Issues and Implementation of C^1 and C^2 Natural Neighbor Interpolation

T. Bobach[1] and M. Bertram[2] and G. Umlauf[1]

[1] IRTG, Geometric Algorithms Group, TU Kaiserslautern, Germany
[2] Computer Graphics Group , TU Kaiserslautern, Germany
{bobach,bertram,umlauf}@informatik.uni-kl.de

Abstract. Smooth local coordinates have been proposed by Hiyoshi and Sugihara 2000 to improve the classical Sibson's and Laplace coordinates. These smooth local coordinates are computed by integrating geometric quantities over weights in the power diagram. In this paper we describe how to efficiently implement the Voronoi based C^2 local coordinates. The globally C^2 interpolant that Hiyoshi and Sugihara presented in 2004 is then compared to Sibson's and Farin's C^1 interpolants when applied to scattered data interpolation.

1 Introduction

Scattered data interpolation is the most prominent application for a family of local coordinates in scattered point sets, which were introduced in 1980 by Sibson [1] based on the natural neighbors in Voronoi diagrams. Another, simpler and less smooth set of local coordinates based on natural neighbors has been proposed as non-Sibsonian or Laplace coordinates [2,3,4]. Hiyoshi et al. [5] generalized Laplace and Sibson's coordinates to C^k continuity over the domain except at the data sites. We will call these coordinates for short Hiyoshi's C^k coordinates. Recently, Hiyoshi [6] proposed a stable computation method for his coordinates, taking a different approach than the one we present in this paper.

For scattered data interpolation the discontinuities of the derivative at the data sites can be solved by using a polynomial of the local coordinates. Sibson [7] first proposed a globally C^1 interpolant that interpolates least squares fitted gradients at the data sites. A more general solution was given by Farin [8] who constructed a C^1 interpolant with quadratic precision using multivariate Bézier simplexes. This was extended by Hiyoshi et al. [9] who used higher degree simplexes to build a globally C^2 interpolant based on Hiyoshi's C^k coordinates.

Generalizations of natural neighbor interpolation exist for transfinite interpolation of circles and line segments [10,11,12] and points scattered on manifolds [13]. Efforts have been made in accelerating the computation of Sibson's interpolant using graphics hardware [14,15].

The main contribution of this paper consists of two parts. *First*, we give guidelines for an efficient and robust implementation of Hiyoshi's C^2 coordinates. A comparison to Hiyoshi's approach [6] is not given for we have only

recently learned of it. *Second*, we compare the resulting interpolant to other smooth scattered data interpolants based on local coordinates, and discuss the importance of derivative estimation in this context. Specifically, we compare Sibson's, Farin's and Hiyoshi's interpolants applied to a downsampled crater lake data set and to data sampled from an analytic function.

We start in Section 2 with some facts about power diagrams as a foundation to briefly restate Hiyoshi's C^k coordinates in Section 3. We then focus on the actual implementation of Hiyoshi's C^2 coordinates in Section 4. The estimation of derivatives up to order two for comparison is discussed in Section 5. Finally, we present a comparison of Sibson's and Farin's C^1 interpolants and Hiyoshi's C^2 interpolant in two different settings in Section 6.

2 Voronoi and Power Diagrams

The scattered data interpolation problem for scalars valued functions is stated as follows. Given a set of data sites $X = \{x_i\}_{i=1,\dots,m} \subset \mathbb{R}^2$ and corresponding data $Z = \{z_i\}_{i=1,\dots,m} \subset \mathbb{R}$, find a function $f : \mathbb{R}^2 \to \mathbb{R}$ satisfying $f(x_i) = z_i$. Further constraints can be stated, e.g. interpolation of given derivatives. Texts on scattered data interpolation in general can be found in [16,17].

Local scattered data schemes compute the interpolated value $f(x)$ by taking a linear combination of the data values in a neighborhood $N(x)$ and weights $\lambda_j(x)$ such that $f(x) = \sum_{j \in N(x)} \lambda_j(x)\, z_j$. The following explains the definition of $N(x)$ in case of natural neighbor interpolation.

