Hausdorff and Higher Order Voronoi diagrams

Evanthia Papadopoulou
University of Lugano
(Università della Svizzera italiana)
IP7: Hausdorff and Higher Order Voronoi diagrams

- IP7: three related themes:
  - Hausdorff Voronoi diagrams
  - Higher order Voronoi diagrams
  - Simplifications in the $L_\infty / L_1$ metrics
    - Algorithms and combinatorial issues
    - Implementation issues
    - VLSI application issues

- Relation to Abstract Voronoi diagrams
Overview

- Motivation: VLSI Critical Area Analysis
- Give background and open problems on each of the three themes
  - Hausdorff Voronoi diagram
  - Higher order Voronoi diagrams
  - Simplifications in $L_\infty / L_1$
Motivation

- Problems originate in VLSI critical area analysis

- **Critical Area**: Measure reflecting the sensitivity of a VLSI design to random defects during manufacturing
  - VLSI design layers of polygons
  - Defect of size $r$: circle of radius $r$. Follows a distribution
  - A defect may cause a short circuit, an open circuit.
  - **Critical Area**: Integral to compute the Probability of Fault given the defect distribution
    - shorts, open faults, via blocks, etc

- **Voronoi CAA**: Industrial tool for Critical Area Analysis based on generalized Voronoi diagrams (IBM-Cadence)
Voronoi CAA

- **Idea**: partition a layout layer into regions where the critical area integral can be computed easily (analytically)
  - integration within each region (easy)

- Problem reduces to variants of Voronoi diagrams
  - e.g. higher order Voronoi, Hausdorff Voronoi

- Then Critical Area Integral can be computed easily
  - $L_{\infty}$: trivial integration -- summation of terms derived from Voronoi edges
Voronoi CAA

- Development of IBM Voronoi CAA based on:
  - $L_\infty$ metric
  - Plane sweep

- Why $L_\infty$?

- Why plane sweep?
L∞ metric

- Practical idea to overcome robustness issues in the construction of ordinary Voronoi diagram involving line-segments

\[ p = (x_p, y_p) \]
\[ d_\infty(p, q) = \max \{|x_p - x_q|, |y_p - y_q|\} \]
\[ q = (x_q, y_q) \]

- **L∞ distance** between \( p, q \): Side of min square touching \( p, q \)
- **L∞ Critical Area** – square defects instead of circles
  - Square defects: very common practical simplification
Why $L_\infty$?

- **Algorithmic degree**
  - Formalizes potential of algorithm for robust implementation
  - Degree $d$: Test computations evaluation of multivariate polynomials of arithmetic degree $\leq d$
  - Test computations require bit precision: $db + O(1)$ (input $b$-bit integers)

- **In-circle test (segments): degree $\leq 40**
  - [Burnikel 96]

- **$L_\infty$ in-circle test (segments): degree $\leq 5**
  - [Papadopoulou & Lee 01]

- **VLSI shapes: typically ortho-45: degree 1**

- **$L_\infty$ Voronoi diagram construction: significantly lower algorithmic degree**
- Robust, faster, easier to derive implementation
Why $L_\infty$ in VLSI?

- VLSI shapes: many axis parallel edges
  - not only axis parallel but in majority

- Rectilinear shapes
  \( L_\infty \) Voronoi diagram \( \equiv \) Straight skeleton of rectilinear polygons

- \( L_\infty \) Voronoi diagram captures in one structure proximity, shape-expansion, and shrinking information
Why plane sweep?

- **Memory consumption**
  - Maintain *only* the *wavefront* (VD near sweep line)
  - Critical area only a local computation given an appropriate Voronoi cell
  - Never keep an entire Voronoi diagram in memory

- Very natural for axis parallel (in majority) data
  - Low algorithmic degree in $L_\infty$

- VLSI layout: huge in flat mode
  - compact hierarchical representation
  - Only see portion intersected by a sweep line at a time.
  - Not a dynamic environment
Hausdorff Voronoi -- Outline

- Define the Hausdorff Voronoi diagram
- State of the art, recent advances
- Open problems
Hausdorff Voronoi Diagram (HVD)

