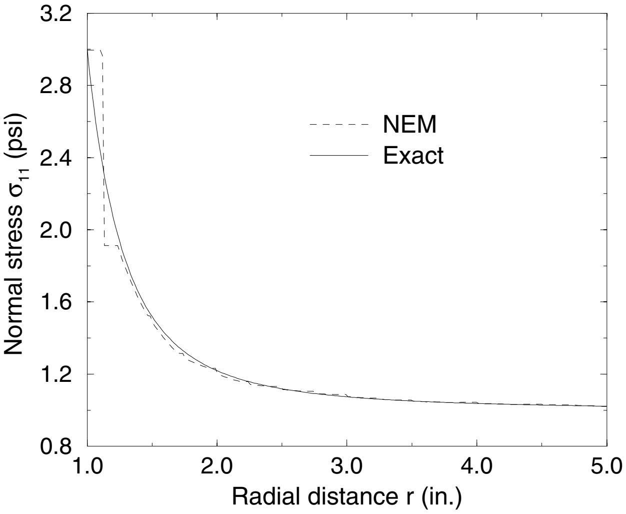
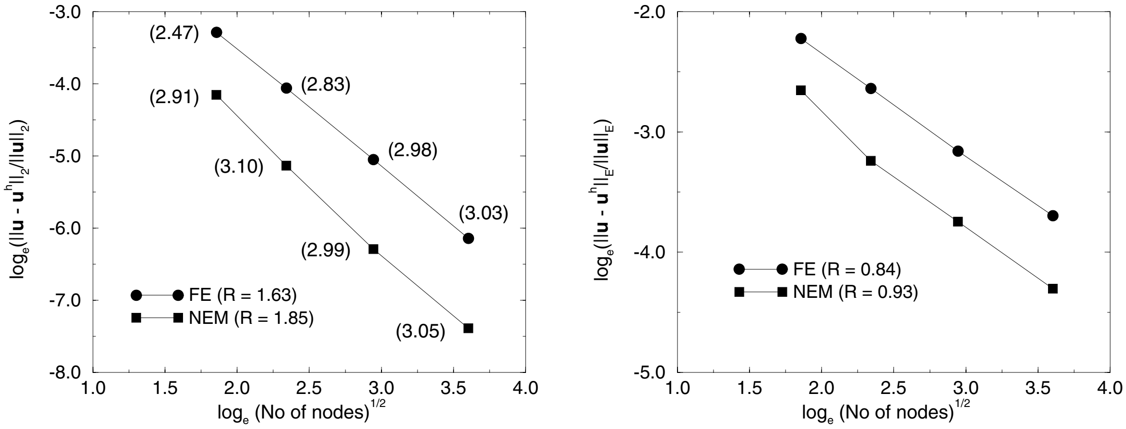


Figure 23. Comparison of σ_{11} for NEM and the exact solution along a radial line ($\theta = 90°$)



(a) (b)

Figure 24. Rate of convergence for the plate with a hole problem. (a) Displacement, and (b) Energy

$$
\begin{aligned}
u_\rho &= \frac{R_i^2 p \rho}{E(R_e^2 - R_i^2)}\Big[1 - \nu + \frac{R_e^2}{\rho^2}(1+\nu)\Big] \\
u_\theta &= 0 \\
u_z &= 0
\end{aligned}
\qquad (46)
$$

where R_i and R_e are the inner and the outer radii of the cylinder, respectively, and p is the applied pressure. The problem is in fact a two-dimensional one that reproduces a state

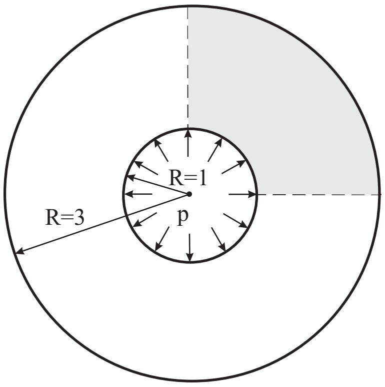


Figure 25. Model for the problem of a hollow cylinder under internal pressure

Method	Number of nodes	$\|e\|_{L_2}$	$\|e\|_E$
α-NEM	166	4.74×10^{-3}	1.90×10^{-1}
FEM	166	6.36×10^{-3}	3.41×10^{-1}
α-NEM	241	2.88×10^{-3}	1.86×10^{-1}
FEM	241	3.51×10^{-3}	3.10×10^{-1}
α-NEM	2076	1.12×10^{-3}	7.05×10^{-2}
FEM	2076	9.56×10^{-4}	1.37×10^{-1}

Table 1. Results for the hollow cylinder problem

of plane strain. In our case, both ends have been fixed in the axial direction to take into account the plane strain assumption. Due to the symmetry, only one quarter of the cylinder has been modeled, as shown in Figure 27. Three clouds of nodes have been used in this case, as shown in Figure 26, with 166, 241 and 1708 nodes, respectively. Material properties were $E = 1.0$ and $\nu = 0.25$.

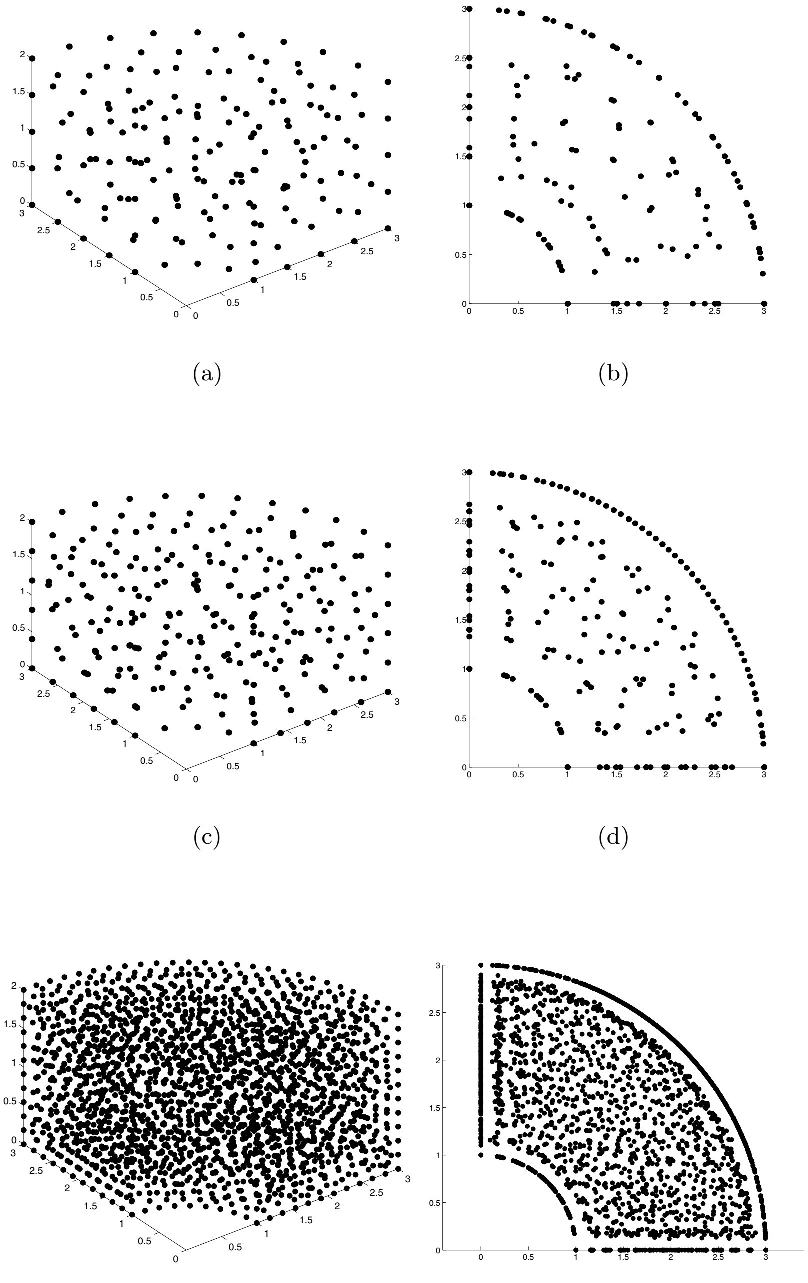
Again, Sibson interpolation in a Galerkin framework was used. Numerical integration of the weak form of the problem was performed by using four points quadrature rule in each tetrahedra. The same quadrature rule was employed in the computation of errors with respect to the analytical solution, which are presented in L_2 and energy norms. These results are included in Table 1.

The convergence results are shown in Figure 28. The three-dimensional results are equivalent to those obtained by Sukumar [106] in 2-d. The 3-d NEM characteristics are a generalization of the results obtained in 2-d [37].

4 ISSUES RELATED TO NUMERICAL INTEGRATION

Numerical integration is a crucial issue in the convergence of Galerkin-based meshless methods. In the NEM the weak form is usually integrated taking the Delaunay triangles as integration cells and performing a traditional Hammer quadrature. This is known to be a source of error [110, 37], since the support of the natural neighbour shape functions can not be exactly decomposed into triangles. In addition, it is known that Gauss quadrature as well as other numerical integration rules can exactly integrate polynomials, whereas natural neighbour shape functions are in general rational functions. These aspects are also a well-known source of error in other meshless methods, for instance EFGM [44].

Two approaches have been investigated in order to improve numerical integration in



(a)

(b)

(c)

(d)

(e)

(f)

Figure 26. Clouds of points for the problem of the cylinder under pressure

the NEM. First, decomposition of integration cells into pieces that exactly describe the support of the shape functions (Section 4.1) and secondly, a stabilized nodal quadrature scheme (Section 4.2).

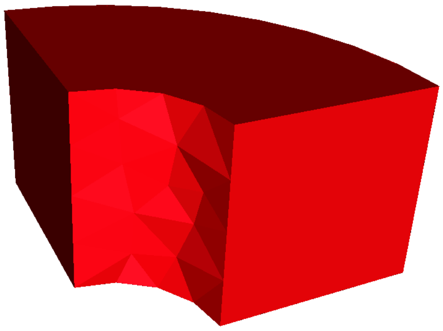
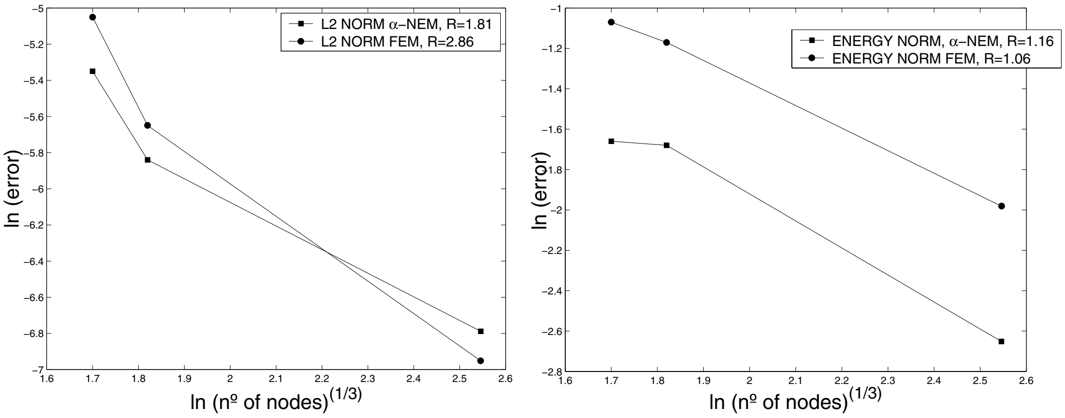


Figure 27. α-shape (166 nodes) used in the simulation of the hollow cylinder under pressure



(a) (b)

Figure 28. Convergence study for the hollow cylinder problem

4.1 Decomposition of the Support of Shape Functions

An alternative approach to the use of background integration cells is to perform an integration over the intersection of supports of each shape function. This approach has also been studied by Atluri and co-workers in [7, 8] for meshless methods based on MLS approximants.

In the NEM, the support of the shape functions is usually formed by the union of a number of circles (the circumcircles of those triangles that neighbour the node considered), as shown in Figure 29. As can be seen, both the nodal support and the intersection of two supports can be decomposed into a set of triangles and circle segments. In the triangles, both Gauss and Hammer quadratures have been tested, whereas in the circle segments a one-to-one transformation between the unit square and the segments was established.

In the approach presented here, numerical integration was performed over the intersec-

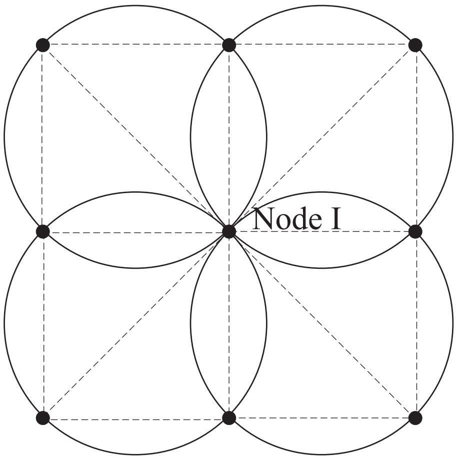


Figure 29. Support of the shape function associated with node I

tions of supports of the shape functions, i.e., in the form

$$\sum_I \sum_J \int\limits_{\Omega_{IJ}} \nabla^s \delta \boldsymbol{v} : \mathbf{C} : \nabla^s \boldsymbol{u} d\Omega = \sum_I \sum_J \int\limits_{\Omega_{IJ}} \delta \boldsymbol{v} \cdot \boldsymbol{b} d\Omega + \int\limits_{\Gamma_t} \delta \boldsymbol{v} \cdot \bar{\boldsymbol{t}} d\Gamma \ \ \forall \delta \boldsymbol{v} \in \mathcal{V} \tag{47}$$

where Ω_{IJ} is a subset of the domain Ω: $\Omega_{IJ} = \omega_I \cap \omega_J \cap \Omega$ with $\omega_I = supp(n_I)$.

With this decomposition, numerical integration of the weak form of the problem is performed as the sum of integrals over triangles and circle segments. Numerical integration in the triangles was performed using Hammer quadrature, whereas for the integration over circle segments a transformation was constructed between the unit square and each specific segment, in the form (Figure 30):

$$x = \frac{1}{2}\Big[\frac{1}{2}(1-\eta)(x_1(1-\xi) + \frac{1}{2}x_2(1+\xi) + (1+\eta)(C_x + R\cos(\frac{1}{2}(1-\xi)\beta + (1+\xi)\alpha))\Big], \tag{48}$$

$$y = \frac{1}{2}\Big[\frac{1}{2}(1-\eta)(y_1(1-\xi) + \frac{1}{2}y_2(1+\xi) + (1+\eta)(C_y + R\sin(\frac{1}{2}(1-\xi)\beta + (1+\xi)\alpha))\Big]. \tag{49}$$

On using the above transformation, Gauss quadrature points were mapped onto an arc of a circle (see Figure 31 for the result of the transformation of a 5 × 5 Gauss quadrature over a square into a circle segment). Application of this quadrature scheme to the patch test is presented in Section 8.

4.2 Stabilized Nodal Quadrature Scheme

Recently, Chen and co-workers [28] have presented a stabilized nodal quadrature scheme with application to moving least squares methods. They adopted a strain smoothing procedure to define the nodal strain operator:

$$\tilde{\varepsilon}^h_{ij}(\boldsymbol{x}_I) = \int_\Omega \varepsilon_{ij}(\boldsymbol{x}) \Phi(\boldsymbol{x}; \boldsymbol{x} - \boldsymbol{x}_I) d\Omega \tag{50}$$

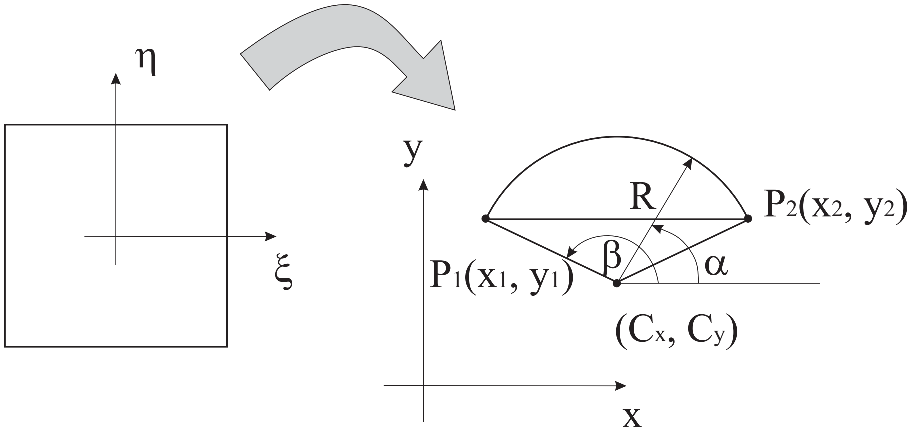


Figure 30. Transformation proposed for the integration on a arc of a circle

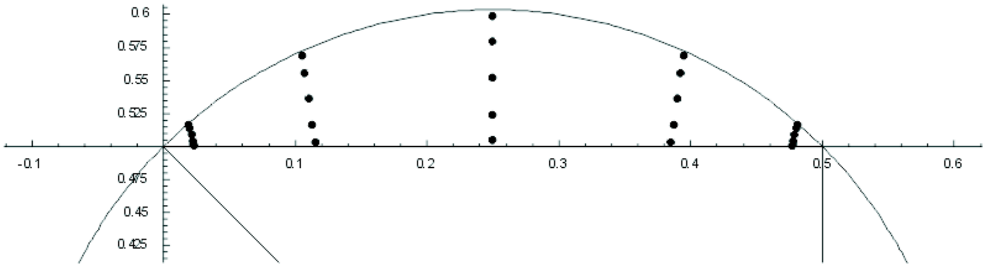


Figure 31. Result of the transformation of 5 × 5 Gauss quadrature over a square into a circle segment

where ε_{ij} is the strain obtained by compatibility assumptions and Φ is a distribution function. In [28], this function was chosen as

$$\Phi(\boldsymbol{x}; \boldsymbol{x} - \boldsymbol{x}_I) = \begin{cases} \frac{1}{A_I} \text{ if } \boldsymbol{x} \in \Omega_I \\ 0 \text{ otherwise} \end{cases} \tag{51}$$

where Ω_I is the Voronoi cell associated with node I (other definitions for the area associated with each node are analysed in [29]) and A_I the corresponding area of this cell. With this definition, the strain smoothing leads to

$$\tilde{\varepsilon}_{ij}^h(\boldsymbol{x}_I) = \frac{1}{2A_I} \int_\Omega \Big(\frac{\partial u_i^h}{\partial x_j} + \frac{\partial u_j^h}{\partial x_i} \Big) d\Omega \tag{52}$$

and on applying the divergence theorem, we obtain

$$\tilde{\varepsilon}_{ij}^h = \frac{1}{2A_I} \int_{\Gamma_I} (u_i^h n_j + u_j^h n_i) d\Gamma \tag{53}$$

where Γ_I is the boundary of the Voronoi cell associated to node I.

This nodal quadrature scheme has rendered excellent results when applied to the RKPM [28, 29], and it is especially well-suited for its application to the NEM, since most of the geometrical entities appearing in its computation (Voronoi cell, circumcenter, etc.) are part of the natural neighbour shape function computations. Hence the additional cost in the implementation of the quadrature scheme is minimal. For a recent and detailed application of the nodal quadrature scheme to the NEM in 2-d, see Yoo *et al.* [124]. A comparison between these two quadrature schemes has been carried out, and the numerical results are presented in the ensuing section.

4.3 Performance of the Numerical Integration

In this section we compare the two approaches presented in Section 4 for the integration of the weak form of the boundary-value problem. The patch test [65], although originally conceived for proving the convergence of non-conforming finite elements, has been widely used in meshless methods to study the accuracy of the numerical integration [44, 110]. In the patch test, a known linear displacement field is imposed on the boundary of a patch of elements. The test is passed if the prescribed field is reproduced (within *machine precision*) in the patch. In the vast majority of meshless methods linear completeness is met by construction, and hence the patch test is expected to be exactly satisfied. Thus, if a method with a linearly complete approximation does not pass the patch test one can readily infer that the failure was due to inaccurate numerical integration.

4.3.1 Two-dimensional patch tests with the support decomposition method

In previous works related to the NEM [110, 37], the patch test was not satisfied to machine precision in both two- and three-dimensions. In this section, the proposed support decomposition method is analysed and a few numerical results are presented. Consider the cloud of nine nodes depicted in Figure 32. The numerical results for the patch test (error in the L_2 error norm) are presented in Table 2.