Denote by $d(\cdot,\cdot)$ the Euclidean distance in $\mathbb{R}^2$. From the data sites X we construct the Voronoi diagram $\mathcal{V}(X) := \{\mathcal{T}_i\}_{i=1,\dots,m}$, which is the partition of space into convex tiles $\mathcal{T}_i$ such that

$$\mathcal{T}_i = \{x \in \mathbb{R}^2 | d(x, x_i) \leq d(x, x_j), i \neq j\}.$$

See [18,19] for a thorough treatment of Voronoi diagrams. Two generators x_i and x_j are called *natural neighbors* if the intersection $\mathcal{T}_i \cap \mathcal{T}_j$ of their associated tiles is not empty. Because generators on the convex hull of X have unbounded tiles, we 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_j by N_j. If at least 3 tiles share a common point, the unique circum-circle through their generators is called *Delaunay circle*, which contains no other generator in its interior.

A power diagram is a Voronoi diagram with weighted generators (x_i, w_i), $w_i \in \mathbb{R}$ and the distance measure $d_p(\cdot,\cdot)$,

$$\big(d_p(x, x_i)\big)^2 = \big(d(x, x_i)\big)^2 - w_i.$$

Then, all notions from the Voronoi diagram carry over: tiles are convex polygons and for uniformly chosen weights, the power diagram coincides with $\mathcal{V}(X)$. For power diagrams tiles can be empty depending on the weights w_i, which does not happen for the special choice of weights w_i in this paper.

Assume we are interested in the natural neighbors N_0 of an arbitrary point $x_0 \in \mathscr{D}$. We treat x_0 as a variable position that moves around in $\mathscr{D}$. Virtually inserting x_0 into X results in a new Voronoi diagram $\mathcal{V}'$ with tiles $\mathcal{T}_i'$. Furthermore, we associate with each generator $x_1, \ldots, x_m$ a zero power weight $w_i := 0$, except for x_0 with power weight $w_0 := -w$. This basically means x_0 loses influence with growing w, and its tile becomes smaller until it vanishes. Note that changing w only affects the shape of $\mathcal{T}_0'(w)$ and $\mathcal{T}_i'(w)$ of its natural neighbors, $i \in N_0$. Although strictly speaking, N_0 also depends on w, we always take it to be as in the standard Voronoi diagram, i.e. $w = 0$. N_0 now only depends on the position of x_0. Expressing x_0 as a linear combination of its neighbors, $x_0 = \sum_{i \in N_0} \lambda_i x_i$, allows the definition of the scattered data interpolant

$$f(x_0) = \sum_{i \in N_0} \lambda_i z_i. \tag{1}$$

Where N_0 is unique, $f(x_0)$ is a C^∞-continuous function. Continuity issues arise whenever N_0 changes, i.e.

1. x_0 crosses a Delaunay circle, or
2. x_0 passes another generator x_i.

The first problem is solved by construction of λ_i, while the second is solved by construction of the interpolant. In the next section we focus on the first problem.

3 Natural Neighbor Coordinates

Natural neighbor coordinates express a point x_0 as a convex combination of its natural neighbors $\{x_i\}_{i \in N_0}$, i.e. $x_0 = \sum_{i \in N_0} \lambda_i x_i$. The weights λ_i are usually derived from the geometry of the Voronoi tiles of x_0 and its neighbors. Laplace coordinates λ_i^0 are defined with respect to the line segments $l_i(w)$ perpendicular to the line connecting x_0 and x_i and bounding $\mathcal{T}_0'(w)$ and the distances $r_i = d(x_0, x_i)$, $i \in N_0$,

$$\tilde{\lambda}_i^0(w) := |l_i(w)|/r_i \quad \text{and} \quad \lambda_i^0(w) := \tilde{\lambda}_i^0(w) / \sum\nolimits_{j \in N_0} \tilde{\lambda}_j^0(w).$$

Laplace coordinates are continuous in $\mathscr{D}$.

