A Note on Natural Neighbor Interpolation and the Natural Element Method (NEM)

N. Sukumar
Theoretical and Applied Mechanics
Northwestern University
IL 60208

May 11, 1997

1 Introduction

Natural neighbor (n-n) interpolation [1, 2, 3] is a multivariate data interpolation method, which has primarily been used in data interpolation and modeling of geophysical phenomena. See [4] for a review on scattered data interpolation in multi-dimensions. Natural neighbor interpolation relies on concepts, such as Voronoi diagrams [5] and Delaunay tessellations [6] in computational geometry [7, 8], in order to construct the interpolant. It is a weighted average method, where the weights are area-based, as opposed to Shepard's interpolant (to cite an example), where distance-based weights are used. In spite of its fairly simple structure, sound theoretical basis for construction, and desirable smoothness properties, n-n interpolation has seemingly received little attention in the area of multivariate data interpolation, when compared to other interpolation/approximation schemes such as Shepard's interpolant, moving least squares approximants, radial basis functions, or Hardy's multiquadrics. Recent work on n-n interpolation [9] and its application to the solution of partial differential equations [10] does exemplify the potential of the Natural Element Method (NEM), and hence begs the question if it could find possible applications in the field of computational solid/fluid mechanics. This note, in the same spirit, attempts to give a brief introduction to the interpolation scheme and its implementation in a Galerkin-based formulation (NEM) to solve partial differential equations. The interested reader can refer to the references (and the references therein) for further details.

2 Voronoi Diagrams and Delaunay Tessellations

The Voronoi diagram and its dual, Delaunay tessellation (covering of a surface with tiles) are geometric constructs that are widely used in the field of computational geometry. For simplicity, and for the purpose of illustration, we consider two-dimensional

Euclidean space $\mathbf{R}^2$; the theory, however, is applicable in a general k-dimensional framework. Consider a set of distinct points (nodes) $P = \{p_1, p_2, \ldots, p_N\}$ in $\mathbf{R}^2$. The tile (also known as Thiessen or Voronoi polygon) of p_n is defined as [1]

$$T_n = \{\mathbf{x} \in \mathbf{R}^2 \ : \ d(\mathbf{x}, \mathbf{x}_n) < d(\mathbf{x}, \mathbf{x}_m) \ \forall \ m \neq n\}, \tag{1}$$

where $d(\, .\, ,\, .)$ is the Euclidean metric. Each tile T_n is the intersection of finitely many open half-spaces, each being delimited by the perpendicular bisector (hyperlane in $\mathbf{R}^k$). In simpler terms, the tile T_n can be viewed as the locus of all points closer to p_n than to any other node. If $N = 2$, it is readily seen that the locus is the perpendicular bisector of the line joining the nodes, while if $N = 3$, the perpendicular bisector of the sides of the triangle (p_1, p_2, p_3) which meet at the circumcenter of the triangle defines the Voronoi diagram. Generalizing, we see that the Voronoi diagram for a set of nodes divides the plane into a set of regions, one for each node, such that any point in a particular region is closer to that region's node than to any other node. The Delaunay triangles are constructed by connecting the nodes whose Voronoi cells have common boundaries. The Delaunay triangulation and the Voronoi diagram are dual structures — each contains the same "information" in some sense, but represented in a different form. See [11] and [12] for details on Voronoi polygons. A nice exposition and comprehensive survey on Voronoi diagrams can be found in [13].

3 Natural Neighbor Interpolation

If T_n and T_m have a common boundary ($(k-1)$-dimensional face in $\mathbf{R}^k$), p_n and p_m are considered as *neighbors*. The notion of a set of 'neighboring nodes' is generalized by the definition of *natural-neighbor* nodes. The natural neighbors of any node are those in the neighboring Voronoi cells, or equivalently, those to which the node is connected by the sides of the Delaunay triangle. The above definition extends if we are interested in finding the natural neighbors of any sampling point $X(\mathbf{x}) \in \mathbf{R}^2$. By including the sampling point X in the Delaunay triangulation, the natural neighbors of X are the set of nodes which are connected to it. It is noteworthy to point out that the number of natural neighbors is a function of position of $X(\mathbf{x})$, and depends on the local nodal density.
Consider an interpolation scheme for a function $u(\mathbf{x})$: $\Omega \subset \mathbf{R}^2 \to \mathbf{R}$, in the form:

$$u^h(\mathbf{x}) = \sum_{I=1}^{n} \phi_I(\mathbf{x}) u_I \,, \tag{2}$$

where u_I $(I = 1, 2, \ldots, n)$ are the function values at the n natural neighbors, and $\phi_I(\mathbf{x})$ are the weights (shape functions in FE) associated with each node. In the context of natural neighbor interpolation, the weights $\phi_I(\mathbf{x})$ are taken as the n-n coordinates of the point $X(\mathbf{x})$ in the plane. Natural-neighbor coordinates were introduced by [1, 2] and may be defined in any number of dimensions. To illustrate, we use an example indicated in [9]. In Figure 1a, five nodes and the sides of the Voronoi

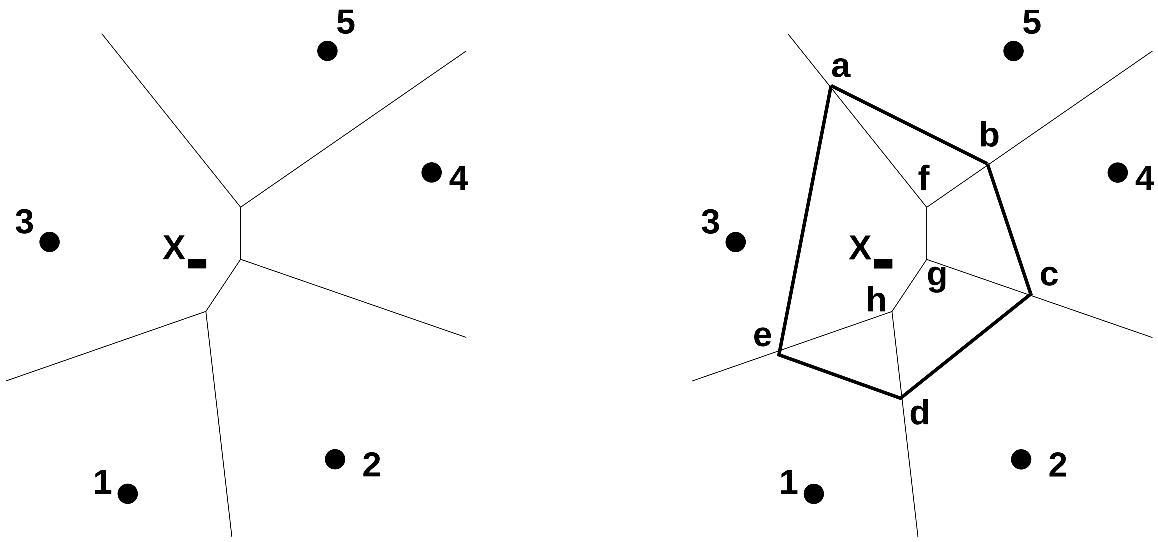

Figure 1: (a) Original Voronoi diagram for five ($n = 5$) neighboring nodes and X; (b) New Voronoi cell about X (dark line)

cells are shown. If a point $X(\mathbf{x})$ is added in cell 3, then a new Voronoi cell can be placed around it (see Figure 1b). The n-n coordinates of X with respect to a neighbor is defined as the ratio of the area of their overlapping Voronoi cells to the total area of the Voronoi cell about X. For instance, in Figure 1b, $\phi_3(\mathbf{x})$ is given by

$$\phi_3(\mathbf{x}) = \frac{A_{afghe}}{A_{abcde}}. \tag{3}$$

The five overlapping regions in Figure 1b are known as *second-order* Voronoi cells, while *abcde* is a *first-order* Voronoi cell. By the above definition of $\phi_I(\mathbf{x})$, the following two properties are self-evident:

$$\sum_{I=1}^{n} \phi_I(\mathbf{x}) = 1, \quad 0 \leq \phi_I(\mathbf{x}) \leq 1\,. \tag{4}$$

Now, if X were to coincide with any node, say node 3 for instance, then it is readily seen that $\phi_3(\mathbf{x}) = 1$ and $\phi_I = 0$, $I \neq 3$. Therefore the n-n coordinates satisfy the Kronecker-delta property:

$$\phi_I(\mathbf{x}_J) = \delta_{IJ} \tag{5}$$

and hence Eq. (2) is an interpolation: $u^h(\mathbf{x}_I) = u_I$.