It can be seen that, even with very high number of quadrature points, the numerical results are not to machine precision (10^{-16}). This behaviour is similar to that observed in the method of finite spheres [41] or in the MLPG method [7], where a large increase in the number of quadrature points is necessary to achieve high levels of accuracy. In fact, beyond a point the savings are minimal and the increase in quadrature points in not practical from a computational viewpoint. We can conclude that the error due to the non-polynomial character of the shape function has a significant influence with little improvement obtained with high quadratures. This is the reason that led us to consider the use of the stabilized conforming nodal integration method [28].

Gauss points (triangles)	Gauss points (arc)	$\|e\|_{L_2}$
2x2	2x2	$2,1527 \cdot 10^{-2}$
4x4	4x4	$2,2793 \cdot 10^{-3}$
10x10	10x10	$4,1850 \cdot 10^{-4}$
20x20	20x20	$1,3944 \cdot 10^{-5}$
50x50	50x50	$6,1493 \cdot 10^{-6}$
70x70	70x70	$2,2505 \cdot 10^{-6}$
100x100	100x100	$5,6665 \cdot 10^{-7}$
150x150	150x150	$4,2297 \cdot 10^{-7}$

Table 2. Error in the L_2 norm for the patch test using the support decomposition method

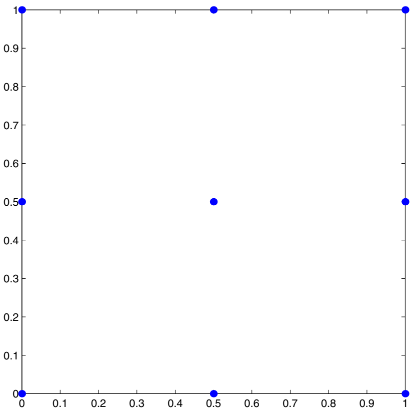


Figure 32. Nine node cloud for the application of the patch test with the support decomposition method

4.3.2 Two-dimensional patch tests with stabilized nodal integration

The technique presented in Section 4.2 was applied to patch tests over several clouds of nodes on a unit square. The nodal discretizations are depicted in Figure 33. We have considered regular as well as irregular (biased towards a corner or a side of the square) nodal distributions.

Grid	Imposed field	$\|e\|_{L_2}$
a	Bilinear **u**	$2,7208 \cdot 10^{-16}$
a	Linear **u**	$2.4343 \cdot 10^{-16}$
b	Bilinear **u**	$4.2407 \cdot 10^{-16}$
c	Bilinear **u**	$2.9842 \cdot 10^{-16}$
a	$\sigma_y^0 = 1$	$7.8614 \cdot 10^{-16}$
d	$\sigma_y^0 = 1$	$9.2009 \cdot 10^{-16}$

Table 3. Error in the L_2 norm for the patch test using the stabilized nodal integration scheme for the grids in Figure 33

In all cases, one point quadrature was employed in the integration along each segment defining the Voronoi cell of interior nodes. For nodes lying on the boundary of the patch or nodes whose cell intersected the boundary, two-points Gauss quadrature was employed.

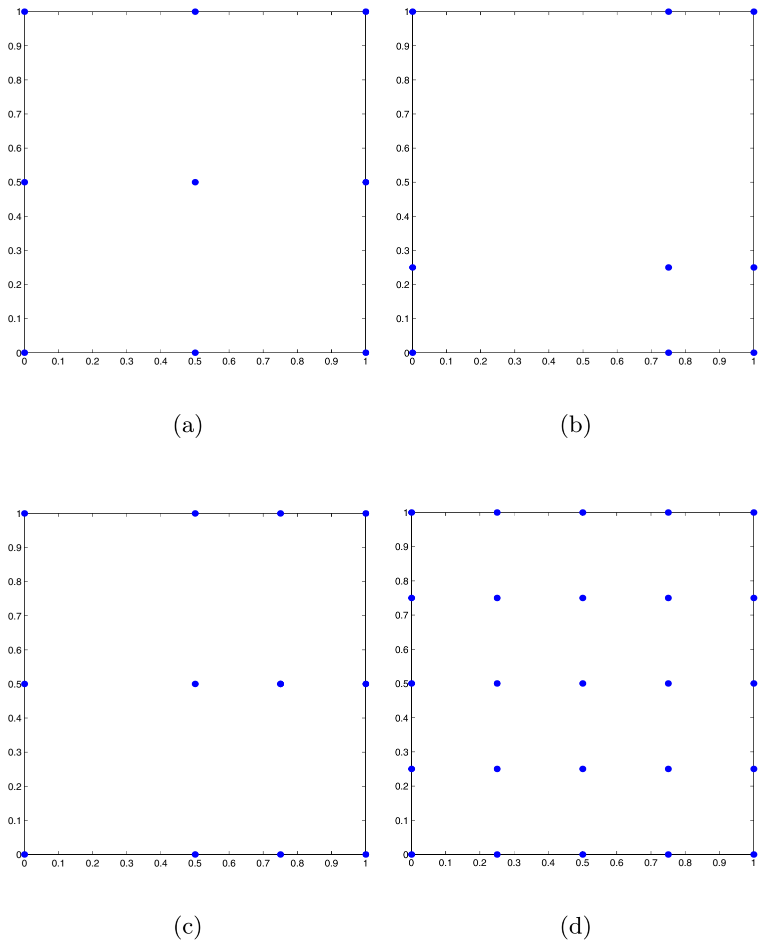
Numerical results for the L_2 error norm (defined in Eq. (54)) are presented in Table 3.

$$\|\boldsymbol{u} - \boldsymbol{u}^h\|_{L_2(\Omega)} = \Big(\int_\Omega (\boldsymbol{u} - \boldsymbol{u}^h) \cdot (\boldsymbol{u} - \boldsymbol{u}^h) d\Omega \Big)^{1/2} \tag{54}$$

Two classes of patch tests are considered, namely displacement and equilibrium patch tests. In the latter case, a constant stress field $\sigma_y^0 = 1$ is imposed, so that the theoretical displacement field is:

$$u_1 = \frac{\nu}{E}(1 - x_1) \tag{55}$$

$$u_2 = \frac{x_2}{E} \tag{56}$$



(a)

(b)

(c)

(d)

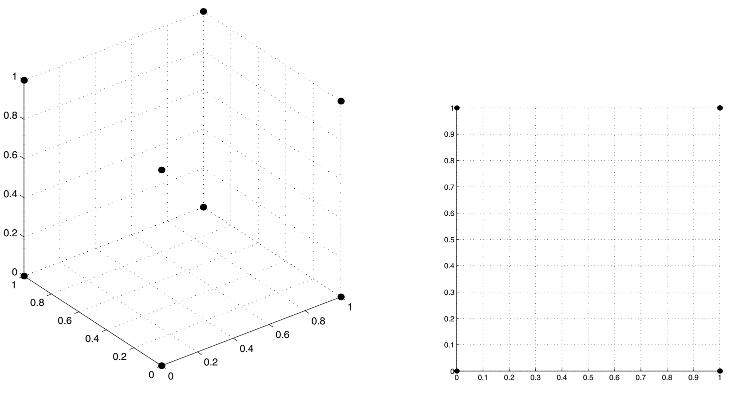
Figure 33. Nodal discretizations used for the patch test with the stabilized nodal integration scheme

It can be observed how these results clearly improve those obtained by the support decomposition method. They are very close to machine precision, both in displacement and equilibrium patch tests; similar results are reported in [124], where the Laplace interpolant is used. Integration over Delaunay triangles (see [110]) also leads to poor results, bounded by 10^{-5} for nodal discretization similar to those considered in this work. Thus it is clear that the use of conforming stabilized nodal quadrature is an appealing approach for the numerical integration in the NEM.

4.3.3 Three-dimensional patch tests

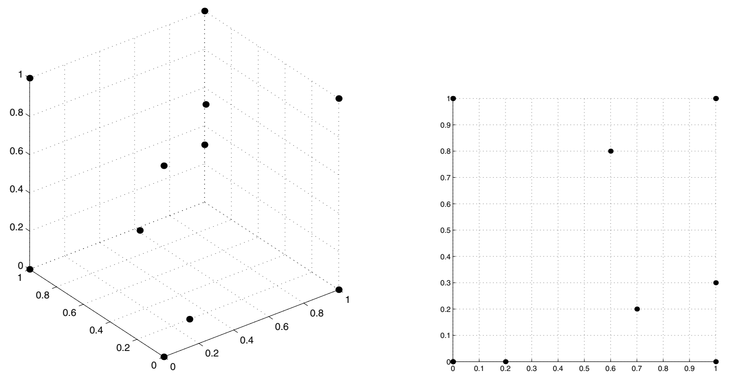
In this section the application of the stabilized conforming nodal integration scheme to the patch test in 3-d is presented. In [37] it was demonstrated that the 3-d patch test was not passed within machine precision if integration over tetrahedra was used; this is analogous to the results obtained in 2-d [110]. For the application of the patch test in 3-d, several nodal grids (unit cube domain) are considered (Figures 34–36). A linear displacement field was imposed on the boundary.

Integration is performed over Voronoi cells facets using either one point quadrature or



(a) (b)

Figure 34. Cloud of points used in the application of the *patch test.* 8 points cloud



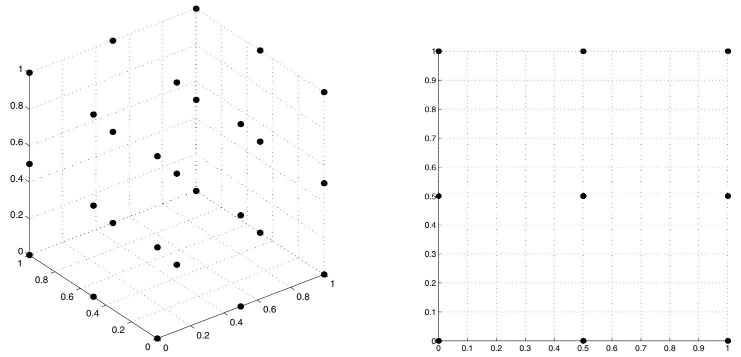
(a) (b)

Figure 35. Cloud of points used in the application of the *patch test.* Irregularly distributed cloud of 9 points

by dividing the facets which are always convex. For instance, see Figure 37a where the Voronoi cells for a cloud of nine nodes is depicted. The facets that intersect the boundary are divided into triangles and three-point quadrature is used in each triangle. In interior facets which do not intersect the boundary, one-point quadrature suffices for good accuracy.

In Table 4, the numerical results are compared to those obtained in [37]. The result for the eight node cloud with four-points quadrature over the tetrahedra is a special case of a symmetric tetrahedralization in which errors are balanced [110, 37]. This special case disappears when non-symmetric quadrature cell distributions are considered.

It can be seen how the use of stabilised nodal integration schemes clearly improves the results obtained. In addition, it must be noted that most of the data necessary to compute the geometric entities involved in the quadrature scheme (i.e., Voronoi cells and their segments) are previously computed through the Lasserre's algorithm, thus obtaining great computational savings.



(a) (b)

Figure 36. Cloud of points used in the application of the *patch test*. 27 points cloud

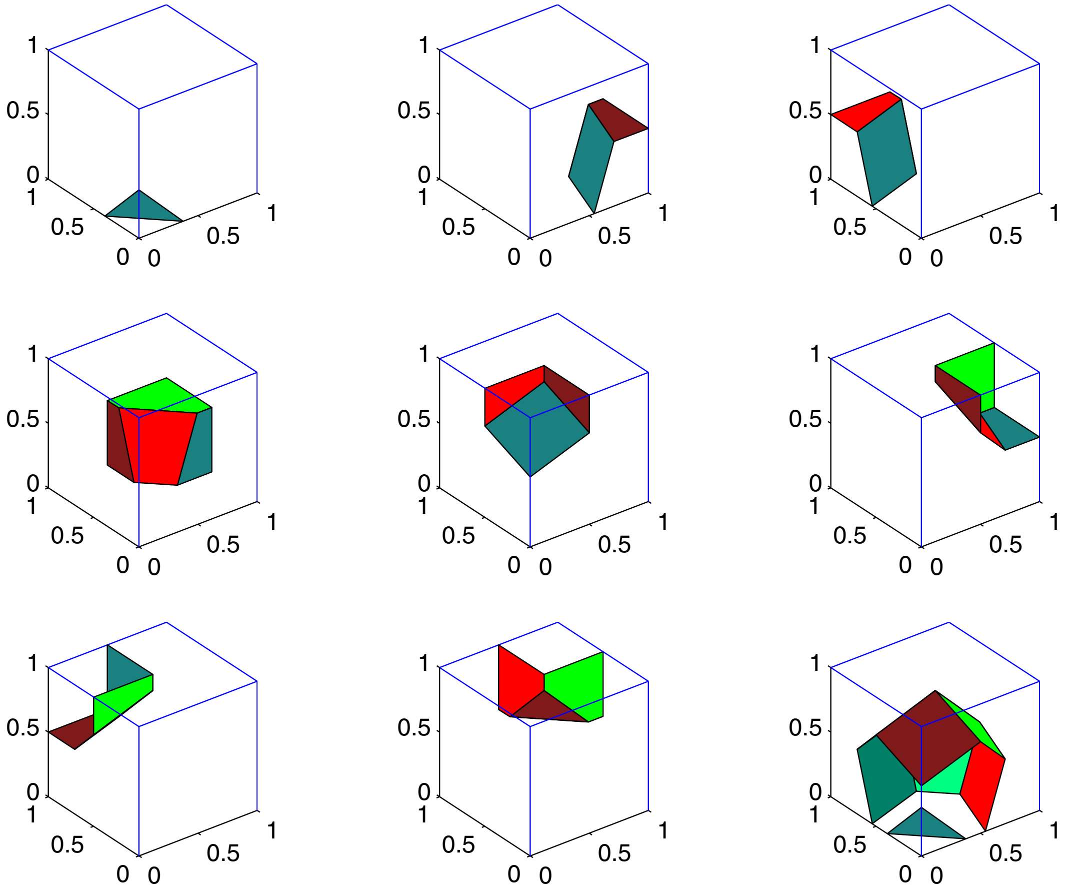


Figure 37. Voronoi cells for the nine point cloud. For simplicity, Voronoi facets lying on the boundary of the cube are not depicted.

Cloud	Number of nodes	Distribution	Integration over tetrahedra	Stabil. nodal quadr.
Fig. 34	8	Regular	$1.7844E-16$	$4.5678E-16$
Fig. 35	9	Irregular	$1.0877E-2$	$2.1298E-16$
Fig. 36	27	Regular	$2.4660E-3$	$1.5796E-16$

Table 4. L_2 error norms in the application of the three-dimensional patch test

5 MIXED INTERPOLATION IN INCOMPRESSIBLE AND NEARLY-INCOMPRESSIBLE ELASTICITY

We review the use of natural neighbour interpolation in a mixed formulation fo applications in incompressible elasticity. The mixed approximation that is presented has been used in [83] and [82] for fluid mechanics problems. An example in 3-d elasticity is presented in Section 5.4.

5.1 Governing Equations and Mixed Formulation

5.1.1 Strong form

Consider an open bounded domain $\Omega \subset \mathbb{R}^2$ with boundary Γ. The strong form for the two-dimensional small displacement elastostatic boundary value problem can be written as [63]:

$$\text{Given } \mathbf{b} : \Omega \to \mathbb{R}^2,\ \bar{\mathbf{u}} : \Gamma_u \to \mathbb{R}^2,\ \text{and } \bar{\mathbf{t}} : \Gamma_t \to \mathbb{R}^2$$

$$\text{Find } \mathbf{u} : \bar{\Omega} \to \mathbb{R}^2,\ p : \bar{\Omega} \to \mathbb{R} \text{ such that}$$

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} + \mathbf{b} = 0 \quad \text{in } \Omega, \tag{57a}$$

$$\boldsymbol{\nabla} \cdot \mathbf{u} + \frac{p}{\lambda} = 0 \quad \text{in } \Omega, \tag{57b}$$

$$\mathbf{u} = \bar{\mathbf{u}} \quad \text{on } \Gamma_u, \tag{57c}$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \bar{\mathbf{t}} \quad \text{on } \Gamma_t, \tag{57d}$$

where the Cauchy stress tensor $\boldsymbol{\sigma}$ is related to the small strain tensor $\boldsymbol{\varepsilon}$ and the hydrostatic pressure parameter p for an isotropic linear elastic material through the constitutive relation:

$$\sigma_{ij} = -p\delta_{ij} + \bar{C}_{ijkl}\varepsilon_{kl}, \tag{57e}$$

$$\bar{C}_{ijkl} = \mu(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}). \tag{57f}$$

with μ the shear modulus.

5.1.2 Weak formulation

We first define the trial and test spaces for the displacements and the pressure. Following [63], we let

$$\mathbf{u} \in \mathcal{S} = \{\mathbf{u} \in \left(H^1(\Omega)\right)^2 \mid \mathbf{u} = \overline{\mathbf{u}} \ \text{ on } \Gamma_u\} \quad \text{(displacement trial solution space)}, \tag{58a}$$

$$\mathbf{v} \in \mathcal{V} = \{\mathbf{v} \in \left(H^1(\Omega)\right)^2 \mid \mathbf{v} = \mathbf{0} \ \text{ on } \Gamma_u\} \quad \text{(displacement test function space)}, \tag{58b}$$

$$p,\, q \in \mathcal{P} = L^2(\Omega) \quad \text{(space of pressures—trial and test functions)}, \tag{58c}$$

where $H^1(\Omega)$ is the Sobolev space of functions with square-integrable first weak derivatives in Ω. The weak form of Eq. (57) in Hermann variational form is posed as [58, 63]:

$$\begin{aligned}
&\text{Find } (\mathbf{u}, p) \in \mathcal{S} \times \mathcal{P} \text{ such that} \\
&\bar{a}(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) = \langle \mathbf{b}, \mathbf{v} \rangle + \langle \bar{\mathbf{t}}, \mathbf{v} \rangle \quad \forall \mathbf{v} \in \mathcal{V}, && \text{(59a)} \\
&b(\mathbf{u}, q) - \frac{1}{\lambda}(p, q) = 0 \quad \forall q \in \mathcal{P}, && \text{(59b)}
\end{aligned}$$

where

$$\bar{a}(\mathbf{u}, \mathbf{v}) = 2\mu \int_\Omega \varepsilon_{ij}(\mathbf{u}) \varepsilon_{ij}(\mathbf{v}) \, d\Omega, \tag{59c}$$

$$b(\mathbf{v}, p) = -\int_\Omega p \boldsymbol{\nabla} \cdot \mathbf{v} \, d\Omega. \tag{59d}$$

In the numerical implementation, finite-dimensional subspaces are used as the trial and test spaces for the displacements and pressure. Let $\mathbf{V}^h \subset \mathbf{V}$ and $\mathbf{V}_0^h \subset \mathbf{V}_0$ be the trial and test spaces for the displacements, and $P^h \subset P$ be the trial and test space for the pressure. The weak form can be written as

$$\begin{aligned}
&\text{Find } (\mathbf{u}^h, p^h) \in \mathbf{V}^h \times P^h \text{ such that} \\
&\bar{a}(\mathbf{u}^h, \mathbf{v}^h) + b(\mathbf{v}^h, p^h) = \langle \mathbf{b}, \mathbf{v}^h \rangle + \langle \bar{\mathbf{t}}, \mathbf{v}^h \rangle \quad \forall \mathbf{v}^h \in \mathbf{V}_0^h, && \text{(60a)} \\
&b(\mathbf{u}^h, q^h) - \frac{1}{\lambda}(p^h, q^h) = 0 \quad \forall q^h \in P^h. && \text{(60b)}
\end{aligned}$$

5.2 Discrete System

Consider the numerical implementation for the mixed natural element method. In a Galerkin procedure, the displacement trial and test functions are interpolated using the same set of shape functions and likewise for the pressure trial and test functions. The trial and test functions are:

$$\mathbf{u}^h(\mathbf{x}) = \sum_{I=1}^n \phi_I(\mathbf{x}) \mathbf{u}_I, \quad \mathbf{v}^h(\mathbf{x}) = \sum_{I=1}^n \phi_I(\mathbf{x}) \mathbf{v}_I, \tag{61a}$$

$$p^h(\mathbf{x}) = \sum_{I=1}^n \psi_I(\mathbf{x}) \bar{p}_I, \quad q^h(\mathbf{x}) = \sum_{I=1}^n \psi_I(\mathbf{x}) \bar{q}_I. \tag{61b}$$

The strain-displacement relation for the displacement trial function can be written as

$$\varepsilon^h(\mathbf{x}) = \sum_{I=1}^n \mathbf{B}_I \mathbf{u}_I, \tag{62}$$

where

$$\mathbf{B}_I = \begin{bmatrix} \phi_{I,1}(\mathbf{x}) & 0 \\ 0 & \phi_{I,2}(\mathbf{x}) \\ \phi_{I,2}(\mathbf{x}) & \phi_{I,1}(\mathbf{x}) \end{bmatrix}. \tag{63}$$

The divergence of the displacement trial solution is given by

$$\nabla \cdot \mathbf{u}^h(\mathbf{x}) = \sum_{I=1}^{n} \tilde{\mathbf{B}}_I \mathbf{u}_I, \tag{64}$$

where

$$\tilde{\mathbf{B}}_I = \Big[\phi_{I,1}(\mathbf{x}) \quad \phi_{I,2}(\mathbf{x})\Big]. \tag{65}$$

On substituting the displacement and pressure trial and test functions in Eq. (60) and using the arbitrariness of displacement and pressure nodal variations, the following discrete system of linear equations is obtained [63]:

$$\begin{bmatrix} \bar{\mathbf{K}} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{M} \end{bmatrix} \begin{Bmatrix} \mathbf{d} \\ \mathbf{p} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f} \\ \mathbf{0} \end{Bmatrix}, \tag{66}$$

where

$$\bar{\mathbf{K}}_{IJ} = \int_{\Omega^h} \mathbf{B}_I^T \bar{\mathbf{C}} \mathbf{B}_J \, d\Omega, \tag{67a}$$

$$\mathbf{G}_{IJ} = -\int_{\Omega^h} \tilde{\mathbf{B}}_I^T \psi_J \, d\Omega, \tag{67b}$$

$$\mathbf{M}_{IJ} = -\frac{1}{\lambda} \int_{\Omega^h} \psi_I \psi_J \, d\Omega, \tag{67c}$$

$$\mathbf{f}_I = \int_{\Omega^h} \phi_I \mathbf{b} \, d\Omega + \int_{\Gamma_t^h} \phi_I \bar{\mathbf{t}} \, d\Gamma. \tag{67d}$$

In the above equations, $\mathbf{d}$ is the vector of nodal displacements and $\mathbf{p}$ is the vector of nodal pressures. The matrix $\bar{\mathbf{K}}$ is symmetric and positive definite, and the matrix $\mathbf{M}$ is symmetric and semi-positive definite ($\mathbf{M} = \mathbf{0}$ when $\nu = 0.5$). The matrix $\mathbf{G}$ corresponds to the discrete gradient operator, and $\mathbf{G}^T$ to its transpose.