Another set of natural neighbor coordinates are Sibson's coordinates λ_i^1 which are C^1 in $\mathscr{D} \setminus X$. Based on the area $v_i(w)$ of $\mathcal{T}_i \cap \mathcal{T}_0'(w)$ they are defined as

$$\tilde{\lambda}_i^1(w) := |v_i(w)| \quad \text{and} \quad \lambda_i^1(w) := \tilde{\lambda}_i^1(w) / \sum\nolimits_{j \in N_0} \tilde{\lambda}_j^1(w).$$

Laplace and Sibson's coordinates are members of a class of natural neighbor coordinates generalized by Hiyoshi et al. [5] to C^k continuity in $\mathscr{D} \setminus X$. The area covered by $\mathcal{T}_0'(w)$ shrinks with growing w until it vanishes for $w_{max} = \inf_w(\mathcal{T}_0(w) = \emptyset)$, see Figure 1(left). Using $v = w_{max} - w$, $k \geq 1$ the recursion

$$\tilde{\lambda}_i^k(u) = \int_0^u \tilde{\lambda}_i^{k-1}(v)\, dv \quad \text{and} \quad \tilde{\lambda}_i^k = \tilde{\lambda}_i^k(w_{max}) \tag{2}$$

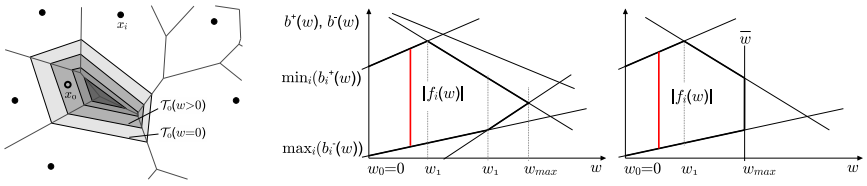


Fig. 1. (left) The Voronoi diagram $\mathcal{V}$ of X (black dots) and the tile $\mathcal{T}_0(w)$ of x_0 for different values of $w \geq 0$. (right) Two examples for $|l_i(w)|$ and its bounding constraints $N_{i,j}y \leq b_{i,j}(w)$.

yields by normalization a relation between the different natural neighbor coordinates,

$$\lambda_i^k := \tilde{\lambda}_i^k / \sum\nolimits_{j \in N_0} \tilde{\lambda}_j^k. \tag{3}$$

These coordinates are called Hiyoshi's coordinates. In $\mathscr{D} \setminus X$ they are C^k in x_0, see [5].

4 Computing Hiyoshi's C^2 Coordinates

In [6] Hiyoshi proposed to compute his coordinates by accumulating the contributions of Delaunay triangles and edges to the value of λ^k. Our approach is different and based on the direct evaluation of equation (2) which requires to express $\lambda^0(w)$ as an analytic expression in x_0 and $\{x_i\}_{i \in N_0}$. Therefore, we first examine how the geometry of $\mathcal{T}_0(w)$ depends on w.

For $w \geq 0$ the distance between $l_i(0)$ and $l_i(w)$ is given by $w/2r_i$. Since the vertices of $\mathcal{T}_0(w)$ are located on the bisectors of $\mathcal{V}$, the length of the line segment $l_i(w)$ is a piecewise linear function on intervals $[w_j, w_{j+1})$, $j \geq 1$,

$$|l_i(w)|_{[w_j, w_{j+1})} = \alpha_{ij} + w\beta_{ij},$$

see for examples Figure 1(right). The tile $\mathcal{T}_0(w) \subset \mathbb{R}^2$ is a convex polygon confined in the intersection of half spaces $n_i x \leq b_i(w), i = 0 \ldots m-1$, $m = |N_0|$ with

$$n_i = (x_i - x_0)/r_i \qquad \text{and} \qquad b_i(w) = c_i + wd_i,$$

with $c_i = (x_i + x_0)n_i/2$ and $d_i = -1/2r_i$. Thus, $\mathcal{T}_0(w)$ can be described by the matrix inequality

$$\mathcal{T}_0(w) = \big\{x \in \mathbb{R}^2 \mid N\,x \leq b(w)\big\}, \quad b(w) \in \mathbb{R}^m, N \in \mathbb{R}^{m \times 2}. \tag{4}$$

