Topological Considerations for the Incremental Computation of Voronoi Diagrams of Straight-Line Segments and Circular Arcs

Martin Held Stefan Huber

University of Salzburg, Austria
Department of Computer Science

March 20, 2008 / EuroCG08, Nancy

Problem

Problem

Devise and implement an algorithm for computing the Voronoi diagram of points, straight-line segments and circular arcs for real-world applications.

Idea

Extend the incremental algorithm of (Imai, 1996) resp. (Held, 2001), which handles points and straight-lines, to circular arcs.

In this talk, we will pick a few topological and graph-theoretical aspects when incrementally constructing Voronoi diagrams.

Basic definitions: Voronoi diagram

Definition (proper input set)

A finite disjoint system $S \subseteq \mathcal{P}(\mathbb{R}^2)$ is called *proper* set of input sites, if

- S consists of points, open segments and open arcs (less than semi-circles),
- S contains the endpoints of the segments and arcs as well.

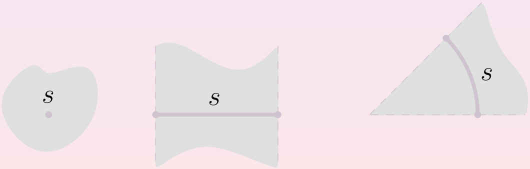


Figure: Cone of influence $\mathcal{CI}(s)$, of a point, segment or arc s.

Basic definitions: Voronoi diagram

Definition (proper input set)

A finite disjoint system $S \subseteq \mathcal{P}(\mathbb{R}^2)$ is called *proper* set of input sites, if

- S consists of points, open segments and open arcs (less than semi-circles),
- S contains the endpoints of the segments and arcs as well.

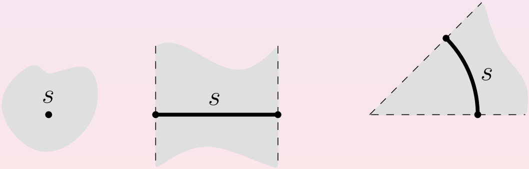


Figure: Cone of influence $\mathcal{CI}(s)$, of a point, segment or arc s.

Voronoi diagrams

Definition (Voronoi cell, polygon, diagram)

Let S be a proper set of input sites, $s \in S$ an input site and d be the Euclidean distance. We define the *Voronoi cell* of s as

$$\mathcal{VC}(s, S) := \mathrm{cl}\{p \in \mathrm{int}\,\mathcal{CI}(s) \ : \ d(p, s) \leq d(p, S \setminus \{s\})\}.$$

The *Voronoi polygon* $\mathcal{VP}(s, S)$ is commonly defined as the boundary of $\mathcal{VC}(s, S)$ and the *Voronoi diagram* is defined as the union of all Voronoi polygons:

$$\mathcal{VD}(S) := \bigcup_{s \in S} \mathcal{VP}(s, S).$$

Voronoi diagrams

Definition (Voronoi cell, polygon, diagram)

Let S be a proper set of input sites, $s \in S$ an input site and d be the Euclidean distance. We define the *Voronoi cell* of s as

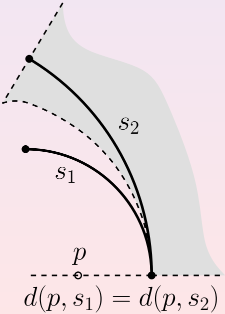
$$\mathcal{VC}(s, S) := \mathrm{cl}\{p \in \mathrm{int}\, \mathcal{CI}(s) \;:\; d(p, s) \leq d(p, S \setminus \{s\})\}.$$

The *Voronoi polygon* $\mathcal{VP}(s, S)$ is commonly defined as the boundary of $\mathcal{VC}(s, S)$ and the *Voronoi diagram* is defined as the union of all Voronoi polygons:

$$\mathcal{VD}(S) := \bigcup_{s \in S} \mathcal{VP}(s, S).$$

Motivating the closure-interior definition

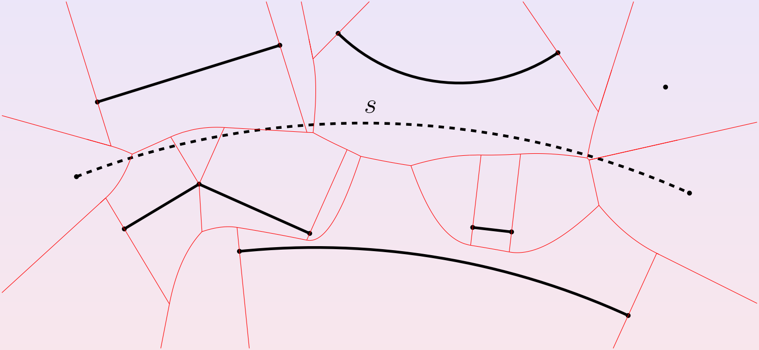
- Suppose we would define a Voronoi cell $\mathcal{VC}(s, S)$ as $\{p \in \mathcal{CI}(s) \ : \ d(p, s) \leq d(p, S \setminus \{s\})\}$.
- All points p from the center of s_1 to the common endpoint of the tangential sites s_1 and s_2 would belong to $\mathcal{VC}(s_2, S)$ as well!



