Linear-size farthest color Voronoi diagrams: conditions and algorithms

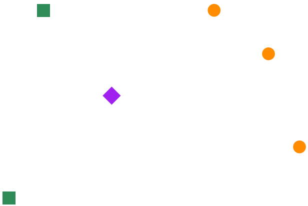
Ioannis Mantas[1] Evanthia Papadopoulou[1]
Vera Sacristán[2] Rodrigo I. Silveira[2]

[1] Università della Svizzera italiana, Lugano, Switzerland
[2] Universitat Politècnica de Catalunya, Barcelona, Spain

19/3/2019 - Utrecht, Netherlands - EuroCG 2019

Color Voronoi Diagrams

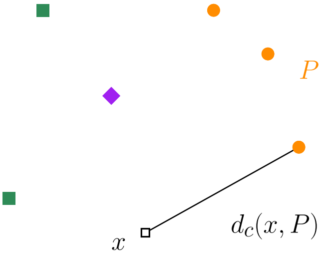
- Family $\mathcal{P}$ of m **clusters** (sets) of points, with n total points.
  $\rightarrow$ Each cluster has a different color.



Color Voronoi Diagrams

- Family $\mathcal{P}$ of m **clusters** (sets) of points, with n total points.
  $\rightarrow$ Each cluster has a different color.

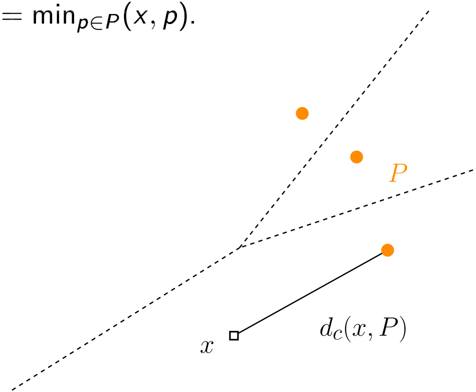
- **Distance** from a point x to a cluster P is
  $d_c(x, P) = \min_{p \in P}(x, p)$.



Color Voronoi Diagrams

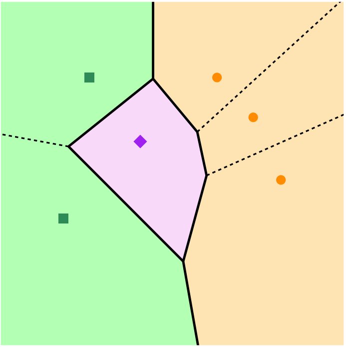
- Family $\mathcal{P}$ of m **clusters** (sets) of points, with n total points.
  $\rightarrow$ Each cluster has a different color.

- **Distance** from a point x to a cluster P is
  $d_c(x, P) = \min_{p \in P}(x, p)$.



Nearest Color Voronoi Diagram (NCVD)

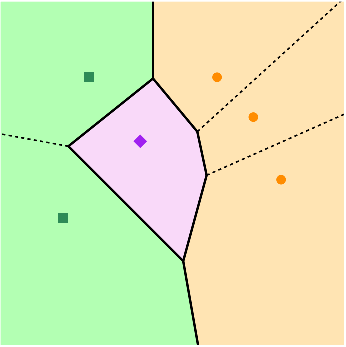
- The **nearest color region** of a cluster $P \in \mathcal{P}$ is: $\{x \in \mathbb{R}^2 \mid d_c(x, P) < d_c(x, Q),\ \forall Q \in \mathcal{P} \setminus \{P\}\}$



Nearest Color Voronoi Diagram (NCVD)

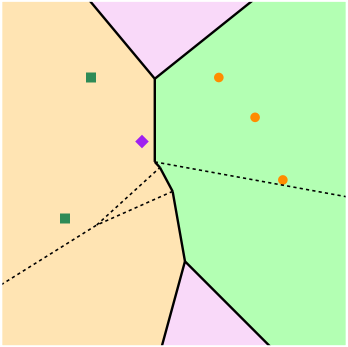
- The **nearest color region** of a cluster $P \in \mathcal{P}$ is:
  $\{x \in \mathbb{R}^2 \mid d_c(x, P) < d_c(x, Q),\ \forall Q \in \mathcal{P} \setminus \{P\}\}$

- NCVD is a **min-min** diagram.



