The weighted farthest color Voronoi diagram on trees and graphs

[Extended Abstract]

Ferran Hurtado, Vera Sacristán
Dept. de Matemàtica Aplicada II
Univ. Politècnica de Catalunya, Barcelona, Spain

Rolf Klein, Elmar Langetepe
Institut für Informatik I
Universität Bonn, Germany

ABSTRACT

Let n point sites be situated on the vertices or edges of a geometric graph G over e edges. Each site can be assigned a multiplicative weight and a color. We discuss the complexity, and provide efficient algorithms for the construction, of the Voronoi diagram in which each point of G belongs to the region of that site which is the closest of the furthest color. Special algorithms are presented for the cases when all colors are identical, when all weights are identical, or when G is a tree.

Keywords

Voronoi diagram, graph, network, locational planning.

1. INTRODUCTION

The Voronoi diagram is one of the most interesting structures in computational geometry; many variants and applications have been described in the surveys [2, 4, 9]. It decomposes a given space, G, into Voronoi regions, one to each element p of a given site set, S, such that the region of site p contains all points of G that are closer to p than to any other site in S. In most cases, G is a space of dimension at least 2.

Some variants of the Voronoi diagram are of particular interest to applications in locational planning. First, each site p can be assigned a positive *weight factor*, c, to model its importance. The distance from p to some point z is then the standard distance $|pz|$ in space G, times c. (Note that a bigger weight factor yields a smaller Voronoi region.) For the Euclidean plane, Aurenhammer and Edelsbrunner [3] have shown that the weighted Voronoi diagram of n point sites can be of complexity $\Theta(n^2)$, and provided an optimal algorithm.

The second variant applies to a situation where each of the n sites represents one of k different types of facilities, like schools, shopping malls, hospitals, etc. A person choosing a new residence might want to have one representative of each type as close by as possible. This problem can be solved by means of the *furthest color* Voronoi diagram. Let us assume that each site is colored according to its type. For each point z of the plane, let $c(i, z)$ denote the closest site of color i. Then, by definition, z belongs to the Voronoi region of the site among $c(1, z), \ldots, c(k, z)$ that is farthest away from z. This distance is minimized by a circle centered at a point on the Voronoi diagram.

For the Euclidean plane, Huttenlocher et al. [7] have shown how to construct this diagram in time $O(kn \log n)$. Abellanas et al. [1] gave efficient algorithms for finding the smallest axis-parallel strip or rectangle containing a point of each color, that were not based on Voronoi diagrams.

In this paper we study both, the weighted and the colored variant, and their combination, in the one-dimensional setting, where G is a geometric tree or graph. So far, only the standard (i. e. unweighted and uncolored) Voronoi diagram on graphs has received attention. In their monograph [9], Okabe et al. introduce the standard Voronoi diagram and discuss its structure. Among other papers, they cite Hakimi et al. [6] who gave an $O(en + m \log m)$ algorithm based on shortest path computations; here e and m denote the number of edges and vertices, respectively. If the sites are vertices of the graph, Mehlhorn [8] avoids the factor n by a more efficient way of using Dijkstra's shortest path algorithm, and Erwig [5] improves on this by giving an $O(e + m + (m - n) \log(m - n))$ algorithm.

Throughout this paper, we consider a geometric tree or graph G of m vertices over e edges, and n point sites that are placed on edges or vertices of G. Just for simplicity we assume that G is planar, and that its edges are realized by straight line segments.

2. WEIGHTED FARTHEST-COLOR VORONOI DIAGRAMS ON A TREE

Consider a tree $T = (V, E, d)$ of size m, where V is the set of vertices, E the set of edges, and $m = |V| = |E| + 1$. For any two points x, y on the tree, $d(x, y)$ indicates the Euclidean distance from x to y along the unique path that connects the two points.

Given a set $S = \{p_1, \ldots, p_n\}$ of n points on the tree, we define the Voronoi diagram of S on the tree, $VD_T(S)$, as a

