Natural Neighbor Interpolation and Order of Continuity

Tom Bobach and Georg Umlauf
University of Kaiserslautern
Computer Science Department / IRTG
D-67653 Kaiserslautern, Germany
bobach|umlauf@informatik.uni-kl.de

Abstract: In this paper we give a survey on natural neighbor based interpolation, a class of local scattered data interpolation schemes that define their support on *natural neighbors* in the Voronoi diagram of the input data sites. We discuss the existing work with respect to common aspects of scattered data interpolation and focus on smoothness of the interpolant.

1 Introduction

Scattered data interpolation (SDI) is the problem of finding an interpolating functional description which is as close as possible to an unknown function for which values are known only at discrete, scattered locations. Among the SDI methods existing so far, those based on natural neighbors possess the best adaption to inhomogeneous sample distributions while only building on a highly local support.

After Sibson introduced natural neighbor coordinates [Sib80] (Sibson's coordinates) and their application to SDI [Sib81], the theory of natural neighbor based local coordinates and SD interpolants built from them has received an in-depth investigation. Piper developed formulas and geometric interpretation of derivatives of Sibson's coordinates [Pip92]. Probably inspired by Sibson's original work, a less smooth type of natural neighbor coordinates (Laplace coordinates) has been proposed independently by several authors [CFL82, Sug99, BIK$^+$97]. An interesting relationship between Laplace and Sibson's coordinates has been found and generalized by Hiyoshi et al. [HS00b], yielding local coordinates of arbitrary continuity except at the data sites.

The problem of transfinite interpolation based on natural neighbor coordinates has been subject to the work of Anton et al. for Sibson's coordinates with respect to points and line segments [AMG98], of Gross et al. with respect to circles and polygons [GF99], and of Hiyoshi et al. for Laplace coordinates with respect to points, line segments, and circles [HS00a].

The geometric definition of natural neighbor coordinates is inappropriate for actual computation, especially in higher dimensions. The two main approaches to solve this are either to reformulate the geometric entities based on the Delaunay neighborhood and algebraic expressions [BS95, Sug99, Hiy05, BBU06b], or to solve the computation approximately on graphics hardware [FEK$^+$05, PLK$^+$06]. Results on the approximate computation of generalized Voronoi diagrams on graphics hardware can be found in [HCK$^+$99].

The interpolation of smooth functions requires additional efforts to overcome derivative discontinuities at the data sites inherent to all natural neighbor based local coordinates. We are only aware of two approaches, one building polynomials of the local coordinates to interpolate derivatives [Sib81, Far90, HS04], the other a construction of non-convex coordinates from a bigger natural neighborhood as explained in [Cla96, Flö03].

If the input data sites are scattered over a manifold rather than its embedding space, the Voronoi diagram and consequently the notion of natural neighbors are subject to a modified metric. That special setting received attention from [BC00, Flö03], where the local coordinate property is established for power diagrams and their restrictions to manifolds.

Outline: We will briefly review the aspects of scattered data interpolation for scalar valued functions in Section 2 to introduce the problems addressed by the natural neighbor based SDI methods which we discuss thereafter, focusing on:

- Smoothness of the local coordinates except at the data sites in Section 3,
- Smoothness of the interpolant at the data sites Section 4,
- Extension of the local coordinates to arbitrarily shaped sites Section 5,
- Extension of the local coordinates to manifolds in Section 6,
- Implementation of natural neighbor interpolation in Section 7.

We end this survey with a classifying summary of all considered methods and a comparison to some other established SDI methods in Section 8.

2 Scattered Data Interpolation

The problem of scattered data interpolation can be stated as follows: given sample data sites $X = \{x_i\}_{i=1\ldots m} \subset \mathbb{R}^n$ and data values $Z = \{z_i\}_{i=1\ldots m} \subset \mathbb{R}$, find a function $f : \mathbb{R}^n \to \mathbb{R}$ that satisfies the interpolation constraint $f(x_i) = z_i$. The subset $\{(x_j, z_j)\}_j$ on which the value at a query position q depends is called the *support* of f at q and leads to the distinction between schemes with *local* and *global* support. While schemes with global support usually have higher smoothness than local ones, their cost of computation makes them inapplicable for large scale data sets.

The aspects that are addressed by a multitude of scattered data interpolation schemes cover, among others:

Support: How is the support determined? If the size of the data set exceeds that of the available RAM, global schemes fail. Local schemes have a small memory footprint and can be computed much more efficiently, but are less smooth.

Smoothness: How often is f continuously differentiable?

Derivative Interpolation: Can f interpolate higher order derivatives at the data sites?

Polynomial Precision: Up to what order does f reproduce polynomials?

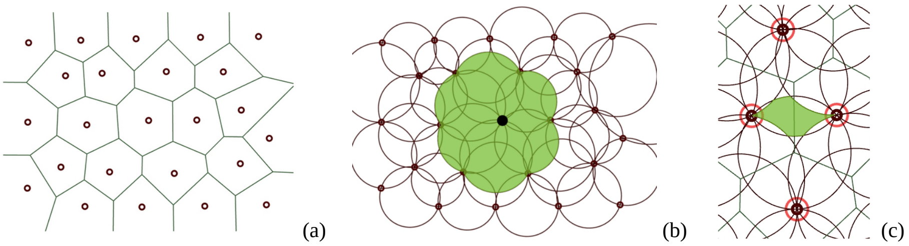

Figure 1: (a) Voronoi Diagram of points in $\mathbb{R}^2$. (b) Points in the shaded region have the bold point as one natural neighbor. (c) Points in the shaded region have the four bold points as natural neighbors.

Transfinite Interpolation: Instead of points can one interpolate to curves or higher-dimensional manifolds? Transfinite interpolation leads to a continuous representation of the input data and usually requires considerably more effort in implementation.

Interpolation on manifolds: Can the interpolation scheme still be applied if the underlying space itself is a manifold, and measurements are subject to other metrics?

Computation: How can f be efficiently evaluated? Naïve implementations of interpolation methods can lead to inacceptable performance. The appropriate implementation of the interpolation schemes is important for its applicability.

Derived values: Exist formulas for derivatives or integrals of f?

Extrapolation: Local schemes typically define only $f|\mathscr{D}$ with $\mathscr{D}$ being the convex hull of X. Does f have a meaningful definition outside $\mathscr{D}$?

Approximation order: If data is sampled from a known function, how close does f get to that function with increasing sampling density? To know the approximation order of a method is to know how well it is suited to model phenomena with a certain class of governing functions.