Prior work

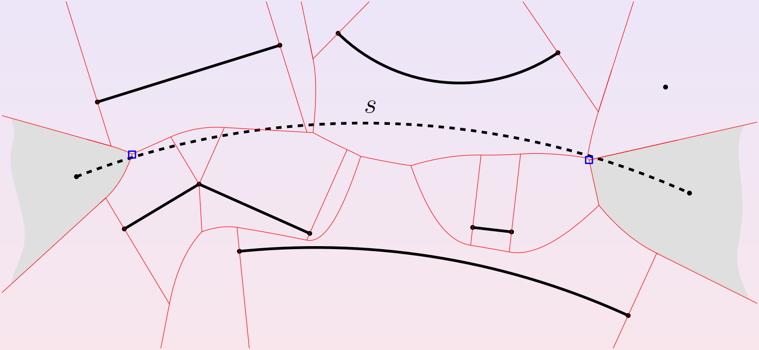
- (Sugihara & Iri, 1992) presented a topology-oriented incremental algorithm for points.
- (Imai, 1996) sketched an extension to segments.
- (Held, 2001) filled missing algorithmic gaps and cast the algorithm into an implementation: VRONI.

Basic incremental algorithm



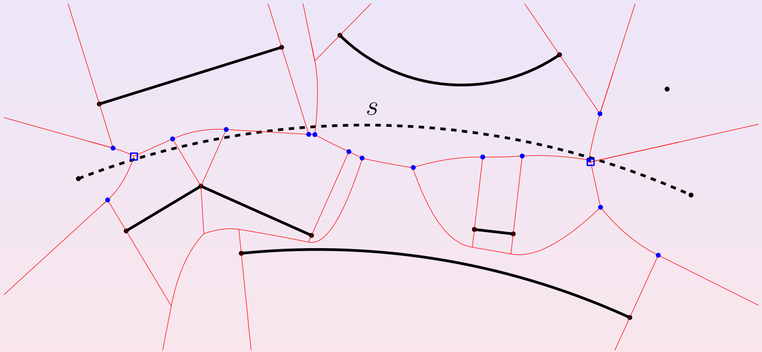
Let $S^+ := S \cup \{s\}$ be a proper set of input sites, with an arc $s \notin S$. Suppose that we already know $\mathcal{VD}(S)$ and we want to insert the arc s into the Voronoi diagram $\mathcal{VD}(S)$.

Basic incremental algorithm



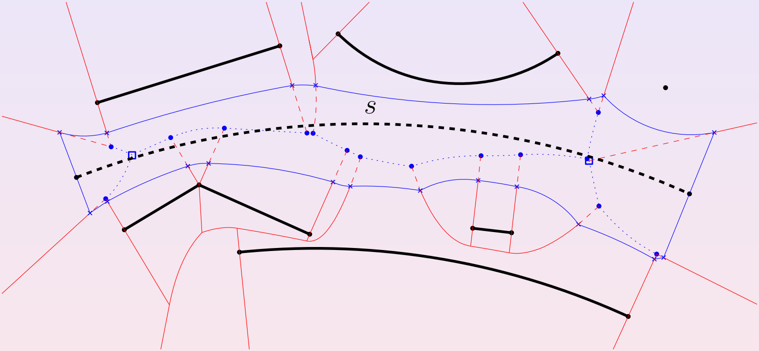
We consider the Voronoi cells of the endpoints of s: within each cell we mark the Voronoi node whose clearance disk is "violated most" by s. We call these two nodes *seed nodes*.

Basic incremental algorithm



Starting from a seed node, recursively mark further nodes if their clearance disk is intersected by s.

Basic incremental algorithm



- Remove marked edges,
- compute new nodes and adapt semi-marked edges,
- connect new nodes with edges.

Question 1

Is there always an appropriate seed node?

Question 2

Is a marked edge always completely in the future Voronoi cell $\mathcal{VC}(s, S^+)$?

Existence of seed node

Lemma

Let p be an endpoint of an arc s. There always exists a node $v \in \mathcal{VP}(p, S)$, with $v \in \mathcal{CI}(s)$, hence $v \in \mathcal{VC}(s, S^+)$.

Based on this lemma, we select a seed node as follows:

- If $\exists v \in \operatorname{int} \mathcal{CI}(s)$ then all nodes in $\mathcal{VP}(p, S) \cap \mathcal{CI}(s)$ are connected by marked edges $\Rightarrow$ we can choose any of them as seed node.
- Otherwise, we distinguish the following cases:
  1. The arc s meets exactly one site s' tangentially in p.
  2. Several sites meet in a common endpoint p.