Table 1: Summary of the results

diagram	measure	line	tree	graph
VD	Complexity	$n-1$	$n-1+m$	$\Theta(n+e)$
	Comp. Time	$\Theta(n\log n)$	$O(m+n\log n)$	$O((n+e)\log n)$
WVD	Complexity	$2n-1$	$\Theta(mn)$	$O(en)$
	Comp. Time	$\Theta(n\log n)$	$O(mn\log n)$	$O(en\log n)$
FCVD	Complexity	$2(n-k)+1$	$2(n-k)+1+m$	$O(ke+n)$
	Comp. Time	$O(n\log n)$	$O(km+n\log n)$	$O((ke+n)\log n)$
WFCVD	Complexity	$O(n\alpha(k))$	$O(mn\alpha(k))$, $\Omega(mn)$	$O(en\alpha(k))$, $\Omega(en)$
	Comp. Time	$O(n\log n+n\alpha(k)\log k)$	$O(mn(\log n+\alpha^2(k)\log k))$	$O(en(\log n+\alpha^2(k)\log k))$

partition $\{VR_T(p_1),\ldots,VR_T(p_n)\}$ of T, where the Voronoi region of each site p_i is the set $VR_T(p_i)$ of all the points x of T that are closer to p_i than to any other site p_j, in terms of the tree distance d. We consider such Voronoi diagram completely described if the following information is known:

- The set of bisector points, i.e. all the boundary points between adjacent Voronoi regions.
- For each bisector point, its location in the tree (the edge to which it belongs, or the vertex it coincides with), the sites of its adjacent Voronoi regions, and the distance to them.
- For each vertex of the tree, it(s) closest site(s) and its distance to it (them).

THEOREM 1. *The Voronoi diagram of a set of n points on a tree of size m has complexity $n+m-1$ and can be computed in optimal $\Theta(m+n\log n)$ time.*

Sketch of the proof: The complexity of the diagram follows from the fact that the Voronoi regions are connected. The construction of the diagram can be obtained by the following algorithm:

1. Sort the sites on each edge, producing the Voronoi partition of each edge between its extreme sites. If there are more than two sites on some edge the two extreme sites are relevant for the next steps.
2. Compute the spanning tree S_T of the remaining sites in T. Let T' be the result of the following operation: First we remove all maximal subtrees u of $T-S_T$; by the *hanging* node $h(u)$ of u we denote the unique node of u that belongs to S_T. Next we replace every every maximal chain of edges of S_T that contains only degree 2 vertices by a single edge of the same length.
3. Partition the remaining sites into two sets of about the same cardinality and compute their Voronoi diagram on the contracted tree T' by divide and conquer.
4. Traverse the original tree, placing the bisectors in their real position, and assign each subtree to the Voronoi region of its hanging node.

The correctness of the algorithm is based on the connectivity of the Voronoi regions, and its complexity analysis strongly relays on the fact that only $O(m)$ sites survive 1 step 1, and that the contracted tree has complexity $O(n)$.

Even on a single line any algorithm producing the Voronoi diagram of a set of n sites solves the ϵ-closeness problem. $\square$

Observation: Notice that for $m=O(1)$ the previuos theorem matches the well known results for the Voronoi diagram of n points on a line.

We now consider the weighted Voronoi diagram $WVD_T(S)$, where each site $p_i\in S$ gets assigned a weight w_i, as a multiplicative factor to its associate distance function. In other words, the weighted Voronoi regions are $WVR(p_i)=\{x\in T\mid w_i{\cdot}d(x,p_i)<w_j{\cdot}d(x,p_j)\ \forall j\neq i\}$. Note that the weighted Voronoi regions do not need to be connected and subtrees do not necessarily belong to the same weighted Voronoi region as their hanging node. Therefore we need more information in the description of the diagrams, i.e. for each edge we keep the bisector points sorted by their order on the edge.

THEOREM 2. *The weighted Voronoi diagram of a set of n points on a tree of size m has complexity $\Theta(nm)$ and can be computed in $O(mn\log n)$ time.*

Sketch of the proof: The complexity of the diagram follows from the known fact that the complexity of the diagram on a line is $O(n)$. In fact, a linear number of bisector points may appear outside the interval given by the two extreme sites and so we can multiply this by adding a branching vertex of degree m in between.

A straightforward adaption of the first algorithm allows to compute the diagram in $O(mn \log n)$ time. $\square$

Observation: The previous bound is tight in the sense that for $m = O(1)$ it fits the known computation results of the weighted Voronoi diagram on a line.

We now generalize the previous Voronoi diagrams in the sense of [1]. Assume that the n sites are classified in k types, modeled by k colors ($k \leq n$). If p denotes a site of color c, its farthest color Voronoi region $FCVR_T(p)$ is formed by all the points x of T for which c is the farthest color and p is the closest c-colored site.

THEOREM 3. *The farthest-color Voronoi diagram of n points of k colors on a tree of complexity m has complexity $2(n-k)+m+1$ and can be computed in $O(km + n \log n)$ time.*

Sketch of the proof: The complexity of the $FCVD$ is $2(n-k)+m+1$, since the regions are connected and the $FCVD$ of n points of k colors on a line can be proved to have complexity $2(n-k)+1$.