Non-scalar values: Can function values $Z \not\subset \mathbb{R}$, e.g. $Z \subset \mathbb{R}^d$, be interpolated? For scalar data at the input sites, the space of possible functions can be described using linear combinations of neighborhood data. This does not necessarily hold for non-scalar data, where more sophisticated blending functions may be needed.

3 Natural Neighbor Coordinates

The group of interpolation schemes we discuss here exploits geometric identities of natural neighbors, building an interpolant by applying the same identity to the data values. We first repeat some facts about Voronoi diagrams and local coordinates, then focus on Laplace, Sibson's and Hiyoshi's coordinates.

3.1 Voronoi Diagrams

Let X be data sites that act as generators of a Voronoi diagram, and $d(\cdot, \cdot)$ the Euclidean distance on $\mathbb{R}^n$. The resulting Voronoi diagram (Figure 1(a)) is the partition of space into convex tiles $\mathcal{V}(X) = \{\mathscr{T}_i\}_{i=1\ldots m}$, $\bigcup_{i=1\ldots m} \mathscr{T}_i = \mathbb{R}^n$, with

$$\mathscr{T}_i = \{x \in \mathbb{R}^n | d(x, x_i) \leq d(x, x_j), i \neq j\}. \tag{1}$$

Two generators x_i and x_j are called *natural neighbors* if their associated tiles share a non-empty hyperface $s_{ij} := \mathscr{T}_i \cap \mathscr{T}_j$. To ensure boundedness of the tiles, we will restrict our considerations to the interior $\mathscr{D}$ of the convex hull of X. We denote the set of indices of the natural neighbors for a generator x_i by N_i. If $n+1$ or more tiles share a common point, the unique circumsphere through their generators is called *Delaunay sphere*, since it contains no other generator in its interior. For an overview on Voronoi diagrams the reader may refer to [Aur91, OBSC00].

We denote by $x_0 \in \mathscr{D}$ an arbitrary point, called *query point*, and define $\mathcal{V}(X \cup \{x_0\}) =: \{\mathscr{T}_i'\}_{i=0\ldots m}$. All notions from the Voronoi diagram carry over to x_0, and $\mathscr{T}_0'$ is called the *virtual tile* of the query position.

3.2 Local Coordinates

As long as the set of data sites X is not degenerate and the query point x_0 lies inside its convex hull, it is also always contained in the convex hull of its natural neighbors $\{x_i\}_{i \in N_0}$. Since $\{x_i\}_{i \in N_0}$ is in general position, i.e. contains $n+1$ affinely independent points, we can express x_0 in terms of *generalized barycentric coordinates* with respect to its natural neighbors

$$\text{(local coordinates)} \qquad x_0 = \sum\nolimits_{i \in N_0} \lambda_i(x_0) x_i, \tag{2a}$$

$$\text{(partition of unity)} \qquad 1 = \sum\nolimits_{i \in N_0} \lambda_i(x_0), \tag{2b}$$

$$\text{(convexity)} \qquad 0 \leq \lambda_i(x_0), \qquad i \in N_0. \tag{2c}$$

Then, (2a) - (2c) guarantee affine invariance for λ, yielding the linear precision scattered data interpolant

$$f(x_0) = \sum\nolimits_{i \in N_0} \lambda_i(x_0) z_i. \tag{3}$$

We will refer to the local coordinates λ_i by the $|N_0|$-tuple λ and omit the argument x_0 for the sake of brevity unless required by context. For $|N_0| = n+1$, λ reduces to the usual barycentric coordinates and f is a linear function. If $|N_0| > n+1$, there are infinitely many choices for λ that satisfy (2a)-(2c).

From (3) it is clear that f is as smooth as λ. Therefore, we concentrate on how to control the smoothness of λ. The sequence of local coordinates we discuss next will be denoted by λ^k and f^k consequently denotes the interpolant (3) based on these coordinates.

3.3 Some Properties of Natural Neighbor Coordinates

Natural neighbor coordinates are based on sizes of geometric entities in the virtual tile $\mathscr{T}_0'$ of the query position x_0. The rate at which these entities change with x_0 basically

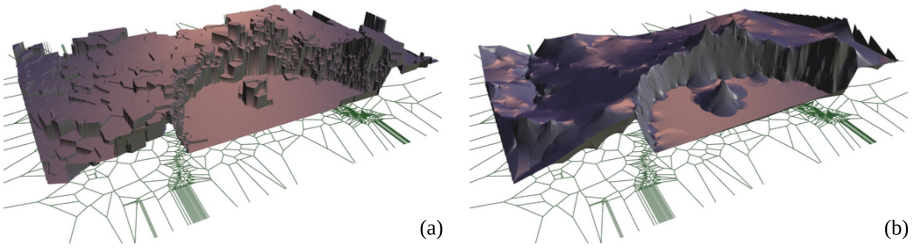

Figure 2: Interpolation of 1493 scattered points sampled from the crater lake data set. The original data set is due to US geological survey with 344 · 463 points. (a) The nearest neighbor interpolant is piecewise constant and discontinuous along the edges of the Voronoi diagram. (b) The Laplace interpolant is continuous with derivative discontinuities along the Delaunay circles.

determines the smoothness of the coordinates. Whenever the query position coincides with a data site, $x_0 = x_i$, these entities are not defined. The coordinates, however, can be continuously extended for $x_0 \to x_i$, yielding C^0 continuity at the data sites.

The region of influence for each data value z_i is the set of points which have x_i as natural neighbor, i.e. for which the contribution of z_i is not zero. This region is just the union of all Delaunay spheres passing through x_i, depicted in Figure 1(b) for the center data site. The regions of constant neighborhood, i.e. where N_0 does not change, are all areas that are bounded by Delaunay spheres, as depicted in Figure 1(c). These regions are considerably more complex than those appearing e.g. in barycentric interpolation in Delaunay tessellations, where they are polygonal domains.

3.4 Nearest Neighbor Interpolation

A simple scattered data interpolation scheme that uses the Voronoi diagram of data sites is *nearest neighbor interpolation*. The weights used here are simply defined as

$$\lambda_i^{-1} := \begin{cases} 1, & x_0 \in \mathscr{T}_i, \\ 0, & \text{otherwise.} \end{cases}$$

Thus, f is discontinuous along the Voronoi edges, as can clearly be seen in Figure 2(a).