Farthest Color Voronoi Diagram (FCVD)

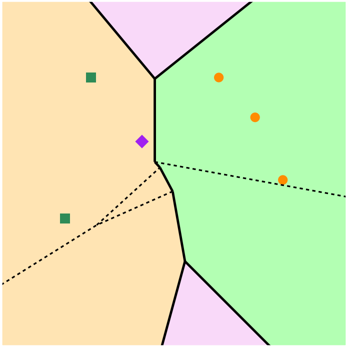
- The **farthest color region** of a cluster $P \in \mathcal{P}$ is:
  $\{x \in \mathbb{R}^2 \mid d_c(x, P) > d_c(x, Q),\ \forall Q \in \mathcal{P} \setminus \{P\}\}$



Farthest Color Voronoi Diagram (FCVD)

- The **farthest color region** of a cluster $P \in \mathcal{P}$ is:
  $\{x \in \mathbb{R}^2 \mid d_c(x, P) > d_c(x, Q),\ \forall Q \in \mathcal{P} \setminus \{P\}\}$

- FCVD is a **max-min** diagram.



FCVD History

- Construction algorithm $O(mn \log n)$.
  Worst case complexity $\Omega(mn)$ - $O(mn\alpha(mn))$.
  [Huttenlocher, Kedem and Sharir 1993]

$m = |\mathcal{P}|$
$n = \sum_{P_i \in \mathcal{P}} |P_i|$

FCVD History

- Construction algorithm $O(mn \log n)$.
  Worst case complexity $\Omega(mn)$ - $O(mn\alpha(mn))$.
  [Huttenlocher, Kedem and Sharir 1993]

- Worst case combinatorial complexity $\Theta(mn)$.
  [Abellanas et al. 2006]

$m = |\mathcal{P}|$
$n = \sum_{P_i \in \mathcal{P}} |P_i|$

FCVD History

- Construction algorithm $O(mn \log n)$.
  Worst case complexity $\Omega(mn)$ - $O(mn\alpha(mn))$.
  [Huttenlocher, Kedem and Sharir 1993]

- Worst case combinatorial complexity $\Theta(mn)$.
  [Abellanas et al. 2006]

- Special cases.
  [Bae 2012, Claverol et al. 2017, Iacono et al. 2017]

$m = |\mathcal{P}|$
$n = \sum_{P_i \in \mathcal{P}} |P_i|$

Motivation - Applications

- **Minimum Hausdorff distance** between two sets of points. [Huttenlocher et al. 1993]
- **Facility location** with multiple types of facilities. [Abellanas et al. 2006]
- Euclidean Bottleneck **Steiner tree**. [Bae et al. 2010]
- **Sensor deployment** in wireless networks. [Lee et al. 2010]
- **Stabbing circles** for segments. [Claverol et al. - 2017]

FCVD relation to Hausdorff Voronoi Diagram

- **Hausdorff Voronoi Diagram**
  A **min-max** diagram - the *dual* of the FCVD.

FCVD relation to Hausdorff Voronoi Diagram

- **Hausdorff Voronoi Diagram**
  A **min-max** diagram - the *dual* of the FCVD.
  Extensively studied:
  - → Envelopes in 3 dimensions [Edelsbrunner et al. 1989]
  - → Divide and Conquer [Papadopoulou & Lee 2004]
  - → Plane Sweep [Papadopoulou 2004]
  - → Randomized Incremental [Arseneva & Papadopoulou 2018]

FCVD relation to Hausdorff Voronoi Diagram

- **Hausdorff Voronoi Diagram**
  A **min-max** diagram - the *dual* of the FCVD.
  Extensively studied:
  → Envelopes in 3 dimensions [Edelsbrunner et al. 1989]
  → Divide and Conquer [Papadopoulou & Lee 2004]
  → Plane Sweep [Papadopoulou 2004]
  → Randomized Incremental [Arseneva & Papadopoulou 2018]

- FCVD can be computed in $O(n^2)$, by adapting the algorithm of [Edelsbrunner et al. 1989].