Now, l_i is represented by $n_i x = b_i(w)$ which can be used to eliminate $x_j, j = 1, 2$, from (4). This is equivalent to a projection of $l_i(w)$ along the j-th coordinate direction. With a new matrix $N_{i,j}$ and right-hand side $b_{i,j}(w)$ the projected line segment is given by

$$l_{i,j}(w) = \big\{y \in \mathbb{R} \mid \;\; N_{i,j}y \leq b_{i,j}(w)\big\} \tag{5}$$

which is an intersection of intervals in $\mathbb{R}$. Thus, the length of l_i can be computed as

$$|l_i(w)| = |l_{i,j}(w)|/|n_{i,j}|, \tag{6}$$

where $n_{i,j}$ is the j-th component of n_i.

In case there is a line segment $l_k, k \neq i$, parallel to l_i there is an upper bound $\overline{w} > 0$ for w such that $l_{i,j}(w)$ is not empty. This upper bound $\overline{w}$ is given by the row of $N_{i,j}y \leq b_{i,j}(w)$ representing the constraint imposed by l_k. Otherwise, we set $\overline{w}$ to infinity.

Classifying the inequalities in (5) as upper and lower bounds for the interval $l_{i,j}(w)$, distinguished by superscript + and -,

$$y \leq c_k^+ + wd_k^+ =: b_k^+(w) \quad \text{and} \quad y \geq c_l^- + wd_l^- =: b_l^-(w)$$

yield for the length of $l_{i,j}(w)$

$$|l_{i,j}(w)| = \max_{w \leq \overline{w}} \big\{0, \min_k\{b_k^+(w)\} - \max_l\{b_l^-(w)\}\big\}.$$

This can be efficiently computed by appropriate sorting, intersecting and merging. According to (6) the length of $l_i(w)$ is now represented by a list $(w_j, \alpha_{ij}, \beta_{ij})$.

The integral yielding Hiyoshi's C^2 coordinates can be retrieved by applying recursion (2),

$$\tilde{\lambda}_i^2 = \int_0^{w_{max}} \int_0^u \tilde{\lambda}_i^0(w_{max} - v)\, dv\, du. \tag{7}$$

Because $\tilde{\lambda}_i^0$ is piecewise linear this integration can be done per interval $[w_j, w_{j+1})$. Denote by $S_j(u)$ the inner integral of (7) for $u \in [w_j, w_{j+1})$, that is of the form

$$\begin{aligned} S_{-1}(u) &:= 0, \\ S_j(u) &= S_{j-1}(w_j) + \big((u^2 - w_j^2)d_j/2 + (u - w_j)\, c_j\big)/r_i. \end{aligned}$$

This gives for $\tilde{\lambda}_i^2$ the expression

$$\begin{aligned} \tilde{\lambda}_i^2 = \sum\nolimits_j \Big(&(w_{j+1}^3 - w_j^3)\ d_j/6r_i + (w_{j+1}^2 - w_j^2)\ c_j/2r_i + \\ &(w_{j+1} - w_j)\ \big(S_{j-1}(w_j) - (w_j^2 d_j - 2w_j c_j)/2r_i\big)\Big). \end{aligned}$$

Remark 1. Because $|l_i(w)|$ is piecewise linear the recursive integral reduction in [5] is not applicable.

5 Derivative Estimation

The interpolant derived in (1) is as smooth as the coordinate functions. For natural neighbor coordinates, $\lambda_i^k(x_0)$ is only C^0 continuous at the generators.

Sibson proposed an involved construction of a polynomial in $\lambda_i^1(x_0)$ to also interpolate gradients at the generators, thus constructing a globally C^1 continuous interpolant in [7]. Later, Farin used multivariate Bézier simplexes over local coordinates to interpolate arbitrary higher order derivatives at the generators, and presented a C^1 implementation with quadratic precision in [8]. This approach can be generalized to arbitrary degrees of smoothness and applies to all regular, convex barycentric coordinate functions. The C^2 implementation of this idea was used by Hiyoshi to construct a global C^2 interpolant based on $\lambda_i^2(x_0)$ in [9].

