Approximating Voronoi Diagrams with Voronoi Diagrams

Gary L. Miller
glmiller@cs.cmu.edu

Todd Phillips
tp517@cs.cmu.edu

Donald R. Sheehy
dsheehy@cs.cmu.edu

1 Introduction

The tremendous usefulness of Voronoi diagrams is tempered by their worst-case $O(n^{\lceil d/2 \rceil})$ size blowup. This makes them an obvious target for approximation, and indeed, several methods have been proposed that produce linear size approximations to the Voronoi diagram supporting logarithmic-time approximate nearest neighbor queries. All such methods use quadtrees to approximate the Voronoi cells. But what if the input does not have a "bad" Voronoi diagram? There is a huge gap between the best-case and the worst case complexity. Sometimes, the exact solution is both simpler and more precise than an approximation (Figure 1).

We present a new method for constructing approximate Voronoi diagrams that uses the Voronoi diagram of a superset of the input as an approximation to the true Voronoi diagram. The approximate Voronoi cells are unions of Voronoi cells of the superset. If the input has a Voronoi diagram with good aspect ratio (and thus has linear size) then the approximate Voronoi diagram we produce will simply be the Voronoi diagram of the input. The size of the diagram is $O(n \log \Delta)$ where Δ is the *spread* of the input (the ratio of largest to smallest interpoint distances). Moreover, it supports approximate nearest neighbor queries in time $O(\log \Delta)$. We also discuss methods for eliminating the dependence on the spread in the size and reducing it to $O(\log n)$ for queries. The construction will be based on sparse meshing technology [4].

Formally, a $(1+\varepsilon)$-*approximate Voronoi diagram* of an input set $N \in \mathbb{R}^d$ is a spatial decomposition of $\mathbb{R}^d$ into $n = |N|$ pieces with the property that exactly one input point p_i lies in each piece, V_i, and for each $x \in V_i$, $|x - p_i| \leq (1+\varepsilon)|x - p_j|$ for all $p_j \in N$.

In related work, Har-Peled introduced a method based on a quadtree construction in which the approximate Voronoi cells are unions and differences of quadtree cells[3]. The nearest neighbor search reduces to searching the quadtree and assigning a near input point to each quadtree cell. This approach was later refined by Sabharwal et al. [7]. More quadtree based methods were presented by Arya and Malamatos [1] and again by Arya et al. [2].

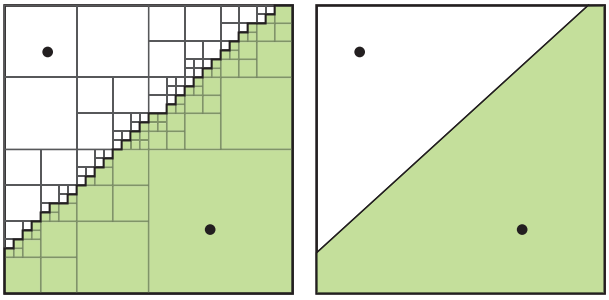


Figure 1: Sometimes, an approximation is not necessary. Here, the diagram on the right achieves both greater simplicity and greater precision.

2 History DAGs

The history DAG is a point location structure that models the incremental changes to a cell complex. We focus on history DAGs over the sequence of Voronoi Diagrams induced by prefixes $P_i \subset N$, where $P_i = \{p_1, \ldots, p_i\}$ for some ordering $(p_1, \ldots, p_n)$ of N (Figure 2, left).

The history DAG has a node (i, j) for each $1 \leq i \leq j \leq n$ corresponding to $\mathrm{Vor}_j(p_i)$, the Voronoi cell of p_i in the Voronoi diagram of P_j. It has a directed edge from (i, j) to $(k, j+1)$ if and only if $\mathrm{Vor}_j(p_i) \cap \mathrm{Vor}_{j+1}(p_k) \neq \emptyset$.

Nearest neighbor queries are answered by searching the DAG (Figure 2, right). The longest possible path in the history DAG is known as the *depth* of the DAG and bounds the worst-case time for a query.

If δ is the maximum degree of any Voronoi cell at any time during the incremental construction, then the history DAG has $O(\delta m)$ vertices and edges.

3 Construction

Let $\mathrm{AVD}(N)$ denote the approximate Voronoi diagram constructed for input $N \subset \mathbb{R}^d$. We use the Sparse Voronoi Refinement meshing algorithm (SVR) to produce a superset $M \supset N$. SVR can be instrumented to keep track of the nearest neighbor among the inputs of each new vertex.

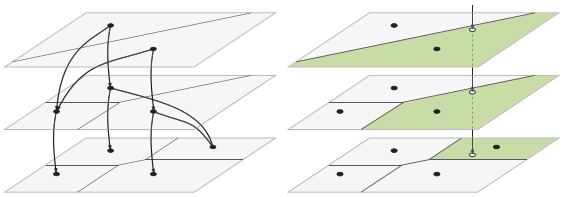


Figure 2: **Left:** The link structure of a simple history DAG. **Right:** A search through the history DAG.

