On Farthest-Site Voronoi Diagrams of Line Segments and Lines in Three and Higher Dimensions*

Gill Barequet†  Evanthia Papadopoulou‡

Abstract

We show that the number of 3-dimensional cells in the farthest-site Voronoi diagram of \( n \) segments (or lines) in \( \mathbb{R}^d \) is \( \Theta(n^2) \) in the worst case, and that the diagram can be computed in \( O(k \log n) \) time, where \( k \) is the complexity of the diagram, using \( O(k) \) space. In \( \mathbb{R}^d \), the number of \( d \)-dimensional cells in the diagram is \( \Theta(n^{d-1}) \) in the worst case.

1 Introduction and Preliminaries

We address farthest-site Voronoi diagrams of \( n \) lines or segments in \( \mathbb{R}^d \) (\( d \geq 3 \)). (The complexity of the respective nearest-neighbor diagrams is a long-standing open problem; the known bounds are \( \Omega(n^2) \) and \( O(n^{d+1}) \).) We show that the number of \( d \)-dimensional cells (and any unbounded feature of any dimension) of both diagrams is \( \Theta(n^{d-1}) \) in the worst case, and that they can be constructed in three dimensions in \( O(k \log n) \) time, where \( k \) is the complexity of the diagram, using \( O(k) \) space. Our main tool is the Gaussian map of the cells of the diagrams.

By the term “head” of a ray \( r \) we mean an infinite connected portion of \( r \) free of any intersections with features of the diagram under consideration.

Definition 1 Let \( M \) be a subdivision of \( \mathbb{R}^d \) in which: (1) all \( d \)-D cells are unbounded; (2) heads of all parallel rays (not in \((d-1)\)-D cells) are contained in the same cell. The association of rays with cells implies a subdivision of the \( d \)-D sphere of directions, called the Gaussian Map of \( M \) and denoted by \( \text{GM}(M) \).

Let \( \text{FVD}(O) \) denote the farthest-site Voronoi diagram of a set \( O \). Throughout the paper we will consider sets of \( n \) line segments or lines. The following is a generalization of a similar observation in \( \mathbb{R}^2 \) [2].

Figure 1: Supporting planes

Observation 1 Let \( S \) be a set of segments in \( \mathbb{R}^d \). The skeleton of \( \text{FVD}(S) \) is a directed acyclic graph (DAG),\(^3\) having only unbounded cells.

\[ \square \]

2 Segments in \( \mathbb{R}^3 \)

We make two general-position assumptions: (1) No plane passes through four endpoints of (not necessarily distinct) segments; and (2) No point in space is equidistant from five segments.

Definition 2 Let \( S \) be a set of segments in \( \mathbb{R}^3 \). A plane \( P \) is called a supporting plane of \( S \) if (i) \( P \) passes through three endpoints of (either three or two) segments of \( S \); (ii) One closed halfspace bounded by \( P \), denoted by \( P^+ \), fully contains the (3 or 2) segments subject of (i); and (iii) The other open halfspace, \( P^- \), intersects the interiors of all other segments in \( S \).

The endpoints of \( s \in S \) are denoted by \( s^{(1)} \) and \( s^{(2)} \). Let \( P \) be a supporting plane. At infinity, along the direction \( \vec{r} \) perpendicular to \( P \) and within \( P^- \), we find points equidistant from the supported segments, the farthest neighbors of the points along \( \vec{r} \). Figures 1(a,b) show supporting planes passing through three endpoints of three or two distinct segments, resp.

In the full version we show that \( \text{FVD}(S) \) has a tree-like structure. We define the Gaussian map of \( \text{FVD}(S) \) as follows. By Observation 1, every direction in \( \mathbb{R}^3 \) can be associated with an unbounded region in \( \text{FVD}(S) \), imposing a planar subdivision, \( \text{GM}(\text{FVD}(S)) \), on the sphere of directions, having two types of vertices: (1) 3-leg: corresponding to 3 endpoints of 3 distinct segments; and (2) Tricycle:

\(^*\)Work on this paper by the second author has been supported in part by the Swiss National Science Foundation grant SNF 20GG21-134355, under the ESF EUROCORES program EuroGIGA/VORONOI. Our preliminary note [3] contains much weaker results.