All these interpolants depend on derivative information being available, like in the truncated Taylor expansion of f at x_i,

$$Z_i(x) = z_i + \mathbf{g}_i x + \frac{1}{2} x^T H_i x. \tag{8}$$

In order to estimate first order derivatives, we adopt Sibson's weighted least squares (WLS) fit of gradients $\mathbf{g}_i$ [7]:

$$\sum\nolimits_{j \in N_i} s_{i,j} |z_i + \mathbf{g}_i x - z_j|^2 \to \min. \tag{9}$$

The weights for the neighbors are chosen to be $s_{i,j} := \lambda_j^1 / d(x_i, x_j)$ and $s_{i,j} := 1/d(x_i, x_j)$ where λ_j^1 is not defined. Since the interpolation property fixes $f(x_i) = z_i$, fitting $\mathbf{g}_i$ has two degrees of freedom and $|N_i|$ constraints. With an average of six constraints, $\mathbf{g}_i$ is overdetermined and the least squares solution well defined in general.

To estimate derivative information up to order two in a first approach we fit (8) to $\{z_j\}_{j \in N_i}$, finding $\mathbf{g}_i$ and H_i such that

$$\sum\nolimits_{j \in N_i} s_{i,j} |Z_i(x_j) - z_j|^2 \to \min. \tag{10}$$

The five unknowns are only well-defined for $|N_i| \geq 5$ and $\{x_j\}_{j \in N_i}$ in non-degenerate positions. Even for $|N_i| = 5$, the averaging nature of the WLS method is lost and the interpolant becomes very sensitive to the local data configuration, as can be seen in Figure 3(e).

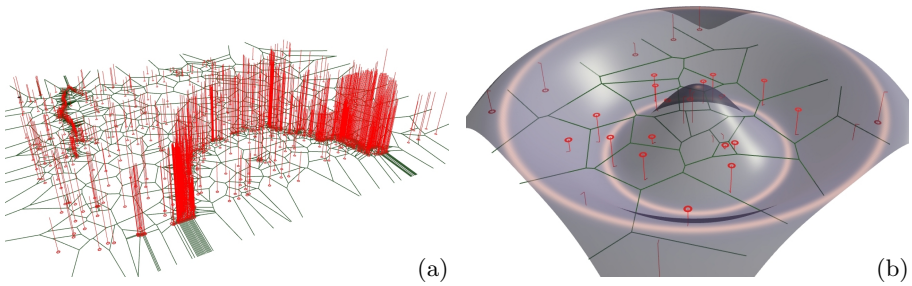
To stabilize the process a larger neighborhood could be used. As Sibson pointed out, the choice of weights for the WLS method is important. But $s_{i,j}$ has no obvious extension for generators other than the natural neighbors.

In a second approach, we employed a different method that naturally incorporates a larger neighborhood and only depends on $s_{i,j}$. First, we determine $\tilde{\mathbf{g}}_i$ as in (10). Then, we find $\mathbf{g}_i$ and H_i such that

$$\sum\nolimits_{j \in N_i} s_{i,j} (|Z_i(x_j) - z_j|^2 + \|\nabla Z_i(x_j) - \tilde{\mathbf{g}}_i\|^2) \to \min. \tag{11}$$

6 Natural Neighbor Interpolation

The scattered data interpolation methods we are going to compare interpolate given scalar values $Z = \{z_i\}_i$, gradients $G = \{\mathbf{g}_i\}_i$, and Hessians $H = \{H_i\}_i$ at the data sites:



(a) (b)

Fig. 2. Input data sets. (a) The original crater lake data set is due to US Geological Survey with $344 \cdot 463$ points. The input to the method is a reduced version with 1493 scattered points. (b) $\cos \|x\|$ sampled at 31 positions.

Sibson's C^0 interpolant for Z: This interpolant is C^0 at X and C^1 in $\mathscr{D} \setminus X$ [7].

Sibson's C^1 interpolant for Z and G: This interpolant is C^1 in $\mathscr{D}$ and reproduces spherical quadratics [7].

