Line Voronoi diagram based interpolation and application to digital terrain modelling

François Anton ♭♯, Darka Mioc ♮ and Christopher M. Gold √
♭Department of Computer Science
University of British Columbia
203-2366 Main Mall, Vancouver, BC, V6T 1Z4, Canada
fanton@cs.ubc.ca
♯Projet PRISME
INRIA Sophia Antipolis
B.P. 93, 06902 Sophia Antipolis, France
♮Département des Sciences Géomatiques
Université Laval
Pavillon Casault, Ste-Foy, QC, G1K 7P4, Canada
dmioc@cs.ubc.ca
√Department of Land Surveying and Geo-Informatics
The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
ChristopherGold@Voronoi.Com

Abstract

Local coordinates based on the Voronoi diagram are used in natural neighbour interpolation to quantify the "neighbourliness" of data sites. In this paper, we propose a generalisation of the local coordinates and natural neighbour interpolation based on the ordinary Voronoi diagram, to the Voronoi diagram for a set of points and oriented line segments in the Euclidean plane. This generalisation allows one to model linear discontinuities that are not modelable through current interpolation techniques. We present the application of the natural neighbour interpolation based on these local coordinates to digital terrain modelling. In our case, the data sites can be either points or oriented line segments. We use this natural neighbour interpolation in order to interpolate elevations from the neighbouring data sites, and construct a digital terrain model. We show an example of use of this extended interpolation technique for the modelling of linear vertical faults, dams or bridges.

1 INTRODUCTION

Local coordinates based on the Voronoi diagram were introduced in [Sibso80], and they have not been extended to generalised Voronoi diagrams up to now except in [Anton98]. Local coordinates based on the Voronoi diagram have been used in natural neighbour interpolation [Sibso81] (also studied in [Gold94] as stolen area interpolation), to quantify the "neighbourliness" of data sites. The properties of these local coordinates have been extensively studied in [Farin90] and [Piper93]. The natural neighbour or stolen area interpolation technique has been extended from ordinary Voronoi diagrams to Voronoi diagrams for sets of points and line segments in [Anton98]. Anton et *al.* [Anton98] extended the results presented in Gold and Roos [Gold94], by providing direct vectorial formulas for the first order and second order derivatives for the stolen area. The analysis presented in [Anton98] generalises the analysis of Piper [Piper93] based on the formalism of partial derivatives, to the formalism of derivatives of a function on a normed space.

Even though the Voronoi diagram for a set of points and oriented line segments has never been defined formally (i.e. the meaning of oriented line segments), it has been used in Gold [Gold94]. The objects in this generalised Voronoi diagram are either points or pairs of oriented line segments (see section 8.6.1 in [Berge77] for the definition of oriented lines). We review this data structure in Section 2. The representation of line segments in Digital Terrain Modelling (DTM) has been traditionally done using constrained Delaunay triangulations (see [Jones98, Li99]). In all these DTMs, the line segments were represented as constrained edges in the Delaunay triangulation, but their elevation could not be taken into account for the interpolation, because the only objects that had elevations were points. This research brings novelty in the modelling of topographic artifacts represented by line segments (e.g. thalwegs, crests, faults), because in this interpolation technique, line segments are data objects that have an elevation (in fact an elevation for each oriented line segment).

In Section 3, we present the stolen area interpolation technique based on the Voronoi diagram for a set of points and oriented line segments, and in Section 4, we present its application to topographic modelling.

2 THE VORONOI DIAGRAM FOR A SET OF POINTS AND ORIENTED LINE SEGMENTS

Let us first introduce the definition of the Voronoi diagram for a set of sites (i.e. objects) in the Euclidean affine space of dimension n.

Definition 1. Let $\mathcal{O}$ be a set of sites in Euclidean affine space of dimension n. For each site o of $\mathcal{O}$, the Voronoi cell $V(o)$ of o is the set of points that are closer to o than to other sites of $\mathcal{O}$. The Voronoi diagram $V(\mathcal{O})$ is the space partition induced by Voronoi cells.