- **Given**: A set $S$ of **clusters of points** in the plane
- **Distance** between a point $t$ and a cluster $P$ is measured according to the **farthest (maximum) distance** between $t$ and all points in $P$

$$d_f(t,P) = \max \{d(t,p), \forall p \in P\}$$

- **Compute**: Voronoi diagram of $S$ (**nearest**), such that the distance between a point $t$ and a cluster $P$ is measured as the **farthest** distance $d_f(t,P)$
- **Min-Max** type of Voronoi diagram
Farthest distance between $t$ and $P$ $d_f(t,P) \equiv$ Hausdorff distance between $t$ and $P$

d_f(t,P) = \max \{d(t,p), \forall p \in P\}
Hausdorff Distance

- Hausdorff distance from $P$ to $Q$: 
  $$h(P,Q) = \max_{p \in P} \min_{q \in Q} d(p,q)$$ 
  $h(P,Q) \neq h(Q,P)$

- Hausdorff distance between $P$, $Q$: 
  $$d_h(P,Q) = \max \{h(P,Q), h(Q,P)\}$$

- Hausdorff distance between $P$, $q$: 
  $$d_h(q,P) = \max_{p \in P} d(p,q) = d_f(q,P)$$
Cases of Interest

- **Cases:**
  1. Clusters of **disjoint** convex hulls
  2. Clusters of **non-crossing** convex hulls
  3. **Arbitrary** clusters of points (small number of crossings)

**Non-crossing:** convex hulls admit $\leq 2$ supporting segments.

**Crossing:** convex hulls admit $> 2$ supporting segments.
Hausdorff Voronoi diagram – example

non-crossing clusters

- Subdivision into Hausdorff Voronoi regions
  \[
  \text{region}(P) = \{ x \mid d_f(x,P) < d_f(x,Q), \ \forall Q \in S, \ Q \neq P \}
  \]
  \(\text{region}(P)\): subdivided by farthest Voronoi diagram of \(P\)
A Hausdorff Voronoi region need not be connected if clusters are crossing.
Connection to Abstract Voronoi Diagrams

- The Hausdorff Voronoi diagram (in general form) is not a case of Abstract Voronoi diagrams (in their current form)

  - Arbitrary clusters: disconnected bisectors, disconnected Voronoi regions
    - not a case of (current) abstract Voronoi diagrams

  - AVDs apply in case of non-crossing clusters (cases 1,2)
Hausdorff Voronoi diagram -- History

The cluster Voronoi diagram:

- Several combinatorial bounds
  - Case 1 -- size $O(n)$, $n = \# \text{pts on CHs of } S$
  - Case 3 -- general case -- size $O(n^2\alpha(n))$
  - $n$ intersecting segments: size $\Omega(n^2)$
- $O(n^2\alpha(n))$-algorithm to construct HVD of arbitrary point clusters
  - $\alpha$ is the inverse Ackermann’s function
  - using 3D envelopes of piecewise linear functions

Closest covered set diagram:

- Clusters of disjoint convex hulls – general convex metrics
- Expected $O(kn \log n)$ – algorithm using Abstract VDs
  - $k$: time to compute Hausdorff bisector of 2 convex polygons
Hausdorff Voronoi diagram – History

- The Hausdorff Voronoi diagram

  - Tight combinatorial bound: \( O(n+m') \)
    - \( n \) = # pts on convex hulls of \( S \)
    - \( m' \) = # crossings -- supporting segments between crossing clusters

  - Plane sweep algorithm: \( O((n+K)\log n) \)
    - \( K \) reflects # crossings and pairs of *interacting* clusters
    - \( K = O(n^2) \) (but small in practice)
    - Room for improvement

  - \( L_\infty \) version implemented in IBM Voronoi CAA

- D&C approach

  - [Papadopoulou, Algorithmica 04]
  - [P., Trans. CAD 01]
  - [Papadopoulou & Lee IJCGA 04]
Hausdorff Voronoi diagram – History

- [Dehne, Maheshwari & Taylor, ICPP'06] :
  - Parallel algorithm for clusters of non-crossing convex hulls