Summary of results

- **Structural properties** of the FCVD.

Summary of results

- **Structural properties** of the FCVD.
- **Sufficient conditions** for FCVD to have $O(n)$ combinatorial complexity.

Summary of results

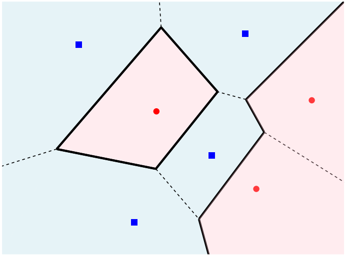
- **Structural properties** of the FCVD.
- **Sufficient conditions** for FCVD to have $O(n)$ combinatorial complexity.
- **Construction algorithms** when these condition are met.

Color bisectors

- The **color bisector** of clusters P and Q is:
  $b_c(P, Q) = \{x \in \mathbb{R}^2 \mid d_c(x, P) = d_c(x, Q)\}$

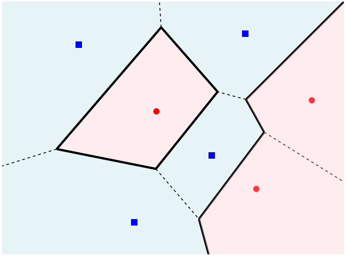
Color bisectors

- The **color bisector** of clusters P and Q is:
  $b_c(P,Q) = \{x \in \mathbb{R}^2 \mid d_c(x,P) = d_c(x,Q)\}$
- $b_c(P,Q)$ is a **subgraph of the diagram** $VD(P \cup Q)$.



Color bisectors

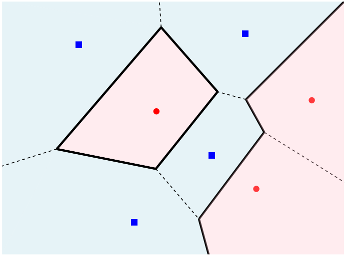
- The **color bisector** of clusters P and Q is:
  $b_c(P,Q) = \{x \in \mathbb{R}^2 \mid d_c(x,P) = d_c(x,Q)\}$
- $b_c(P,Q)$ is a **subgraph of the diagram** $VD(P \cup Q)$.



$\rightarrow$ It consists of cycles and unbounded chains.

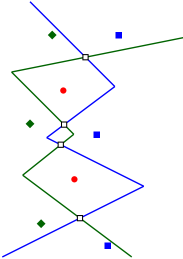
Color bisectors

- The **color bisector** of clusters P and Q is:
  $b_c(P, Q) = \{x \in \mathbb{R}^2 \mid d_c(x, P) = d_c(x, Q)\}$
- $b_c(P, Q)$ is a **subgraph of the diagram** $VD(P \cup Q)$.



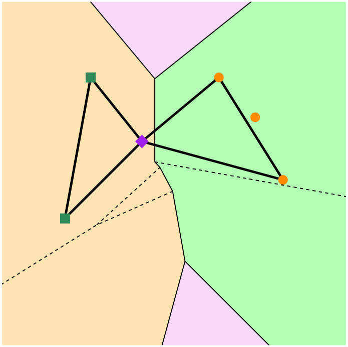
→ It consists of cycles and unbounded chains.

→ Two bisectors may intersect linearly many times.