3.5 Laplacian Interpolation

A set of C^{n-2}-smooth local coordinates has been proposed by different authors as Laplace- or Non-Sibsonian coordinates [CFL82, Sug99, BIK+97]. Denote by $\sigma_i := \text{vol}^{n-1}(s_{0i})$ the $n-1$-dimensional area of the hyperface shared by $\mathscr{T}_0'$ and $\mathscr{T}_i'$, and $r_i := d(x_0, x_i)$. Then Laplace coordinates λ^0 are defined as

$$\hat{\lambda}_i^0 := \sigma_i / r_i \qquad \text{and} \qquad \lambda_i^0 := \hat{\lambda}_i^0 / \sum\nolimits_{j \in N_0} \hat{\lambda}_j^0.$$

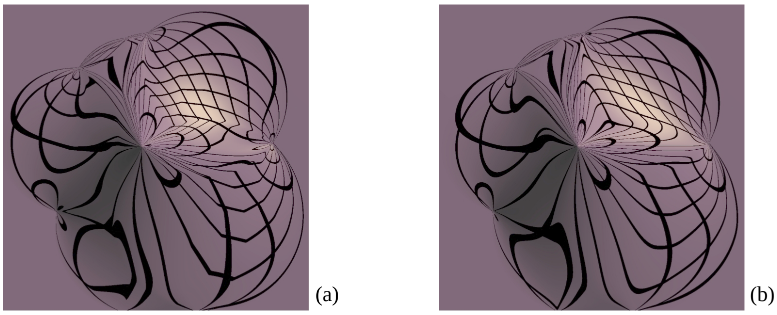

Figure 3: Reflection lines on the basis function of Sibson's (a) and Hiyoshi's C^2 (b) coordinates, seen from above. (a) The cusps indicate the C^2 discontinuities at the Delaunay circles. (b) Due to C^2 continuity away from the data sites, the cusps have vanished.

These coordinates and the resulting interpolant f^0 are continuous in $\mathscr{D}$ and have derivative discontinuities at the generators. For $X \subset \mathbb{R}^n$, we find that λ_i^0 is C^{n-2} on the Delaunay spheres. For $n = 2$ this results in the C^0 artifacts that can be seen in Figure 2(b). Different proofs for λ_i^0 satisfying (2a) have been given in [HS00b, BIK$^+$97].

3.6 Sibson's Interpolation

The first appearance of natural neighbor based local coordinates is due to Sibson [Sib80], who extended this to scattered data interpolation in [Sib81]. Sibson's coordinates are based on the volumes $\nu_i := \mathrm{vol}^n(\mathscr{T}_0' \cap \mathscr{T}_i)$ via

$$\hat{\lambda}_i^1 := \nu_i, \qquad \text{and} \qquad \lambda_i^1 := \hat{\lambda}_i^1 / \textstyle\sum_{j \in N_0} \hat{\lambda}_j^1.$$

The fact that ν_i is a n-dimensional volume results in λ^1 being C^{n-1} continuous except at the generators, where they are still only C^0. See Figure 3(a) for an example.

Properties of Sibson's coordinates received close attention by Farin [Far90] and Piper [Pip92]. With c_i denoting the centroid of s_{0i} their explicit formula for the gradient of λ_i is

$$\nabla \lambda_i^1 = \sigma_i (c_i - x_0)/r_i. \tag{4}$$

Different proofs for λ_i^1 satisfying (2a) have been given by [Sib80, Pip92, HS00b].

3.7 Hiyoshi's Interpolation

Hiyoshi and Sugihara [HS00b] proposed a generalization of Laplace and Sibson's coordinates based on an integral of σ_i. In [Hiy05] Hiyoshi restated this as

$$\hat{\lambda}_i^k := \frac{1}{(k-1)!} \int_{p \in \mathscr{T}_i \cap \mathscr{T}_0'} \big((x_0 - x_i) \cdot (p - c_i)\big)^{k-1} |dp|,$$

$$\lambda_i^k := \hat{\lambda}_i^k / \sum\nolimits_{j \in N_0} \hat{\lambda}_j^k .$$

For $k = 0, 1$, Hiyoshi's coordinates coincide with Laplace and Sibson's coordinates. For $k > 1$, these coordinates are C^{k+n-2} in $\mathscr{D} \setminus X$. As Hiyoshi pointed out in [Hiy05], the limit $k \to \infty$ does not lead to C^∞ coordinates but to the piecewise linear interpolant on the Delaunay tessellation.

4 Smooth Interpolation at the Data Sites

The previous section considered interpolants building on a linear combination of data values by local coordinates as in (3), resulting in derivative discontinuities at x_i. Here we consider basically two approaches to overcome this. One is to construct polynomials of the local coordinates to control the derivatives at the data sites. Another is to construct local coordinates from a larger neighborhood, which results in smooth, non-convex weights. To apply the first approach, the derivatives at the data sites need to be known. Otherwise they can be estimated using the approach described in [BBU06b].

In the remainder of this section we will denote by f^{ab} an interpolant based on local coordinates λ^a which has smoothness C^b at the data sites.

4.1 Sibson's C^1 Interpolant

In [Sib81] Sibson described a construction of a C^1 interpolant. He generates gradients ∇_i based on the weighted least squares plane through the neighboring data values, which are then interpolated by blending first order functions with the help of coordinates λ_i^1. With $r_i := d(x_0, x_i)$, $\gamma_i := \lambda_i^1 / r_i$, define

$$\zeta_i := z_i + (x_0 - x_i)^T \nabla_i \quad \text{and} \quad \zeta := \left(\sum\nolimits_{i \in N_0} \gamma_i \zeta_i\right) / \left(\sum\nolimits_{i \in N_0} \gamma_i\right),$$

$$\alpha := \left(\sum\nolimits_{i \in N_0} \lambda_i^1 r_i\right) / \left(\sum\nolimits_{i \in N_0} \gamma_i\right) \quad \text{and} \quad \beta := \sum\nolimits_{i \in N_0} \lambda_i^1 (r_i)^2 .$$

Blending this with Sibson's C^0 interpolant $f^1(x_0)$ yields Sibson's C^1 interpolant

$$f_{Sib}^{11}(x_0) = (\alpha f^1(x_0) + \beta \zeta)/(\alpha + \beta),$$

which does not easily generalize to higher orders of continuity.

4.2 Farin's C^1 Interpolant

A much more general approach which is not restricted to natural neighbor coordinates but can be applied to all local coordinates having properties (2a)-(2c) was proposed by Farin [Far90]. λ can be seen as barycentric coordinates in a l-variate Bézier simplex which projects to the convex hull of $\{x_i\}_{i \in N_0}$, where $l = |N_0|$. In Bézier simplexes it is easy to model directional derivatives at the vertices x_i by appropriately choosing the Bézier control net. From prescribed derivatives at x_i, a certain number of control points is fixed.