5.3 Displacement and Pressure Interpolation

We consider the interpolation schemes for the displacements and the pressure in the mixed natural element method. The displacement vectors $\mathbf{u}^h(\mathbf{x}) : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^2$ are interpolated using the Sibson shape functions, and can be written in the form:

$$\mathbf{u}^h(\mathbf{x}) = \sum_{I=1}^{n} \phi_I(\mathbf{x}) \mathbf{u}_I, \tag{68}$$

where $\mathbf{u}_I$ ($I = 1, 2, \ldots, n$) are the vectors of nodal displacements at the n natural neighbours, and $\phi_I(\mathbf{x})$ are the shape functions associated with each node.

The pressure field is interpolated by an interpolation scheme akin to Eq. (68):

$$p^h(\mathbf{x}) = \sum_{I=1}^{n} \psi_I(\mathbf{x}) \bar{p}_I, \tag{69}$$

where $\psi_I(\mathbf{x})$ is the shape function associated with node I and $\bar{p}_I$ are the nodal pressures. We consider $\psi_I(\mathbf{x})$ to be of the general form

$$\psi_I(\mathbf{x}) = \frac{\big(\phi_I(\mathbf{x})\big)^k}{\sum_{J=1}^{n} \big(\phi_J(\mathbf{x})\big)^k}, \tag{70}$$

where k is a non-negative integer and n is the number of natural neighbours for the point $\mathbf{x}$. The case $k = 1$ reduces to natural neighbour interpolation, and all other k render shape functions that form a partition of unity, with the resulting interpolant satisfying only constant completeness. For $k > 1$, numerical studies show severe Poisson locking results; the cases $k = 0$ and $k = 1$ alleviated locking in the near incompressible limit and hence merit investigation [106]. These two cases are considered for the interpolation of the pressure field, and Eq. (69) takes the specific forms:

$$p^h(\mathbf{x}) = \sum_{I=1}^{n} \frac{1}{n} \bar{p}_I, \tag{71a}$$

$$p^h(\mathbf{x}) = \sum_{I=1}^{n} \phi_I(\mathbf{x}) \bar{p}_I. \tag{71b}$$

Since the shape functions $\phi_I(\mathbf{x})$ have compact support [50] and are $\mathcal{C}^0$, the shape functions $\psi_I(\mathbf{x}) = 1/n$ are $\mathcal{C}^{-1}$. An alternate $\mathcal{C}^{-1}$ pressure interpolant can be constructed using the well-known concept of nearest neighbours in computational geometry [96]. This leads to the interpolation scheme known as nearest neighbour interpolation. In this approach, if $\mathbf{x} \in T_I$, where T_I is the Voronoi polygon of node I, then we assign the nodal pressure $\bar{p}_I$ to $p(\mathbf{x})$. In essence, the pressure field is assumed to be a constant over the Voronoi polygon T_I. In the context of natural neighbour interpolation, if $\phi_I = \|\boldsymbol{\phi}\|_\infty$, then $\psi_I(\mathbf{x}) = 1$ and $\psi_J(\mathbf{x}) = 0 \ \forall J \neq I$, and therefore

$$p(\mathbf{x}) = \bar{p}_I \quad \text{if } \mathbf{x} \in T_I. \tag{72}$$

5.4 Example in Nearly Incompressible Elasticity

In this section the behaviour of the proposed mixed approximation, described in Section 5, is analysed through the well known example of a three dimensional beam under bending [117]. The dimensions of the beam are 1.0 × 0.5 × 4.0, and the exact (theoretical) displacements are imposed on the 1.0 × 0.5 rectangular side of the bar. On the opposite side of the bar, a parabolically distributed load of unit magnitude was applied. The material properties are Young's modulus $E = 1$ and the Poisson's ratio ν ranges from 0.3 to 0.4999999. Numerical simulations are conducted using the Sibson interpolation with Gauss quadrature in the tetrahedra. The approximation of the pressure field is constructed by means of standard Sibson approximation of pressure (Eq. 71a) in the case of $\mathcal{C}^0 - \mathcal{C}^0$ approximation or by means of Eq. 71b in the case of $\mathcal{C}^0 - \mathcal{C}^{-1}$ approximation. The end displacement for these simulations is compared to the theoretical solution [117], and also to the FEM. The FE simulations are performed using standard trilinear hexahedral (TLH) finite elements and mixed trilinear displacement-trilinear pressure and trilinear displacement-constant pressure hexahedra, which are integrated using four Gauss points in each element. The normalized end displacement (indicated as a %) is listed in Table 5 for a grid composed of 255 nodes (Figure 38).

As is well-known, severe locking is observed for the standard FEM, and the results clearly improve if a mixed interpolation is used. However, it can be seen that the natural neighbour approximations render slightly better results than the finite elements, and better results are obtained using the $\mathcal{C}^0 - \mathcal{C}^0$ approximation.

The behavior of the mixed NEM is very similar to the mixed $\mathcal{C}^0 - \mathcal{C}^{-1}$ displacement–pressure quadrilateral finite element. The performance of the NEM in near incompressibility is critical for incompressible fluid mechanics problems which is treated in the next section.

	FEM			NEM		
ν	TLH	$\mathcal{C}^0 - \mathcal{C}^0$	$\mathcal{C}^0 - \mathcal{C}^{-1}$	Standard.	$\mathcal{C}^0 - \mathcal{C}^0$	$\mathcal{C}^0 - \mathcal{C}^{-1}$
0.3	98.53	98.81	100.63	99.86	100.12	100.89
0.4	95.45	96.78	97.63	97.29	98.38	99.77
0.4999	0.85	92.07	98.33	9.70	94.30	97.39
0.4999999	-6.85	92.08	98.36	3.26	94.30	100.10

Table 5. Normalized end displacement (expressed as a %) for the 3-d beam bending problem

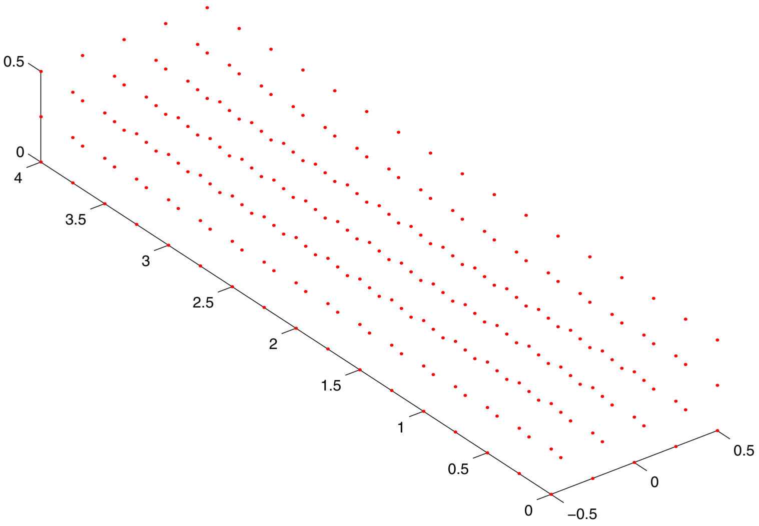


Figure 38. Nodal discretisation for the 3-d beam bending problem

6 UPDATED LAGRANGIAN APPROACH TO FREE SURFACE FLUID MECHANICS PROBLEMS

In general, meshless methods are well-suited for problems with large deformations, where traditional finite element techniques sometimes fail. This is particularly so in 3-d where excessive remeshing is both demanding as well as time-consuming. In this section, we present a few simulations in fluid mechanics. The first application of the natural element method to fluid mechanics problems was conducted by Braun and Sambridge [24]. Further extension to Newtonian as well as non-Newtonian fluid mechanics was carried out in [82], whereas in [83] a comparison between the updated Lagrangian and Eulerian approaches was performed.

6.1 Problem Statement

The model problem in an updated Lagrangian framework is stated as follows:

1. Equilibrium equations (balance of linear momentum in the absence of inertial and body forces):

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \mathbf{0}. \tag{73}$$

2. Incompressibility of the fluid:

$$\boldsymbol{\nabla} \cdot \mathbf{v} = 0. \tag{74}$$

3. Newtonian fluid constitutive equation:

$$\boldsymbol{\sigma} = -p\mathbf{I} + 2\mu\mathbf{D}, \tag{75}$$

where μ is the fluid viscosity, $\mathbf{I}$ the second-order unit tensor, and $\mathbf{D}$ the strain rate tensor.

The weak form associated with the strong form is posed as: find the kinematically admissible velocity field $\mathbf{v} \in \mathcal{U}$ such that

$$\int_{\Omega(t)} \boldsymbol{\sigma} : \mathbf{D}^* d\Omega = \int_{\Gamma_t} \bar{\mathbf{t}} \cdot \mathbf{v}^* d\Gamma \ \ \forall \mathbf{v}^* \in \mathcal{V} \tag{76}$$

$$\int_{\Omega(t)} (-\nabla \mathbf{v} + \varepsilon p) p^* d\Omega = 0 \ \ \forall p^* \in L_2(\Omega(t)), \tag{77}$$

where $\mathcal{U} = \{\mathbf{v} | \mathbf{v} \in H^1(\Omega(t)),\ \mathbf{v}|_{\Gamma_v} = \bar{\mathbf{v}}\}$, $\mathcal{V} = \{\mathbf{v}^* | \mathbf{v}^* \in H^1(\Omega(t)),\ \mathbf{v}|_{\Gamma_v} = \mathbf{0}\}$, and Γ_v is the portion of the boundary of the domain with prescribed velocities. As usual, H^1 and L_2 are the Sobolev and Lebesgue functional spaces, respectively. A penalty parameter ε is introduced to enforce the fluid incompressibility. As in incompressible elasticity, a mixed $\mathcal{C}^0 - \mathcal{C}^{-1}$ interpolation for the velocities and the pressure is chosen.

6.2 Tracking the Fluid Front

An in-depth study on the determination of the shape of the domain at each time step was conducted in [83]. For free-surface flows, such as mould filling simulations, tracking an explicit description of the fluid front is very time-consuming, since care must be taken to determine the position and time when the fluid front disappears after contact of two initially separated portions of the fluid. In [83], an α-shape approach was preferred. At each time step an appropriate α-shape of the domain was used and Eqs. (76) and (77) were solved. Thus, the choice of an appropriate α-value depends not only on the nodal spacing h and its relationship with the desired level of detail of the geometry to be reproduced (see Section 3.2.1), but also on the time increment chosen to solve the problem.

Consider the case of fluid flow between two parallel plates with a cylindrical obstacle (Figure 39). A fixed number of nodes, with prescribed velocity $v_x(x = 0, y) = \frac{4V}{h^2}(h - y)y$ ($V = 0.1$) is added to the problem at each time step to force the flow.

It is clear that, as explained in Section 3.2.1, for the actual volume of the domain to be represented, the parameter α must be chosen such that the minimum radius of curvature of the boundary is properly reproduced. In this case, α is bounded at least by the radius of the cylindrical obstacle, $R = 0.0015$. In the case in which $\alpha > R$ the domain is not well reproduced, which is illustrated in Figure 40a.

From Figure 40, one can readily infer that a bad α-value leads to the lack of fulfillment of the conservation of mass (or volume) throughout the simulation. In Figure 41 a comparison is shown between the analytical volume and the computed one, for three different α values. It is seen that, if α is too big a spurious increase in the volume of the domain is obtained (see Figure 40a). On the other hand, if α is too small a loss of volume is produced as the triangles become distorted by the fluid motion. Thus, α is bounded between the nodal

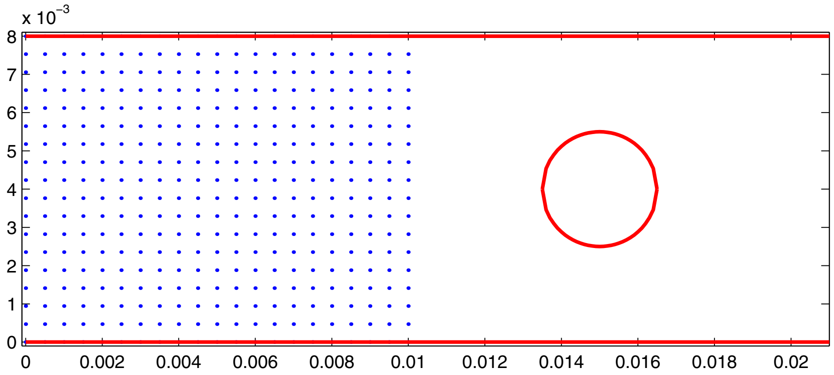
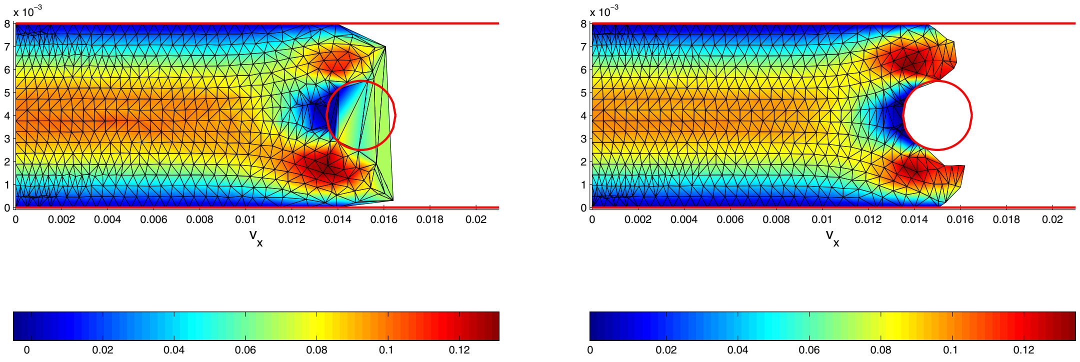


Figure 39. Geometry and initial nodal discretisation for the flow between two plates problem



(a) Too high α value

(b) Appropriate α

Figure 40. Comparison of the effect of a badly chosen α-value

spacing h and the the radius of the cylinder R to obtain a suitable geometric description. Here,

$$h < \alpha < R. \tag{78}$$

Another source of error appears if the time increment Δt used is too large. In this case, if $|\mathbf{v}|\Delta t > h$ the particle motion in a single time step is larger than the nodal spacing h, which leads to an undesired mixture in the fluid. Consider the previous example, in which the nodes on either side of the axis of symmetry are labelled with different colours (Figure 42). Once the obstacle is passed, there will be a “welding” line between the green fluid and the blue one. If Δt is too large the two branches of the fluid can mix, leading to a loss in volume. On the other hand, if α is too large the position of the welding line becomes unclear (see Figure 42a).

Thus, the error in the mass conservation is of order $\mathcal{O}(\alpha) \approx \mathcal{O}(h)$. It can be concluded that the α-NEM approximation conserves mass as the cloud density h is reduced. Finally,

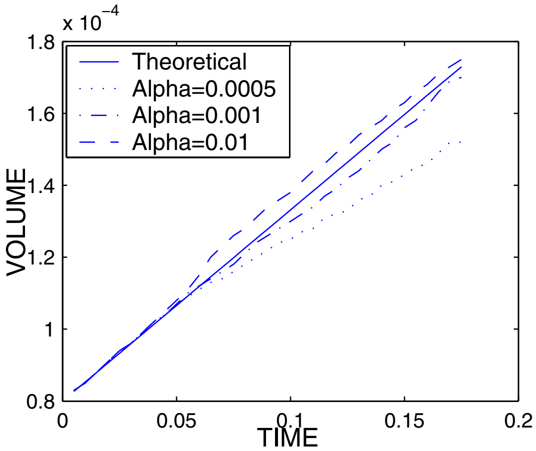
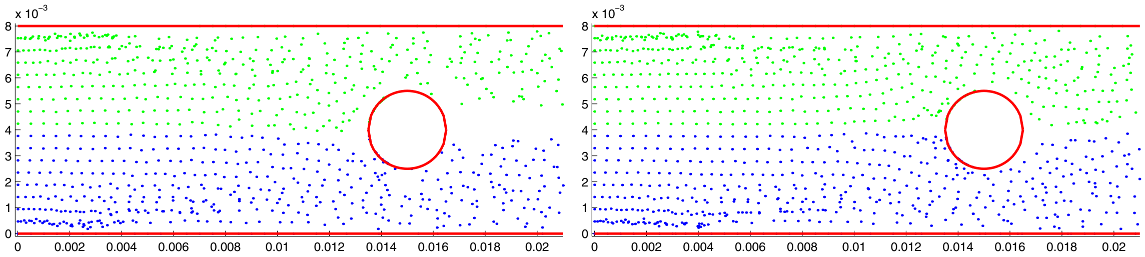


Figure 41. Comparison between the theoretically computed volume and the actually computed for three different α values



(a) Too high α value

(b) Appropriate α

Figure 42. Effect of the time increment in the final geometry of the domain

to avoid fluid interpenetration the time step must satisfy

$$\Delta t = \min_{\mathbf{x}} \left\{ \frac{\alpha(\mathbf{x}, t)}{|\mathbf{v}(\mathbf{x}, t)|} \right\}. \tag{79}$$

6.3 Mould Filling Simulation

In this section the application of natural neighbour Galerkin methods to the simulation of complex free surface fluid flows (often referred to as *moving boundary* problems) is demonstrated. Mould filling is a prototypical example of a moving-boundary problem. Flow during mould filling is a very important process in industry, since it controls the occurrence of defects in the final manufactured product.

In the simulation of mould filling processes several strategies can be found in the literature. One of the most widely-used technique is that of fixed mesh (Eulerian) finite elements. Tracking of the flow front position is carried out by means of a volume of fluid (VOF) technique, in which a new variable (*presence of fluid* function) is introduced whose evolution is governed by a purely advective (transport) equation. This introduces additional numerical difficulties in the treatment of such equations, as well as some artificial diffusion in the flow front position. For a detailed discussion and comparison of these methods, the interested reader can refer to [83] and [76].

On the other hand, the use of Lagrangian (*moving mesh*) techniques is also possible. These have the important property of being able to transport variables related to the fluid history in a straightforward manner by the method of characteristics [83]. However, if traditional FE techniques are used, the need for remeshing and the mapping of state variables from one mesh to the other is necessary, which introduces numerical errors in the final solution and an increase in computational cost notably in 3-d. To avoid such problems, meshless techniques appear as an attractive choice to tackle these kinds of problems.

