

Sibson and non-Sibsonian interpolants for elliptic partial differential equations

N. Sukumar*

Princeton Materials Institute, Princeton University, Princeton, NJ 08544, USA

Abstract

The Natural Element Method (NEM) is a meshless Galerkin method which has shown promise in the area of computational mechanics. In earlier applications of NEM [1–3], natural neighbor (Sibson) coordinates [4] were used to construct the trial and test functions. Recently, Belikov and co-workers [5] proposed a new interpolation scheme (non-Sibsonian interpolation) based on natural neighbors. In this paper, we present the Sibson and the non-Sibsonian interpolants, and discuss their use in a Galerkin scheme for the solution of elliptic PDEs. In particular, by choosing the non-Sibsonian interpolant, the exact imposition of essential boundary conditions in a meshless method is realized.

Keywords: Meshless methods; Natural element method; Non-Sibsonian interpolation; Natural neighbor interpolation; Essential boundary conditions

1. Introduction

In the natural element method [1], Sibson coordinates [4] are used to construct the trial and test functions. Recent studies on NEM [2,3] have shown that natural neighbor coordinates, which are based on the Voronoi tessellation of a set of nodes, are an appealing choice to construct meshless interpolants. In [5], a new interpolation scheme (non-Sibsonian interpolation) based on natural neighbors was proposed. In this paper, we discuss the construction and use of the Sibson and the non-Sibsonian interpolation schemes for the solution of elliptic PDEs. Two significant advantages of the non-Sibsonian interpolant over the Sibsonian interpolant are its computational efficiency in 2 dimensions, which is expected to carry over to 3 dimensions, and the ability to exactly impose essential boundary conditions on the boundaries of convex and non-convex domains [6].

2. Sibson and non-Sibsonian interpolation

2.1. C^0 shape function

Natural neighbors and natural neighbor (Sibson) interpolation was introduced by Sibson [4] as a means for data fitting and smoothing. Consider a set of distinct nodes $\mathbf{N} = \{n_1, n_2, \dots, n_M\}$ in \mathbb{R}^2 . The Voronoi diagram of the set \mathbf{N} is a sub-division of the plane into regions T_I , such that any point in T_I is closer to n_I (nearest neighbor) than to any other node $n_J \in \mathbf{N}$ ($J \neq I$). The region T_I is the Voronoi cell of n_I :

$$T_I = \{\mathbf{x} \in \mathbb{R}^2 : d(\mathbf{x}, \mathbf{x}_I) < d(\mathbf{x}, \mathbf{x}_J) \forall J \neq I\}, \quad (1)$$

where $d(\mathbf{x}_I, \mathbf{x}_J)$ is the Euclidean distance between \mathbf{x}_I and \mathbf{x}_J . The Delaunay triangulation is the geometric dual of the Voronoi diagram (Fig. 1a). If we insert a point \mathbf{x} into the set \mathbf{N} , then the natural neighbors of the point \mathbf{x} are defined as follows. If the point \mathbf{x} lies within the circumcircle of a Delaunay triangle t with vertices n_I – n_J – n_K , then n_I , n_J , and n_K are natural neighbors of \mathbf{x} . Natural neighbor (Sibson) shape function is based on the first- and second-order Voronoi diagram and is defined by the ratio of polygonal areas (polyhedral volumes) in 2 dimensions (3 dimensions): $\phi_I(\mathbf{x}) = A_I(\mathbf{x})/A(\mathbf{x})$, where $A(\mathbf{x})$ is the area of the first-order Voronoi cell about \mathbf{x} and $A_I(\mathbf{x})$ is the area

* Tel.: +1 (609) 258-0494; Fax: +1 (609) 258-6878; E-mail: nsukumar@princeton.edu

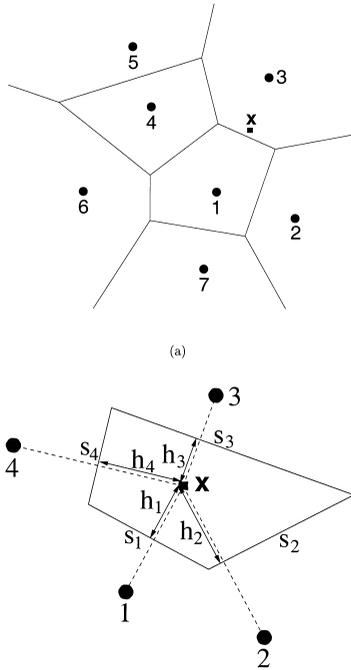


Fig. 1. Construction of natural neighbors and the non-Sibsonian interpolant: (a) original Voronoi diagram and \mathbf{x} , and (b) first-order Voronoi cell of point \mathbf{x} .

of overlap between the first-order Voronoi cell about node I and $A(\mathbf{x})$. For details on the construction and properties of the Sibson interpolant, the interested reader can see [1,2,4].