Farin's C^1 interpolant for Z and $\tilde{G}$: This interpolant is C^1 in $\mathscr{D}$, and has quadratic precision [8].

Hiyoshi's C^2 interpolant for Z, G, and H: This interpolant is C^2 in $\mathscr{D}$, and has cubic precision [9].

We apply the interpolants and derivative estimation to two data sets. The first is a reduced elevation map of the Crater Lake, see Figure 2(a). The second data set is randomly sampled from the function $\cos \|x\|$ with 0 centered to the shown domain, see Figure 2(b). For both data sets, G in Sibson's and Farin's interpolants has been estimated using equation (9). G and H for Hiyoshi's interpolant have been estimated using both equation (10) and equation (11) to illustrate stability issues. For the analytic data set we furthermore considered the exact values for G and H.

Figure 3(a) shows the Crater Lake data set with piecewise linear interpolation on a Delaunay triangulation for comparison to an established method. Sibson's C^0 interpolant in Figure 3(b) is well suited to capture rough features while Sibson's and Farin's C^1 interpolants in Figure 3(c,d) better deal with smooth regions, giving the best overall impression. There is no noticeable difference between these two methods. If the derivative information for Hiyoshi's C^2 interpolant is aquired from equation (10), strong oscillations occur as in Figure 3(e). Using equation (11) to estimate G and H stabilizes the interpolant in Figure 3(f) but still tends to produce more wriggles than necessary.

The computational complexity of the interpolants is directly related to the order of smoothness they achieve. Sibson's and Farin's C^1 interpolants perform very similar both in quality of the interpolant and in running time, because both interpolate the same gradients and differ only in the polynomial in λ_i^1. Serious performance issues in Hiyoshi's method were observed for the construction of the quintic Bézier control net for large neighborhoods, i.e. $|N_i| > 20$.

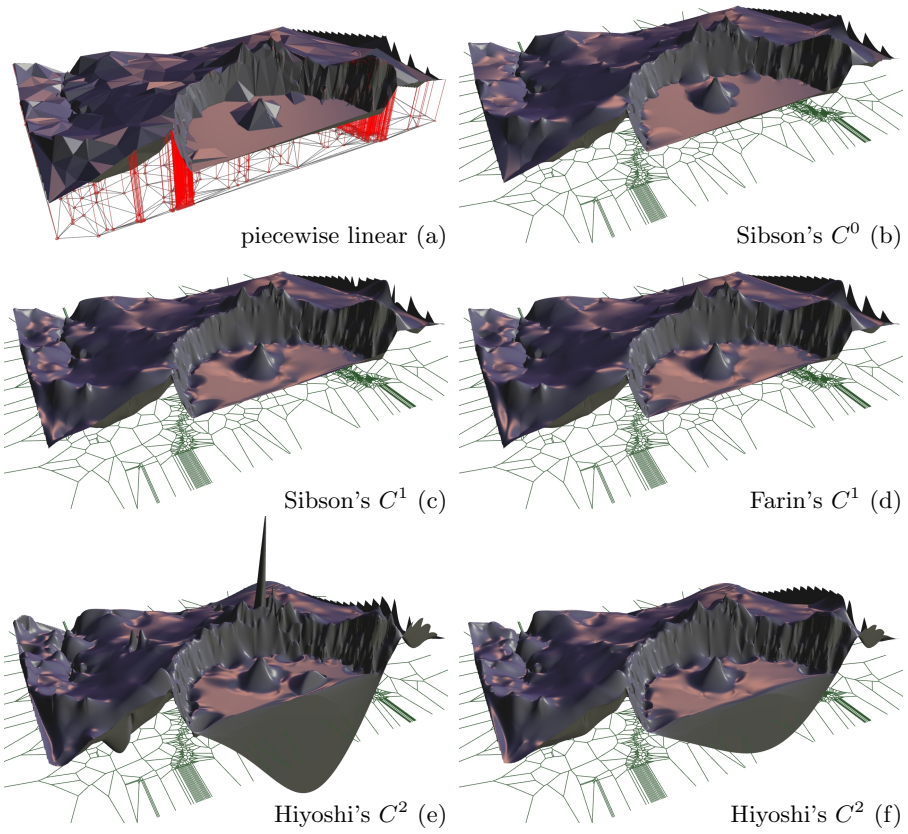


Fig. 3. The crater lake data set with 1493 points is approx. 1% of the original data set. (a) Delaunay tessellation, (b) Sibson's C^0, (c) Sibson's C^1, (d) Farin's C^1, (e),(f) Hiyoshi's C^2 with G,H estimated based on equation (10) and (11).