In this section we analyse the problem of mould filling by means of natural neighbour Galerkin methods. We consider the filling of a theoretical mould whose geometry and dimensions are shown in Figure 43. Again, standard Sibson interpolation is used throughout the simulation.

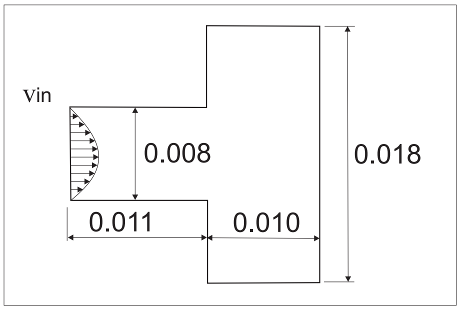


Figure 43. Geometry and dimensions of the mould

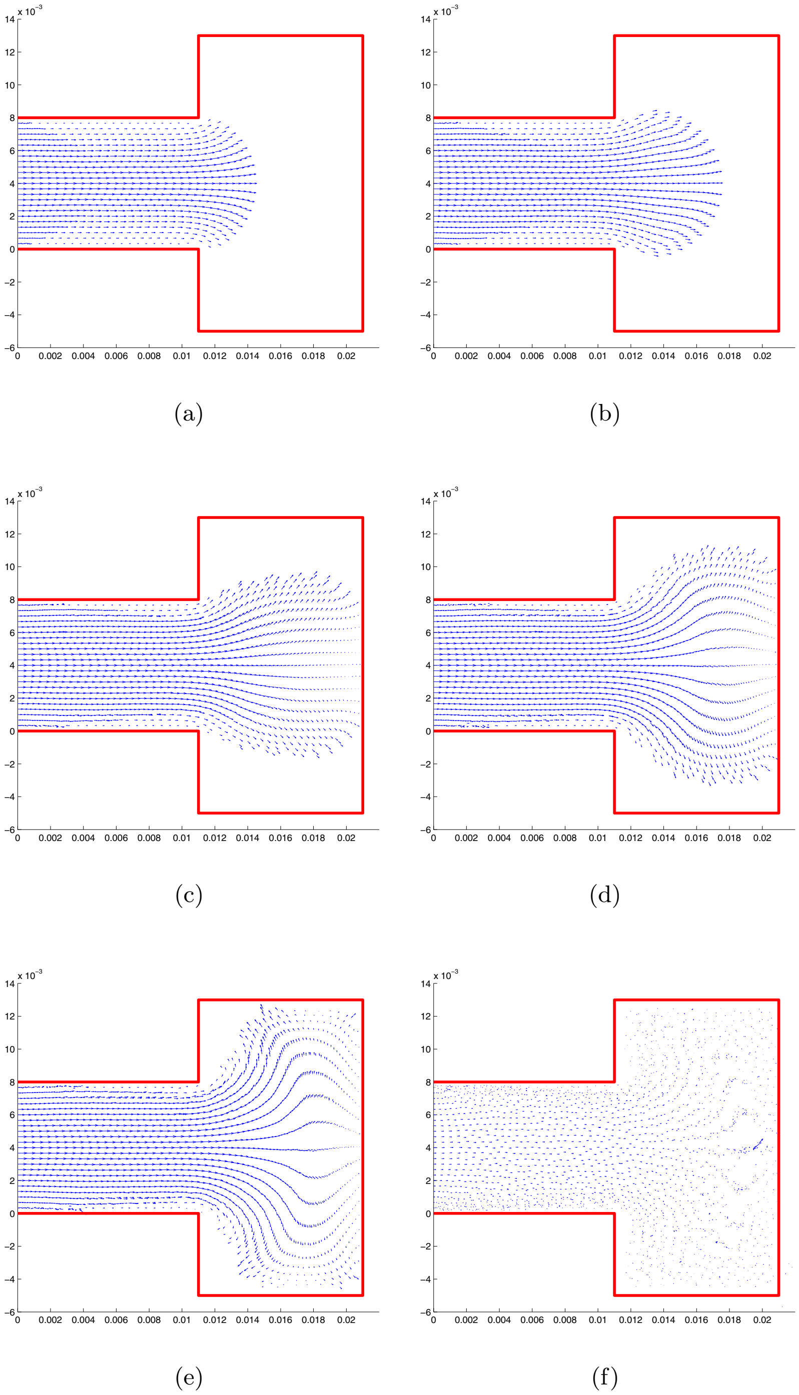
The mould is filled with a number of Delaynay triangulation algorithms with velocity $v(x=0,y)=\frac{4V}{h^2}(h-y)y$ ($V=0.1\ m/s$) until it is completely filled. At each time step the α-shape of the domain is extracted, provided that a sufficiently refined initial cloud of nodes has been considered. No nodes (apart from those being incorporated from the inlet) have been added to the model and thus, in this sense, no *remeshing* is involved. An updated Lagrangian framework is adopted to avoid numerical problems, so the natural neighbours for each node are determined at each time step. This leads to lower computational costs, since there exist a number of Delaunay trianglation algorithms (see [47]) that are of near-optimal complexity $\mathcal{O}(n\log n)$, where n is the number of nodes.

No slip boundary conditions are applied to the whole mould and viscosity μ was set to unity. The initial cloud consists of 357 regularly distributed nodes, whereas the final one has 2341 nodes. Several nodal velocity fields are shown in Figures 44a–44f.

It can be seen that at the last time step (Figure 44f) the velocity field tends to zero velocity everywhere, since the mould is full and the fluid remains incompressible. In Figure 45 the α-shape of the fluid geometry at time $t=60\times0.05$ is depicted. Note that the Delaunay triangulation is by no means apt to be used in a traditional FEM simulation, due to the high distortion of the triangles. In addition, the position of the flow front is clear.

In order to verify the conservation of mass equation given in Eq. (74), the computed volume of the fluid after the constant inlet flow is compared to the theoretical result in Figure 46. It can be seen how the predicted volume is very accurate and only some minor discrepancies appear over the last few steps of the simulation, where the α-shape is unable to exactly reproduce sharp corners of ninety degrees (a volume slightly higher than the actual one is obtained). Of course, more accurate predictions can be obtained by using a finer cloud of nodes.

This example points to the fact that natural neighbour Galerkin methods are very well suited for applications in which the mesh suffers high distortion and traditional finite



(a)

(b)

(c)

(d)

(e)

(f)

Figure 44. Six velocity fields at different time steps during the filling simulation

element simulations need significant remeshing. This problem is even more acute in three-dimensional problems, where high quality 3-d mesh generators are not readily available for moving-boundary problems.

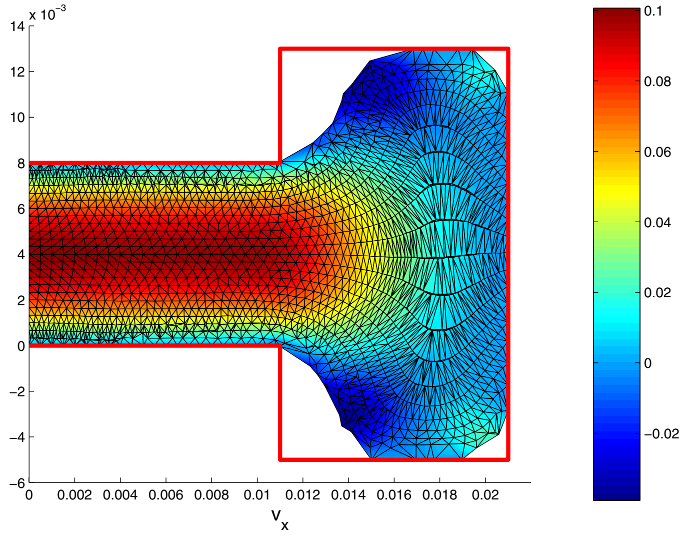


Figure 45. Horizontal velocity contour plot over the α-shape of the fluid at the 60^{th} time step

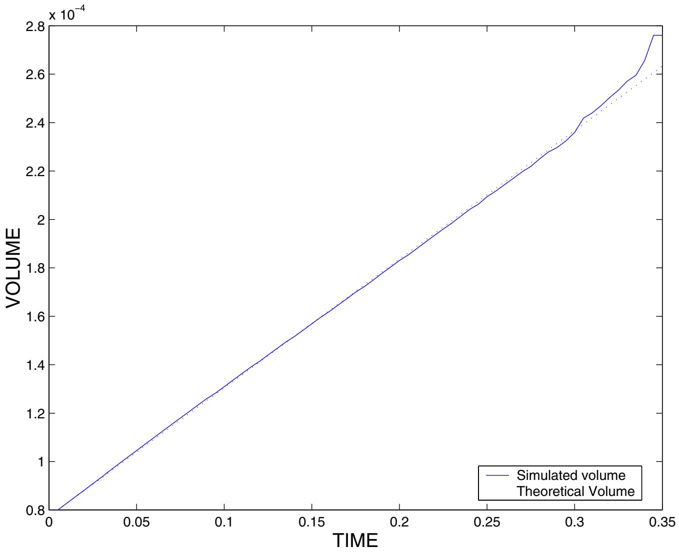


Figure 46. Computed volume of the fluid through the filling process compared with the Theoretical one

7 NATURAL NEIGHBOUR FINITE DIFFERENCE SCHEME ON IRREGULAR GRIDS

Although this paper is primarily related to Galerkin applications, we also review here the application of natural neighbour interpolation to the construction of a finite difference scheme, which was first proposed in [108].

The construction of finite difference schemes on non-uniform grids in one-dimension has been extensively studied [116, 55, 70, 123, 81, 69], but the extension to irregular grids in 2-d and 3-d is still an open and active area of research. Finite difference schemes are attractive when the domain is regular and uniform nodal spacing suffices. Finite elements are particularly useful when dealing with boundary-value problems on arbitrarily-shaped geometries with nonlinear material behaviour, and where p- or h-adaptivity is desirable. Finite volume schemes are based on a volume integral formulation of the original partial differential equation (PDE) on a set of finite (control) volumes that partitions the domain. This technique proves to be well-suited to deal with physical conservation laws (mass, momentum, and energy), such as those that arise in fluid dynamics, groundwater flow and contamination, and for shock-capturing applications.

Classical finite difference schemes are typically based on regular grids, and hence are restricted to problem domains with regular geometry. Due to the wide applicability of finite difference schemes for solving initial/boundary value problems in many areas of mathematical physics, there have been many contributions towards extending the methodology on irregular grids. Building on prior work due to Jensen [67], Liszka and Orkisz [78] proposed a generalized finite difference method (GFDM) on irregular grids. The central issue addressed in [78] was the appropriate selection of the computational cell (star-shaped domain) that surrounds a node, so as to yield a well-conditioned linear system of equations. In [78], a Taylor series approximation was used, whereas Breitkopf *et al.* [25] adopted moving least squares approximants [73] in the GFDM to derive the discrete form for the differential operators.

7.1 Difference Approximation for the Diffusion Operator

The Voronoi tessellation and its dual the Delaunay triangulation are often used to discretize a continuum. The Voronoi cell provides a natural domain of influence for a given node, and hence it is commonly used in numerical methods such as the finite volume and the finite element method. In the Voronoi Cell Finite Element Method (VCFEM) [54], the Voronoi tessellation is used to represent the material microstructure and a finite element formulation is developed on the Voronoi cells. The numerical method is used for multiscale analysis of heterogeneous materials.

We adopt the Voronoi cell as the computational cell and use an integral balance to derive the finite difference scheme for the diffusion operator. The motivation for the above is derived from prior work on the Laplace interpolant [15, 111] and from [51], where prescriptions are presented for vector identities (such as the gradient and divergence) on a lattice. The approach we pursue bears close connection to that pursued within the context of a finite volume scheme [86], as well as in integrated finite differences [90] which is extensively used in hydrogeology. The finite difference scheme can be viewed as a generalization of the approach in [90] to Voronoi grids.

As out model problem, we consider the following 2-dimensional steady-state diffusion equation with Dirichlet boundary conditions:

$$-\mathcal{L}u(\mathbf{x}) = -\boldsymbol{\nabla}\cdot\big(\kappa(\mathbf{x})\boldsymbol{\nabla}u(\mathbf{x})\big) = f(\mathbf{x}) \quad \text{in} \quad \Omega \subset \mathbb{R}^2 \tag{80a}$$

$$u(\mathbf{x}) = g(\mathbf{x}) \quad \text{on } \partial\Omega, \tag{80b}$$

where $\boldsymbol{\nabla}$ is the gradient operator, Ω is an open set in 2-d, and $\partial\Omega$ is the boundary of Ω. If $\kappa \equiv 1$, then $\mathcal{L}$ is the Laplacian operator. The above diffusion equation, which is written in flux form, is used to describe diffusion processes such as heat conduction, mass transfer, flow through porous media, or the potential in electrostatics. The function $g(\mathbf{x})$ is the specified boundary temperature on $\partial\Omega$ in heat conduction (Dirichlet boundary condition).

The fundamental requirement for the solution of diffusion or convection-diffusion equations using finite element [34], finite volume methods [94], or finite differences is that the approximation must satisfy a discrete maximum principle—non-physical local extrema must not be present in the numerical solution, for otherwise spurious oscillations or even divergence can result. A scheme that satisfies a maximum principle is also known as a monotone scheme. This requirement stems from the fact that in a diffusion problem, the flow (fluid or heat) must travel from regions of higher potential to regions of lower potential, and hence the numerical approximation must mirror the same behaviour. The need to satisfy the discrete maximum principle renders the task of constructing monotone finite difference and finite volume schemes on arbitrary unstructured grids as non-trivial.

The model diffusion problem in Eq. (80) is solved using a finite difference method, or equivalently a point collocation scheme [1]. The discrete form for the model problem in Eq. (80) is written as:

$$-\mathcal{L}_h u(\mathbf{x}_I) = f(\mathbf{x}_I), \quad I = 1, 2, \ldots, M \ (\mathbf{x}_I \in \Omega) \tag{81a}$$

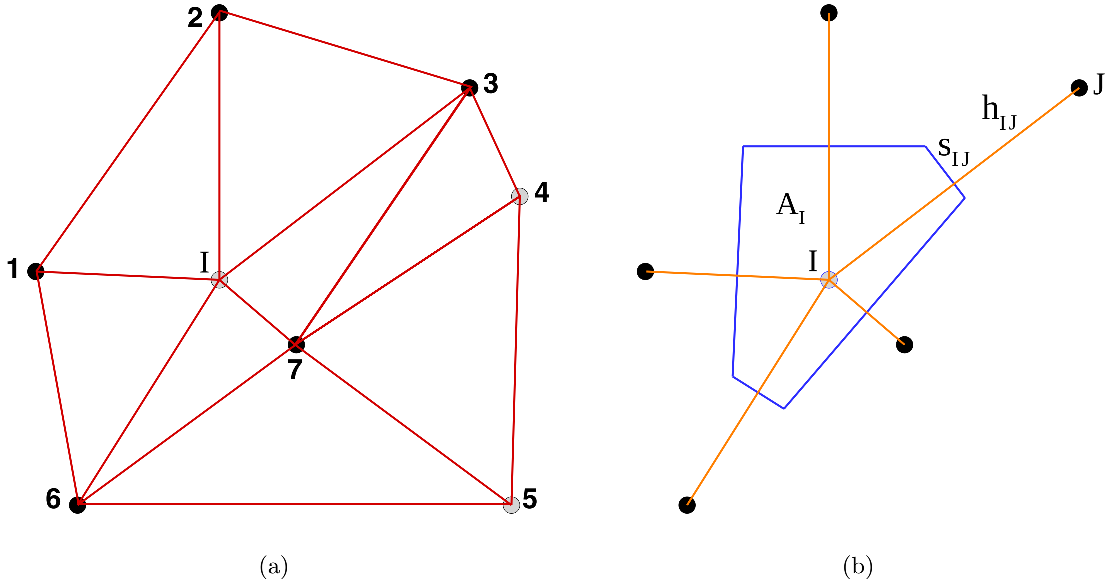
$$u(\mathbf{x}_I) = g(\mathbf{x}_I) \quad \mathbf{x}_I \in \partial\Omega, \tag{81b}$$

where M is the number of the nodes in the domain, $\mathcal{L}_h$ is the discrete diffusion operator, and u_h is the finite difference approximation (h denotes a measure of the nodal spacing).

Consider the domain Ω shown in Figure 1c which is reproduced in Figure 47a. The point p that was added to the tessellation is now assumed to denote a node (say I). We can write the balance (conservation) law for the negative divergence of the flux ($\mathbf{q} = -\kappa\boldsymbol{\nabla}u$) over the Voronoi cell A_I (see Figure 47b) in the form:

$$[-\boldsymbol{\nabla}\cdot\big(\kappa(\mathbf{x})\boldsymbol{\nabla}u\big)]_I = -\lim_{A_I\to 0}\frac{\int_{A_I}\kappa(\mathbf{x})\boldsymbol{\nabla}u\cdot\mathbf{n}\,d\Omega}{A_I} = -\lim_{A_I\to 0}\frac{\int_{\partial A_I}\kappa(\mathbf{x})\dfrac{\partial u}{\partial n}\,d\Gamma}{A_I}, \tag{82}$$

where A_I is the area of the Voronoi cell of node I, and Gauss's (divergence) theorem has been invoked to convert the volume integral to a surface integral.



(a) (b)

Figure 47. Finite difference approximation at node I. (a) Node I in the triangulation; and (b) Voronoi cell of node I and its natural neighbours

On using a cell-based central difference approximation for the normal derivative of u (see Figure 47b), and the harmonic average [90] of the nodal diffusivities for the diffusive coefficient $\kappa(\mathbf{x})$, we can write the discrete operator in Eq. (81) as:

$$(\mathcal{L}_h u)_I = \frac{\int\limits_{\partial A_I} \kappa(\mathbf{x}) \dfrac{\partial u_h}{\partial n} \, d\Gamma}{A_I} = \frac{1}{A_I} \sum_{J=1}^{n} \kappa_{IJ} \frac{(u_J - u_I)}{h_{IJ}} s_{IJ}, \quad \frac{2}{\kappa_{IJ}} = \frac{1}{\kappa_I} + \frac{1}{\kappa_J}, \tag{83}$$

where n is the number of natural neighbours for node I ($n = 5$ in Figure 47b), h_{IJ} is the distance between nodes I and J, and s_{IJ} is the length of the Voronoi edge associated with nodes I and J (Figure 47b). Using Eq. (8) and after some algebraic simplification, we obtain:

$$(\mathcal{L}_h u)_I = \frac{1}{A_I} \left[\sum_{J=1}^{n} \beta_{IJ} u_J - \beta_I u_I \right], \quad \beta_I = \sum_{J=1}^{n} \beta_{IJ}, \tag{84a}$$

$$\beta_{IJ} = \kappa_{IJ} \alpha_{IJ}, \quad A_I = \frac{1}{4} \sum_{J=1}^{n} s_{IJ} h_{IJ}, \tag{84b}$$

where α_{IJ} is the Laplace weight. The above expression is consistent with the prescription introduced for the discrete Laplacian ($\kappa \equiv 1$) on a random lattice [51]. The right-hand side of Eq. (81a) is just f_I. Hence, the discrete system for the finite difference (or collocation) scheme is:

$$\mathbf{K}\mathbf{u} = \tilde{\mathbf{f}}, \tag{85a}$$

$$\mathbf{K}_{II} = \beta_I, \ \mathbf{K}_{IJ} = -\beta_{IJ} \ (I \neq J), \quad \tilde{\mathbf{f}}_I = f_I A_I, \quad (\mathbf{x}_I \in \Omega), \tag{85b}$$

$$u_I = g(\mathbf{x}_I), \quad \mathbf{x}_I \in \partial\Omega. \tag{85c}$$

If $\kappa \equiv 1$, we obtain the Poisson equation. Using

$$\alpha_{IJ} = \frac{s_{IJ}}{h_{IJ}}, \quad \alpha_I = \sum_{J=1}^{n} \alpha_{IJ}, \tag{86}$$

the discrete system can now be written as:

$$\mathbf{K}_{II} = \alpha_I, \ \mathbf{K}_{IJ} = -\alpha_{IJ} \ (I \neq J), \ \tilde{\mathbf{f}}_I = f_I A_I, \ (\mathbf{x}_I \in \Omega), \tag{87a}$$

$$u_I = g(\mathbf{x}_I), \ \mathbf{x}_I \in \partial\Omega. \tag{87b}$$

7.2 Discrete Nodal Gradient Operators

In a Galerkin method, the natural neighbour-based interpolation scheme in Eq. (10) is used to evaluate $u^h(\mathbf{x})$ at any point within the domain Ω. In moving to a discrete (finite-difference) setting, we can reconstruct the same picture by imagining that a node (say I that is located at $\mathbf{x}_I \equiv \mathbf{x}$) has been removed and is then inserted into the grid (see Figure 47a). Clearly, depending on the context, one can see (conceptually) the equivalence between defining the approximation at a point p (continuum perspective) vis-à-vis that at a node (discrete/lattice perspective) for a given nodal discretization. This viewpoint extends to the evaluation of the discrete nodal gradient too.