- There is the reverse type of Voronoi diagram (farthest-nearest type)
  - [D. P. Huttenlocher, K. Kedem, and M. Sharir D&CG 93]
  - [Abellanas, Hurtado, Icking, Klein, Langetepe, Ma, Palop, Sacristan EuroCG01]
  - Farthest Polygon Voronoi diagrams
    - [Cheong, Everett, Glisse, Gudmundsson, Hornus, Lazard, Lee, Na. ESA 07]
Hausdorff Voronoi diagram - Recent improvements

- The $L^\infty$ Hausdorff Voronoi diagram revisited [Papadopoulou & Xu ISVD11]

1. Structural complexity: $\Theta(n+m)$
   - $n$ is number of clusters (equiv. to rectangles in $L^\infty$)
   - $m$ is number of essential crossings

   A crossing $(P,Q)$ is essential iff there is a min enclosing square of $P$, induced by (side of) $Q$, empty of other rectangles
Hausdorff Voronoi diagram - Recent improvements

1. The $L_\infty$ Hausdorff Voronoi diagram revisited
   [Papadopoulou & Xu ISVD11]

2. Improved bound on plane sweep construction

   - $O((n+M) \log n)$, $M$ is number of special crossings called potentially essential, $m \leq M$, $M=m$, in worst case

   - $O(n \log n)$ for non-crossing clusters

Bound achieved in two ways

- 2 – pass sweep.
  - Preprocessing step (plane sweep) -- point-dominance in $\mathbb{R}^3$
- Wavefront: augmented balanced binary tree
HVD: non-crossing case -- state of the art

- Size: $O(n)$, $n$: # points on cluster convex hulls

- No algorithm with optimal time complexity exists in $L_2$, even in the non-crossing case
  - Expected $O(kn \log n)$ construction time using AVDs
    - $k$: time to compute Hausdorff bisector between two clusters
  - $O(n^2)$ using duality and envelopes in 3D
  - $O((n+K) \log n)$ using plane sweep or D&C, $K$ current bound $O(n^2)$

- $L_\infty$:
  - Expected $O(n \log n)$ time using AVDs
  - $O(n \log n)$ time $O(n)$ space using plane sweep (2 passes)
  - same bound by D&C AVDs (I think)
HVD: non-crossing case

- Difficulty even in non-crossing case: Sites not enclosed in their Voronoi regions
  - Plane sweep: difficult to identify point of min-priority in a Voronoi region (special events generated)
HVD – general case: state of the art

- **Size** $O(n+m')$
  - $n$: # points on cluster convex hulls
  - $m'$: # crossings -- supporting segments between crossing clusters

- Can now show $\Theta(n+m)$, $m$: # essential crossings

- **Construction algorithms**
  - No optimal algorithm exists
  - $O(n^2)$ using envelopes in 3D
  - $O((n+K) \log n)$ using plane sweep, $K$ currently $O(n^2)$

- **Desirable**: $O((n+m)\log n)$ algorithm for HVD construction
**Crossing**

- **Crossing**: pair of diagonals in \(\text{CH}(P \cup Q)\), one in \(P\) one in \(Q\), that cross and whose 3 endpoints induce an enclosing circle of \(P \cup Q\)

- **Distinct crossings** -- distinct enclosing circles
  - \(\#\) distinct crossings = \(\frac{1}{2}\) (\# supporting segments between \(P, Q\))
Essential crossing

- **Essential crossing:** corresponding enclosing circle encloses no cluster other than P and Q
HVD – general case: state of the art

- Size $O(n+m')$
  - $n$: # points on cluster convex hulls
  - $m'$: # crossings -- supporting segments between crossing clusters

- Can now show $\Theta(n+m)$, $m$: # essential crossings

- Construction algorithms
  - No optimal algorithm exists
  - $O(n^2)$ using envelopes in 3D
  - $O((n+K) \log n)$ using plane sweep, $K$ currently $O(n^2)$

- Desirable: $O((n+m)\log n)$ algorithm for HVD construction
Hausdorff Bisector

- Hausdorff bisector \( b(P,Q) \):
  - locus of points equidistant from \( P \) and \( Q \) \( (d_f(t,P) = d_f(t,Q)) \)
  - Sub-graph of \( FVD(P \cup Q) \)