The remaining control points can be chosen arbitrarily without interfering with the interpolation property. Exploiting the concept of degree elevation, these can be chosen to yield polynomial precision. In the following, we will denote by D_v the directional derivative along v. By $\alpha \in \mathbb{N}^n$, $\sum \alpha_i = d$, we denote the n-dimensional multi-index that enumerates the d-th degree Bézier control points b_α. The indexes of the vertices of the Bézier simplex are denoted by α^i, where the i-th entry is set to d, and e^i is a multi-index with zero entries except for the i-th component which is one.

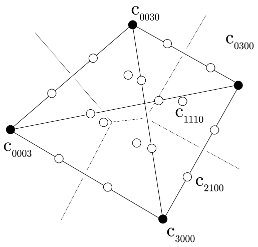

Figure 4: Planar projection of the control net of a cubic Bézier simplex in $\mathbb{R}^3$.

Farin presented the implementation of the above idea for cubic Bézier simplexes over λ^1 to interpolate gradients ∇_i, yielding a globally C^1 interpolant. By setting the directional derivatives in each vertex x_i towards its neighbor x_j,

$$D_{x_j - x_i} = \nabla_i (x_j - x_i), \qquad j \in N_0 \setminus \{x_i\},$$

we constrain x_i and all inner control points $b_{\alpha^i - e^i + e^j}$, to be coplanar, i.e.

$$b_{\alpha^i} = z_i, \qquad \text{and} \qquad b_{\alpha^i - e^i + e^j} = z_i + \frac{1}{3}(x_j - x_i)^T \nabla_i \text{ for } i \neq j.$$

This fixes all control points except those on simplex faces. By degree elevation for Bézier simplices, these are chosen to ensures quadratic precision of the resulting interpolant, see [Far90, Flö03]. Let $\beta = e^i + e^j + e^k$ for $i < j < k$, then b_β is an inner control point. Set $u = \frac{1}{3}(b_{\alpha^i} + b_{\alpha^j} + b_{\alpha^k})$ and $v = \frac{1}{6}\sum_{a,b \in \{i,j,k\}, a<b} b_{\beta - e^a + e^b}$, the average of the remaining fixed control points, then $b_\beta = \frac{3}{2}v - \frac{1}{2}u$ yields quadratic precision for the interpolant. The resulting interpolant inherits C^1 continuity on $\mathscr{C} \setminus X$ from λ^1 and is given by

$$f^{11}_{Far}(x_0) := b^3(\lambda^1).$$

4.3 Hiyoshi's C^2 Interpolant

Applying the above approach to quintic Bézier simplexes over λ^2, [HS04] present a construction of control points that matches derivatives up to order two given by the *Hessian* $\mathcal{H}_i$ at generator x_i. Let i, j, k be mutually distinct and $d_{ij} = x_j - x_i$, then

$$\begin{aligned} b_{\alpha^i} &= z_i, \\ b_{\alpha^i - e^i + e^j} &= z_i + \frac{1}{5}{\nabla_i}^T d_{ij}, \\ b_{\alpha^i - 2e^i + 2e^j} &= z_i + \frac{2}{5}{\nabla_i}^T d_{ij} + \frac{1}{20}{d_{ij}}^T \mathcal{H}_i d_{ij}, \\ b_{\alpha^i - 2e^i + e^j + e^k} &= z_i + \frac{1}{5}({\nabla_i}^T d_{ij} + {\nabla_i}^T d_{ik}) + \frac{1}{20}{d_{ij}}^T \mathcal{H}_i d_{ik} \end{aligned}$$

fix the control points based on the prescribed derivatives. Cubic precision of the resulting interpolant is given by the choice of the remaining control points based on the degree elevation principle, see [HS04].

4.4 Clarkson's Interpolation

One special kind of local coordinates that does not directly fit definition (2) is an idea of Clarkson [Cla96, Flö03]. It is based on the two-ring neighborhood of the query position and is designed to reproduce spherical quadratics, i.e. functions of the form $x \mapsto a\|x-b\|^2$, $a \in \mathbb{R}, b \in \mathbb{R}^n$. It is the only approach so far that has a really implicit C^1 construction and does not depend on prescribed derivative information. Clarkson's local coordinates differ significantly from those of Section 3:

- They depend on $\bigcup_{i \in N_0} N_i$, i.e. the two-ring neighborhood of x_0.
- They are not convex, i.e. (2c) does not hold.
- They seem to be C^1 at x_i, which is not yet proved.

5 Transfinite Interpolation

In this section we discuss methods to interpolate line segments, polygons and circular arcs instead of points. Non-point generators lead to *generalized Voronoi diagrams*, and the geometric primitives that constitute the local coordinates from Section 3 are no longer convex polygons. The main consequence of this generalization is an increased complexity in both data handling and the computation of the interpolant, which also seems to be the reason that research in this direction has been restricted to two dimensions so far.

Interestingly, transfinite interpolation enables us to impose discontinuities along the manifold data sites by using different values for each side.

This section will first explain the main differences between ordinary Voronoi diagrams and such with curves as generators, before taking a closer look at how the identities from Section 3 extend to the transfinite case.

5.1 Generalized Voronoi Diagrams in 2D

Assume the data sites are points $x_i \in X$ and non-intersecting curves $c_i \in C$, where i runs over the combined set of data sites $X \cup C$. Definition (1) still holds with a modified distance function,

$$d(x, c_i) = \min_{q \in c_i} \|x - q\|.$$

We denote the bisectors $\mathscr{T}_i \cap \mathscr{T}_j$ by e_{ij}. Tiles induced by points are still convex, while for curves this is in general not true. As in the ordinary Voronoi diagram, the virtual insertion of a new point x_0 into $\mathcal{V}(X)$ results in a new, convex tile $\mathscr{T}_0'$. An example of a generalized Voronoi diagram and the virtual tile (shaded) can be seen in Figure 5(a). The shape of the bisectors between the various elements of the Voronoi diagram is at least as complicated as that of the elements itself. Thus, an exact computation of areas and lengths seems feasible only for simple shapes of the generators. For arbitrary shapes, the Voronoi diagram can be approximated using graphics hardware, see Section 7.3.