Cluster Hull

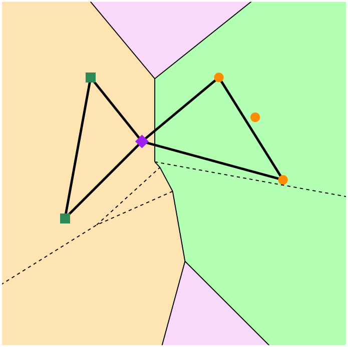
The **Cluster Hull** is a closed (non-simple) **polygonal chain** that characterises the unbounded faces of the FCVD.



Cluster Hull

The **Cluster Hull** is a closed (non-simple) **polygonal chain** that characterises the unbounded faces of the FCVD.

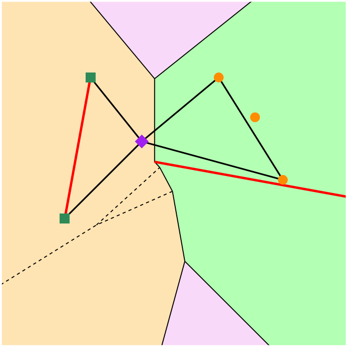
- Introduced for the Hausdorff VD [Papadopoulou & Lee 2004].
- 1 to 1 correspondence: **hull edges** - **unbounded edges** of FCVD.



Cluster Hull

The **Cluster Hull** is a closed (non-simple) **polygonal chain** that characterises the unbounded faces of the FCVD.

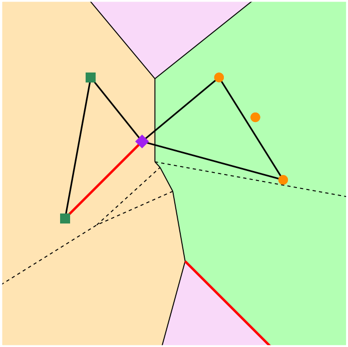
- Introduced for the Hausdorff VD [Papadopoulou & Lee 2004].
- 1 to 1 correspondence: **hull edges** - **unbounded edges** of FCVD.



Cluster Hull

The **Cluster Hull** is a closed (non-simple) **polygonal chain** that characterises the unbounded faces of the FCVD.

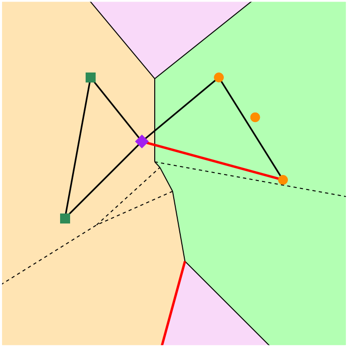
- Introduced for the Hausdorff VD [Papadopoulou & Lee 2004].
- 1 to 1 correspondence: **hull edges** - **unbounded edges** of FCVD.



Cluster Hull

The **Cluster Hull** is a closed (non-simple) **polygonal chain** that characterises the unbounded faces of the FCVD.

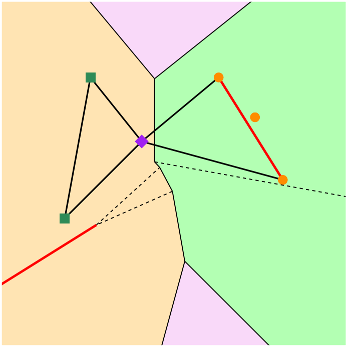
- Introduced for the Hausdorff VD [Papadopoulou & Lee 2004].
- 1 to 1 correspondence: **hull edges** - **unbounded edges** of FCVD.



Cluster Hull

The **Cluster Hull** is a closed (non-simple) **polygonal chain** that characterises the unbounded faces of the FCVD.

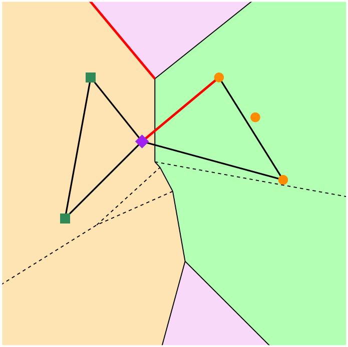
- Introduced for the Hausdorff VD [Papadopoulou & Lee 2004].
- 1 to 1 correspondence: **hull edges** - **unbounded edges** of FCVD.