The discrete nodal gradient at a node can be approximated using the following two approaches:

1. The gradient at node I is evaluated by assuming that node I is inserted into the tessellation and a discrete approximation for $u_{,j}$ at $\mathbf{x}_I$ ($\mathbf{x} \equiv \mathbf{x}_I$) is computed. Referring to Eq. (11), we can approximate the gradient of a scalar-valued function u at $\mathbf{x}_I$ as

$$\frac{\partial u^h}{\partial x_j}(\mathbf{x}_I) = \sum_{J=1}^{n} w_J^j(\mathbf{x}_I) u_J = \sum_{J=1}^{n} \phi_{J,j}(\mathbf{x}_I) u_J \quad (j = 1, 2), \tag{88}$$

   where $n = 5$ for node I in Figure 47 and the weight w_J^j is the derivative of the Sibson or the Laplace shape function in the x_j-coordinate direction.

2. As a second approach, we consider the non-local smoothing procedure (finite volume averaging) for the strain operator that was proposed by Chen and co-workers [28]. The displacement gradient can be written as

$$\frac{\partial u^h}{\partial x_j}(\mathbf{x}_I) = \frac{\int\limits_{A_I} u^h_{,j}(\mathbf{x})\, d\Omega}{A_I} = \frac{\int\limits_{\partial A_I} u^h n_j \, d\Gamma}{A_I}, \tag{89}$$

   where Gauss's divergence theorem has been used. In [28], numerical integration was used to evaluate the surface integral. For the special case of a regular grid, one can approximate the integral using an average weighted-value for u^h on each Voronoi edge. The choice $u^h(\mathbf{x}_m) = (u_I + u_J)/2$ renders an exact result (central-difference scheme) for a linear u field on a regular grid:

$$\frac{\partial u^h}{\partial x_j}(\mathbf{x}_I) = \frac{\sum\limits_{J=1}^{n} (u_I + u_J) n_j^J s_{IJ}}{2A_I}, \quad A_I = \frac{1}{4} \sum_{J=1}^{n} s_{IJ} h_{IJ}, \tag{90}$$

   where n_j^J $\big($for e.g., $n_1^{IJ} = (x_J - x_I)/h_{IJ}\big)$ is the j-th component of the unit outward normal to the Voronoi edge common to nodes I and J.

7.3 Theoretical Analysis

As indicated previously and also mentioned in prior studies [106, 111], the Sibson and the Laplace interpolant reduce to linear finite element interpolation in 1-dimension. In 1-d, $s_{IJ} \equiv 1$ since the Voronoi facet is a point and all interior nodes have two neighbours (Figure 48). Hence, the stiffness matrix in the finite difference scheme is identical to the one arising in finite elements:

$$\mathbf{K}_{II} = \frac{1}{h_I} + \frac{1}{h_{I+1}}, \quad \mathbf{K}_{IJ} = -\frac{1}{h_I}\ (J = I - 1), \quad \mathbf{K}_{IJ} = -\frac{1}{h_{I+1}}\ (J = I + 1). \tag{91}$$

However, the right-hand side $f(x)$ is treated differently in both methods—in finite elements, a weighted-integral value $\int_\Omega f N_I \, dV$ is assigned to the Ith row in the external force vector, whereas in the difference scheme, collocation at the node is used. Hence, the solution obtained by the two methods will in general be different.

An interesting theoretical aspect is that of consistency on non-uniform grids which has been extensively studied for cell- and vertex-centered difference schemes. Consistency in finite difference schemes ensures that, in the limit, when the grid spacing tends to zero, the difference between the finite difference scheme for the differential operator and the continuous form of the same is zero. We use the Taylor series expansion to study consistency in 1-d for the Laplace operator. Consider a direction n with unit vector $\mathbf{n} = (\pm 1, 0)$ in 1-d

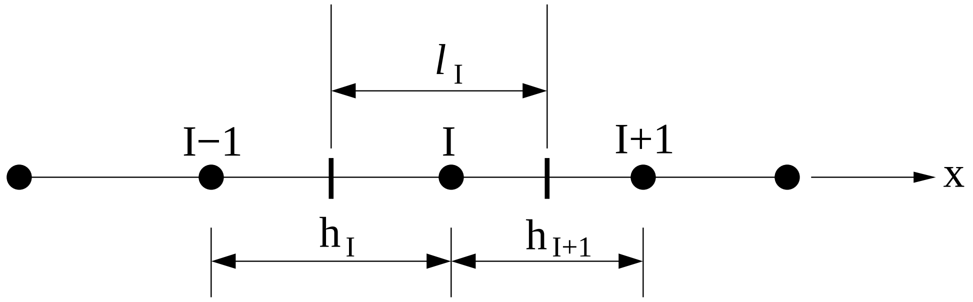


Figure 48. Voronoi cell for a non-uniform grid in 1-d

and let h be a scalar. The Taylor series approximation for a function $u(x)$ at $\tilde{x} = x \pm h$ is written as:

$$u(x \pm h) = u(x) \pm u'(x)h + u''(x)\frac{h^2}{2} \pm u'''(x)\frac{h^3}{6} + \mathcal{O}(h^4). \tag{92}$$

Now, let us consider the finite difference approximation for the Laplacian operator at node I. Using Eq. (92), the Taylor series expansion at neighbours $I-1$ and $I+1$ are:

$$u_{I-1} = u_I - u'(x_I)h_I + u''(x_I)\frac{h_I^2}{2} - u'''(x_I)\frac{h_I^3}{6} + \mathcal{O}(h_I^4), \tag{93a}$$

$$u_{I+1} = u_I + u'(x_I)h_{I+1} + u''(x_I)\frac{h_{I+1}^2}{2} + u'''(x_I)\frac{h_{I+1}^3}{6} + \mathcal{O}(h_{I+1}^4). \tag{93b}$$

On multiplying both sides of Eqs. (93a) and (93b) by $1/h_I$ and $1/h_{I+1}$, respectively, and adding the result, we obtain

$$\frac{u_{I-1}}{h_I} + \frac{u_{I+1}}{h_{I+1}} = u_I\left(\frac{1}{h_I} + \frac{1}{h_{I+1}}\right) + u''(x_I)\left(\frac{h_I + h_{I+1}}{2}\right) + u'''(x_I)\frac{h_{I+1}^2 - h_I^2}{6} + \mathcal{O}(h_{mI}^3), \tag{94}$$

where $h_{mI} = \max(h_I, h_{I+1})$. On using the definition of α_{IJ} and α_I that appear in Eq. (86), and since $\ell_I = (h_I + h_{I+1})/2$, Eq. (94) simplifies to

$$\frac{1}{\ell_I}\left[\sum_{J=1}^{2}\alpha_{IJ}u_J - \alpha_I u_I\right] = u''(x_I) + u'''(x_I)\left(\frac{h_{I+1} - h_I}{3}\right) + \mathcal{O}(h_{mI}^2). \tag{95}$$

On using Eq. (87) and noting that $u_I = u(x_I)$ is the exact nodal value of u, the above equation can be re-written as

$$\mathcal{L}u - \mathcal{L}_h u = -u'''(x_I)\left(\frac{h_{I+1} - h_I}{3}\right) - \mathcal{O}(h_{mI}^2) \tag{96}$$

and if $h_I \neq h_{I+1}$ (non-uniform grid), then

$$\mathcal{L}u - \mathcal{L}_h u = -\mathcal{O}(h_{mI}) \tag{97}$$

and hence in the limit that the grid spacing tends to zero, we have

$$\lim_{h_{mI}\to 0} \mathcal{L}u - \mathcal{L}_h u = 0 \tag{98}$$

which shows that *first-order* consistency is obtained on non-uniform grids. If the nodal spacing is uniform (say h), then from Eq. (96) we note that the coefficient of u''' also vanishes, and hence *second-order* consistency is established.

The traditional treatment of consistency and convergence via a Taylor series expansion on regular grids is not readily extendable to non-uniform grids. The notion of *flux consistency* appears to be important in the development of error estimates [69] for difference schemes on Cartesian grids. Kreiss *et al.* [70] coined the term *supraconvergence* for schemes that converge at a higher-order than the local truncation error; supraconvergence has received a lot of attention in the numerical analysis literature for node- and cell-centered finite difference schemes. In 2-d, the difference scheme is inconsistent (zeroth-order) on irregular grids, but second-order convergence in u is attained. A detailed theoretical analysis with supportive convergence tests is presented in [108].

7.4 Comparisons to Classical Finite Differences

Referring to the 1-d grid in Figure 48 and on using Eq. (87), the discrete approximation for the Laplacian ($\mathcal{L}u = u''$) at node I can be written as:

$$(\mathcal{L}_h u)_I = \frac{2}{h_I + h_{I+1}} \left[\frac{1}{h_I} u_{I-1} + \frac{1}{h_{I+1}} u_{I+1} - \left(\frac{1}{h_I} + \frac{1}{h_{I+1}} \right) u_I \right], \tag{99}$$

or

$$(\mathcal{L}_h u)_I = \frac{2}{h_I h_{I+1}(h_I + h_{I+1})} \big(h_{I+1} u_{I-1} + h_I u_{I+1} - (h_I + h_{I+1}) u_I \big). \tag{100}$$

If $h_I = h_{I+1} = h$ (uniform grid), we obtain the classical central-difference approximation:

$$(\mathcal{L}_h u)_I = \frac{u_{I-1} + u_{I+1} - 2u_I}{h^2}, \tag{101}$$

and if for example $h_I = 3\Delta x/4$, $h_{I+1} = 3\Delta x/2$, then

$$(\mathcal{L}_h u)_I = \frac{16}{27\Delta x^2} (2u_{I-1} - 3u_I + u_{I+1}), \tag{102}$$

which is identical to the expression given in [115], where cell-centered finite difference schemes are derived on non-uniform grids. In [115], Taylor series expansion and the method of undetermined coefficients are used to find the weights so that the coefficient of the first derivative vanishes, coefficient of the second derivative is unity, etc. Hence, Eq. (100) is a generalized 1-d finite difference approximation for the Laplacian on non-uniform grids.

We now explore the difference scheme in 2-d. Rectangular and hexagonal grids are used in finite difference methods. In lattice models, Monte Carlo simulations are typically carried out on square and triangular lattices (hexagonal grid) in which periodic boundary conditions are assumed.

Claim 1. The Voronoi cell finite difference approximation for the Laplacian reduces to the classical finite difference method for rectangular and hexagonal grids.

Proof. The first part is shown for a rectangular grid with nodal spacings h_1 and h_2 in the coordinate directions (Figure 49). Using Eq. (84), we can write the finite difference approximation for the Laplacian ($\mathcal{L}_h = \nabla_h^2$) at node I as:

$$\nabla_h^2 u = \frac{1}{A_I}\left[\sum_{J=1}^{4}\alpha_{IJ}u_J - \alpha_I u_I\right], \quad \alpha_{IJ} = \frac{s_{IJ}}{h_{IJ}}, \quad \alpha_I = \sum_{J=1}^{4}\alpha_{IJ}. \tag{103}$$

From Figure 49, we note that $A_I = h_1 h_2$, $s_{IJ} = h_2$, $h_{IJ} = h_1$ $(J = 1,2)$, $s_{IJ} = h_1$, $h_{IJ} = h_2$ $(J = 3,4)$, and hence $s_{IJ}/h_{IJ} = h_2/h_1$ $(J = 1,2)$ and $s_{IJ}/h_{IJ} = h_1/h_2$ $(J = 3,4)$. The above equation reduces to

$$\nabla_h^2 u = \frac{1}{h_1 h_2}\left[\frac{h_2}{h_1}(u_1 + u_2) + \frac{h_1}{h_2}(u_3 + u_4) - 2\left(\frac{h_1}{h_2} + \frac{h_2}{h_1}\right)u_I\right], \tag{104}$$

which simplifies to

$$\nabla_h^2 u = \frac{u_1 + u_2 - 2u_I}{h_1^2} + \frac{u_3 + u_4 - 2u_I}{h_2^2} \tag{105}$$

which is the five-point finite difference approximation for the Laplacian on a rectangular grid. If $h_1 = h_2 = h$, then

$$\nabla_h^2 u = \frac{u_1 + u_2 + u_3 + u_4 - 4u_I}{h^2} \tag{106}$$

which is the well-known five-point stencil for the Laplacian on a square grid. The finite difference weights for a square grid are shown in Figure 50a.

Consider the hexagonal grid shown in Figure 50b. Referring to Eq. (103), we can once again write the finite difference approximation for the Laplacian at node I as:

$$\nabla_h^2 u = \frac{1}{A_I}\left[\sum_{J=1}^{6}\alpha_{IJ}u_J - \alpha_I u_I\right], \quad \alpha_{IJ} = \frac{s_{IJ}}{h_{IJ}}, \quad \alpha_I = \sum_{J=1}^{6}\alpha_{IJ}. \tag{107}$$

For a hexagonal grid, $h_{IJ} = h$, $s_{IJ} = s$, and hence $\alpha_{IJ} = s/h$ $\forall J$. In addition, using Eq. (84b), we have $A_I = 6sh/4$. Hence, the above equation can be written as

$$\nabla_h^2 u = \frac{2}{3sh}\left[\sum_{J=1}^{6}\frac{s}{h}u_J - \frac{6s}{h}u_I\right], \tag{108}$$

and therefore

$$\nabla_h^2 u = \frac{2}{3h^2}(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 - 6u_I) \tag{109}$$

which is the seven-point stencil (hexagonal grid) for the classical finite difference method. The finite difference weights for this case are shown in Figure 50b. □

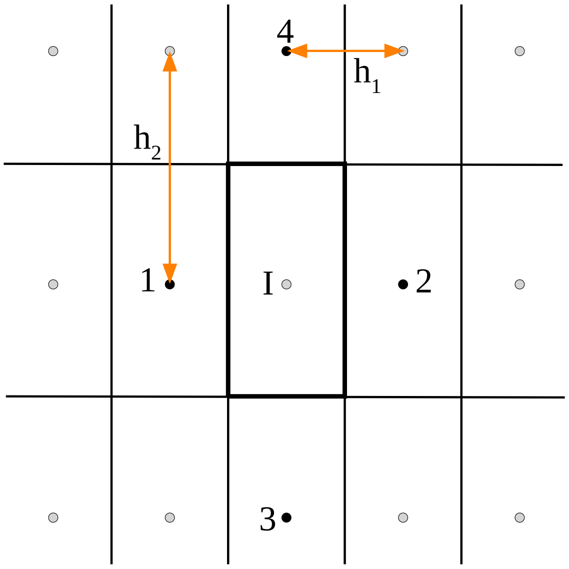


Figure 49. Finite difference approximation for the Laplacian on a rectangular grid with nodal spacings h_1 and h_2 in the coordinate directions. The Voronoi cell of node I is indicated by the four dark lines that form a rectangle of area $h_1 h_2$

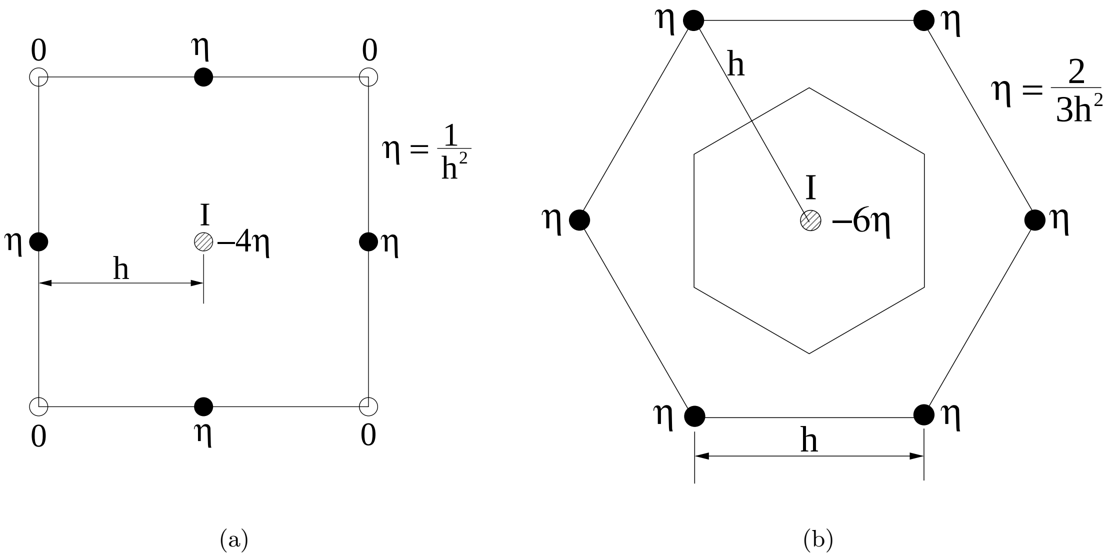


Figure 50. Finite difference weights for the Laplacian on regular grids. Filled circles are the natural neighbours of the interior node and the weights for each neighbour are indicated. (a) Square grid; and (b) Hexagonal grid

7.5 Finite Difference Examples

We explore the application of the difference scheme to the Poisson equation in 1-d and 2-d. In 1-d, two elliptic problems with Dirichlet boundary conditions are considered:

$$-u'' = \pi^2 \sin(\pi x) \quad \text{in} \quad \Omega = (0,1), \quad u(0) = u(1) = 0, \tag{110a}$$

$$-u'' + u = 0 \quad \text{in} \quad \Omega = (0,1), \quad u(0) = 2,\ u(1) = e + \frac{1}{e}. \tag{110b}$$

The exact solutions for the above problems are: $u(x) = \sin(\pi x)$ and $u(x) = e^x + e^{-x}$, respectively. In our analysis, the L_∞ (max or sup) discrete norm is defined as:

$$\|u - u_h\|_{\infty,\Omega} = \max_{I=1,\dots,M} |u(x_I) - u_h(x_I)|, \tag{111}$$

where u and u_h are the exact and the finite difference solutions, respectively. In addition, the L_∞ discrete norm for the truncation error is given by

$$c\|\tau\|_{\infty,\Omega} = \max_{I=2,\dots,M-1} |\tau_I|, \tag{112a}$$

where

$$\tau_I = -\frac{1}{h_I}u(x_{I-1}) + \left(\frac{1}{h_I} + \frac{1}{h_{I+1}}\right)u(x_I) - \frac{1}{h_{I+1}}u(x_{I+1}) - \pi^2\sin(\pi x_I) \tag{112b}$$

is the truncation error at node I for Eq. (110a) and

$$\tau_I = -\frac{1}{h_I}u(x_{I-1}) + \left(\frac{1}{h_I} + \frac{1}{h_{I+1}}\right)u(x_I) - \frac{1}{h_{I+1}}u(x_{I+1}) + u(x_I) \tag{112c}$$

is the truncation error at node I for Eq. (110b). In the above equations, $u(\cdot)$ is the nodal value of the exact solution, and the definitions for h_I and h_{I+1} are shown in Figure 48. The sup-norms are evaluated for different maximum cell size $\ell_m = \max_I \ell_I$, where $\ell_I = (h_I + h_{I+1})/2$, and the rate of convergence is estimated by the slope on a log-log plot.

We considered non-uniform grids in 1-d to test the order of convergence of the truncation error and also of u [69]. To this end, a uniform random number between 20 and 200 was chosen to be the grid size (number of nodes M). Then, the spatial coordinate of the M nodes was set by picking $M-2$ random numbers between zero and unity (two nodes were assigned the coordinates $x = 0$ and $x = 1$). To conduct the convergence study on non-uniform grids, we executed 1000 independent simulations (trials) for each example problem.