\(^†\)Dept. of Computer Science, The Technion—Israel Inst. of Technology, Haifa 32000, Israel, barequet@cs.technion.ac.il

\(^‡\)Faculty of Informatics, Università della Svizzera italiana, Lugano, Switzerland, evanthia.papadopoulou@usi.ch

\(^3\)This needs a proper orientation of the edges, as explained in Section 2.2.2.
corresponding to 3 endpoints of 2 segments. Edges in GM(FVD(S)) are traced by two types of plane rotations: (I) About a segment; and (II) About the line defined by 2 endpoints of 2 distinct segments. Hence, we also have two types of edges referred to as edges of Type I and Type II. Edges of Type I connect pairs of tricycle vertices, and they do not correspond to unbounded 2-D faces (bisectors) in the Voronoi diagram. Edges of Type II correspond to bisectors of pairs of segments. The structures which include Type-I features are called the augmented Gaussian map and Voronoi diagram. Figure 2 shows a schematic example of such a map. The faces of \( s \in S \) can have two types of subfaces in the augmented map, one for each endpoint of \( s \).

### 2.1 Complexity

**Theorem 1** Let \( S \) be a set of \( n \) line segments in \( \mathbb{R}^3 \). The worst-case complexity of GM(FVD(S)) is \( \Theta(n^2) \).

**Proof.** Lower bound. We show the lower bound by an example. Let \( S \) contain \( n \) segments connecting almost-antipodal points on a sphere. Consider the great circles traced on GM(FVD(S)) by rotating planes around the segments. All planes, in all orientations, are supporting. The great circles intersect in \( \Theta(n^2) \) points, all of which are features in the map.

**Upper bound.** The upper bound is obtained by a standard duality transformation which considers the upper envelope of the dual entities of segments, i.e., 3-D wedges that fully contain the halfplanes bounded from above by their apaxes. Refer to Figure 3. Let us count the vertices of the envelope found on the slopes of the wedges. Given a slope of a wedge \( w \) with apex \( a \), the shape of the intersection of another wedge with \( w \) is either (for \( w_1 \)) a 2-D wedge or (for \( w_2 \)) a wedge trimmed from above by \( a \). The complexity of the upper envelope \( E \) of the (full) 2-D wedges is \( \Theta(n) \) in the worst case [4]. Trimming \( E \) by \( a \) at most doubles the complexity of \( E \), and anyway the trimming portions of \( a \) do not appear in the final envelope. In addition, another wedge \( w' \) may partially or fully contain the apex \( a \), in which case \( w' \) takes over \( w \) and features are then counted on the slope of \( w' \). Hence, the slope of \( w \) contains \( O(n) \) vertices of the upper hull of the wedges, for a total of \( O(n^2) \) vertices on all slopes. \( \square \)

In fact, a single segment can capture the complexity of the entire Gaussian map.

**Theorem 2** Let \( S \) be a set of \( n \) line segments in \( \mathbb{R}^3 \). The complexity of the region of a single segment of \( S \) in GM(FVD(S)) is \( \Theta(n^2) \) in the worst case.

**Theorem 3** Let \( S \) be a set of \( n \) line segments in \( \mathbb{R}^3 \). The number of 3-dimensional cells in FVD(S) is \( \Theta(n^2) \) in the worst case.

**Proof.** Since all 3-dimensional cells in FVD(S) are unbounded (Observation 1), and there are no “tunnels” in FVD(S) (this is shown in the full version), there is a bijection between 3D cells in FVD(S) and 2D faces in GM(FVD(S)). By Theorem 1, the number of the latter is \( O(n^2) \), and this quantity is attained in the worst case. The claim follows. \( \square \)

### 2.2 Algorithms

#### 2.2.1 Computing the Gaussian map

First, we compute GM(FVD(S)). As in two dimensions [7], this can be done by using divide-and-conquer. In the merge step, we need to combine two Gaussian maps.

One crucial point to note is that if the complexity of each map is quadratic with \( n \), then so is the complexity of their overlay. Indeed, every intersection of two edges of the maps \( M_1, M_2 \) can be charged to either a unique original vertex that does not exist in the final map, or to a unique new vertex in the resulting map. As a result, identifying the intersection points can be done in \( O(n^2) \) time [5].

A second point to note is how we compute the merged map from the overlay of \( M_1, M_2 \). In constant time we classify each original vertex as “valid” or “invalid” according to whether it remains in the merged map. Given a vertex \( v \in M_1 \) (representing a supporting plane \( \pi \)), we know from the overlay which face \( f \in M_2 \) contains \( v \). The face \( f \) corresponds to an endpoint \( p \) of a segment in \( S \). The validity of \( v \) is determined by which of the two halfspaces delimited by \( \pi \) contains \( p \). We also need to find new vertices of the combined map, which can appear only on existing overlay edges. At most one new vertex appears
between two existing (old or overlay) vertices. The identities and locations of new vertices are determined by the validity/invalidity of neighboring old vertices. While detecting new vertices, we also find new edges, thus, we can also form the faces of the combined map. This can also be done in $O(n^2)$ time.

Hence, the time needed to compute the Gaussian map satisfies the recursion $T(n) = 2T(n/2) + O(n^2)$, whose solution is $O(n^2)$.