Cluster Hull

The **Cluster Hull** is a closed (non-simple) **polygonal chain** that characterises the unbounded faces of the FCVD.

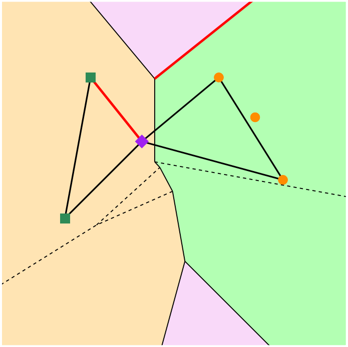
- Introduced for the Hausdorff VD [Papadopoulou & Lee 2004].
- 1 to 1 correspondence: **hull edges** - **unbounded edges** of FCVD.



Cluster Hull

The **Cluster Hull** is a closed (non-simple) **polygonal chain** that characterises the unbounded faces of the FCVD.

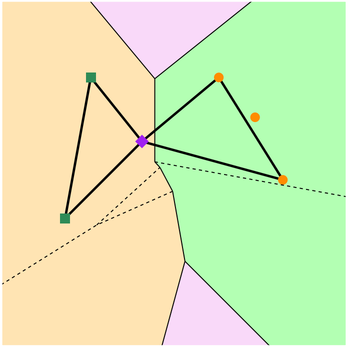
- Introduced for the Hausdorff VD [Papadopoulou & Lee 2004].
- 1 to 1 correspondence: **hull edges** - **unbounded edges** of FCVD.



Cluster Hull

The **Cluster Hull** is a closed (non-simple) **polygonal chain** that characterises the unbounded faces of the FCVD.

- Introduced for the Hausdorff VD [Papadopoulou & Lee 2004].
- 1 to 1 correspondence: **hull edges** - **unbounded edges** of FCVD.



Abstract Voronoi diagrams

Abstract Voronoi diagrams [Klein 1989]

Voronoi diagrams are **defined on a system of a bisectors** that satisfy a set of axioms. For every $\mathcal{P}' \subseteq \mathcal{P}$:

$A1$. Each bisector is an unbounded Jordan curve.

$A2$. Each nearest neighbor region is non-empty and connected.

$A3$. The union of all nearest neighbor regions covers the entire plane.

Abstract Voronoi diagrams

Abstract Voronoi diagrams [Klein 1989]

Voronoi diagrams are **defined on a system of a bisectors** that satisfy a set of axioms. For every $\mathcal{P}' \subseteq \mathcal{P}$:

$A1$. Each bisector is an unbounded Jordan curve.

$A2$. Each nearest neighbor region is non-empty and connected.

$A3$. The union of all nearest neighbor regions covers the entire plane.

A family of clusters $\mathcal{P}$ is called **admissible**, if the color bisectors satisfy axioms A1-A3.

Admissible families

Proposition - Structure and complexity

If $\mathcal{P}$ is an admissible family, then FCVD($\mathcal{P}$) is a tree of $O(n)$ total combinatorial complexity.

Admissible families

Proposition - Structure and complexity

If $\mathcal{P}$ is an admissible family, then FCVD($\mathcal{P}$) is a tree of $O(n)$ total combinatorial complexity.

Proposition - Condition

A linearly-separable family $\mathcal{P}$ is admissible, if and only if each region in NCVD($\mathcal{P}$) is connected.

Admissible families

Proposition - Structure and complexity

If $\mathcal{P}$ is an admissible family, then FCVD($\mathcal{P}$) is a tree of $O(n)$ total combinatorial complexity.