P: red cluster
Q: blue cluster
\( b(P,Q) \): black curve
Hausdorff Bisector

- Hausdorff bisector $b(P,Q)$:
  - locus of points equidistant from $P$ and $Q$ ($d_f(t,P) = d_f(t,Q)$)
  - Sub-graph of $FVD(P \cup Q)$

P: red cluster
Q: blue cluster
$b(P,Q)$: black curve
Hausdorff Bisector -- Properties

- Sub-graph of FVD(P∪Q)
- Collection of edge-disjoint monotone chains extending to infinity

**Non-crossing case**: single chain
- unbounded portions: bisectors of supporting segments between CH(P), CH(Q)

**General case**: One chain for each pair of (distinct) consecutive supporting segments in CH(P∪Q)
- One brunch for each distinct “crossing”
Hausdorff Voronoi diagram – abstract terms

- **Given**: Set of farthest Voronoi diagrams (FVDs)
  - Each FVD represents a cluster of objects
    - cluster of points
    - cluster of line-segments
    - FVD: tree structure of linear size

- **Compute**: Voronoi diagram of S in the **nearest** sense; distance between a point and a cluster is measured in the **farthest** sense
  - Merge FVDs in nearest sense
  - Distance between a point t and an object is measured in the ordinary way (nearest)
HVD Open Problems

- Desirable to design an $O((n+m)\log n)$ algorithm for HVD construction
  - seems doable (generalizing upon $L_\infty$ result)

- Desirable to extend to clusters of line-segments
  - Appears in the VLSI CA application
  - A cluster is no longer characterized by its convex hull but it is characterized by a “farthest hull”
  - Farthest line-segment Voronoi diagram
    - convex hull no longer relevant
    - remains a **tree structure** of **linear complexity**
    - farthest Voronoi regions may be disconnected
  - Main properties seem to remain
HVD open problems

- Unify under the notion of abstract Voronoi diagrams
  - Farthest Abstract Voronoi diagram remains a tree structure of linear complexity
Higher Order Voronoi diagrams -- overview

- Definitions
- Recent advances on $L_\infty$ version
- Higher order Voronoi diagram of line segments
- Open problems
Higher order Voronoi diagrams

- **k^{th} order Voronoi region**: maximal locus of points with the same k nearest neighbors

- **k^{th} order Voronoi region**: locus of points closest to a cluster of k sites (k-cluster)
  - $d(t,P) = d_f(t,P)$, P: cluster of k sites
Order-3 Voronoi diagram

- \( P = \{5,6,7\} \), \(|P|=3\)
- \( \text{reg}(P) \): locus of points with the same 3 nearest sites \(\{5,6,7\}\)
- \( \text{reg}(P) \): locus of points closer to \(P\) than any other triplet of sites
- \( d(t,P) = d_f(t,P) \)
- \( \text{reg}(P) \): subdivided by \( \text{FVD}(P) \)
Relation k-VD ~ HVD

- Order-k Voronoi diagram (k-VD) \( \equiv \) Hausdorff Voronoi diagram of all subsets of k sites
  - k-cluster – subset of k sites

- k-VD(S) \( \equiv \) HVD(S\(_k^\prime\)) \( \equiv \) HVD(S\(_k^{''}\))
  - S: set of sites
  - S\(_k^\prime\): “valid” k-clusters -- have non-empty regions in k-VD
  - S\(_k^{''}\): all possible k-clusters

- S\(_k^\prime\) (valid k-clusters): readily available from (k-1)-VD
Relation $k$-VD ~ HVD

- $(k-1)$-VD provides $S'_k$ (all relevant $k$-clusters)

- Iterative methods to compute $k$-VD($S$) for increasing $k$ -- not hard
  - $k$-VD provides all valid ($k+1$)-clusters
Nice Interpretation in $L_\infty$

- $L_\infty$ k-VD = additively weighted Voronoi diagram of axis-parallel Voronoi edges in (k-1)-VD
  - each Voronoi edge weighted with distance from owner (constant)

- $L_\infty$ k-VD construction (given (k-1)-VD) identical to 1-VD
  - Very simple by plane sweep
  - Basis of IBM Voronoi CAA development using plane sweep.