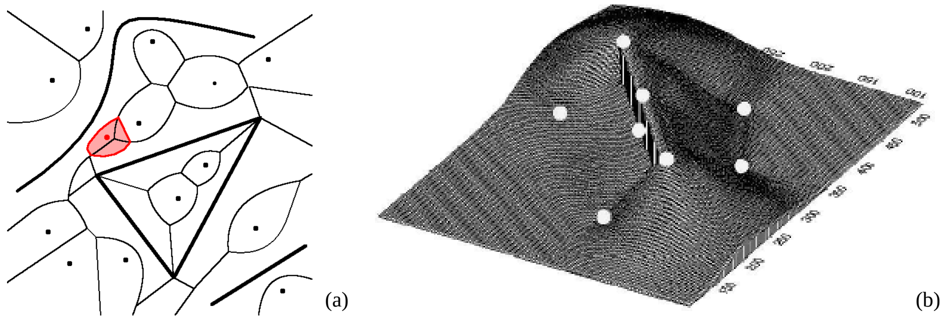

Figure 5: (a) The Voronoi diagram of a set of points, line segments and general curves (drawn bold). The virtual tile of a new point is shaded. Picture courtesy of [Hof99]. (b) Transfinite interpolation of a directed line segments and a couple of points. Picture courtesy of [AMG04].

5.2 Interpolating Data on Line Segments

If the data sites are line segments, there are bisectors between lines, between points, and between points and lines, where endpoints of line segments also count as points. The bisectors are parabolic arcs between the interior of a line segment and a point, while all other bisectors remain linear. In practice, the endpoints of a line segment are treated as separate generators, which leads to a partition of its Voronoi tile into tiles for its directed half edges and its end points, as shown in Figure 6(b).

To account for the continuous nature of the data sites, local coordinates in the transfinite setting have their identity expressed similar to

$$x_0 = \sum_{i\in N_0^X} \lambda_i x_i \; + \sum_{i\in N_0^C} \int_{q\in c_i} \lambda_i(q) q \, |dq|, \tag{5}$$

with $N_0^X \cup N_0^C$ being the union of point shaped and line shaped neighbor indices, and $\lambda_i(q)$ denoting a scalar weight function over the length of c_i. The interpolant thus is

$$f(x_0) = \sum_{i\in N_0^X} \lambda_i z_i \; + \sum_{i\in N_0^C} \int_{q\in c_i} \lambda_i(q) z_i(q) \, |dq|, \tag{6}$$

with $z_i(q)$ being the scalar value distribution over the site.

In [GF99], the interpolation of arbitrary functions along polygons is solved. Each subtile $\mathscr{T}_0' \cap \mathscr{T}_i$ can be interpreted to have a certain thickness above x_i, which is nonzero only where the subtile projects to x_i. The $\lambda_i(q)$ are taken to be this thickness, normalized by the overall area, and define a meaningful density for the accumulation of data values. The application of this interpolant to the data distributed along the non-convex polygon in Figure 6(a) is shown in Figure 6(b).

In [AMG98, AMG04], the same approach has been implemented for arbitrary arrangements of non-intersecting line segments and points. Although they restrict their approach to a linear data distribution along the lines, the approach of [GF99] can also be applied to

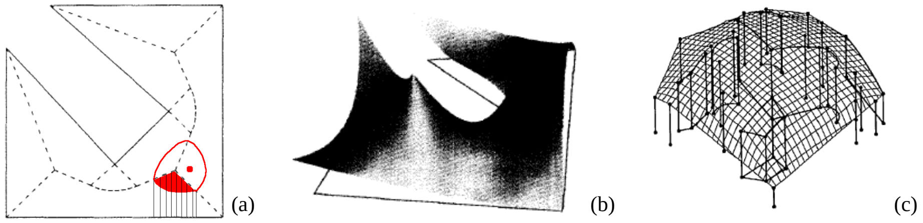

Figure 6: Transfinite interpolation of curves. (a) The Voronoi diagram of a polygon, the contribution of the lower subtile depicted by the thin lines. (b) Transfinite interpolation of the boundary values. (c) Transfinite interpolation of a collection of points, line segments and circular arcs. Pictures (a), (b) courtesy of [GF99], (c) courtesy of [HS00a].

interpolate to arbitrary scalar functions over the sites. By allowing different values on both sides of the line segments, they are able to faithfully model discontinuities as they arise in e.g. geology. See Figure 5(b) for an example.

While the last two approaches focus on a generalization of Sibson's coordinates, [HS00a] generalizes Laplace interpolation to arrangements of multiple classes of curves. The main difference lies in the definition of $\lambda_i(q)$, which can now be interpreted as a density function over the bisectors boundary of $\mathscr{T}_0'$. The result of this interpolant applied to an arrangement of points, line segments, and circles is shown in Figure 6(c).

5.3 Interpolating Data on Circles, Lines and Points

In case the input consists of data distributed over a circle, the tile $\mathscr{T}_0'$ is an ellipsis. Consequently, Sibson's transfinite interpolant takes a simple form. Let x_1 be the circle centered at 0, $z_1(\Theta)$ the data, parameterized over $\Theta \in [0, 2\pi)$, and $x_0 = (\rho, \theta)$ be expressed in polar coordinates with respect to 0. Then in [GF99] a Sibson's transfinite interpolant on circles is defined as

$$f(x_0) = \frac{(1-\rho^2)^{3/2}}{2\pi} \int_0^{2\pi} \frac{z_1(\Theta)}{(\rho\cos(\theta-\Theta)-1)^2} d\Theta \qquad \begin{cases} 0 \le \rho < 1 \\ 0 \le \theta \le 2\pi \end{cases} .$$

Based on a similar idea, [HS00a] formulated an identity and the thereby defined interpolant for Laplace coordinates.

6 Natural Neighbor Coordinates on Manifolds

The Voronoi diagram is defined by a set of points and a distance measure. For points on a manifold, this definition still holds, at the expense of potentially non-convex tiles due to a non-Euclidean metric, see [LL00]. The manifold setting results in bisectors of arbitrary complexity and computing areas (volumes) becomes tedious for non-trivial geometries. To the author's best knowledge, there has been no work carried out on natural neighbor based interpolation on continuous manifolds.

In [BC00], however, it is shown that if the manifold has a sufficiently dense sampling, a less complicated approach is possible. The data sites on the manifold induce a Voronoi diagram in the manifold's embedding space, $\mathbb{R}^n$. The intersection of that Voronoi diagram and the manifold gives a partition of the manifold that locally converges to the Euclidean Voronoi diagram when the sampling density goes to infinity. Furthermore, the main result in [BC00] states that Sibson's identity holds for an infinitely dense sampling of the surface.

Based on this work, natural neighbor based interpolation on point clouds issued from manifolds is developed in [Flö03]. As a main result, a point on a manifold can be expressed in local coordinates in the tangent plane at that point, given the manifold is sampled densely enough. The intersection of the n-dimensional Voronoi diagram of the data sites with the tangent plane defined by the normal vector at the query position produces a power diagram in the tangent plane. [Flö03] proves Sibson's identity for power diagrams and develops natural neighbor coordinates for point clouds.