### 2.2.2 Computing the Voronoi diagram

Second, we compute $\text{FVD}(S)$ when $\text{GM(\text{FVD}(S))}$ is already known. To this aim we generalize the two-dimensional method of Aurenhammer et al. [2], resulting in a $O(k \log n)$-time algorithm, where $k$ is the complexity of the diagram.

The weight of every point $p$ in space is the radius of the smallest ball centered at $p$ and intersecting (or touching) all segments in $S$. Like in two dimensions [2, 8], the graph structure of the diagram can be regarded as a rooted DAG (with leaves at infinity), where the root is the point with minimal weight. Any path on the DAG, leading from a leaf to the root, consists of points in decreasing order of weight.

We start from $g = \text{GM(\text{FVD}(S))}$ and “collapse” it inward, discovering Voronoi vertices in order. The weight of an event (a candidate vertex) is the radius of the minimum sphere touching (from the outside) the four segments associated with the vertex and not intersecting any other segment. The events are stored in a priority queue, $Q$, and are handled in a decreasing order of weight. $Q$ is initialized by all faces in $\text{GM(\text{FVD}(S))}$, corresponding to unbounded cells in $\text{FVD}(S)$. A realized vertex is the meeting point of three Voronoi faces, represented in $g$ by three edges. The algorithm maintains $\text{FVD}(S)$, $g$, and $Q$, processing two types of events (Figure 4). Events of both types result in a vertex shared by 4 edges and 6 faces. The difference is in the number of features which exist/remain before/after handling the event. Each of the $O(k)$ events triggers a constant number of operations, each of which is handled in $O(\log n)$ time.

In summary, we have:

**Theorem 4** The farthest-site Voronoi diagram of $n$ line segments in $\mathbb{R}^3$ can be computed in $O(k \log n)$ time, where $k$ is the complexity of the diagram, using $O(k)$ space.

### 3 Lines in $\mathbb{R}^3$

We now analyze the Gaussian map and farthest-site Voronoi diagram of a set $L$ of $n$ lines in $\mathbb{R}^3$. The lines are assumed to be in general position in the sense that no two lines are parallel. As with segments, standard perturbation arguments show that this assumption is not necessary since the worst-case complexity is never attained in a degenerate situation. The map $\text{GM(\text{FVD}(L))}$ is defined as for segments, however, in this case we have rotations of planes of Type I only, i.e., rotations around lines in $L$. All these planes are supporting planes. Instead of rotations of Type II we have the bisectors of the respective lines. While rotating a plane around a line $\ell \in L$, the directions normal to the plane trace a great circle $c(\ell)$ on the sphere of directions. Along these directions we find points for which $\ell$ is the farthest out of all lines in $L$. An intersection point of two great circles $c(\ell_1), c(\ell_2)$, for $\ell_1, \ell_2 \in L$, is a vertex with anomaly in $\text{GM(\text{FVD}(L))}$. Although this point corresponds to a direction perpendicular to two parallel planes containing $\ell_1$ and $\ell_2$, clearly one of them is farther than the other line from points along the common normal. Each such vertex with anomaly is actually the intersection of four great circles: $c(\ell_1)$, $c(\ell_2)$, and the two circles representing the bisection of $\ell_1$ and $\ell_2$ in $\text{GM(\text{FVD}(L))}$.

**Theorem 5** Let $L$ be a set of $n$ lines in $\mathbb{R}^3$ in general position. The complexity of $\text{GM(\text{FVD}(L))}$ is $\Theta(n^2)$.

**Proof.** The lower bound is set by the vertices with anomaly. The upper bound is set as follows. Let $p_1 = \pi_1(\ell), p_2 = \pi_2(\ell)$ denote the two antipodal points (on the sphere of directions) representing a line $\ell \in L$. In addition, denote by $r$ the point representing a direction $\vec{r}$. The line in $L$ farthest from points along $\vec{r}$ (close to infinity) is defined by the angles formed by $\vec{r}$ and the lines. Specifically, $r$ is in the region of $(p_1, p_2)$ for which $\min(d(r, p_1), d(r, p_2))$ is maximized over all $\ell \in L$, where $d(r, p_1)$ is the spherical distance. Assume w.l.o.g. that $d(r, p_1) \leq d(r, p_2)$. Then, by definition, $d(r, p_1) \leq \pi/2$ and $d(r, p_2) \geq \pi/2$. Therefore, the sought map is the $n$th level in the arrangement of the Voronoi surfaces corresponding to the $2n$ points on the sphere, whose complexity is $O(n^2)$.

Likewise, the number of 3-D cells in the Voronoi diagram is also quadratic with the number of lines.