Since in our experiments, Sibson's and Farin's interpolants show no noticeable difference, the results for the Cosine data set only show Farin's interpolant.

The sample data distribution for the Cosine data set is shown in Figure 2(b). The difference between Farin's interpolant with exact derivatives and with G estimated as in equation (9) is clearly visible in Figure 4(a,b). The sample density in the lower left corner is so low that the gradient could not be reliably estimated, yielding a badly behaved interpolant. In Figure 4(c) derivatives from equation (10) again appear very sensitive and result in Hiyoshi's interpolant being wriggly. Figure 4(d), in contrast, shows Hiyoshi's interpolant with the stabilized method from equation (11). Note how well Hiyoshi's interpolant performs on exact derivative information in Figure 4(e). Sibson's C^0 interpolant in Figure 4(f) is unable to reproduce the smooth underlying function, but it does not oscillate at all.

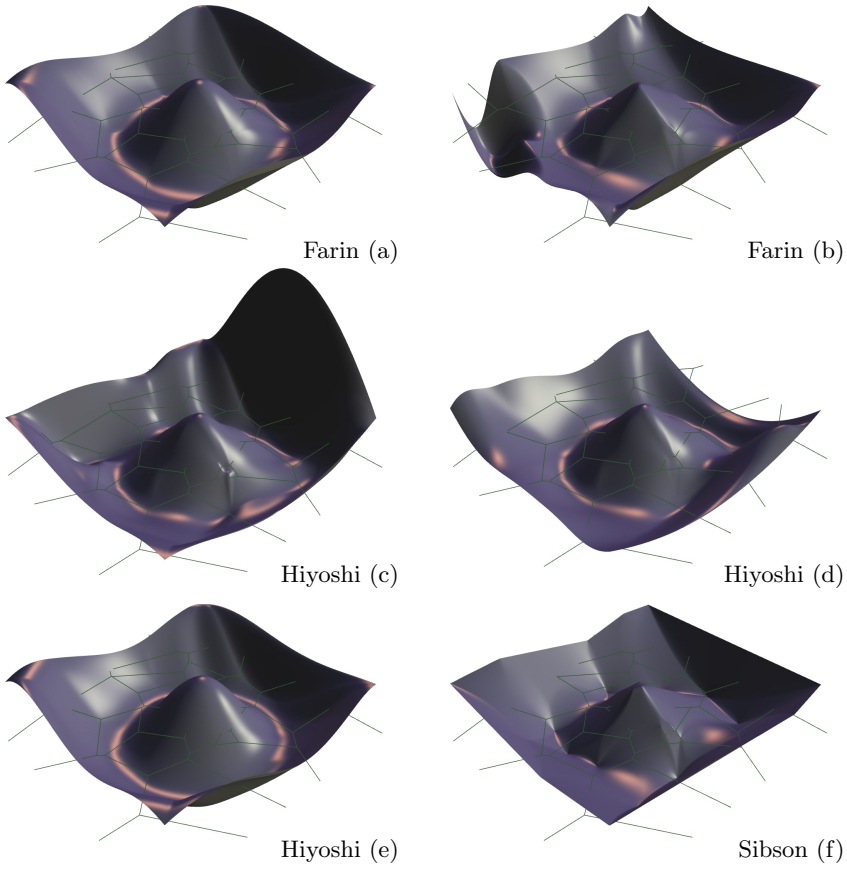


Fig. 4. (a), (b) Farin's interpolant with exact and estimated derivatives. (c), (d) Hiyoshi's interpolant with G, H estimated using (10) resp. (11), (e) Hiyoshi's interpolant with exact derivatives, (f) Sibson's C^0 interpolant.