Proposition - Condition

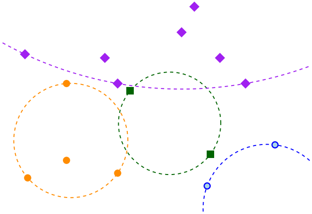
A linearly-separable family $\mathcal{P}$ is admissible, if and only if each region in NCVD($\mathcal{P}$) is connected.

Corollary - Admissible check

We can check if a family $\mathcal{P}$ is admissible in $O(n \log n)$ time.

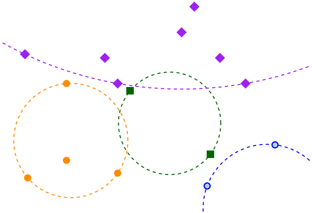
Disk-separable families

A family of clusters $\mathcal{P}$ is **disk-separable** if for every $P \in \mathcal{P}$ there exists a disk containing P and no point from other cluster.



Disk-separable families

A family of clusters $\mathcal{P}$ is **disk-separable** if for every $P \in \mathcal{P}$ there exists a disk containing P and no point from other cluster.

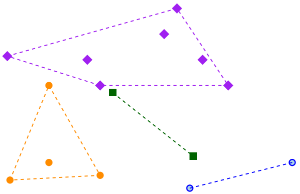


Proposition - Sufficient condition

If a family $\mathcal{P}$ is disk-separable, then $\mathcal{P}$ is also admissible.

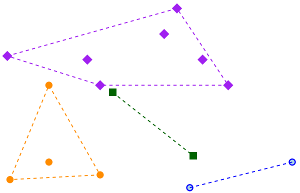
Linearly-separable families

A family of clusters $\mathcal{P}$ is **linearly - separable** if all clusters have pairwise disjoint convex hulls.



Linearly-separable families

A family of clusters $\mathcal{P}$ is **linearly - separable** if all clusters have pairwise disjoint convex hulls.

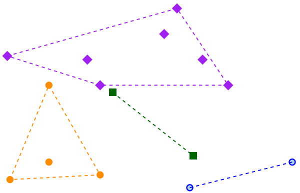


Lemma - Unbounded faces

If $\mathcal{P}$ is a linearly-separable family, then FCVD($\mathcal{P}$) has $O(m)$ unbounded faces.

Linearly-separable families

A family of clusters $\mathcal{P}$ is **linearly - separable** if all clusters have pairwise disjoint convex hulls.



Lemma - Unbounded faces

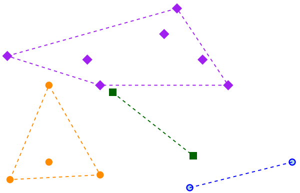
If $\mathcal{P}$ is a linearly-separable family, then FCVD($\mathcal{P}$) has $O(m)$ unbounded faces.

Lemma - Bounded faces

If $\mathcal{P}$ is a linearly-separable family, then FCVD($\mathcal{P}$) has $O(n + s(\mathcal{P}))$ bounded faces.

Linearly-separable families

A family of clusters $\mathcal{P}$ is **linearly - separable** if all clusters have pairwise disjoint convex hulls.



Lemma - Unbounded faces

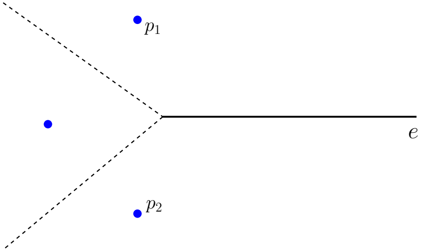
If $\mathcal{P}$ is a linearly-separable family, then FCVD($\mathcal{P}$) has $O(m)$ unbounded faces.