Let us now consider a set of points and line segments $\mathcal{O} = \{O_1, ..., O_s\}$ in the Euclidean plane. The distance from a point M to an object O_i is defined as: either the Euclidean distance between the two points if the object is a point, or $d(M, O_i) := \inf_{P \in O_i} d_e(M, P)$ where d_e denotes the Euclidean distance between two points, otherwise. The Voronoi cell $V(O_i)$ of O_i is the set of points that are closer (in the sense of the distance between a point and an object defined just above) to O_i than to other sites $O_j : j \neq i$ of $\mathcal{O}$. The Voronoi diagram for a set of points and oriented line segments is the space partition induced by these Voronoi cells; it is a generalised Voronoi diagram.

3 THE NATURAL NEIGHBOUR INTERPOLATION

In this section, we will make a brief introduction to the natural neighbour interpolation work developed by Anton et al. [Anton98]. We have a set $O = \{O_1, ..., O_s\}$ of neighbouring data objects, at which we know the elevation, and we want to interpolate the elevation at some unknown location M in the convex hull of O. If the object is a line segment, we know that the elevation varies linearly on each oriented line segment. In order to interpolate the elevation at M from the values at neighbouring data sites, we compute the local coordinates of M.

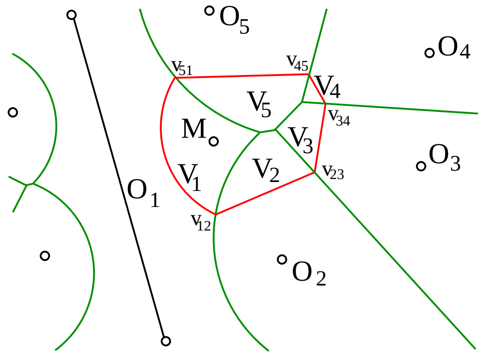

Figure 1: Natural neighbour interpolation

These local coordinates are defined as follows: $u_k(M) = \frac{\lambda_k(M)}{\sum_i \lambda_i(M)}$, where $\lambda_k(M)$ is the area of the intersection (e.g. the region marked as V_2 is the intersection of the old tile of O_2 with the tile of M on Fig. 2) of the "old" tile of O_k and the "new" tile of M. The vectorial expression for the Voronoi vertex (circumcentre) of O_i, O_j, and M is:

$$\overrightarrow{v_{i\,i+1}} = \overrightarrow{m_{i\,i+1}} + \frac{\overrightarrow{O_{i+1}M} \cdot \overrightarrow{O_iM}}{2\overrightarrow{n_{i\,i+1}} \cdot \overrightarrow{O_iM}} \overrightarrow{n_{i\,i+1}},$$

where $m_{i\,i+1}$ is the middle of $[O_iO_{i+1}]$, and $n_{i\,i+1} = \begin{pmatrix} O_{i+1,2} - O_{i,2} \\ O_{i,1} - O_{i+1,1} \end{pmatrix}$ where $O_{i,1}$ designates the x coordinate of the object O_i and $O_{i,2}$ designates the y coordinate of the object O_i.

From this expression, we get that the Voronoi vertex is defined, continuous and differentiable except at data sites, and its derivative at the point M is:

$$D\overrightarrow{v_{i\,i+1}}(M) = \frac{d\overrightarrow{M} \cdot \overrightarrow{v_{i\,i+1}M}}{\overrightarrow{n_{i\,i+1}} \cdot \overrightarrow{O_iM}} \overrightarrow{n_{i,i+1}}.$$

The continuity of the Voronoi vertex can be extended at data points since the limit of the Voronoi vertex when the interpolation point goes towards a data point is the point at infinity in the direction of the bisector.

In order to determine $\lambda_k(M)$, we decompose the corresponding area in triangles (see Fig. 3): $v_{k-1\,k}$, $v_{k\,k+1}$, $C_{k\,1}$ and $v_{k-1\,k}$, $C_{k\,j}$, $C_{k\,j+1}$, where $C_{k\,j}$ is the i^{th} Voronoi vertex of $V(O_k) \cap V(M)$ in the counterclockwise orientation from $\overrightarrow{v_{k-1\,k}v_{k\,k+1}}$ and we get the following result:

$$2\lambda_k(M) = det\left(\overrightarrow{v_{k-1\,k}v_{k\,k+1}}, \overrightarrow{v_{k-1\,k}C_{k\,1}}\right) + \sum_{j=1}^{J_k} det\left(\overrightarrow{v_{k-1\,k}C_{k\,j}}, \overrightarrow{v_{k-1\,k}C_{k\,j+1}}\right).$$