It must be stressed that although for Sibson's, Farin's and Hiyoshi's interpolants, C^1 and C^2 continuity are guaranteed by construction, the quality of the resulting elevation map very much depends on the given derivative information. With exact derivative information as in the Cosine data set, all interpolants produce results close to the input data, with Hiyoshi's method performing best. Without derivative information available, the outcome differs depending on the nature of the represented data and the method used for derivative estimation.

Especially for data stemming from heterogeneous sources like terrains, there seems to be no best approach, but rather the demand for interpolants with adaptive smoothness. Furthermore, since most scattered data comes without derivative information, more sophisticated estimation of the latter could improve the applicability of the interpolants in practice.

Acknowledgements

This work was supported by the German Science Foundation (DFG grant number GRK 1131) as part of the International Graduate School (IRTG) on "Visualization of Large and Unstructured Data Sets."

References

1. Sibson, R.: A vector identity for the Dirichlet tessellation. Mathematical Proceedings of Cambridge Philosophical Society **87** (1980) 151–155
2. Christ, N.H., Friedberg, R., Lee, T.D.: Weights of links and plaquettes in a random lattice. Nuclear Physics B **210(3)** (1982) 337–346
3. Belikov, V., Ivanov, V., Kontorovich, V., Korytnik, S., Semenov, A.: 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** (1997) 9–15
4. Sugihara, K.: Surface interpolation based on new local coordinates. Computer Aided Design **13** (1999) 51–58
5. Hiyoshi, H., Sugihara, K.: Voronoi-based interpolation with higher continuity. In: Symposium on Computational Geometry. (2000) 242–250
6. Hiyoshi, H.: Stable computation of natural neighbor interpolation. In: Proc. of the 2^{nd} Int. Sym. on Voronoi Diagrams in Science and Engineering. (2005) 325–333
7. Sibson, R.: A brief description of natural neighbor interpolation. Interpreting Multivariate Data (1981) 21–36
8. Farin, G.: Surfaces over Dirichlet tessellations. Computer Aided Geometric Design **7** (1990) 281–292
9. Hiyoshi, H., Sugihara, K.: Improving the global continuity of the natural neighbor interpolation. In: ICCSA (3). (2004) 71–80
10. Gross, L., Farin, G.E.: A transfinite form of Sibson's interpolant. Discrete Applied Mathematics **93** (1999) 33–50
11. Hiyoshi, H., Sugihara, K.: An interpolant based on line segment Voronoi diagrams. In: JCDCG. (1998) 119–128
12. Anton, F., Mioc, D., Gold, C.: 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)
13. Boissonat, J.D., Flötotto, J.: A coordinate system associated with points scattered on a surface. Computer-Aided Design **36** (2004) 161–174
14. Fan, Q., Efrat, A., Koltun, V., Krishnan, S., Venkatasubramanian, S.: Hardware-assisted natural neighbor interpolation. In: Proc. 7th Workshop on Algorithm Engineering and Experiments (ALENEX). (2005)
15. Park, S., Linsen, L., Kreylos, O., Owens, J., Hamann, B.: Discrete sibson interpolation. In: IEEE Trans. on Vis. and Comp. Graphics 12. Volume 2. (2006) 243–253
16. Lodha, S.K., Franke, R.: Scattered data techniques for surfaces. In: Proc. of Dagstuhl Conf. on Sci. Vis., IEEE Comp. Soc. Press (1999) 182–222
17. Wendland, H.: Scattered Data Approximation. Volume 17 of Cambridge Monographs on Appl. and Comp. Math. Cambridge University Press (2004)
18. Aurenhammer, F.: Voronoi diagrams - a survey of a fundamental geometric data structure. ACM Computing surveys **23** (1991) 345–405
19. Okabe, A., Boots, B., Sugihara, K., Chiu, S.N.: Spatial Tessellations: Concepts and applications of Voronoi diagrams. Wiley series in probability and statistics. John Wiley & Sons Ltd (2000)