Since $\phi_I(\mathbf{x})$ is only non-zero in the union of the n circles that pass through the vertices of the Delaunay triangles about node I, the n-n interpolation is a local (interpolating) scheme. In addition, the $\phi_I(\mathbf{x})$ are continuosly differentiable (C^∞) everywhere except at the nodes, $\mathbf{x}_I$, where all the derivatives are discontinuous [1]. The proof of the above in one-dimension is trivial. In 1D, it is readily derived that the ϕ_I are precisely linear FE shape functions: $\phi_1 = 1-x$, $\phi_2 = x$, where $x \in (0, 1)$.

Hence FEM is realized (by the n-n interpolation) in 1D, where it is well-known that the displacement derivative is discontinuous at nodes (**element boundaries** in 1D).

The natural neighbor coordinates $\phi_I(\mathbf{x})$ also satisfies an important property known as the Local Coordinate Property (LCP) — geometrical coordinates are interpolated exactly [1]:

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

Since the interpolant satisfies linear consistency, in the context of elastostatics, the approximation (trial function) can exactly represent rigid-body motion and linear displacement fields.

3.1 Governing Equations and Weak Form

In order to study a model PDE, let us consider small displacement elastostatics, which is governed by the equation of equilibrium:

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} + \mathbf{b} = 0 \text{ in } \Omega \tag{7}$$

where

$$\boldsymbol{\sigma} = \mathbf{C} : \boldsymbol{\varepsilon} \,, \quad \boldsymbol{\varepsilon} = \boldsymbol{\nabla}_s \mathbf{u} \,. \tag{8}$$

In the above equations, $\Omega \subset \mathbf{R}^2$ is the domain of the body, $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\boldsymbol{\varepsilon}$ is the small strain tensor, $\mathbf{b}$ is the body force per unit volume, $\mathbf{C}$ is the material moduli tensor, $\mathbf{u}$ is the displacement, $\boldsymbol{\nabla}$ is the gradient operator, and $\boldsymbol{\nabla}_s$ is the symmetric gradient operator.

The essential and natural boundary conditions are

$$\mathbf{u} = \bar{\mathbf{u}} \text{ on } \Gamma_u \,, \quad \mathbf{n} \cdot \boldsymbol{\sigma} = \bar{\mathbf{t}} \text{ on } \Gamma_t \,, \tag{9}$$

where Γ is the boundary of Ω, and $\bar{\mathbf{u}}$ and $\bar{\mathbf{t}}$ are prescribed displacements and tractions, respectively.

The weak form (principle of virtual work) is

$$\int_\Omega \nabla_s \delta\mathbf{v} : \boldsymbol{\sigma} \, d\Omega = \int_\Omega \delta\mathbf{v} \cdot \mathbf{b} \, d\Omega + \int_{\Gamma_t} \delta\mathbf{v} \cdot \bar{\mathbf{t}} \, d\Gamma \quad \forall \; \delta\mathbf{v} \in \mathcal{H}^1_E(\Omega) \tag{10}$$

On substituting the trial and test functions in the above equation and using the arbitrariness of nodal variations, the following discrete system of equations is obtained:

$$\boldsymbol{K}\mathbf{d} = \mathbf{f}^{ext} \,, \tag{11}$$

where

$$\mathbf{K}_{IJ} = \int_\Omega \mathbf{B}_I^T \mathbf{D} \mathbf{B}_J \, d\Omega \,, \quad \mathbf{f}_I^{ext} = \int_{\Gamma_t} \mathbf{\Phi}_I^T \bar{t} \, d\Gamma + \int_\Omega \mathbf{\Phi}_I^T \mathbf{b} \, d\Omega \,. \tag{12}$$

In the above equation, $\mathbf{\Phi}_I$ is the shape function vector and $\mathbf{B}_I$ is the matrix of shape function derivatives.

4 Implementation of the Natural Element Method (NEM)

The implementation of the Natural Element Method (NEM) by means of a Galerkin-based procedure is parallel to that adopted in FEM or Element-Free Galerkin (EFG) method [14], with the key distinction that separates the three is in the construction of the shape functions $\phi_I(\mathbf{x})$ and their derivatives.