Lemma - Bounded faces

If $\mathcal{P}$ is a linearly-separable family, then FCVD($\mathcal{P}$) has $O(n + s(\mathcal{P}))$ bounded faces. (and overall combinatorial complexity)

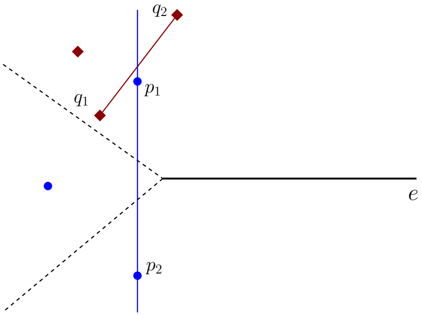
Straddling number

A Voronoi edge e of $VD(P)$, part of bisector(p_1, p_2), is **straddled** by a cluster Q,



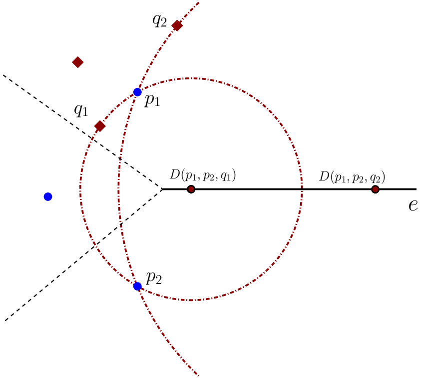
Straddling number

A Voronoi edge e of $VD(P)$, part of bisector(p_1, p_2), is **straddled** by a cluster Q, if $\exists q_1, q_2 \in Q$ such that:
i)The line through p_1, p_2 intersects the segment $\overline{q_1, q_2}$.



Straddling number

A Voronoi edge e of $VD(P)$, part of bisector(p_1, p_2), is **straddled** by a cluster Q, if $\exists q_1, q_2 \in Q$ such that:
i)The line through p_1, p_2 intersects the segment $\overline{q_1, q_2}$.
ii)The centers of $D(p_1, p_2, q_1)$ and $D(p_1, p_2, q_2)$ lie on e.

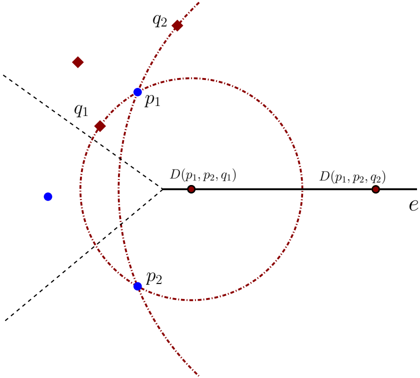


Straddling number

A Voronoi edge e of $VD(P)$, part of bisector(p_1, p_2), is **straddled** by a cluster Q, if $\exists q_1, q_2 \in Q$ such that:

i)The line through p_1, p_2 intersects the segment $\overline{q_1, q_2}$.

ii)The centers of $D(p_1, p_2, q_1)$ and $D(p_1, p_2, q_2)$ lie on e.



The number of all clusters that straddle e is $s(e)$.

The **straddling number** of $\mathcal{P}$ is

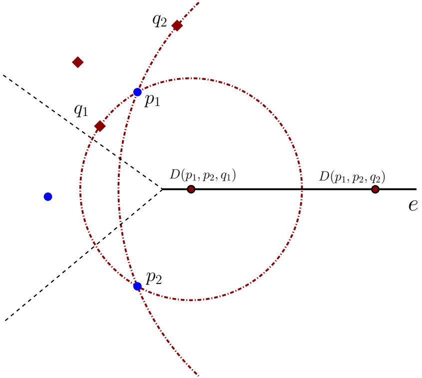
$$s(\mathcal{P}) = \sum_{P \in \mathcal{P}} \sum_{e \in VD(P)} s(e).$$

Straddling number

A Voronoi edge e of $VD(P)$, part of bisector(p_1, p_2), is **straddled** by a cluster Q, if $\exists q_1, q_2 \in Q$ such that:

i)The line through p_1, p_2 intersects the segment $\overline{q_1, q_2}$.

ii)The centers of $D(p_1, p_2, q_1)$ and $D(p_1, p_2, q_2)$ lie on e.