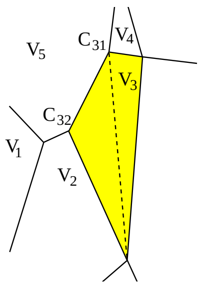

Figure 2: Decomposition into triangles

Therefore, the local coordinates are defined, continuous, and differentiable everywhere except at data sites, and we get: $2D\lambda_k\left(M\right) = det\left(d\overrightarrow{v_{k-1\;k}}, \overrightarrow{C_{k\;J_k} v_{k\;k+1}}\right) + det\left(d\overrightarrow{v_{k\;k+1}}, \overrightarrow{v_{k-1\;k} C_{k\;1}}\right)$. The continuity of the local coordinates can be extended at data points since the limit of the local coordinate when the interpolation point goes towards a data point is 1.

By the chain rule ($D\lambda_k\left(M\right) = \nabla\lambda_k\left(M\right) \cdot d\overrightarrow{M}$), we get the direct formula for the gradient of the area stolen from O_k by M. We extend the local coordinates to line segments, by considering that in the computation of the area stolen by M, we integrate the areas stolen by M, to each portion of length $d\lambda$ of the line segment.

4 THE INTERPOLATION PROCESS

In order to interpolate the elevation at a point, the algorithm locates an edge of the triangle of the dual of the Voronoi diagram (i.e. Delaunay triangulation) for a set of points and oriented line segments in which the given point lies. Then, it determines if the given point is a vertex or lies on the line segment corresponding to a vertex of the triangle in which it lies. If this is the case, it means that the given point is one of the data points, or it lies on a data line segment. In the first case, the elevation at that point is the same as the elevation at the data point. In the second case, we assume that the elevation at the point can take both of the elevations linearly interpolated from the elevations at the extremities of the two oriented line segments. If it is not the case, the algorithm computes the list of objects (corresponding to Delaunay triangulation vertices) from which the point would steal some area if it was inserted in the Delaunay triangulation. This is done without inserting the point in the Delaunay triangulation.

Starting from the located edge, and visiting the three edges of the enclosing triangle, the algorithm tests whether the given edge is safe (an edge is not safe if it should be swapped, resulting in a triangle swap by exchange of the common edge of two adjacent triangles [Guiba85]). If an edge is safe, then it is added to the circular list of safe edges enclosing the interpolated point. If an edge is not safe, the edge having the same origin and immediately after it in the clockwise orientation (the edge pointed by its Oprev operator [Guiba85]), and the edge having the same destination and immediately after it in the counterclockwise orientation (the edge pointed by its Dnext operator [Guiba85]) are successively checked. The safe edges detected by this algorithm are inserted in a circular list in the counterclockwise order. The origin of the edge pointed by the Rot operator [Guiba85] of each edge of this circular list is a natural neighbour of the point being interpolated. Then, the areas stolen by the interpolated point from each associated neighbour are computed. The interpolated elevation is the weighted average of the elevations of the neighbours by the areas stolen from the neighbours. The construction of the Quad-Edge data structure requires $O\left(n\, logn\right)$ worst case time, where n is the number of sampled points. Each interpolation requires $O\left(logn\right)$ amortised worst case time.

5 EXPERIMENTAL RESULTS

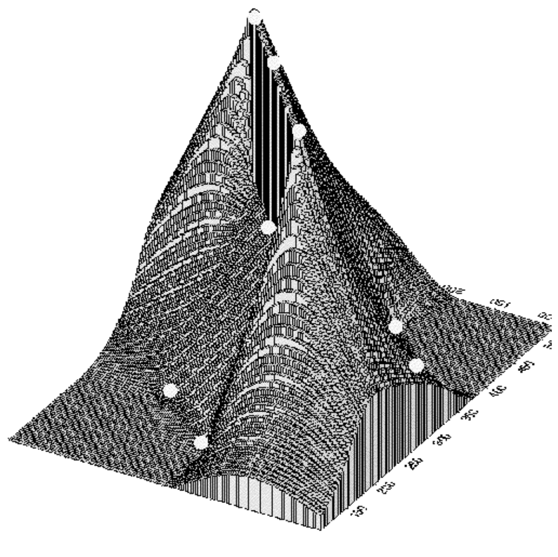

Figure 3: An example of Digital Terrain Model obtained by the natural neighbour interpolation based on the Voronoi diagram for a set of points and oriented line segments, the data objects are marked by plain disks