**Theorem 6** Let $L$ be a set of $n$ lines in $\mathbb{R}^3$ in general position. The number of 3-dimensional cells in $\text{FVD}(L)$ is $\Theta(n^2)$.
Proof. The lower bound is implied by the lower bound on the complexity of the Gaussian map. The upper bound is shown similarly to that for segments (Thm. 3): The number of 3-D cells in the Voronoi diagram is equal to the number of faces in the Gaussian map. Applying Theorem 5 completes the proof. □

Remarks
(1) Another way to show the upper bound on the number of cells in $FVD(S)$ is by growing lines by scaling up continuously segments of a unit length. Occasionally the number of cells changes, but it can never exceed its worst case, $\Theta(n^2)$. Thus, the limit number of cells cannot exceed that of any finite scale of the segments.
(2) One can formulate $GM(FVD(L))$ by defining a “cluster” for every pair of points that represent a line, and computing the so-called “farthest color Voronoi diagram” of these clusters. The complexity of this diagram is $O(NK)$ [1], where $N$ is the total number of points and $K$ is the number of clusters. In our case $N = 2n$ and $K = n$, implying the claimed bound.

Regarding the Gaussian map as the farthest-site Voronoi diagram of clusters of size 2, we can apply the method of Huttenlocher et al. [6] which runs in $O(V \log V)$ time, where $V$ is the number of vertices in the map, yielding an $O(n^2 \log n)$-time algorithm. However, as for line segments, we can compute the Gaussian map by D&C, where the main difference is in the constant-time validity/invalidity check of vertices. We get the same recursive formula which solves to $\Theta(n^2)$. Furthermore, given the Gaussian map, one can compute the Voronoi diagram in additional $O(k \log n)$ time, where $k$ is the complexity of $FVD(L)$, exactly as for segments. We summarize:

Theorem 7 The farthest-site Voronoi diagram of $n$ lines in $\mathbb{R}^d$ can be computed in $O(k \log n)$ time, where $k$ is the complexity of $FVD(L)$, using $O(k)$ space. □

4 Higher Dimensions

Again, the Voronoi diagrams have a tree-like structure, and so we can apply the same machinery.

Theorem 8 Let $S$ be a set of $n$ segments in $\mathbb{R}^d$ ($d$ fixed). The worst-case compl. of $GM(FVD(S))$, as well as the number of $d$-D cells in $FVD(S)$, is $\Theta(n^{d-1})$.

Proof. (sketch) Let us start with $GM(FVD(S))$. The lower bound is obtained, like in 3D, by a generalization of an example in [2]: Put $n-1$ segments connecting almost-antipodal segments on a $d$-D sphere, and one short segment near the center of the sphere. The latter segment may have $\Omega(n^{d-1})$ ($d-1$)-dimensional cells in $GM(FVD(S))$. A matching upper bound is proven by induction on the dimension (details omitted). Similarly to the 3-D case, there is a bijection between the number of $d$-dimensional cells in $FVD(S)$ and ($d-1$)-dimensional cells in $GM(FVD(S))$, proving the claim about the former type of cells.

Theorem 9 Let $L$ be a set of $n$ lines in $\mathbb{R}^d$ (for a fixed dimension $d$) in general position. The combinatorial complexity of $GM(FVD(L))$, as well as the number of $d$-dimensional cells in $FVD(L)$, is $\Theta(n^{d-1})$.

Proof. (sketch) The complexity of the Gaussian map of the Voronoi diagram of lines is at most the complexity for a set of segments. Indeed, given a set of lines $L$, one can replace $L$ by a set $S$ of unit segments supported by the lines in $L$, then scale up continuously the segments around their midpoints. Denote by $S_c$ the set $S$ thus scaled by $c$. While doing this, the complexity of $GM(FVD(S_c))$, for any fixed $c$, cannot be $\omega(n^{d-1})$. In conjunction, there exists $c_0$ s.t. for any $c > c_0$, the map $GM(FVD(S_c))$ has the same topological structure as $GM(FVD(S_{c_0}))$. The value of $c_0$ is chosen so as to make all features of $GM(FVD(S_{c_0}))$ well separated on the sphere of directions (setting $c_0$ large enough), so that larger values of $c$ move the features of $GM(FVD(S_c))$ in a controlled way, avoiding crashes of features into each other. Thus, in the limit, when $c \to \infty$, the complexity of $GM(FVD(L))$ is also $O(n^{d-1})$. This, in turn, implies that the number of $d$-D cells in $FVD(L)$ is $O(n^{d-1})$. A matching lower bound is obtained by fixing the lines so that their directions (points on one of the two hemispheres) correspond to vertices of a cyclic polytope, attaining the claimed complexity. □

References