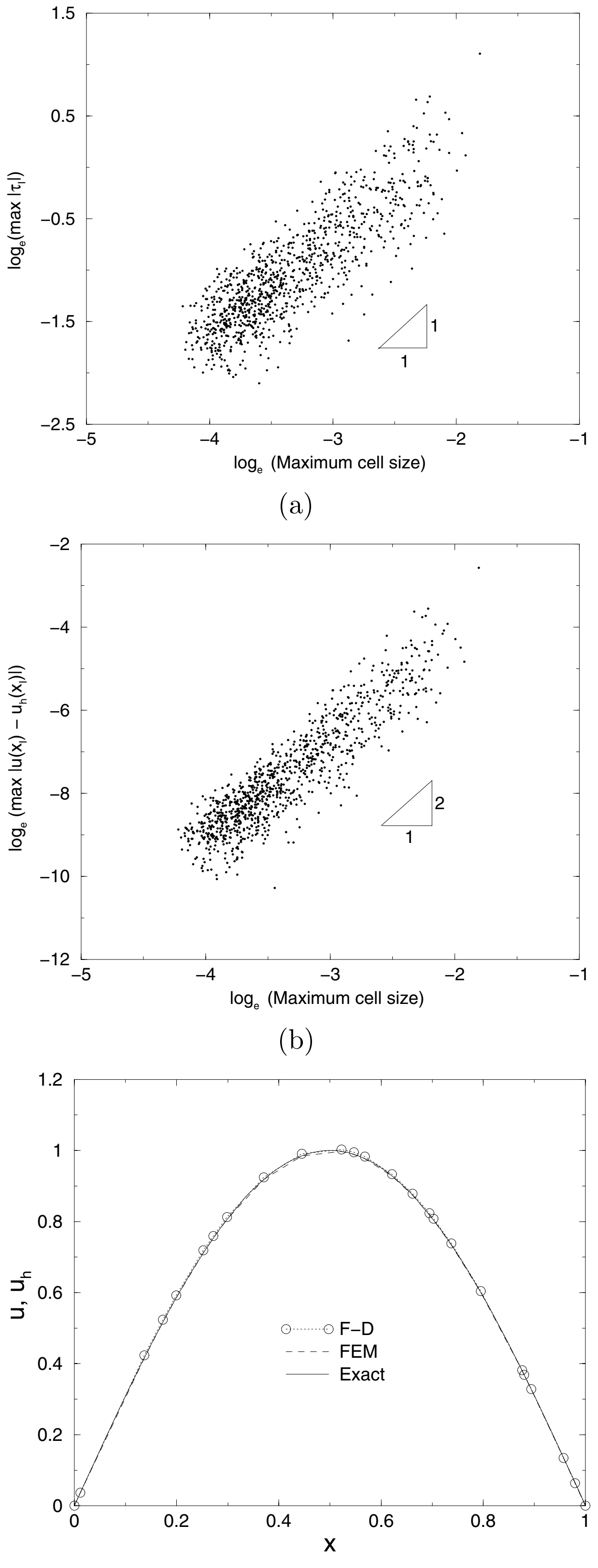
In Figure 51, the results for the Poisson problem (Example I) are presented and in Figure 52, numerical simulations results for the elliptic problem (Example II) in Eq. (110b) are shown. In Figures 51a and 52a, it is clearly observed that the truncation error decreases as $\mathcal{O}(h)$, whereas in Figures 52b and 52b, the rate of convergence in u is clearly observed to be 2. This is in agreement with previous studies on finite volume schemes on non-uniform grids [70, 69]. Hence, the *supraconvergence* behaviour is observed with u converging at a higher rate than the truncation error. Comparisons of the finite-difference, finite element, and the exact solution on a random grid of 25 nodes are shown in Figures 51c and 52c.

In 2-d, we consider the Laplace equation with Dirichlet boundary conditions. As a model problem, and to explore the performance of the proposed technique on a non-rectangular domain, we consider an annulus which is bounded between two concentric cylinders of radius $r = 1$ and $r = 2$:

$$-\nabla^2 u(r,\theta) = 0 \quad \text{in} \quad 1 < r < 2 \tag{113a}$$

$$u(1,\theta) = \bar{u}_1(\theta), \; u(2,\theta) = \bar{u}_2(\theta), \tag{113b}$$

which in electrostatics is the model problem for the potential in the presence of a charge distribution in the domain—u is the potential, f is the charge density, and $\mathbf{E} = -\boldsymbol{\nabla} u$ (flux) is the electric field (force per unit charge).

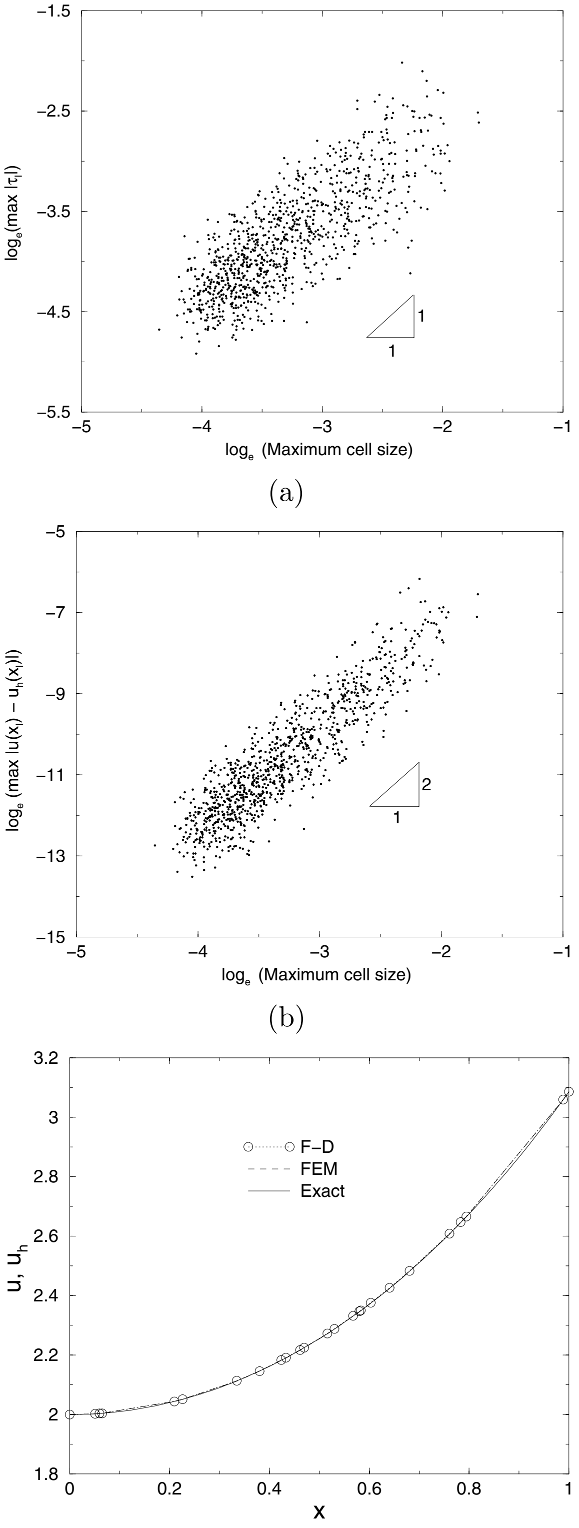


(a)

(b)

(c)

Figure 51. Poisson problem in 1-d (Example I). (a) Convergence of truncation error; (b) Convergence of solution error; and (c) Comparison of numerical and exact solutions



(a)

(b)

(c)

Figure 52. Elliptic problem in 1-d (Example II). (a) Convergence of truncation error; (b) Convergence of solution error; and (c) Comparison of numerical and exact solutions

A convergence study is conducted in 2-d. In the analysis, we define the L^2 norm as:

$$\|u - u_h\|_{2,\Omega} = \sqrt{A_I \sum_{I=1}^{M} \big(u(\mathbf{x}_I) - u_h(\mathbf{x}_I)\big)^2}, \tag{114}$$

where u and u_h are the exact and the finite difference solutions, respectively, and A_I is the area of the Voronoi cell of node I. The discrete L_2 norm in Eq. (114) is defined analogous to its continuous counterpart—$\|u\|_{L_2(\Omega)} = \left(\int_\Omega u^2 \, d\Omega\right)^{1/2}$. Numerical results are presented for two problems:

1. The Laplace equation with the potential a constant on both the inner and the outer boundaries ($\bar{u}_1 = 0$, $\bar{u}_2 = 1$) is considered. Due to symmetry, the Laplace equation reduces to the 1-d radial equation: $u_{rr} + u_r/r = 0$ with the exact solution

$$u(r) = \frac{\ln(r)}{\ln(2)}. \tag{115}$$

2. As a second test, we consider the Laplace equation with boundary conditions that vary with θ: $\bar{u}_1 = 1 + \cos\theta$, and $\bar{u}_2 = 2 + \cos\theta$. The exact solution using separation of variables can be written as:

$$u(r,\theta) = 1 + \frac{\ln(r)}{\ln(2)} + \left[\frac{r}{3} + \frac{2}{3r}\right]\cos\theta. \tag{116}$$

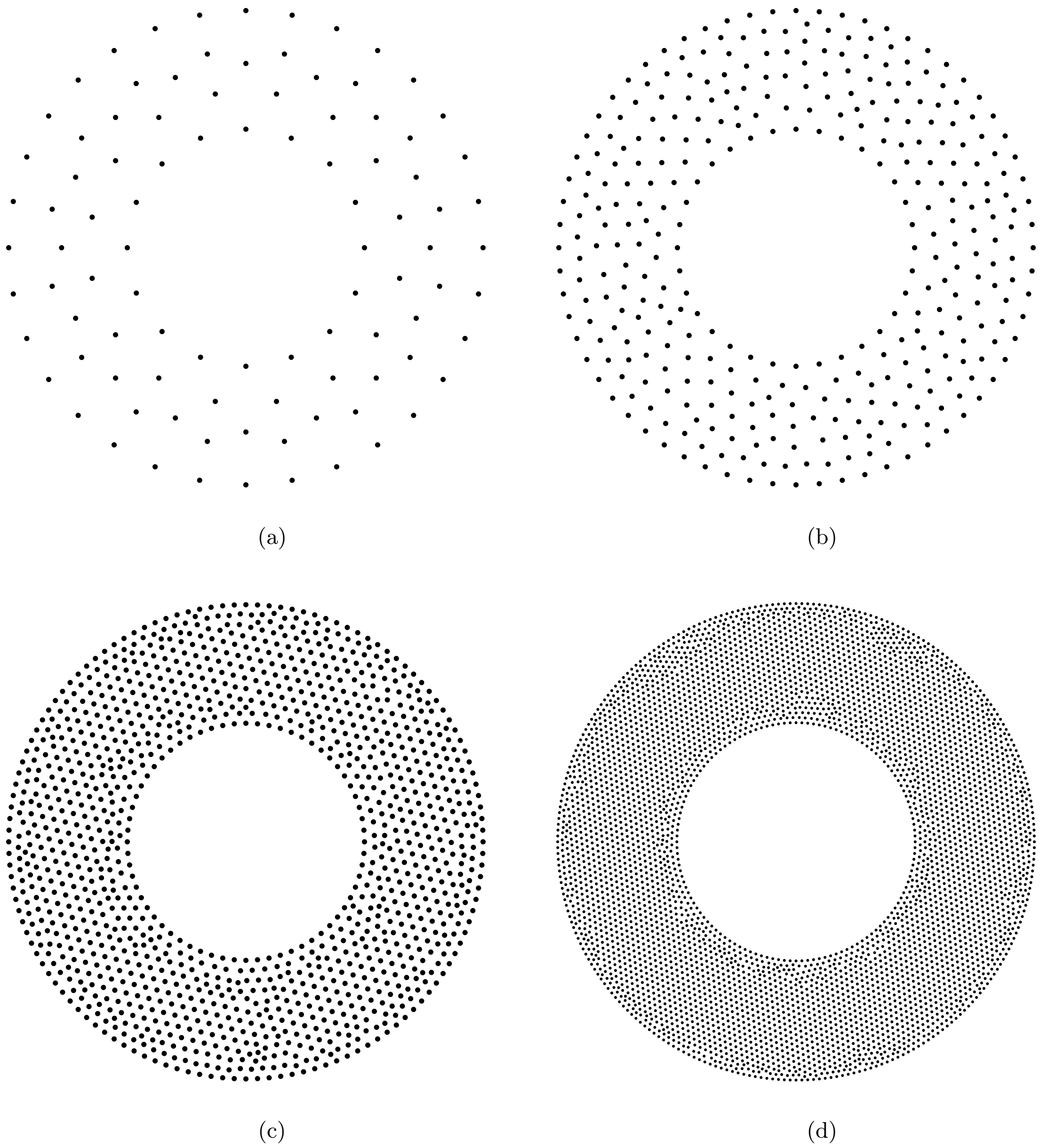
In order to conduct the convergence study, a Delaunay triangulation mesh generator [12] is used so that quasi-uniform grids for a user-specified desired value of h (grid spacing) are obtained. The algorithm is based on comparing an actual local length scale ℓ (e.g., element width, circum-radius) to the desired length scale specified by a scalar variable which is known as the length density function ρ [12]. Four nodal grids are used to carry out a convergence study. In generating the nodal grids, a constant value of $\rho = h$ is specified on both the inner radius ($r = 1$) as well as the outer radius ($r = 2$). The nodal grids used in the analysis are shown in Figure 53 with the average grid spacing $h = \pi/8$, $\pi/16$, $\pi/32$, and $\pi/64$. The relative error in the L_2 norm is plotted in Figure 54, and it is seen that the scheme is second-order accurate.

8 NUMERICAL EXAMPLES

In this section, we focus on two applications with the α-NEM: first, a 2-d simulation of metal cutting in the presence of plasticity and large deformations is carried out; and secondly, we pursue the simulation of complex biological tissues behaviour.

8.1 Simulation of Metal Cutting

As noted for the mould filling simulation example, machining is a process that involves complex physical phenomena (contact, large deformations, heat transfer, etc.). Metal cutting simulations have also been accomplished from a variety of viewpoints: Lagrangian, Eulerian and also Arbitrary Lagrangian Eulerian (ALE) simulations can be found in the literature. For a good review on the state-of-the-art in machining simulations, the interested reader can see [10]. In [30] a review of fixed and moving mesh techniques in such simulations is presented.



(a)

(b)

(c)

(d)

Figure 53. Nodal grids for the Laplace problem in an annulus. (a) 96 nodes; (b) 346 nodes; (c) 1257 nodes; and (d) 4744 nodes

If the Lagrangian (either total or updated) framework with finite elements is chosen to simulate such processes, the need for frequent remeshing becomes necessary. If, on the other hand, an Eulerian framework is preferred (see [30]) advective terms appear as a consequence of the motion and special techniques must be used in order to accurately locate the boundaries of the domain. The use of meshless methods alleviate the burden associated with remeshing to model these phenomena, which permits an updated Lagrangian simulation to be performed in a relatively straightforward manner.

In order to validate the use of natural neighbour Galerkin methods in such problems, we

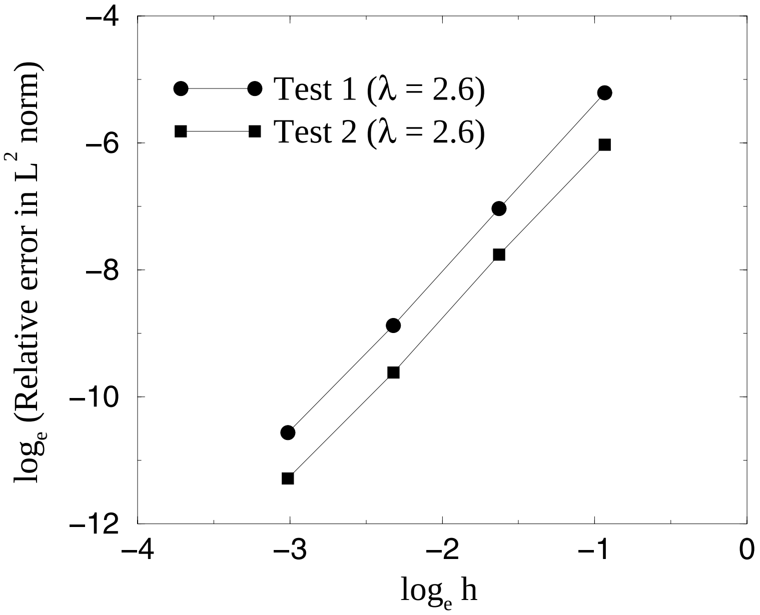


Figure 54. Rate of convergence λ for the 2-d Laplace problem

have defined a benchmark example, in which heat production due to friction is neglected and a Norton-Hoff model of plasticity (viscoplastic flow with zero yield stress) is used. Thus, the equations of the problem can be stated as:

1. Equilibrium equations with no inertial and body forces:

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \mathbf{0}. \tag{117}$$

2. Incompressibility of the material (the whole specimen is in the plastic regime)

$$\boldsymbol{\nabla} \cdot \mathbf{v} = 0, \tag{118}$$

where $\mathbf{v}$ is the velocity field.

3. Generalised Newtonian material behaviour:

$$\boldsymbol{\sigma} = -p\mathbf{Id} + 2\mu(\mathbf{d})\mathbf{d} \tag{119}$$

where p represents the pressure field, $\mathbf{I}$ is the second-order identity tensor, μ is the viscosity that depends on the strain rate tensor $\mathbf{d}$ (non-linear behaviour). The Norton-Hoff plasticity model assumes the following relationship between viscosity and strain rate:

$$\mu(\mathbf{d}) = \mu_0 \Big(\sqrt{2\mathbf{d} : \mathbf{d}} \Big)^{n-1}, \tag{120}$$

where μ_0 is the so-called consistency coefficient and n the pseudoplasticity coefficient which in this case is set to 0.3.

Given the non-linear behaviour of this equation, a Newton-Raphson linearisation scheme is used. Thus, the linearized weak form is: find the kinematically admissible velocity field $\mathbf{v} \in \mathcal{S}$ such that

$$\int_{\Omega(t)} \Big(-\delta p \mathbf{Id} + 2\Big(\frac{\partial \mu(\mathbf{d})}{\partial \mathbf{d}} : \delta\mathbf{d}\Big)\mathbf{d} + 2\mu(\mathbf{d})\delta\mathbf{d}\Big) : \delta\mathbf{d}^* d\Omega = \\ - \int_{\Omega(t)} (-p\mathbf{Id} + 2\mu(\mathbf{d})\mathbf{d}) : (\delta\mathbf{d})^* d\Omega \ \ \forall \delta\mathbf{v}^* \in \mathcal{V} \tag{121}$$

$$\int_{\Omega(t)} \Big(-\nabla(\delta\mathbf{v}) + \varepsilon\delta p\Big)(\delta p^*) d\Omega = 0 \ \ \forall \delta p^* \in L_2(\Omega(t)), \tag{122}$$

where $\mathcal{S} = \{\mathbf{v}|\ \mathbf{v} \in \big(H^1(\Omega(t))\big)^2,\ \mathbf{v}|_{\Gamma_v} = \mathbf{v}_g\}$, $\mathcal{V} = \{\mathbf{v}|\mathbf{v} \in \big(H^1(\Omega(t))\big)^2,\ \mathbf{v}|_{\Gamma_v} = \mathbf{0}\}$, and with Γ_v the portion of the boundary of the domain with prescribed velocities. A penalty parameter ε is introduced to enforce the material incompressibility. Again, we choose a mixed $\mathcal{C}^0 - \mathcal{C}^{-1}$ interpolation for the velocities and the pressure.

The initial geometry consisting of 3197 nodes is shown in Figure 55. The cutting tool, represented by a filled polygon on the right in the same figure, is assumed to be perfectly rigid. Slipping contact is assumed to appear between the tool and the specimen.