On Figs. 3 and 4, we can see the results of the use of our natural neighbour interpolation based on the Voronoi diagram for a set of points and oriented line segments for simulating linear discontinuities like faults, bridges, brakes of slope, or dams. Fig. 3 shows

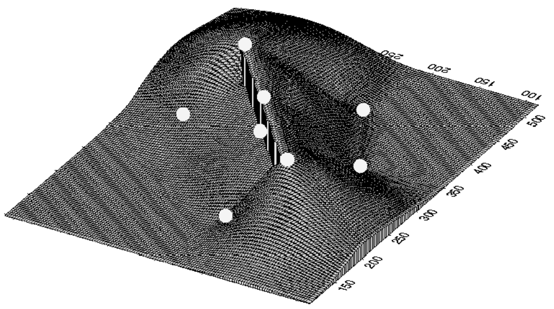

Figure 4: The surface obtained from the same data after using an Hermitian smoothing function, the scale has been reduced in order to see the whole of the top of the surface

the modelization of a vertical fault as the result of the interpolation based on the Voronoi diagram for a set of points and one pair of oriented line segment. Fig. 4 shows the use of a smoothing function on top of the interpolation in order to get a smoother surface. The smoothing function we used is an Hermitian interpolation function [Davis75]. It applies on the local coordinates the following change: the new local coordinate $u'_k(M) = 3u_k(M)^2 - 2u_k(M)^3$.

6 DISCUSSION

We have presented an example of use of the natural neighbour interpolation technique based on the Voronoi diagram for a set of points and oriented line segments for digital terrain modelling. Our future work will try to use this interpolation technique for surface reconstruction.

References

[Anton98] F. Anton, C.M. Gold, and D. Mioc. Local coordinates and interpolation in a Voronoi diagram for a set of points and line segments. In *The Voronoi Conference on Analytic Number Theory and Space Tillings*, pages 9–12, 1998.

[Berge77] Marcel Berger. *Géométrie. Vol. 2.* CEDIC, Paris, 1977. Espaces euclidiens, triangles, cercles et sphères. [Euclidian spaces, triangles, circles and spheres].

[Davis75] Philip J. Davis. *Interpolation and approximation.* Dover Publications Inc., New York, 1975. Republication, with minor corrections, of the 1963 original, with a new preface and bibliography.

[Farin90] Gerald Farin. Surfaces over Dirichlet tessellations. *Comput. Aided Geom. Design*, 7(1-4):281–292, 1990. Curves and surfaces in CAGD '89 (Oberwolfach, 1989).

[Gold94] C.M. Gold and T Roos. Surface modelling with guaranteed consistency - an object-based approach. In J. Nievergelt, T. Roos, H-J. Schek, and P. Widmayer, editors, *Proceedings: IGIS'94: Geographic Information Systems,Monte Verita, Ascona, Switzerland*, volume 884 of *Lecture Notes in Computer Science*, pages 70–87. Springer-Verlag, Berlin, 1994.

[Guiba85] Leonidas J. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *ACM Trans. Graph.*, 4(2):74–123, April 1985.

[Jones98] C. B. Jones and Mark Ware J. Nearest neighbour search for linear and polygonal objects with constrained triangulations. In T. K. Poiker and N. Chrisman, editors, *Proceedings of the 8th International Symposium on Spatial Data Handling*, pages 13–21. International Geographical Union, Geographic Information Science Study Group, 1998.

[Li99] T. S. Li and S. M. Wong. Quadrilateral mesh generation using constrained Delaunay triangulation. *Internat. J. Appl. Sci. Comput.*, 6(1):33–38, 1999.

[Piper93] B. Piper. Properties of local coordinates based on Dirichlet tessellations. In *Geometric modelling*, pages 227–239. Springer, Vienna, 1993.

[Sibso80] R. Sibson. A vector identity for the Dirichlet tesselation. *Math. Proc. Camb. Phil. Soc.*, 87:151–155, 1980.

[Sibso81] R. Sibson. A brief description of natural neighbour interpolation. In Vic Barnet, editor, *Interpreting Multivariate Data*, pages 21–36. John Wiley & Sons, Chichester, 1981.