As for the construction, the $FCVD$ of n points on a line can be computed in optimal $\Theta(n \log n)$ time by a careful use of the fact that the diagram is the projection of the upper envelope of some monochrome polygonal lines of slopes 1 and -1. The tree structure adds a factor k at each of its vertices. $\square$

Observation: Recall that the above bounds fit the results of Theorem 1 when $k = 1$, as well as the mentioned line results when $m = O(1)$. It is also worth to say that the previous theorem improves a straightforward application of the known results on envelopes [10].

We finally consider the weighted furthest-color Voronoi diagram on a tree, $WFCVD_T(S)$, a generalization of the $FCVD_T(S)$ where each site has a weight assigned, in the usual multiplicative way.

THEOREM 4. *The complexity of the weighted farthest-color Voronoi diagram of n points of k colors on a tree of complexity m is bounded above by $O(mn\alpha(k))$, where $\alpha(k)$ is the inverse of Ackermann's function, and from below by $\Omega(mn)$, and it can be computed in $O(mn(\log n + \alpha^2(k) \log k)$ time.*

Sketch of the proof: For any value of m, n and k, we can construct a tree T, with size m and arbitrary degrees, and a set S of n points of k colors, with an arbitrary number of points of each color, such that the $WFCVD_T(S)$ has complexity $(m-1)n$.

The upper bound is obtained by applying Corollary 6.3 of [10] to each edge of the tree.

The algorithm that constructs the diagram combines the k monochrome weighted Voronoi diagrams. $\square$

Observation: When all the sites are of the same type ($k = 1$) this result matches that of Theorem 2. On the other hand, for the case of the tree being a line ($m = O(1)$), there is still space for a small improvement, since the running time of the algorithm turns out to have an extra $\alpha(k)$ factor.

3. EXTENSION TO GRAPHS

In case of graphs things are a bit different. For example the shortest path between two points is no longer unique. Despite such effects similar arguments can be applied.

Let e denote the number of egdes of a graph $G = (V, D)$. The Voronoi diagram on graphs has complexity $O(n+e)$. Let n_i denote the number of sites of an edge i. Each edge is divided into at most $n_i + 2$ pieces; two bisectors may have sites outside i. Summing up yields $\sum_i (n_i + 2) = n + 2e$.

For computing the diagram we again apply a divide and conquer approach. Divide the set of sites into two subsets of about the same cardinality, compute their Voronoi diagrams on the graph. Fortunately, the merge step can be done in $O(e)$. The site of one of the diagrams wins at a vertex, i.e. it is the closest site to the vertex. This closest site may build new bisectors with some sites of the other diagram. Fortunately, this can be done vertex by vertex, every edge may be concerned.

Similar techniques and adaptions work also for the more complicated cases.

4. REFERENCES

[1] M. Abellanas, F. Hurtado, C. Icking, R. Klein, E. Langetepe, L. Ma, B. Palop, and V. Sacristán. Smallest color-spanning objects. In *Proc. 9th Annu. European Sympos. Algorithms*, volume 2161 of *Lecture Notes Comput. Sci.*, pages 278–289. Springer-Verlag, 2001.

[2] F. Aurenhammer. Voronoi diagrams: A survey of a fundamental geometric data structure. *ACM Comput. Surv.*, 23(3):345–405, Sept. 1991.

[3] F. Aurenhammer and H. Edelsbrunner. An optimal algorithm for constructing the weighted Voronoi diagram in the plane. *Pattern Recogn.*, 17:251–257, 1984.

[4] F. Aurenhammer and R. Klein. Voronoi diagrams. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 201–290. Elsevier Science Publishers B.V. North-Holland, Amsterdam, 2000.

[5] M. Erwig. The graph voronoi diagram with applications. *Networks*, 36:156–163, 2000.

[6] S. L. Hakimi, M. Labbé, and E. Schmeichel. The Voronoi partition of a network and its implications in

location theory. *ORSA J. Comput.*, 4(4):412–417, 1992.

[7] D. P. Huttenlocher, K. Kedem, and M. Sharir. The upper envelope of Voronoi surfaces and its applications. *Discrete Comput. Geom.*, 9:267–291, 1993.

[8] K. Mehlhorn. A faster approximation algorithm for the steiner problem in graphs. *Inform. Process. Lett.*, 27:125–128, 1988.

[9] A. Okabe, B. Boots, K. Sugihara, and C. S-N. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams.* John Wiley & Sons, Chichester, UK, 2nd edition, 2000.

[10] M. Sharir and P. K. Agarwal. *Davenport-Schinzel Sequences and Their Geometric Applications.* Cambridge University Press, New York, 1995.