- Extends to $L_2$ – weights non-constant

- Simple plane sweep construction of k-VD (given (k-1)-VD)
  - iterative for increasing k – application interest in low k
Open Problem

- **Open problem**: Compute k-VD directly (no iteration)
  - time $\sim$ size of k-VD
  - Difficulty: participating k-clusters are not known in advance
  - Those with undounded regions may be “easier” to determine
  - Those with bounded regions seem more difficult

- Yes in $L_\infty$ metric

[Liu, Papadopoulou, Lee ESA11]
k-VD Recent advances

[Liu, Papadopoulou, Lee ESA11]

- A direct paradigm for computing k-VD introduced
- Applied efficiently in $L_\infty$ metric
- Method based on data structures for segment dragging queries

- Two open problems:
  1. Apply direct paradigm in $L_2$ **efficiently** (circle dragging queries)
  2. Avoid the use of advanced data structures
     I believe possible ($L_\infty$) -- Investigate plane sweep
k-VD Recent advances

- **Observation**
  
  $O(k(n-k))$ bound on k-VD is **not tight** in $L_\infty (L_1)$ for large $k$.
  
  For $k=n-1$, size of $L_\infty$ k-VD is $O(1)$

- $L_\infty (L_1)$ : size of k-VD is $O((n-k)^2)$
  
  [Liu, Papadopoulou, Lee ESA11]

- $L_\infty (L_1)$ : size of k-VD is $O(\min\{k(n-k),(n-k)^2\})$
Open problems

- Iterative plane sweep construction – not hard
- **Open problem**: Can we transform the iterative plane sweep into a direct plane sweep method?
  - (at least in $L_\infty$)?

- More generally, direct methods to compute k-VD
  - time complexity $\sim$ size of k-VD
Higher order VD of line-segments

- Higher order Voronoi diagram of line segments
  - ignored so far

- Input: Set S of arbitrary line segments
  - planar straight line graph

- Different versions of the problem
  1. (less interesting): Each line segment consists of 3 sites: 2 endpoints, open portion of line segments
  2. (more interesting): Each line segment is one entity including its endpoints
Version 1

- less interesting
- kind of tedious

FVD: order-(3n-1) diagram

Equivalent to FVD of points (segment endpoints)

- As k increases open line-segment portions disappear
- Easy to design an iterative approach of increasing order
  - kind of tedious
- possible speed ups
Version 2 (interesting)

- Each line segment is one entity including its endpoints
- Farthest line-segment VD: order n-1 VD
- Order-k Voronoi regions can be disconnected
  - for any k>1
- Bisectors can be two-dimensional if segments are non-disjoint (e.g. input a planar straight-line graph)
  - Appropriate definitions
Example: 2-VD of line segments

- Voronoi region of a pair of segments can be disconnected
Some Observations

- Bounded regions of a \( k \)-cluster of line segments can be disconnected in many faces
  - example of \( \Omega(n) \)
  - Nevertheless, each such face corresponds to a distinct edge of \((k-1)\)-VD

- Unbounded regions can be disconnected in \( \Theta(k) \) faces
  - simple extension from [Aurenhammer, Drysdale, Kraser, IPL 06]
Challenges

- **Question**: Does the $O(k(n-k))$ bound hold for the order-$k$ line segment VD?
  - despite disconnected regions?