The number of all clusters that straddle e is $s(e)$.

The **straddling number** of $\mathcal{P}$ is

$$s(\mathcal{P}) = \sum_{P \in \mathcal{P}} \sum_{e \in VD(P)} s(e).$$

Corollary - Condition

If a linearly-separable family $\mathcal{P}$ has $s(\mathcal{P}) = O(n)$, then FCVD$(\mathcal{P})$ has $O(n)$ combinatorial complexity.

Algorithm description

Divide & Conquer algorithm

1. **Split** family of clusters $\mathcal{P}$ in two parts $\mathcal{P}_L$, $\mathcal{P}_R$.
2. **Recursively compute** FCVD($\mathcal{P}_L$) and FCVD($\mathcal{P}_R$)
3. **Merge** FCVD($\mathcal{P}_L$), FCVD($\mathcal{P}_R$) into FCVD($\mathcal{P}$).

Algorithm description

Divide & Conquer algorithm

1. **Split** family of clusters $\mathcal{P}$ in two parts $\mathcal{P}_L$, $\mathcal{P}_R$.
2. **Recursively compute** FCVD($\mathcal{P}_L$) and FCVD($\mathcal{P}_R$)
3. **Merge** FCVD($\mathcal{P}_L$), FCVD($\mathcal{P}_R$) into FCVD($\mathcal{P}$).

The **merge curve** consists of bounded & unbounded components.
For each component:

a. **Find a starting point**.
b. **Trace the component**.

Merge curve construction

a. **Finding starting points.**

b. **Tracing components.**

Merge curve construction

a. **Finding starting points.**
   - $\rightarrow$ Unbounded components can be found in $O(n)$ time using the cluster hull.
   - $\rightarrow$ Bounded components ???

b. **Tracing components.**

Merge curve construction

a. **Finding starting points.**

$\rightarrow$ Unbounded components can be found in $O(n)$ time using the cluster hull.

$\rightarrow$ Bounded components ???

b. **Tracing components.**

$\rightarrow$ A component M can be traced in $O(|M|)$ time.

Algorithms

Theorem - Admissible families

If $\mathcal{P}$ is an **admissible family**, FCVD($\mathcal{P}$) can be constructed in $O(n \log n)$ time.

Algorithms

Theorem - Admissible families

If $\mathcal{P}$ is an **admissible family**, FCVD($\mathcal{P}$) can be constructed in $O(n \log n)$ time.

Key: FCVD($\mathcal{P}$) is a tree, so there are no bounded components.

Algorithms

Theorem - Admissible families

If $\mathcal{P}$ is an **admissible family**, FCVD($\mathcal{P}$) can be constructed in $O(n \log n)$ time.

Key: FCVD($\mathcal{P}$) is a tree, so there are no bounded components.

Theorem - Linearly-separable families

If $\mathcal{P}$ a **linearly-separable family** where each cluster has $O(1)$ straddles, FCVD($\mathcal{P}$) can be constructed in $O(n \log^2 n)$.

Algorithms

Theorem - Admissible families

If $\mathcal{P}$ is an **admissible family**, FCVD($\mathcal{P}$) can be constructed in $O(n \log n)$ time.

Key: FCVD($\mathcal{P}$) is a tree, so there are no bounded components.

Theorem - Linearly-separable families

If $\mathcal{P}$ a **linearly-separable family** where each cluster has $O(1)$ straddles, FCVD($\mathcal{P}$) can be constructed in $O(n \log^2 n)$.

Key: The bounded components can be found in $O(n \log n)$ time [Iacono et al. 2017].

Future work

- **Settle FCVD complexity** of linearly-separable families.
  $\rightarrow$ We conjecture it is $\Theta(mn)$ in the worst case.

- **Design** $o(n^2)$ **algorithm** when FCVD has $O(n)$ complexity.

Thank you for your attention!