In addition to Eq. (1), define $t_{IJ} = \{\mathbf{x} \in \bar{T}_I \cap \bar{T}_J, J \neq I\}$, $\bar{T}_I = T_I \cup \partial T_I$ is the closure of set T_I , and t_{IJ} may be an empty set. If $d(\mathbf{x}_I, \mathbf{x}_J) \neq 0$, then [5]

$$\sum_{J \neq I} \frac{|t_{IJ}| |\mathbf{x}_J|}{d(\mathbf{x}_I, \mathbf{x}_J)} = \mathbf{x}_I \left[\sum_{J \neq I} \frac{|t_{IJ}|}{d(\mathbf{x}_I, \mathbf{x}_J)} \right], \quad (2)$$

where $|\cdot|$ denotes the Lebesgue measure (length) in \mathbb{R} . In terms of the notation used above, the non-Sibsonian shape function $\phi_I(\mathbf{x})$ is defined as

$$\phi_I(\mathbf{x}) = \frac{\frac{|t_{xI}|}{d(\mathbf{x}, \mathbf{x}_I)}}{\sum_{J=1}^n \frac{|t_{xJ}|}{d(\mathbf{x}, \mathbf{x}_J)}}. \quad (3)$$

In 2 dimensions, the non-Sibsonian shape function $\phi_I(\mathbf{x})$ takes the form:

$$\phi_I(\mathbf{x}) = \frac{\alpha_I(\mathbf{x})}{\sum_{J=1}^n \alpha_J(\mathbf{x})}, \quad \alpha_J(\mathbf{x}) = \frac{s_J(\mathbf{x})}{h_J(\mathbf{x})}, \quad (4)$$

where $s_I(\mathbf{x})$ is the length of the Voronoi edge associated with node I , and $h_I(\mathbf{x})$ is the perpendicular distance between the Voronoi edge of node I to \mathbf{x} (Fig. 1b). In Fig. 2, the non-Sibsonian shape function is shown for the center

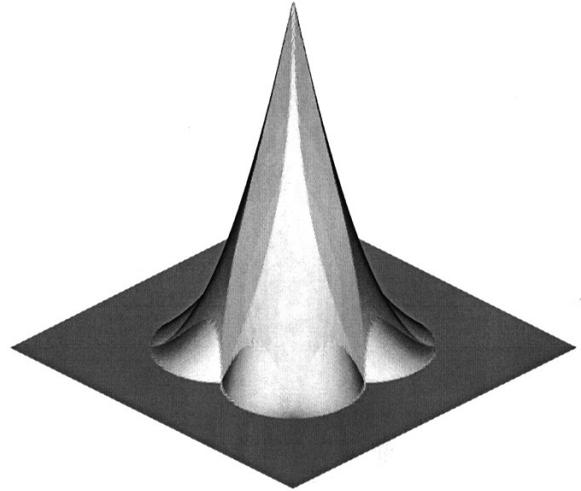


Fig. 2. Non-Sibsonian shape function.

node in a 5×5 regular grid. For a scalar-valued function $u(\mathbf{x})$, the interpolation scheme using the non-Sibsonian shape function takes the form:

$$u^h(\mathbf{x}) = \sum_{I=1}^n \phi_I(\mathbf{x}) u_I. \quad (5)$$

The computational complexity of non-Sibsonian shape function in 2 dimensions depends only on the ratio of length measures, whereas the Sibson shape function is based on the ratio of polygonal areas. The computations becomes even more favorable and tractable in 3 dimensions for the non-Sibsonian interpolant. The non-Sibsonian shape functions are strictly positive, interpolate nodal data, and are also linearly complete [5].

The Sibsonian interpolant is precisely linear on the boundary of convex domains, but not exactly so if the boundary is part of a non-convex domain [2]. For the non-Sibsonian interpolant, there are no such restrictions. This is so, since for a point $\mathbf{x} \in \partial\Omega$, $s_I(\mathbf{x})$ is always finite for interior nodes, whereas it is unbounded for nodes on $\partial\Omega$; hence the nodal shape functions of interior nodes are zero. Hence, by choosing the non-Sibsonian interpolant, the exact imposition of essential boundary conditions in a meshless method is realized. An alternative approach using α -shapes in NEM to satisfy the imposition of essential boundary conditions was proposed in [7].

2.2. C^1 shape function

By embedding Sibson coordinates in the Bernstein-Bézier representation of a cubic simplex, Farin [8] proposed a $C^1(\Omega)$ natural neighbor interpolant. By using the Sibson coordinate in Farin’s construct, and carrying out the transformations outlined in [3], we arrive at a $C^1(\Omega)$ Sibson interpolant for PDEs.

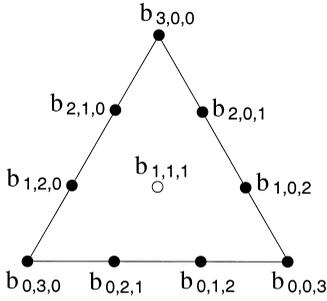


Fig. 3. Cubic Bernstein-Bézier triangular patch.