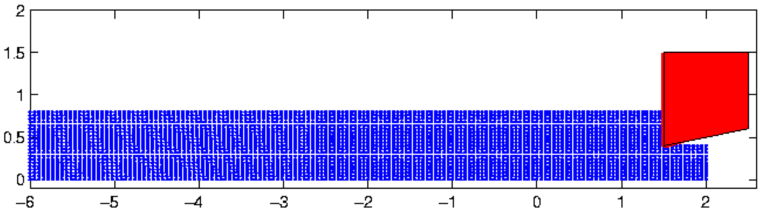


Figure 55. Initial geometry of the cloud of points for the cutting simulation

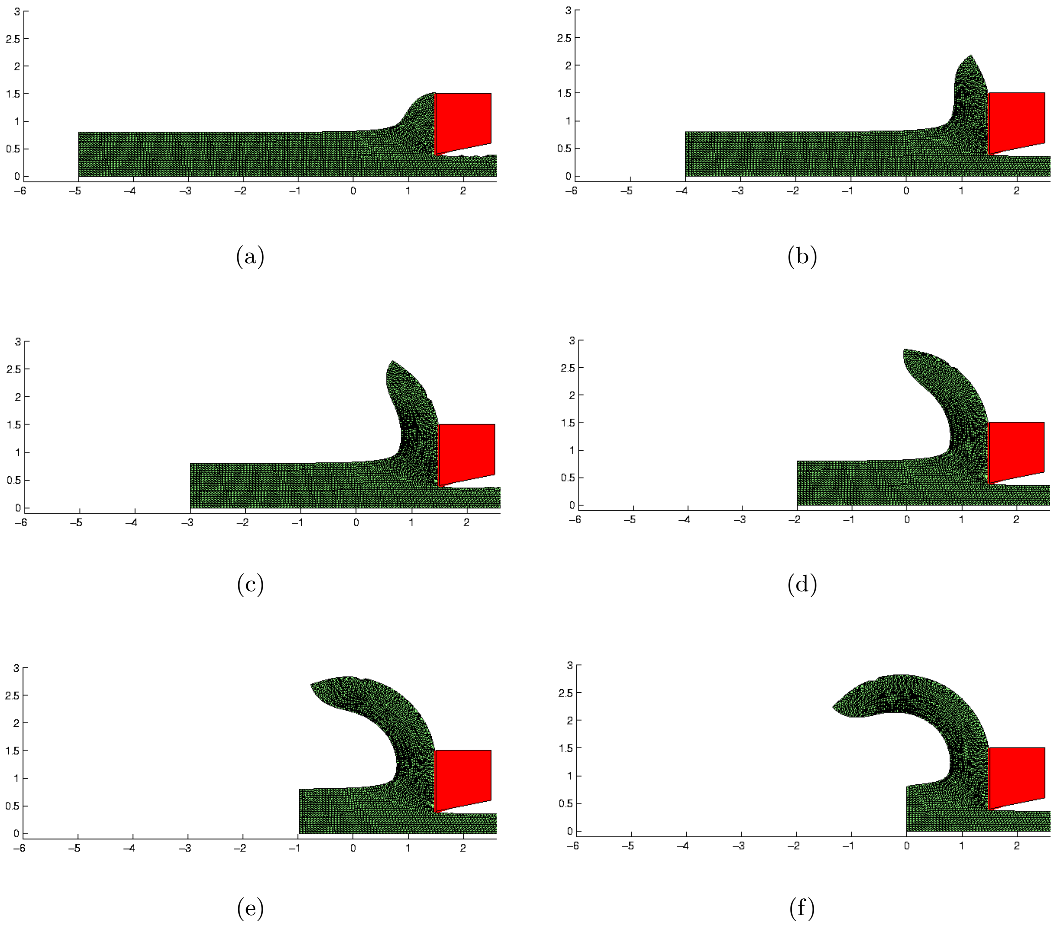
Prescribed displacements are applied to the left side of the specimen, so as to generate a unit displacement in one unit of time. The time increment chosen to solve the problem is $\Delta t = 0.01$. Finally, parameter μ_0 was set to 1.0. The resultant α-shapes of the cutted specimen for time increments $\Delta t = 100, 200, 300, 400, 500, 600$ are depicted in Figure 56.

In Figure 57 the equivalent strain rate (i.e., $\sqrt{(2d_{ij}d_{ij})}$) at an intermediate time step is shown. It can be seen that, per expectations, there is a definite gradient towards the cutting edge.

The predicted shape of the chip is in good agreement with those in the literature for a continuous chip (see [26] for an analysis of the chip geometry). As opposed to many finite element simulations [26, 84], no remeshing is needed throughout the simulation and the initial cloud remains the same without adding or deleting nodes. This is even more important in three-dimensional simulations, where mesh-quality is of utmost importance and the computational cost of meshing becomes a significant component in the overall simulation time.

8.2 Simulation of the Poroelastic Behaviour of the Human Menisci

In this example we show how the use of meshless methods such as natural neighbour Galerkin methods can help to alleviate the burden associated with numerical simulations in biomechanics. In biomechanics, data for numerical simulations is often obtained after a volume reconstruction from Computed Tomography (CT) scans or Magnetic Resonance Images (MRI). In addition, these data are frequently presented in the form of *voxels*, a



(a)

(b)

(c)

(d)

(e)

(f)

Figure 56. α-shapes for time increments 100(a), 200(b), 300(c), 400(d), 500(e) and 600(f) in the cutting simulation

highly structured grid of nodal values. These values are related to the physical properties of the simulation to be performed, such as the bone density in the case of CT scans. Thus, meshing is frequently a very time-consuming task that can be performed only by advanced users. In Garcia-Aznar, Cueto and Doblaré [52] the α-NEM simulation of bone internal remodeling was studied and compared to FE results.

Here we review the simulation of the human menisci behaviour previously presented in [38]. Human soft tissues can be treated as a two-phase material, composed of a solid phase representing around the 20-30% of the tissue weight, and a fluid phase, composed of water and mineral salts. Most of this fluid can be displaced by the application of a pressure gradient, which is critical to the final behaviour of the menisci [88]. Equations describing the articular cartilage behaviour [125] are now briefly introduced.

8.2.1 Problem statement

An incompressible solid (non-miscible phase) and an immiscible and incompressible fluid phase are considered. Volume fractions of the solid and fluid phases are denoted by ϕ^s and ϕ^f, respectively. A Lagrangian description of the motion has been chosen, so that the solid

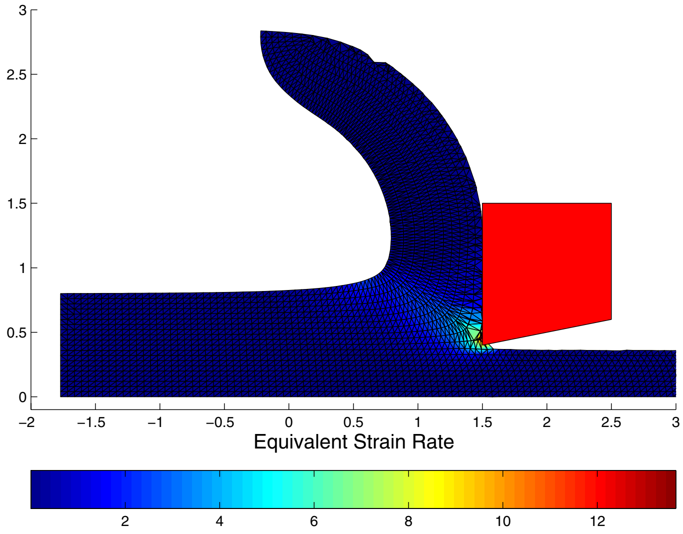


Figure 57. Equivalent strain rate for an intermediate time step in the cutting simulation

phase configuration can be described by

$$\mathbf{x}^s = \varphi^s(\mathbf{X}^s, t) = \mathbf{X}^s + \mathbf{u}^s(\mathbf{X}^s, t) \tag{123}$$

The fluid phase displacement, relative to the solid phase, will be denoted by **w**, where **u** is the motion of the solid phase. This relative displacement can be obtained after the fluid flow as:

$$w_i = \frac{Q}{S \cdot n} \tag{124}$$

where the porosity, n, has been introduced. The equation of equilibrium for the biphasic system is as follows:

$$\sigma_{ij,j} + \rho b_i - \rho \ddot{u}_i - \rho_f \left(\frac{\partial \dot{w}_i}{\partial t} + \dot{w}_k \dot{w_{i,k}} \right) = 0 \tag{125}$$

where b_i are the volumetric body forces and $\ddot{u}_i$ the solid phase acceleration, which is usually neglected. The last term in Eq. (125) takes into account the fluid phase acceleration in Eulerian form, including the convective term.

The fluid phase equation of equilibrium is

$$-p_{,i} + \rho_f b_i - \rho_f \left(\ddot{u}_i + \frac{1}{n} \left(\frac{\partial \dot{w}_i}{\partial t} + \dot{w}_k \dot{w}_{i,k} \right) \right) + \frac{R_i}{n} = 0 \tag{126}$$

where $\frac{R_i}{n} = k_{ij}^{-1} \dot{w}_j$ are the seepage forces that are opposed to the fluid flow, and k_{ij} represents the permeability of the solid, which is usually not a constant.

The rate-dependent constitutive equation, written in terms of solid phase velocities and in an Eulerian description, is generally non-linear, taking the form

$$\dot{\sigma}_{ij} = \dot{\sigma}'_{ij} - \alpha \delta_{ij} \dot{p} = D_{ijkl} \dot{\varepsilon}_{kl} + \dot{w}_{ik} \sigma_{kj} + \dot{w}_{jk} \sigma_{ki} - \alpha \delta_{ij} \dot{p}. \tag{127}$$

In incremental form and in a Lagrangian description, we have

$$d\sigma_{ij} = d\sigma'_{ij} - \alpha\delta_{ij}dp \tag{128}$$

with α a coefficient that depends on the porosity and is close to unity for porous materials, and σ' represents the so-called effective stress.

Finally, the fluid mass conservation law is imposed in the form

$$\frac{\partial W_w}{\partial t} = 0, \tag{129}$$

where W_w is the fluid weight, γ_w its specific weight, and

$$W_w = \int_\Omega n\gamma_w d\Omega. \tag{130}$$

Equivalently,

$$\int_\Omega \Big[\frac{\partial}{\partial t}(n\gamma_w) + \boldsymbol{\nabla}\cdot(n\gamma_w\mathbf{v}')\Big]d\Omega = 0, \tag{131}$$

where $\mathbf{v}'$ is the actual fluid velocity. In local form, it renders

$$\frac{\partial n}{\partial t} + \boldsymbol{\nabla}\cdot\mathbf{v} = 0. \tag{132}$$

If the equation (132) is expressed in terms of volumetric strain and fluid particle motion, we arrive at

$$\dot{\theta} = -\dot{w}_{i,i} = \alpha\dot{\varepsilon}_{ii} + \frac{1}{C}\dot{p} - \delta_{ij}D_{ijkl}\frac{\dot{\varepsilon}^0_{kl}}{3K_s}. \tag{133}$$

The $\alpha\dot{\varepsilon}_{ii}$ term is the solid phase contribution to volumetric strain and the second and third terms usually tend to infinity. The coefficient of compressibility C is given by

$$\frac{1}{C} = \frac{n}{K_f} + \frac{\alpha - n}{K_s}. \tag{134}$$

Equations (123) to (134) form a system of equations whose unknowns are $\mathbf{u}$, $\mathbf{w}$ and p, and they can be solved by taking into account appropriate boundary conditions. Finally, in order to avoid the use of the variable $\mathbf{w}$, a new variable is introduced, namely

$$U_i = u_i + \frac{w_i}{n} \tag{135}$$

which in small displacement theory represents the total fluid displacement with respect to the solid phase, plus that due to the fluid velocity in the pores. To perform the time integration a generalized mid-point rule has been chosen. Thus, the displacement variable is approximated as

$$\mathbf{U}_{n+1} = \mathbf{U}_n + \dot{\mathbf{U}}_{n+\alpha}\Delta t \tag{136}$$

$$\dot{\mathbf{U}}_{n+\alpha} = (1-\alpha)\dot{\mathbf{U}}_n + \alpha\dot{\mathbf{U}}_{n+1}, \tag{137}$$

where $\Delta t > 0$ and $\alpha \in [0,1]$.

8.2.2 Numerical results

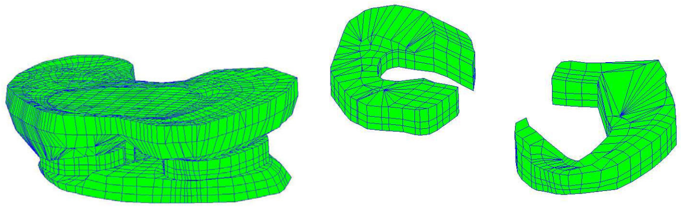
A comparison between the FEM and the NEM is conducted. In these simulations, the *distal* part of the femur, *proximal* part of the tibia and the menisci have been considered. In the FE simulations the whole model has been discretized into finite elements, whereas in the α-NEM simulations, menisci have been discretized by means of a cloud of points, maintaining FE discretisation in the bone. A FE–NEM coupling procedure is adopted. Material properties are described in Table 6. For the menisci, a Poisson's ratio of 0.3, Young's modulus of 80 MPa, permeability coefficient k=2.787 $\times 10^{-10}$ms^{-1} and specific weight of 9.81 $\times 10^3$N/m^3 have been considered.

MATERIAL	Young's mod. (MPa)	Poisson's ratio
BONE (TIBIA)	18500	0.3
BONE (FEMUR)	17500	0.3

Table 6. Material properties for the bone

Displacement boundary conditions with all nodes fixed was applied at the tibia (lower part of the model shown in Figure 58(a)), whereas the load (628.8N) was distributed over the upper part of the femur. This load corresponds to the maximum value achieved during walking by a human male of 75 kg weight. Zero pore pressure at the free boundary is considered as the essential boundary condition on the variable p.

Details of the menisci are shown in Figure 58(b), and the cloud of points for the FE–NEM coupling is shown in Figure 59. In these results the well-known (see, for instance [88]) behaviour of the menisci has been reproduced. It can be seen how the menisci act as a "pillow" in the load transmission from the femur to the tibia.



(a) (b)

Figure 58. FE mesh of a human knee and detail of the menisci

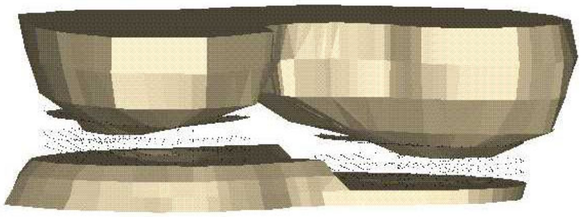


Figure 59. Cloud of points for the FE-NEM coupling in the human knee simulation

First, we present the finite element results. Effective and total stresses and pore pressure are depicted in Figure 60. Pore pressure is depicted at four nodes placed along the meniscus thickness. We observe that the results for the NEM exhibit similar behaviour (see Figure 61) to that obtained with the FEM. Difference of the order of 3–5 percent are obtained, thus demonstrating the potential of the α-NEM in biomechanical simulations. Contour maps of pore pressure at the menisci are shown in Figure 62.

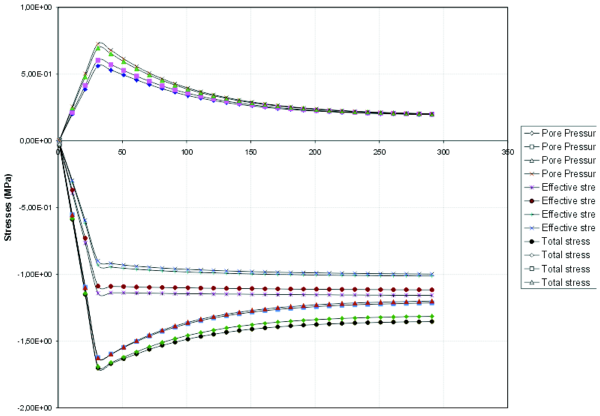


Figure 60. Time variation of stresses at four nodes along the thickness of the meniscus (FE results)

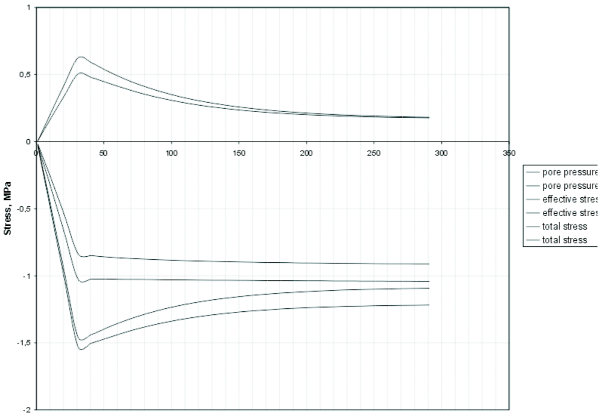


Figure 61. Time variation of stresses at two nodes along the thickness of the meniscus (NE results)

We again emphasize the advantages of using meshless methods in biomechanical numerical simulations. No user-time is required in the mesh generation stage, thus making the process virtually automatic.

9 CONCLUSIONS

Meshless methods have constituted an active field of research during the past decade, and have lead to many new and novel developments within computational mechanics. One of these methods, coined as the natural element method (NEM) or in its generality referred to as natural neighbour Galerkin methods, presents a few distinct and attractive features

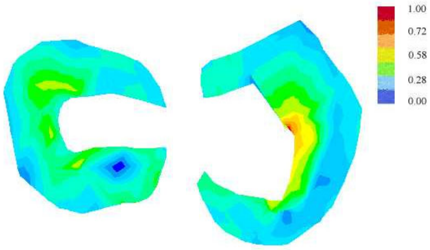


Figure 62. Pore pressure (MPa) contours at the menisci (FE-NE results)

among meshless methods. The review of the NEM and its applications in solid and fluid mechanics has been the subject of this paper.

Natural neighbour Galerkin methods use natural neighbour interpolation (either Sibson or Laplace interpolation) to construct the Galerkin discrete system of equations. These interpolate nodal data and are precisely linear on the boundary, and hence the imposition of essential boundary conditions can be carried out as is done in finite elements This is in contrast to many meshless methods (such as element-free Galerkin method or those based on radial basis functions), in which the interpolating character is absent.

A series of problems were presented to demonstrate the accuracy and capability of natural neighbour Galerkin methods. In particular, they provide an alternative approach to finite elements in large deformation problems. Updated Lagrangian approaches to engineering problems such as mould filling and metal cutting simulations were presented, and the simulations showed the advantages of the NEM.

An analysis of the performance of three distinct numerical integration methods was performed. From the traditional Delaunay triangles-based integration that lead to significant loss in accuracy for the patch test with nodal refinement, to stabilized nodal integration that show superior performance, a means to carry out accurate numerical integration was presented. The use of the Laplace interpolant in the development of a finite difference scheme for the diffusion equation was also described. In conclusion, it is the authors' belief that natural neighbour-based techniques provide an appealing choice for many engineering problems and are a potential alternative to finite elements as well as some of the other meshless methods in computer modelling and simulation of complex phenomena in solid and fluid mechanics.

ACKNOWLEDGMENTS

N. S. is grateful to Dr. David Watson and Dr. Andrei Yu Semenov for helpful discussions on the Sibson and Laplace interpolants, respectively. N. S. also thanks Dr. Tim Baker for providing the Delaunay triangulation code, which was used to generate the grids considered in the convergence study for the finite-difference scheme. E. C. acknowledges the help and guidance provided by Professor Francisco Chinesta in the development of the fluid flow examples presented in this paper.

References

1 N. R. Aluru (2000). A point collocation method based on reproducing kernel approximations. *International Journal for Numerical Methods in Engineering*, **47**, 1083–1121.

2 N. R. Aluru and G. Li (2001). Finite cloud method: A true meshless technique based on a fixed reproducing kernel approximation. *International Journal for Numerical Methods in Engineering*, **50**(**10**), 2373–2410.

3 Nina Amenta, Marsahll Bern and Manolis Kamvysselis (1998). A new Voronoi-based surface reconstruction algorithm. In *Siggraph 98*, 415–421.

4 Nina Amenta, Marshall Bern and David Eppstein (1998). The crust and the beta-skeleton: combinatorial curve reconstruction. *Graphical Models and Image Processing*, **60**/2:2, 125–135.

5 H. Askes, J. Pamin and R. de Borst (2000). Dispersion analysis and element-free Galerkin solutions of second- and fourth-order gradient-enhanced damage models. *International Journal for Numerical Methods in Engineering*, **49**(**6**), 811–832.

6 S. N. Atluri and T. Zhu (1998). A new meshless local Petrov-Galerkin (MLPG) approach to nonlinear problems in computer modeling and simulation. *Computer Modeling and Simulation in Engineering*, **3**, 187–196.

7 S.N. Atluri, H. G. Kim and J. Y. Cho (1999). A critical assesment of the truly meshless local Petrov-Galerkin and local boundary integral equation methods. *Computational Mechanics*, **24**, 348–372.

8 S.N. Atluri and T. Zhu (2000). New concepts in meshless methods. *International Journal for Numerical Methods in Engineering*, **47**, 537–556.

9 F. Aurenhammer (1996). Voronoi diagrams – a survey of a fundamental geometric data structure. *ACM Transactions on Mathematical Software*, **23**, 469–483.

10 Various authors (2000). Special issue on machining simulation. *International Journal of Forming Processes*, **3**(**1-2**).

11 I. Babuška and J. M. Melenk (1997). The partition of unity method. *International Journal for Numerical Methods in Engineering*, **40**, 727–758.

12 T. J. Baker (1994). Triangulations, mesh generation and point placement strategies. In D. A. Caughey and M. M. Hafez, editors, *Frontiers of Computational Fluid Dynamics*, pages 101–115, New York, John Wiley & Sons.

13 P. D. Beale and D. J. Srolovitz (1988). Elastic fracture in random materials. *Physical Review B*, **37**(**10**), 5500–5507.