A computational procedure to evaluate the shape functions (n-n coordinates) is outlined in [3], which is extended by [9] to compute the derivatives of the interpolating function $u^h(\mathbf{x})$. In the application of n-n interpolation to PDEs (fluid-structure interaction), [10] adopted Lasserre's recursive formula [15] to compute the area (volume of a convex polyhedral in $\mathbf{R}^k$) of the second-order Voronoi cells. The shape functions $\phi_I(\mathbf{x})$ as well as their derivatives $\phi_{I,j}(\mathbf{x})$ are computed. In [10], it is pointed out that Lasserre's formula [15] is more robust than the one due to [3], which breaks-down if the point $X(\mathbf{x})$ lies along the edge of a Delaunay triangle. Expressions for the shape functions and their derivatives are provided [3, 16, 9, 10].

5 A Study in Contrast: FE, EFG, and NEM

A comparison of FEM, EFG, and NEM follows:

1. The interpolant in both FEM and NEM possess similar properties (see Section 3), and the interpolants are exactly the same in 1D. The EFG trial function does possess linear consistency but is not an interpolant; hence, Lagrange multipliers or a coupling with FE is required to satisfy the essential boundary conditions.

2. A "similar" (in principle) mesh structure (triangulation) is used in both FE and NEM, while the approximant in EFG (a "meshless method") is constructed solely on the basis of a set of nodes and a weight function with compact support.

3. In all three methods, the interpolant/approximant has a local character because of the compact support of the shape functions $\phi_I(\mathbf{x})$.

4. In FE, the functional that is integrated over the domain Ω in the weak form is constructed using a polynomial trial function and hence we can obtain an accurate estimate of the integral using Gauss-Legendre quadrature (within machine precision and round-off errors). As opposed to FE, in both NEM and EFG (moreso probably in EFG), the shape functions are not polynomials in general. Hence numerical integration is an issue and the question of optimum/accurate quadrature arises, which is still an open issue. In [10], three-point Gauss-Legendre quadrature was found to be sufficient to obtain accurate results.

5. In the discrete system $\mathbf{Kd} = \mathbf{f}$, the stiffness matrix $\mathbf{K}$ is banded in FEM, while in both EFG and NEM, $\mathbf{K}$ is not banded in general.

6. The computational cost (apart from numerical issues) is an important factor in the feasibility and usability of a numerical method. Finite elements, apart from their nice local (polynomial) properties, are a computationally attractive choice because of the very fast execution times that are attainable. The key time-consuming steps in the NEM are:

   (a) Dirichlet tessellation (Voronoi diagram) of the nodes — for some typical algorithms, see [17, 18, 19]. In 2D, Fortune's 'sweepline' algorithm [18] is considered to be one of the fastest, while in higher dimensions, the Qhull package [20] based on the 'quickhull' algorithm [19] appears to be the unanimous choice. Both the above are $\mathcal{O}(n \log n)$ algorithms.

   (b) Search for natural neighbors for a sampling point $X(\mathbf{x})$. The walking-triangle algorithm due to Lawson [21], which is a $\mathcal{O}(n)$ algorithm, is a suitable choice.

   [10] used the Qhull package [20] for the Dirichlet tessellation and Lawson's algorithm [21] for the neighbor-search. They have reported [10] execution times for NEM that are about a factor of two slower than quadratic finite elements.

6 Applications in Computational Mechanics

In the preceding sections, a concise description of natural neighbor interpolation and its implementation in the context of a Galerkin procedure to solve PDEs was delineated. Having at least given a flavor (I hope) of the interpolation scheme, one can be undoubtedly ambitious in suggesting its possible applications within the field of computational mechanics:

1. Patch test: The first step in the testing of a numerical method for the elliptic operator. The patch test is a sufficient condition to prove the convergence of a numerical method [22].

2. Smooth boundary-value problems in elastostatics (2D) — plate with a circular/elliptical hole, beam in bending, etc.

3. Benchmark crack problems in 2D: The edge-crack, center-crack, and shear crack problems, where symmetric boundary conditions can be imposed. The case of inclined cracks (no planes of symmetry) might pose a problem, since the whole domain has to be modeled — non-convex domain!