Let $\mathbf{i} = (i_1, i_2, \dots, i_n)$ be a multi-index with norm $|\mathbf{i}| = i_1 + i_2 + \dots + i_n$. In addition, let $\Phi = (\phi_1, \phi_2, \dots, \phi_n)$, where ϕ_I are the Sibson shape functions, be the barycentric coordinate of a simplex $\delta \in \mathbb{R}^{n-1}$. A $C^1(\Omega)$ Bernstein-Bézier surface interpolant of degree-3 over the simplex δ can be written as [8]

$$u^h(\Phi) = \sum_{|\mathbf{i}|=3} B_{\mathbf{i}}^3(\Phi) b_{\mathbf{i}}, \tag{6}$$

where $b_{\mathbf{i}}$ is the Bézier ordinate associated with the control point $\mathbf{i}/3$, and

$$(\phi_1 + \phi_2 + \dots + \phi_n)^3 = \sum_{|\mathbf{i}|=3} B_{\mathbf{i}}^3(\Phi), \tag{7}$$

$$B_{\mathbf{i}}^3(\Phi) = \binom{3}{\mathbf{i}} \phi_1^{i_1} \phi_2^{i_2} \dots \phi_n^{i_n},$$

where $\binom{m}{\mathbf{i}}$ is the multinomial coefficient, and it is noted that there are $n^2 + \binom{n}{3}$ terms on the right-hand side of Eq. (6). The control points (circles) and Bézier ordinate values ($b_{\mathbf{i}}$) for a cubic Bernstein-Bézier triangular patch are shown in Fig. 3.

The interpolant in Eq. (6) is suitable for surface data approximation; however, for the numerical solution of PDEs by a Galerkin procedure 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. To meet the desired goal, we re-cast Eq. (6) in the following form (matrix-vector notation):

$$u^h(\Phi) = \{\mathbf{B}(\Phi)\}^T \{\mathbf{b}\} = \{\mathbf{B}(\Phi)\}^T [\mathbf{T}] \{\mathbf{u}\} = \{\Psi(\Phi)\}^T \{\mathbf{u}\}. \tag{8}$$

In Eq. (8), $\{\Psi(\Phi)\}^T = \{\psi_1(\Phi), \psi_2(\Phi), \psi_3(\Phi), \dots, \psi_{3n-2}(\Phi), \psi_{3n-1}(\Phi), \psi_{3n}(\Phi)\}$, and $\{\mathbf{u}\}^T = \{u_1, \theta_{1x}, \theta_{1y}, \dots, u_n, \theta_{nx}, \theta_{ny}\}$, where $u_I = u(\mathbf{x}_I)$ are the nodal function values, and $\theta_{Ix} = u_{,x}(\mathbf{x}_I)$ and $\theta_{Iy} = u_{,y}(\mathbf{x}_I)$ are the nodal gradient values. The transformation matrix $[\mathbf{T}]$ maps the nodal function and gradient values to the Bézier ordinates [3].

On constructing $[\mathbf{T}]$ and carrying out the matrix-vector multiplication indicated in Eq. (8), we can express the

$C^1(\Omega)$ trial function in standard shape function notation as:

$$u^h(\mathbf{x}) = \sum_{i=1}^{3n} \psi_i(\mathbf{x}) u_i, \tag{9}$$

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 u_I , θ_{Ix} , and θ_{Iy} , respectively. Further details on the transformation matrix $[\mathbf{T}]$ and the shape function computations can be found in [3].

In the Galerkin implementation for the Poisson equation or linear elastostatics (second-order), the trial and test functions take the form shown in Eq. (5), whereas for the biharmonic equation (fourth-order), Eq. (9) is used. On using the weak form and a standard Galerkin procedure, the discrete system is obtained.

Acknowledgments

The author thanks Jeff Erickson for pointing out the work of Hiyoshi and Sugihara [9], who have also independently proposed the non-Sibsonian interpolant.

References

- [1] Braun J, Sambridge M. A numerical method for solving partial differential equations on highly irregular evolving grids. *Nature* 1995;376:655-660.
- [2] Sukumar N, Moran B, Belytschko T. The natural element method in solid mechanics. *Int J Num Methods Eng* 1998;43(5):839-887.
- [3] Sukumar N, Moran B. C^1 natural neighbor interpolant for partial differential equations. *Num Methods Part Diff Eq* 1999;15(4):417-447.
- [4] Sibson R. A vector identity for the Dirichlet Tessellation. *Math Proc Cambridge Philos Soc* 1980;87:151-155.
- [5] Belikov VV, Ivanov VD, Kontorovich VK, Korytnik SA, Semenov AYU. The non-Sibsonian interpolation: a new method of interpolation of the values of a function on an arbitrary set of points. *Comput Math Math Phys* 1997;37(1):9-15.
- [6] Sukumar N, Moran B, Semenov AYU, Belikov VV. Natural neighbor Galerkin methods. *Int J Num Methods Eng* 2001; 50(1):1-27.
- [7] Cueto E, Doblaré M, Gracia L. Imposing essential boundary conditions in the natural element method by means of density-scaled α -shapes. *Int J Num Methods Eng* 2000;49(4): 519-546.
- [8] Farin G. Surfaces over Dirichlet tessellations. *Computer Aided Geometric Design* 1990;7(1-4):281-292.
- [9] Hiyoshi H, Sugihara K. Two generalizations of an interpolant based on Voronoi diagrams. *Int J Shape Modeling* 1999;5(2):219-231.