7 Implementation of Natural Neighbor Interpolation

Natural neighbor based interpolants are based on an underlying identity that provides generalized barycentric coordinates in the natural neighbors. The definition of those local coordinates is motivated geometrically on the Voronoi diagram of the input data sites. The computation, however, can often be carried out in a more elegant and also more stable way. These approaches can be classified as geometric, algebraic and approximate. For simple settings, the geometric approach is still feasible. For higher dimensions, higher orders of continuity and more complex input data sites, algebraic and approximate approaches yield more efficient and more stable solutions.

In the rest of this section we describe the computation of natural neighbor coordinates, since they are the main building block for all interpolants in this survey. The implementation of the C^1 and C^2 constructions at the data sites from Section 4 is straightforward.

7.1 Geometric Computation

The dual to the Voronoi diagram is the Delaunay tessellation. Therefore, evaluation and traversal of the Voronoi diagram of a set of points can be carried out on its Delaunay tessellation, for which the adjacency information is known. One drawback of this approach is the numerical instability in cases where both nominator and denominator in the formulas of Section 3 tend to infinity.

Laplace and Sibson's coordinates relate to areas and volumes of intersections of Voronoi tiles which are easily computed in two dimensions, and implementations are known for three dimensions as well [Owe]. In case of line segment shaped data sites, the constrained Delaunay tessellation can be used. Input data sites of arbitrary shape are difficult to handle in classical geometric data structures and usually require more intricate representations of the Voronoi diagram. The common solution to this is a tessellation of the input data sites, once again allowing for the constrained Delaunay tessellation to be applied.

In three or more dimensions, the data structures required for storing the Delaunay

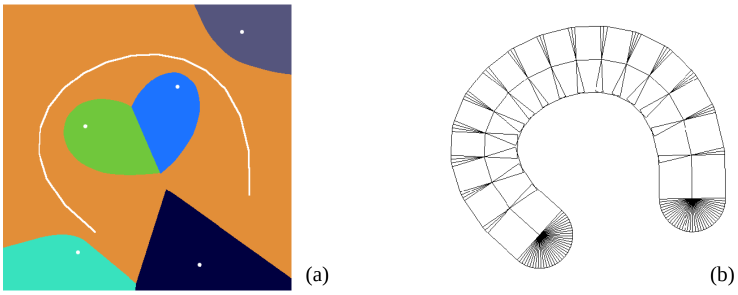

Figure 7: (a) Generalized Voronoi diagram computed on graphics hardware. (b) Polygonal approximation of a distance function. Pictures courtesy of [HCK$^+$99].

tessellation and its adjacency graph become very complex, and traversing the topological neighborhood becomes error prone and cumbersome.

7.2 Algebraic Computation

For algebraic computation the explicit construction of the Voronoi diagram is avoided, and computation is carried out directly on the geometric entities by which it is defined. Note that except for Hiyoshi's approach [Hiy05], also algebraic approaches suffer numerical instabilities when both nominator and denominator tend to infinity.

To compute Laplace coordinates in the two-dimensional setting, the calculation presented by Sugihara only assumes the natural neighbors of the query position to be given in counterclockwise order [Sug99]. The resulting identity also holds in the more general case of star-shaped neighborhoods, making this approach robust against topological inconsistencies as they appear from numerical noise.

Watson [Wat92] explains Sibson's coordinates as the signed decomposition of the area of a triangle, spanned by the circumcenters of the involved triangles. Building on this idea, [Hiy05] proposed a way to stably compute λ^k in $\mathbb{R}^2$ by encoding the construction of Voronoi entities into algebraic expressions in Delaunay entities that circumvent numerical instabilities based on zero denominators as they might appear in the equations in Section 3.

A straightforward computation of Laplace and Sibson's coordinates in any dimension exists once the one-ring Delaunay neighborhood is known. The content of the corresponding tile facets and tile intersections can be expressed as an intersection of half-spaces that are entirely defined by the query position and its Delaunay neighbors [BS95]. Thus, the computation of Laplace and Sibson's coordinates reduces to the determination of Delaunay neighbors [Wat81] and volume computation in n dimensions [BEF00]. This approach has been applied to derive a construction of Hiyoshi's coordinates in $\mathbb{R}^2$ in [BBU06b]. Note that the average number of Delaunay neighbors grows exponentially with dimension, and so does the complexity of volume computations.

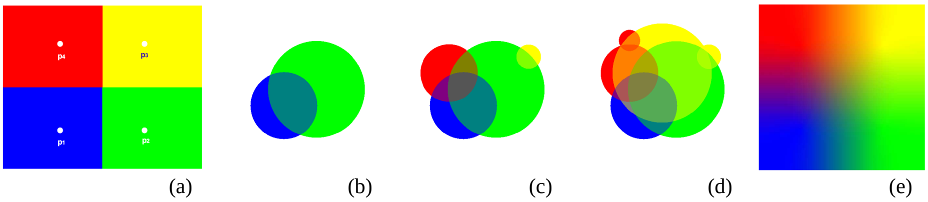

Figure 8: Discrete computation of Sibson's interpolant for the setting in (a). Each point in the domain gives rise to a disc colored with the value of the nearest generator and the distance to that generator as its radius, depicted in (b)-(d). The translucent overlay of all discs is the discrete Sibson's interpolant. Pictures courtesy of [PLK+06].

7.3 Approximate Computation

By allowing a certain error for the local coordinates, an approximate formulation of natural neighbor coordinates can be given based on a discretization of the Voronoi diagram. The discrete version of the two-dimensional Voronoi diagram can efficiently be computed by rendering appropriate primitives to frame- and z-buffer, utilizing the capabilities of recent graphics hardware.

The computation of generalized Voronoi diagrams with the help of graphics hardware is discussed in [HCK+99]. Basically, the distance function of each of the generators is represented by a geometric object. Rendering these leaves the minimum distances in the z-buffer and the associated generator in the color buffer. An example is shown in Figure 7.

The computation of Sibson's coordinates can now be performed by counting pixels in the approximate Voronoi diagram with added query position [FEK+05]. However, this does not allow for an efficient or stable evaluation of Laplace or Hiyoshi's C^k coordinates.

If, instead of evaluating single point queries, a whole field is to be evaluated, the influence of the data values at the generators can directly be distributed to the domain in a more efficient manner. The way described in [FEK+05] requires the Delaunay triangulation to be known, while [PLK+06] do without tessellation at all solely using a k-d tree to provide nearby points. This is illustrated in Figure 8.