4. The higher order smoothness (or regularity) of the NEM interpolant in $\Omega \backslash \mathbf{x}_I$ allows one to consider higher order PDEs [10]. The FE interpolant, $u^h_{FE}(\mathbf{x}) \in C^0$, since $u^h_{FE}(\mathbf{x})$ is continuous but its gradients are discontinuous across element

boundaries (edges of the Delaunay triangle). The approach used by [23] to successfully avoid locking in beams and plates by using cardinal splines on regular grids can be explored using NEM in the context of irregular nodal arrangements. The jumps in the displacement gradients at the nodes do not pose a problem since the numerical integration is carried out inside the triangle.

5. Finite deformation: Mesh distortion and mesh entanglement which are the bane of FE are not issues in NEM. The density of the nodes dictates the quality of the n-n interpolation and not the Delaunay triangles.

6. Fluid-structure interaction: See [10].

7. Moving interfaces ...

7 Acknowledgements

The author thanks Dr. David Watson for his comments and insights on natural neighbor interpolation.

References

[1] R. Sibson. A vector identity for the Dirichlet tesselation. *Mathematical Proceedings of the Cambridge Philosophical Society*, 87:151–155, 1980.

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

[3] D. F. Watson. *Contouring: A Guide to the Analysis and Display of Spatial Data.* Pergamon Press, Oxford, 1992.

[4] P. Alfeld. Scattered data interpolation in three or more variables. In T. Lyche and L. L. Schumaker, editors, *Mathematical Methods in Computer Aided Geometric Design*, pages 1–34, San Diego, 1989. Academic Press.

[5] G. M. Voronoi. Nouvelles applications des paramètres continus à la théorie des formes quadratiques. deuxième Mémoire: Recherches sur les parallélloèdres primitifs. *J. Reine Angew. Math.*, 134:198–287, 1908.

[6] B. Delaunay. Sur la sphère vide. A la memoire de Georges Voronoi. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskih i Estestvennyh Nauk*, 7:793–800, 1934.

[7] F. Preparata and M. Shamos. *Computational Geometry: An Introduction.* Springer-Verlag, New York, N.Y., 1985.

[8] J. O'Rourke. *Computational Geometry in C.* Cambridge University Press, 1994.

[9] M. Sambridge, J. Braun, and H. McQueen. Geophysical parameterization and interpolation of irregular data using natural neighbors. *Geophysical Journal International*, 122:837–857, 1995.

[10] J. Braun and M. Sambridge. A numerical method for solving partial differential equations on highly irregular evolving grids. *Nature*, 376:655–660, 1995.

[11] B. N. Boots. *Voronoi (Thiessen) Polygons.* Geo Books, Norwich, UK, 1986.

[12] A. Okabe, B. Boots, and K. Sugihara. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams.* John Wiley & Sons, Chichester, England, 1992.

[13] F. Aurenhammer. Voronoi diagrams — a survey of a fundamental geometric data structure. *ACM Transactions on Mathematical Software*, 23:469–483, 1996.

[14] T. Belytschko, Y. Y. Lu, and L. Gu. Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering*, 37:229–256, 1994.

[15] J. B. Lasserre. An analytical expression and an algorithm for the volume of a convex polyhedron in R^n. *Journal of Optimization Theory and Applications*, 39(3):363–377, 1983.

[16] D. F. Watson. *nngridr: An implementation of natural neighbor interpolation.* David Watson, 1994.

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

[18] S. Fortune. A sweepline algorithm for Voronoi diagrams. *Algorithmica*, 2:153–174, 1987.

[19] C. B. Barber, D. P. Dobkin, and H. T. Huhdanpaa. The Quickhull algorithm for convex hull. *ACM Transactions on Mathematical Software*, 22:469–483, 1996.

[20] C. B. Barber and H. T. Huhdanpaa. *Qhull (Computer Program).* Available at http://www.geom.umn.edu/software/qhull/, The Geometry Center, University of Minnesota, 1300 South Second Street, Minneapolis, MN 55454, 1994.

[21] C. L. Lawson. Software for C^1 surface interpolation. In J. R. Rice, editor, *Mathematical Software III*, volume 3, New York, N.Y., 1977. Academic Press.

[22] G. Strang and G. Fix. *An Analysis of the Finite Element Method.* Prentice-Hall, Englewood Cliffs, N.J., 1973.

[23] B. Donning. Meshless methods for shear-deformable beams and plates. Master's thesis, Northwestern University, 1997.