14 V. V. Belikov, V. D. Ivanov, V. K. Kontorovich, S. A. Korytnik and A. Yu. Semenov (1997). The non-Sibsonian interpolation: A new method of interpolation of the values of a function on an arbitrary set of points. *Computational Mathematics and Mathematical Physics*, **37**(**1**), 9–15.

15 V. V. Belikov and A. Yu. Semenov (1997). New non-Sibsonian interpolation on arbitrary system of points in Euclidean space. In *15th IMACS World Congress, Numerical Mathematics*, Volume 2, 237–242, Wissen Tech. Verlag, Berlin.

16 V. V. Belikov and A. Yu. Semenov (2000). Non-Sibsonian interpolation on arbitrary system of points in Euclidean space and adaptive isolines generation. *Applied Numerical Mathematics*, **32**(**4**), 371–387.

17 T. Belytschko, Y. Krongauz, D. Organ, M. Fleming and P. Krysl (1996). Meshless methods: An overview and recent developments. *Computer Methods in Applied Mechanics and Engineering*, **139**, 3–47.

18 T. Belytschko, W. K. Liu and B. Moran (2000). *Finite Elements for Nonlinear Continua and Structures*. John Wiley and Sons, New York.

19 T. Belytschko, Y. Y. Lu and L. Gu (1994). Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering*, **37**, 229–256.

20 J.-D. Boissonnat and F. Cazals (2001). Natural neighbour coordinates of points on a surface. *Computational Geometry: Theory and Applications*, **19**(**2–3**), 155–173.

21 J. E. Bolander, Jr and S. Berton (2002). Shrinkage induced cracking in cement composite overlays. In H. A. Mang, F. G. Rammerstorfer, and J. Eberhardsteiner, editors, *Proceedings of the Fifth World Congress on Computational Mechanics*, Vienna, Austria.

22 J. E. Bolander, Jr and S. Saito (1998). Fracture analyses using spring networks with random geometry. *Engineering Fracture Mechanics*, **61**, 569–591.

23 A. Bowyer (1981). Computing Dirichlet tessellations. *Computer J.*, **24**, 162–166.

24 J. Braun and M. Sambridge (1995). A numerical method for solving partial differential equations on highly irregular evolving grids. *Nature*, **376**, 655–660.

25 P. Breitkopf, G. Touzot and P. Villon (2000). Double grid diffuse collocation method. *Computational Mechanics*, **25**(**2/3**), 199–206.

26 E. Ceretti, L. Filice and F. Micari (2000). Analysis of the chip geometry in orthogonal cutting of mild steel. In *4th international ESAFORM conference on Material Forming*, Liege, Belgium.

27 B. Chazelle and L. Palios (1990). Triangulating a non-convex polyhedron. *Disc. and Comput. Geometry*, **5**, 505–526.

28 J. S. Chen, C. T. Wu, S. Yoon and Y. You (2001). A stabilized conforming nodal integration for Galerkin meshfree methods. *International Journal for Numerical Methods in Engineering*, **50**, 435–466.

29 J.-S. Chen, C.-T. Wu, S. Yoon and Y. You (2002). Non-linear version of stabilized conforming nodal integration for galerkin meshfree methods. *International Journal for Numerical Methods in Engineering*, **53**(**12**), 2587–2615.

30 F. Chinesta, Ph. Lorong, D. Ryckelinck, G. Coffignal, M. Tourantier, M. A. Martínez, E. Cueto and M. Doblaré (2002). Thermomechanical cutting model discretisation: Eulerian or Lagrangian, mesh or meshless? In *Esaform Conference proceedings*, Krakow.

31 N. H. Christ, R. Friedberg and T. D. Lee (1982). Gauge-theory on a random lattice. *Nuclear Physics B*, **210**(**3**), 310–336.

32 N. H. Christ, R. Friedberg and T. D. Lee (1982). Random lattice field-theory – general formulation. *Nuclear Physics B*, **202**(**1**), 89–125.

33 N. H. Christ, R. Friedberg and T. D. Lee (1982). Weights of links and plaquettes in a random lattice. *Nuclear Physics B*, **210**(**3**), 337–346.

34 I. Christie and C. Hall (1984). The maximum principle for bilinear elements. *International Journal for Numerical Methods in Engineering*, **20**, 549–553.

35 Y. A. Chu and B. Moran (1995). A computational model for nucleation of solid-solid phase transformations. *Modelling and Simulation in Materials Science and Engineering*, **3**, 455–471.

36 R. Clough and J. L. Tocher (1965). Finite element stiffness matrices for analysis of plates in bending. In *Proceedings of the 1st Conference on Matrix Methods in Structural Mechanics*, Wright-Patterson AFB.

37 E. Cueto, B. Calvo and M. Doblaré (2002). Modeling three-dimensional piece-wise homogeneous domains using the α-shape based Natural Element Method. *International Journal for Numerical Methods in Engineering*, **54**, 871–897.

38 E. Cueto, J. Cegoñino, B. Calvo and M. Doblaré (2003). On the imposition of essential boundary conditions in Natural Neighbour Galerkin methods. *Communications in Numerical Methods in Engineering*, **19** (**5**), 361–376.

39 E. Cueto, M. Doblaré and L. Gracia (2000). Imposing essential boundary conditions in the Natural Element Method by means of density-scaled α-shapes. *International Journal for Numerical Methods in Engineering*, **49**(**4**), 519–546.

40 S. De and K. J. Bathe (2000). The method of finite spheres. *Computational Mechanics*, **25**, 329–345.

41 S. De and K. J. Bathe (2001). Towards an efficient meshless computational technique: the method of finite spheres. *Engineering Computations*, **18**, 170–192.

42 C. de Boor (1987). B-form basics. In G. Farin, editor, *Geometric Modeling: Algorithm and New Trends*, pages 131–148, SIAM, Philadelphia, PA.

43 C. de Veubeke (1968). A conforming finite element for plate bending. *International Journal for Solids and Structures*, **4**, 95–108.

44 J. Dolbow and T. Belytschko (1999). Numerical Integration of the Galerkin Weak Form in Meshfree Methods. *Computational Mechanics*, **23**, 219–230.

45 C. A. Duarte and J. T. Oden (1996). An *H-p* adaptive method using clouds. *Computer Methods in Applied Mechanics and Engineering*, **139**, 237–262.

46 H. Edelsbrunner, D. G. Kirkpatrick and R. Seidel (1983). On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, **IT-29**(**4**), 551–559.

47 H. Edelsbrunner and E. Mücke (1994). Three dimensional alpha shapes. *ACM Transactions on Graphics*, **13**, 43–72.

48 G. Farin (1986). Triangular Bernstein-Bézier patches. *Computer Aided Geometric Design*, **3**, 83–127.

49 G. Farin (1990). *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*. Academic Press, New York, second edition.

50 G. Farin (1990). Surfaces over Dirichlet tessellations. *Computer Aided Geometric Design*, **7**(**1–4**), 281–292.

51 R. Friedberg and M. Ruiz (1984). Vector algebra on a lattice. *Physical Review D*, **29**(**12**), 2916–2918.

52 J. M. García, E. Cueto and M. Doblaré (2000). Simulation of bone internal remodeling by means of the alpha-shapes-based natural element method. In *ECCOMAS Conference*, Barcelona.

53 L. Gavete, J. J. Benito, S. Falcón and A. Ruiz (2000). Penalty functions in constrained variational principles for element free Galerkin method. *European Journal of Mechanics. A/ Solids*, **19**, 699–720.

54 S. Ghosh and S. Moorthy (1995). Elastic-plastic analysis of arbitrary heterogeneous materials with the Voronoi cell finite-element method. *Computer Methods in Applied Mechanics and Engineering*, **121**(**1–4**), 373–409.

55 V. Girault (1974). Theory of a finite difference method on irregular networks. *SIAM Journal on Numerical Analysis*, **11**(**2**), 260–282.

56 B. Grünbaum (1967). *Convex Polytopes.* John Wiley and Sons, New York.

57 S. C. Harris, D. Williams and R. Sibson (1999). Scaling random walks on arbitrary sets. *Mathematical Proceedings of the Cambridge Philosophical Society*, **125**(**3**), 535–544.

58 L. R. Hermann (1965). Elasticity equations for incompressible and nearly incompressible materials by a variational theorem. *AIAA Journal*, **163**, Nos. **1-4**, 1896–1900.

59 H. Hiyoshi and K. Sugihara (1999). Two generalizations of an interpolant based on Voronoi diagrams. *International Journal of Shape Modeling*, **5**(**2**), 219–231.

60 H Hiyoshi and K. Sugihara (2000). Voronoi-based interpolation with higher continuity. In *Proceedings of the 16th Annual ACM Symposium on Computational Geometry*, pages 242–250.

61 H. Hiyoshi and K. Sugihara (2002). Improving continuity of Voronoi-based interpolation over Delaunay spheres. *Computational Geometry*, **22**, 167–183.

62 A. Huerta and S. Fernández-Méndez (2000). Enrichment and coupling of the finite element and meshless methods. *International Journal for Numerical Methods in Engineering*, **48**, 1615–1636.

63 T.J.R. Hughes (1987). *The Finite Element Method.* Prentice-Hall, New York.

64 B. Irons (1969). A conforming quartic triangular element for plate bending. *International Journal for Numerical Methods in Engineering*, **1**, 29–45.

65 B. Irons and A. Razzaque (1972). Experience with the patch test for convergence of finite elements. In A. K. Aziz, editor, *The mathematical Foundations of the Finite Element Method with Applications to Partial Differential Equations*, New York. Academic Press.

66 A. Jagota and S. J. Bennison (1994). Spring-network and finite-element models for elasticity and fracture. In K. K. Bardhan, B. K. Chakrabarti, and A. Hansen, editors, *Nonlinearity and Breakdown in Soft Condensed Matter. (Springer Lecture Notes in Physics 437)*, pages 186–201, Springer, Berlin.

67 P. S. Jensen (1972). Finite difference techniques for variable grids. *Computers and Structures*, **2**, 17–29.

68 N. L. Jones, S. J. Owens and E. C. Perry (1995). Plume characterization with natural neighbor interpolation. In *Proceedings GEOENVIRONMENT 2000, Geotechnical Engineering and Environmental Engineering Divisions/ASCE*, pages 331–345, New York, N.Y.

69 W. P. Jones and K. R. Menzies (2000). Analysis of the cell-centred finite volume method for the diffusion equation. *Journal of Computational Physics*, **165**, 45–68.

70 H.-O Kreiss, T. A. Manteuffel, B. Swartz, B. Wendroff and A. B. White, Jr. (1986). Supraconvergent schemes on irregular grids. *Mathematics of Computation*, **47**(**176**), 537–554.

71 Y. Krongauz (1996). *Application of Meshless Methods to Solid Mechanics.* PhD thesis, Northwestern University, Evanston, IL.

72 P. Krysl and T. Belytschko (1996). Analysis of thin plates by the element-free Galerkin method. *Computational Mechanics*, **17**, 26–35.

73 P. Lancaster and K. Salkauskas (1981). Surfaces generated by moving least squares methods. *Mathematics of Computation*, **37**, 141–158.

74 J. B. Lasserre (1983). An analytical expression and an algorithm for the volume of a convex polyhedron in $\mathbb{R}^n$. *Journal of Optimization Theory and Applications*, **39**(**3**), 363–377.

75 C. L. Lawson (1977). Software for C^1 surface interpolation. In *J. R. Rice (Ed.) Mathematical Software III. Vol. 3.* Academic Press, N.Y.

76 R. W. Lewis and K. Ravindran (2000). Finite element simulation of metal casting. *International Journal for Numerical Methods in Engineering*, **47**, 29–59.

77 S. Li and W. K. Liu (2002). Meshfree and particle methods and their applications. *Applied Mechanics Review*, **55**(**1**), 1–34.

78 T. Liszka and J. Orkisz (1980). The finite difference method at arbitrary irregular grids and its application in applied mechanics. *Computers and Structures*, **11**, 83–950.

79 W. K. Liu, S. Jun and Y. F. Zhang (1995). Reproducing kernel particle methods. *International Journal for Numerical Methods in Engineering*, **20**, 1081–1106.

80 Charles T. Loop and Tony D. DeRose (1989). A multisided generalization of Bézier surfaces. *ACM Transactions on Graphics*, **8**(**3**), 204–234.

81 T. A. Manteuffel and A. B. White, Jr. (1986). The numerical solution of second-order boundary value problems on nonuniform meshes. *Mathematics of Computation*, **47**(**176**), 511–535.

82 M. A. Martínez, E. Cueto, M. Doblaré and F. Chinesta (2001). A meshless simulation of injection processes involving short fibers molten composites. *International Journal of Forming Processes*, **4**(**3-4**), 217–236.

83 M. A. Martínez, E. Cueto, M. Doblaré and F. Chinesta (2003). Fixed mesh and meshfree techniques in the numerical simulation of injection processes involving short fiber suspensions. *Journal for Non Newtonian Fluid Mechanics*, **115**, 51–78.

84 T. Marusich and M. Ortiz (1995). Modeling and simulation of high-speed machinning. *International Journal for Numerical Methods in Engineering*, **38**(**21**), 3675–3694.

85 J. M. Melenk and I. Babuška (1996). The partition of unity finite element method: Basic theory and applications. *Computer Methods in Applied Mechanics and Engineering*, **139**, 289–314.

86 I. D. Mishev (1998). Finite volume methods on Voronoi meshes. *Numerical Methods for Partial Differential Equations*, **14**, 193–212.

87 J. J. Monaghan (1988). An introduction to SPH. *Computer Physics Communications*, **48**, 89–96.

88 C. V. Mow, S. C. Kuei, W. M. Lai and C. G. Armstrong (1980). Biphasic creep and stress relaxation of articular cartilage in compression: theory and experiments. *Journal of Biomechanical Engineering*, **102**, 73–84.

89 Y. X. Mukherjee and S. Mukherjee (1997). The boundary node method for potential problems. *International Journal for Numerical Methods in Engineering*, **40**(**5**), 797–815.

90 T. N. Narasimhan and P. A. Witherspoon (1976). An integrated finite difference method for analyzing fluid flow in porous media. *Water Resources Research*, **12**(**1**), 57–64.

91 B. Nayroles, G. Touzot and P. Villon (1992). Generalizing the finite element method: Diffuse approximation and diffuse elements. *Computational Mechanics*, **10**, 307–318.

92 E. Oñate, S. Idelsohn, O. C. Zienkiewicz, R. L. Taylor and C. Sacco (1996). A stabilised finite point method for analysis of fluid mechanics problems. *Computer Methods in Applied Mechanics and Engineering*, **139**, 315–346.

93 S. J. Owens (1992). An implementation of natural neighbor interpolation in three dimensions. Master's thesis, Brigham Young University.

94 N. B. Petrovskaya (2001). Modification of a finite volume scheme for Laplace's equation. *SIAM Journal on Scientific Computing*, **23**(**3**), 891–909.

95 P. Piper (1993). Properties of local coordinates based on Dirichlet tessellations. In G. Farin, H. Hagen, and H. Noltemeier, editors, *Geometric Modelling*, volume 8, pages 227–239, Wien New York. Springer-Verlag.

96 F. Preparata and M. Shamos (1985). *Computational Geometry: An Introduction*. Springer-Verlag, New York.

97 J. Rupert and R. Seidel (1992). On the difficulty of tetrahedralizing three-dimensional non-convex polyhedra. *Disc. and Comput. Geometry*, **7**, 227–253.

98 M. Sambridge, J. Braun and H. McQueen (1995). Geophysical parameterization and interpolation of irregular data using natural neighbors. *Geophysical Journal International*, **122**, 837–857.

99 M. S. Sambridge, J. Braun and H. McQueen (1996). Computational methods for performing natural neighbor interpolation in two and three dimensions. In R. L. May and A. K. Easton, editors, *Proceedings of the Seventh Biennial Conference on Computational Techniques and Applications (CTAC95)*, pages 685–692.

100 E. Schönhardt (1928). Über die zerlegung von dreieckspolyedern in tetraeder. *Math. Annalen*, **98**.

101 D. Shepard (1968). A two-dimensional interpolation function for irregularly spaced points. In *ACM National Conference*, pages 517–524.

102 J. R. Shewchuk (2000). Sweep Algorithms for constructing higher-dimensionnal Constrained Delaunay Triangulations. In *Proceedings of the Sixteenth Annual Symposium on Computational Geometry, Hong Kong*. Association for computing Machinery.

103 R. Sibson (1980). A vector identity for the Dirichlet tesselation. *Mathematical Proceedings of the Cambridge Philosophical Society*, **87**, 151–155.

104 R. Sibson (1981). A brief description of natural neighbor interpolation. In V. Barnett, editor, *Interpreting Multivariate Data*, pages 21–36, Chichester. John Wiley.

105 K. Sugihara (1999). Surface interpolation based on new local coordinates. *Computer-Aided Design*, **31**, 51–58.

106 N. Sukumar (1998). *The Natural Element Method in Solid Mechanics*. Ph.D. thesis, Theoretical and Applied Mechanics, Northwestern University, Evanston, IL, U.S.A., June.

107 N. Sukumar (2001). Sibson and non-Sibsonian interpolants for elliptic partial differential equations. In K. J. Bathe, editor, *Proceedings of the first MIT Conference on Fluid and Solid Mechanics*, volume 2, pages 1665–1667, Amsterdam, The Netherlands, Elsevier Press.

108 N. Sukumar (2002). Voronoi cell finite difference method for the diffusion operator on arbitrary unstructured grids. *International Journal for Numerical Methods in Engineering*, accepted for publication.

109 N. Sukumar (2003). Voronoi cell finite difference method for the diffusion operator on arbitrary unstructured grids. *International Journal for Numerical Methods in Engineering*, **57**(**1**), 1–34.

110 N. Sukumar, B. Moran and T. Belytschko (1998). The natural element method in solid mechanics. *International Journal for Numerical Methods in Engineering*, **43**(**5**), 839–887.

111 N. Sukumar, B. Moran, A. Yu. Semenov and V. V. Belikov (2001). Natural neighbor Galerkin methods. *International Journal for Numerical Methods in Engineering*, **50**(**1**), 1–27.

112 D. Sulsky, Z. Chen and H. L. Schreyer (1994). A particle method for history-dependent materials. *Computer Methods in Applied Mechanics and Engineering*, **118**, 179–186.

113 B. Szabo and I. Babuška (1991). *Finite Element Analysis*. John Wiley & Sons.

114 M. Teichmann and M. Capps (1998). Surface reconstruction with anisotropic density-scaled alpha shapes. In *Proceedings of the 1998 IEEE Visualization Conference*.

115 J. W. Thomas (1995). *Numerical Partial Differential Equations: Finite Difference Methods*. Springer-Verlag, New York.

116 A. N. Tikhonov and A. A. Samarskii (1962). Homogeneous difference schemes on non-uniform nets. *U.S.S.R Comput. Math. and Math. Phys.*, **2**, 927–953.

117 S. Timoshenko and J. N. Goodier (1972). *Teoría de la Elasticidad*. Editorial Urmo.

118 L. Traversoni (1994). Natural neighbor finite elements. In *International Conference on Hydraulic Engineering Software, Hydrosoft Proceedings*, volume 2, pages 291–297. Computational Mechanics Publications.

119 D. F. Watson (1981). Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes. *The Computer Journal*, **24**(**2**), 167–172.

120 D. F. Watson (1992). *Contouring: A Guide to the Analysis and Display of Spatial Data*. Pergamon Press, Oxford.

121 D. F. Watson (1994). *nngridr: An implementation of natural neighbor interpolation*. David Watson.

122 D. F. Watson and G. M. Philip (1987). Neighborhood-based interpolation. *Geobyte*, **2**(**2**), 12–16.

123 A. Weiser and M. F. Wheeler (1998). On convergence of block-centered finite differences for elliptic problems. *SIAM Journal on Numerical Analysis*, **25**(**2**), 351–375.

124 J. Yoo, B. Moran and J.-S. Chen (2003). Nodal natural neighbor methods. *International Journal for Numerical Methods in Engineering*, in press.

125 O. C. Zienkiewicz, C. Humpheson and W. Lewis (1977). A unified approach to soil mechanics problems (including plasticity and viscoplasticity). *Finite Elements in Geomechanics. G. Gudehus Ed.*, pages 151–178.

Please address your comments or questions on this paper to:
International Center for Numerical Methods in Engineering
Edificio C-1, Campus Norte UPC
Grand Capitán s/n
08034 Barcelona, Spain
Phone: 34-93-4016035; Fax: 34-93-4016517
E-mail: onate@cimne.upc.es