8 Summary and Comparison

The properties of all schemes discussed in this paper are summarized in Table 8. A discussion of the four blocks is given below.

Point Based Interpolation Schemes: Schemes with global smoothness have only been proposed for the setting of data sites. Both Farin's and Sibson's C^1 constructions operate on Sibson's coordinates and yield fairly straightforward implementations. Farin's construction has quadratic precision and adapts to a wider range of input constellations.

Hiyoshi's C^2 scheme provides a high quality interpolant but is computationally expensive and tedious to implement even though explicit guidelines for its implementation in $\mathbb{R}^2$ exist. The C^2 construction at the data sites requires the construction of quintic Bézier control nets and bears considerable combinatorial complexity. In our experiments, we found

	Shape of x_i $^{(6)}$	Smoothness of λ in $\mathscr{D} \setminus X$	Deriv. at x_i	Smoothness at x_i	Gener. to $\mathbb{R}^d$	Gener. to C^k	Precision	Comput. Compl.$^{(4)}$	Support$^{(5)}$	Cont. dep. on X	References
Point based											
Lap. coord.	points	$C^{\mathrm{d}-2}$	-	C^0	+	-	linear	++	1	+	[CFL82, Sug99, BIK$^+$97]
Sib. coord.		$C^{\mathrm{d}-1}$	-	C^0	+	-		++	1	+	[Sib81]
Hiy. coord.		C^k	-	C^0	(+)$^{(1)}$	+		-	1	+	[HS00b]
Sibson C^1		C^0	∇	C^1	+	-	s.q.$^{(3)}$	+	1	+	[Sib81, Far90, Pip92]
Farin C^1		C^1	∇	C^1	+	+ $^{(2)}$	quad.	+	1	+	[Far90]
Hiyoshi C^2		C^2	$\nabla, \mathscr{H}$	C^2	(+)$^{(1)}$	+ $^{(2)}$	cub.	–	1	+	[HS04, Hiy05]
Clarkson C^1		C^1 in $\mathscr{D}$	-	C^1	+	-	s.q.$^{(3)}$	?	2	+	[Cla96, Flö03]
Transfinite											
Gross	pol, ci	C^1	-	C^0	-	-	linear	-	1	+	[GF99]
Anton	pt, li	C^1	-	C^0	-	-		-	1	+	[AMG98, AMG04]
Hiyoshi	pt, li, ca	C^1	-	C^0	-	-		-	1	+	[HS00a]
Manifold											
Flötotto	points	C^1	-	C^0	+	-	n.a.	-	1	n.a.	[BC00, Flö03]
Other meth.											
Nearest n.	points	C^{-1}	-	C^∞	+	-	-	++	0	-	[OBSC00]
FEM	points	C^k	-	C^0	+	-	lin.	++	0	-	-
RBF	points	C^∞	-	C^∞	+	+	polyn.	- -	g	-	-

(1) ongoing research. (2) based on the Bézier simplex approach. (3) spherical quadratics. (4) ++ low, - - high.
(5) $\{012\}$-ring, g(lobal). (6) pt=points, li=lines, pol=polygons, ci=circles, ca=circular arcs.

Table 1: Overview of considered interpolation schemes.

considerable increases in computation time for extreme situations with more than 20 natural neighbors, which very likely becomes an issue in higher dimensions. Besides these drawbacks, the C^2 interpolant provided the best results when applied to data representing a smooth function, which has been verified in [BBU06a].

In contrast to the schemes above, Clarkson's construction does not interpolate prescribed derivatives but achieves C^1 smoothness implicitly. Since the final interpolant is a linear combination of data from the two ring neighborhood it is similar to the other C^0 schemes over natural neighbor coordinates, but requires a larger support and results in non-convex coordinates.

Transfinite Interpolation: Research on transfinite natural neighbor interpolation has so far only concentrated on expressing the identity of Laplace and Sibson's coordinates with respect to line- and circle-shaped generators in two dimensions. Consequently, the resulting interpolants remain C^0 across the data sites. In simple cases like line segments and circular arcs, closed form integration is possible, but more general shapes require approximations. In spite of these restrictions the improved flexibility provided by transfinite interpolation is useful e.g. for fault modeling in geosciences.

Interpolation on Manifolds: Sibson's identity holds on smooth manifolds for an infinitesimal sampling, but not necessarily for arbitrary samples of the manifold. The Voronoi diagram in that non-Euclidean metric does not have the same, simple geometric structure.

However, local restriction of the Euclidean Voronoi diagram to the tangent plane reveals the lower-dimensional power diagram, for which Laplace and Sibson's identity hold. As a result, Sibson's interpolant can be used on manifolds with sufficiently dense sampling.
Other Scattered Data Schemes: Many other scattered data interpolation schemes have been proposed, among them finite elements, radial basis functions with global or local support, subdivision and bivariate splines, all of which have advantages in certain applications. Yet, natural neighbor based interpolation offers a unique combination of the properties

- locality,
- support determined by truly automatic neighborhood,
- continuous dependency on positions of input sites.

Radial basis functions offer very good mathematical properties in terms of approximation order and smoothness and do like natural neighbor based schemes not depend on a particular tessellation. But even the construction of a compactly supported interpolant requires the solution of a global linear system. Finite elements can be constructed with high orders of continuity but are defined over one fixed choice of elements, i.e. the tessellation of the domain, and thus do not continuously depend on the positions of the data sites. Similar arguments apply to bivariate splines and subdivision. In the more relaxed setting of scattered data *approximation* approaches like hierarchical B-splines, thin plate splines exist, or moving least squares exist. Of these, only the latter has properties similar to natural neighbor based schemes and can even be integrated with natural neighbor coordinates as a replacement for the inverse distance weights.

9 Conclusion

Natural neighbor based interpolation offers some unique properties that make it appealing in settings where the sample distribution is inhomogeneous or changes over time. Its local support is an advantage in large scale data processing, its automatic neighborhood definition and the continuous dependency on the data site positions is especially interesting for meshless methods in mechanical engineering and computational fluid dynamics, where they have been successfully applied in two- and three-dimensional settings. Sibson's identity as well as Laplace coordinates have been generalized to data sites of arbitrary shape, which is useful in geological and terrain modeling.

The main drawback of natural neighbor based interpolation so far lies in the lack of smoothness of the local coordinates at the data sites. For point shaped data sites, this can be solved. One remedy is a non-convex coordinate construction by Clarkson. Another are schemes that interpolate prescribed derivatives, which are unknown in most settings and must therefore be estimated. Furthermore, local coordinates with higher smoothness so far only exist for two dimensions and are tedious to implement.