SVR guarantees that the Voronoi cells all have some bound on their aspect ratio. Let $f_S(x)$ be the distance to the second-nearest neighbor of x from the set S. (x is it's own nearest neighbor if $x \in S$. Define the $R_v := \max_{x \in Vor_M(v)} |xv|$[1]. By over-refining the mesh we can guarantee that for all $v \in M$,

$$R_v \leq \frac{\varepsilon}{\varepsilon + 2} f_N(v). \tag{1}$$

The history DAG of the mesh is constructed as part of the SVR algorithm. After the algorithm terminates, the history DAG is kept and used to answer nearest neighbor queries. The depth of the history DAG is guaranteed to be $O(\log \Delta)$ and the degree δ is bounded by a constant τ independent of n.

An approximate nearest neighbor query for N is answered by finding the nearest neighbor among the mesh vertices, M, and returning the nearest input point to that mesh vertex. Indeed the approximate Voronoi cell of a vertex $p \in N$, $\mathrm{AVD}(p)$ is the union of Voronoi cells in the mesh of all mesh vertices in $Vor_N(p)$.

4 Analysis

Theorem 4.1. *Let $M \supset N$ be the superset of the input constructed for $\mathrm{AVD}(N)$. For all x, $\mathrm{NN}_N(\mathrm{NN}_M(x)) \leq (1+\varepsilon)\mathrm{NN}_N(x)$.*

Proof. Let u be the nearest neighbor of x among N. Let v be the nearest neighbor of x from M and u' be the nearest neighbor of v from N, then we seek to show $|xu'| \leq (1+\varepsilon)|xu|$.

If $u = u'$ or $v \in N$, we are done, so suppose $v \notin N$ and $u \neq u'$, thus $f_N(v) \leq |v - u|$. This bound on f and Equation 1 together imply that

$$|vx| \leq R_v \leq \frac{\varepsilon}{\varepsilon+2} f_N(v) \leq \frac{\varepsilon}{\varepsilon+2}|vu| \tag{2}$$

$$\leq \frac{\varepsilon}{\varepsilon+2}(|vx| + |xu|) \leq \varepsilon|xu|. \tag{3}$$

[1]Slight modification is necessary at boundaries.

Collecting terms from this equation then yields $|vx| \leq \varepsilon|xu|$. This fact, the triangle inequality, and the observation $|vu'| \leq |vu|$ imply

$$|xu'| \leq |xv| + |vu'| \leq |xv| + |vu| \tag{4}$$

$$\leq 2|xv| + |xu| \leq (1+\varepsilon)|xu|. \tag{5}$$

□

The following Theorem follows in a straightforward way from the properties of the SVR algorithm [4].

Theorem 4.2. *The size of $AVD(N)$ is $O(n \log \frac{\Delta}{\varepsilon})$. The depth of the corresponding history DAG is $O(\log \frac{\Delta}{\varepsilon})$. Both terms suppress constants that may be exponential in the dimension.*

5 Extensions

We believe it is possible to extend these results to achieve fully linear size approximate Voronoi diagrams by applying ideas from linear-size meshing [5]. We also believe it is possible to replace the $\log \frac{\Delta}{\varepsilon}$ with $\log \frac{n}{\varepsilon}$ by applying methods from geometric separator theory [6]. This is an area of ongoing research.

References

[1] S. Arya and T. Malamatos. Linear-size approximate Voronoi diagrams. In *SODA*, 2002.

[2] S. Arya, T. Malamatos, and D. M. Mount. Space-efficient approximate Voronoi diagrams. In *STOC*, 2002.

[3] S. Har-Peled. A replacement for Voronoi diagrams of near linear size. In *FOCS*, 2001.

[4] B. Hudson, G. Miller, and T. Phillips. Sparse Voronoi Refinement. In *Proceedings of the 15th International Meshing Roundtable*, pages 339–356, Birmingham, Alabama, 2006. Long version available as Carnegie Mellon University Technical Report CMU-CS-06-132.

[5] G. L. Miller, T. Phillips, and D. R. Sheehy. Linear-size meshes. In *CCCG: Canadian Conference in Computational Geometry*, 2008.

[6] G. L. Miller, S.-H. Teng, W. Thurston, and S. A. Vavasis. Finite-element meshes and geometric separators. *SIAM Journal on Scientific Computing*, 19(2):364–386, March 1998.

[7] Y. Sabharwal, N. Sharma, and S. Sen. Nearest neighbor search using point location in balls with applications to approximate Voronoi decompositions. *Journal of Computer and Systems Sciences*, 2006.