- **Work in progress**
  - with Maksym Zavershynskyi – SNF project related to EuroGIGA
Background on O(k(n-k)) bound

- DT Lee’s formula (82) on the number of k-VD regions seems to hold as is in case of line-segments

\[ N_k = (2k - 1)N - (k^2 - 1) - \sum_{i=1}^{k} S_i \]

- \( S_i \): number of unbounded regions in i-VD
- \( N_k \): number of faces in k-VD

- To infer O(k(n-k)) from the formula, combinatorial results on k-sets are needed (it is not immediate from the formula)

- Such results are not available for line-segments
Abstract higher-order Voronoi diagrams

- No abstract version of higher order Voronoi diagrams exists
- Abstract Voronoi diagram (order-1)
- Abstract Farthest Voronoi diagram (order-(n-1))

Open problem: Obtain an abstract notion of order-k VD
  - First: study the case of line segments
An observation

- DT Lee’s analysis (82) on k-VD is given for points in $L_2$
- Seems valid as long as
  - bisectors involved are well behaved
  - FVD is tree structure
  - the tree structure of FVD is crucial in the analysis
- Farthest AVDs have these properties
Current work

- Farthest line segment Voronoi diagram, $L_2, L_\infty$
- Unbounded regions of order-k line segment VD
Farthest line-segment Voronoi diagram

- [Aurenhammer, Drysdale, Kraser IPL 06]

- Not related to convex hull
- Disconnected Voronoi regions
- Tree structure (all regions unbounded)
- Size: $O(n)$

- There can be a segment with $\Theta(n)$ disconnected faces
Farthest line-segment Voronoi diagram

- Farthest line-segment hull: convex-hull like structure that characterizes the unbounded bisectors (regions) of the farthest line-segment Voronoi diagram
Farthest line-segment hull

- A segment $s \in FH$ iff the line through $s$ divides the plane into two open half-planes such that one ($H(s)$) intersects all segments but $s$ (but segments collinear with $s$)
Farthest line-segment hull

- A segment endpoint $p \in FH$ iff there is a line $l$ through $p$ that divides the plane into two open half-planes such that one ($H(l)$) intersects all segments but $s$ (and possibly other segments incident to $p$)
Farthest line-segment hull

- A supporting segment $p_1p_2 \in FH$ iff the line $l$ through $p_1p_2$ divides the plane into two open half-planes such that one ($H(l)$) intersects all segments but those incident to $p_1, p_2$.
Properties

- Supporting segments of FH define the unbounded bisectors
  - like in convex hull
- Non-empty regions: edges and vertices of the farthest hull
- If segments degenerate to points their farthest hull is by definition their convex hull
Gauss Map representation

- Represented as a circular list of ordered unit vectors using the Gaussian Map
Constructing the Farthest line-segment VD

- D&C -- $O(n \log n)$ time -- very simple using the Gmap

- Once farthest hull is available the FVD can be constructed easily
  - points or segments – no longer matters algorithmically
  - simple algorithm in [Aurenhammer, Drysdale, Kraser IPL 06]

- Other algorithmic paradigms for convex hulls may also be adjustable to construct the FH (through GMap)
  - Jarvis march
  - quick hull
  - incremental
(HVD of line segments)

- HVD of point clusters: cluster -- convex hull
  - FVD is a tree

- HVD of clusters of line segments: Each cluster represented by its farthest hull.
  - FVD remains a tree structure
Order-k hull

- Similarly to farthest hull we can define the order-k hull

- order-k hull: cyclic structure that characterizes the unbounded regions and bisectors of k-VD
  - points
  - line-segments (appropriate definitions to include incident segments)

- Constructing the order-k hull can be used towards any D&C approach to construct the order-k Voronoi diagram
Summary IP7 and open problems

- Hausdorff Voronoi diagrams
  - Concrete: point, line segments
  - Efficient algorithm to compute HVD for either points or line segments
    - Points in $L_\infty$ -- recently improved
  - Abstract version
Summary IP7 and open problems

- Higher order Voronoi diagrams
  - Strong relation with Hausdorff Voronoi diagram
  - Higher order line segment Voronoi diagram -- structural properties
    - Algorithms to construct the order-k line-segment hull
  - Iterative plane sweep construction of k-VD
    - ($L^\infty$ implemented in Voronoi CAA)
    - Applies also to line segments
  - Direct methods to compute order-k Voronoi diagrams
    - points, line segments
    - Investigate plane sweep (starting $L^\infty$)
  - Abstract properties of higher order Voronoi diagrams
Summary IP7 and open problems

- Simplifications in the $L_\infty$ metric

- $L_\infty$ line-segment Voronoi diagrams in CGAL
  - Investigate