Some unsolved aspects about natural neighbor based interpolation remain interesting. The smoothness across the data sites in transfinite interpolation could be improved by adopting one of the approaches from the point shaped setting. The implementation of higher-dimensional local coordinates with C^{2+}-smoothness is an open problem and subject to current research. Furthermore, direct formulas for derived values such as integrals would certainly add to the attractivity of natural neighbor based interpolation.

References

[AMG98] F. Anton, D. Mioc, and C. Gold. Local coordinates and interpolation in a Voronoi diagram for a set of points and line segments. In *Proceedings of the 2nd Voronoi Conference on Analytic Number Theory and Space Tillings*, pages 9–12, 1998.

[AMG04] F. Anton, D. Mioc, and C. Gold. Line Voronoi diagram based interpolation and application to digital terrain modelling. In *ISPRS - XXth Congress, Vol.2*. International Society for Photogrammetry and Remote Sensing, 2004.

[Aur91] F. Aurenhammer. Voronoi Diagrams - A survey of a fundamental geometric data structure. *ACM Computing surveys*, 23(3):345–405, 1991.

[BBU06a] T. Bobach, M. Bertram, and G. Umlauf. Comparison of Voronoi Based Scattered Data Interpolation Schemes. In *Proc. Visualization, Imaging and Image Processing*, 2006.

[BBU06b] T. Bobach, M. Bertram, and G. Umlauf. Issues and Implementation of C^1 and C^2 Natural Neighbor Interpolation. In *Proceedings of the* 2^{nd} *International Symposium on Visual Computing*, 2006.

[BC00] J. Boissonnat and F. Cazals. Natural neighbour coordinates of points on a surface. Technical Report 4015, INRIA-Sophia., 2000.

[BEF00] B. Bueler, A. Enge, and K. Fukuda. Exact volume computation for polytopes: a practical study. *DMV Seminar*, 29:131–151, 2000.

[BIK+97] V. Belikov, V. Ivanov, V. Kontorovich, S. Korytnik, and A. Semenov. The Non-Sibsonian interpolation: A new method of interpolation of the values of a function on an arbitrary set of points. *Comp. Math. and Math. Physics*, 37(1):9–15, 1997.

[BS95] J. Braun and M. Sambridge. A numerical method for solving partial differential equations on highly irregular grids. *Nature*, 376:655–660, 1995.

[CFL82] N. H. Christ, R. Friedberg, and T. D. Lee. Weights of links and plaquettes in a random lattice. *Nuclear Physics B*, 210(3):337–346, 1982.

[Cla96] K.L. Clarkson. Convex Hulls: Some Algorithms and Applications. Presentation at Fifth MSI-Stony Brook Workshop on Computational Geometry, 1996.

[Far90] G. Farin. Surfaces over Dirichlet Tessellations. *Computer Aided Geometric Design*, 7:281–292, 1990.

[FEK+05] Q. Fan, A. Efrat, V. Koltun, S. Krishnan, and S. Venkatasubramanian. Hardware-assisted Natural Neighbor Interpolation. In *Proc. 7th Workshop on Algorithm Engineering and Experiments (ALENEX)*, 2005.

[Flö03] J. Flötotto. *A coordinate system associated to a point cloud issued from a manifold: definition, properties and applications*. PhD thesis, Université de Nice-Sophia Antipolis, Sep 2003. http://www.inria.fr/rrrt/tu-0805.html.

[GF99] L. Gross and G. Farin. A transfinite form of Sibson's interpolant. *Discrete Applied Mathematics*, 93:33–50, 1999.

[HCK$^+$99] K. Hoff, T. Culver, J. Keyser, M. Lin, and D. Manocha. Fast Computation of Generalized Voronoi Diagrams using Graphics Hardware. In *Proceedings of ACM SIGGRAPH 1999*, 1999.

[Hiy05] H. Hiyoshi. Stable computation of natural neighbor interpolation. In *Proceedings of the 2^{nd} International Symposium on Voronoi Diagrams in Science and Engineering*, pages 325–333, 2005.

[Hof99] K. Hoff. Fast Computation of Generalized Voronoi Diagrams Using Graphics Hardware. Siggraph'99 Presentation, 1999. http://www.cs.unc.edu/ geom/voronoi/.

[HS00a] H. Hiyoshi and K. Sugihara. An Interpolant Based on Line Segment Voronoi Diagrams. In *JCDCG*, pages 119–128, 2000.

[HS00b] H. Hiyoshi and K. Sugihara. Voronoi-based interpolation with higher continuity. In *Symposium on Computational Geometry*, pages 242–250, 2000.

[HS04] H. Hiyoshi and K. Sugihara. Improving the Global Continuity of the Natural Neighbor Interpolation. In *ICCSA (3)*, pages 71–80, 2004.

[LL00] G. Leibon and D. Letscher. Delaunay Triangulations and Voronoi Diagrams for Riemannian Manifolds. In *Proceedings of the sixteenth annual symposium on Computational geometry*, pages 341–349, 2000.

[OBSC00] A. Okabe, B. Boots, K. Sugihara, and S. Chiu. *Spatial Tessellations: Concepts and applications of Voronoi diagrams*. John Wiley & Sons Ltd, 2000.

[Owe] S.J. Owen. Subsurface Characterization with three-dimensional Natural Neighbor Interpolation, 1993. Available at `http://www.andrew.cmu.edu/user/sowen/natneigh/index.html`.

[Pip92] B.R. Piper. Properties of Local Coordinates Based on Dirichlet Tesselations. In *Geometric Modelling*, pages 227–239, 1992.

[PLK$^+$06] S.W. Park, L. Linsen, O. Kreylos, J.D. Owens, and B. Hamann. Discrete Sibson Interpolation. In *IEEE Transactions on Visualization and Computer Graphics*, volume 12, pages 243–253, 2006.

[Sib80] R. Sibson. A vector identity for the Dirichlet tessellation. *Mathematical Proceedings of Cambridge Philosophical Society*, 87:151–155, 1980.

[Sib81] R. Sibson. A brief description of natural neighbor interpolation. *Interpreting Multivariate Data*, pages 21–36, 1981.

[Sug99] K. Sugihara. Surface interpolation based on new local coordinates. *Computer Aided Design*, 13(1):51–58, 1999.

[Wat81] D.F. Watson. Computing the n-dimensional Delaunay tessellation with application to Voronoi Polytopes. *The Computer Journal*, 24(2):167–172, 1981.

[Wat92] D.F. Watson. *Contouring - A guide to the analysis and display of spatial data*. Pergamon, 1st edition, 1992.