Existence of seed node

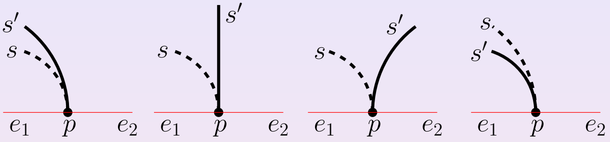
Lemma

Let p be an endpoint of an arc s. There always exists a node $v \in \mathcal{VP}(p, S)$, with $v \in \mathcal{CI}(s)$, hence $v \in \mathcal{VC}(s, S^+)$.

Based on this lemma, we select a seed node as follows:

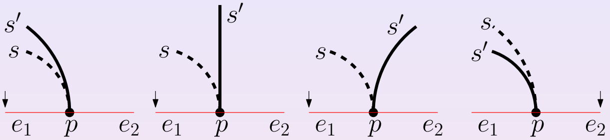
- If $\exists v \in \operatorname{int} \mathcal{CI}(s)$ then all nodes in $\mathcal{VP}(p, S) \cap \mathcal{CI}(s)$ are connected by marked edges $\Rightarrow$ we can choose any of them as seed node.
- Otherwise, we distinguish the following cases:
  1. The arc s meets exactly one site s' tangentially in p.
  2. Several sites meet in a common endpoint p.

Tangential sites



- s meets exactly one site $s' \in S$ tangentially in p.
- $e_1, e_2 \in \mathcal{VP}(p, S)$ originate from p,
- on a supporting line.

Tangential sites



- We do not mark nodes coinciding with input points.
- The other two nodes of e_1, e_2 are the only candidates.
- Based on case analysis: Select proper candidate.

Spikes

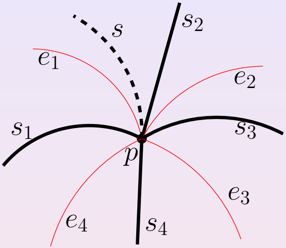


Figure: Left: Geometrical view. Right: Topological view.

Several sites $s_1, s_2 \ldots$ meet in p.

1. Scan the nodes $v \in \mathcal{VP}(p, S)$ and check whether $v \in \mathcal{CI}(s)$ and check for non-zero clearance.
2. If no such node exists...

Spikes

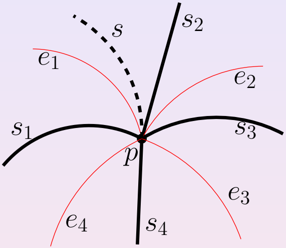


Figure: Left: Geometrical view. Right: Topological view.

- . . . scan edges $e_1, e_2, \ldots$ which are incident to $v_1, v_2, \ldots$.
- Test whether clearance disk of a second node of e_i is intersected by s and choose such a node as seed node.
- Again, take care of tangential sites.

Question 1

Is there always an appropriate seed node?

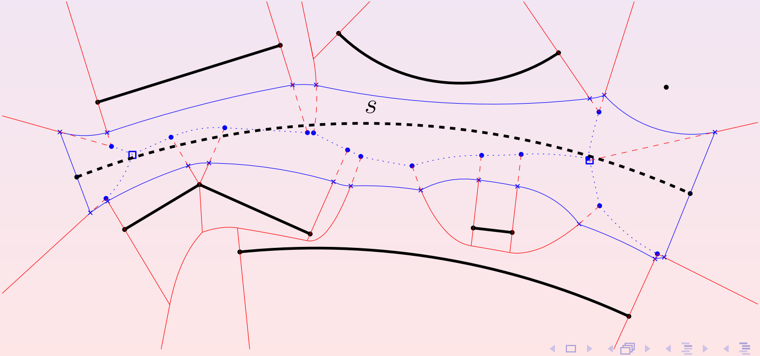
Question 2

Is a marked edge always completely in the future Voronoi cell $\mathcal{VC}(s, S^+)$?

Tree structure of marked edges

Theorem

Let G be the graph corresponding to the nodes and edges of $\mathcal{VD}(S)$ which completely lie in $\mathcal{VC}(s, S^+)$, but do not intersect with $(\mathrm{cl}\, s) \setminus s$. Then G forms a tree.



Tree structure of marked edges

Let T be the graph of marked edges. It can be shown that

- T contains a cycle if and only if T contains an edge which is not completely contained in $\mathcal{VC}(s, S^+)$, and,
- therefore, should be partly preserved.

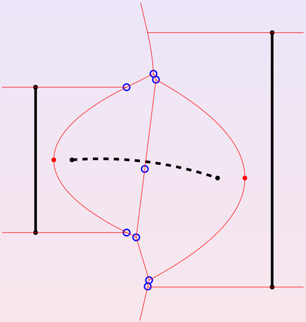
Tree structure of marked edges

Question

Can it happen that T contains an edge which should be partly preserved?

Answer: Actually, it can happen.
Solution: Search for cycles in T and break them up.

Apex splitting



There are known examples (Held, 2001), where a cycle of edges is marked.

Solution: Split every edge at its apex by a degree-2 node, if it contains the apex in its relative interior.

Note

In the sequel, we assume that no edge contains the apex in its relative interior. (Apex splitting)

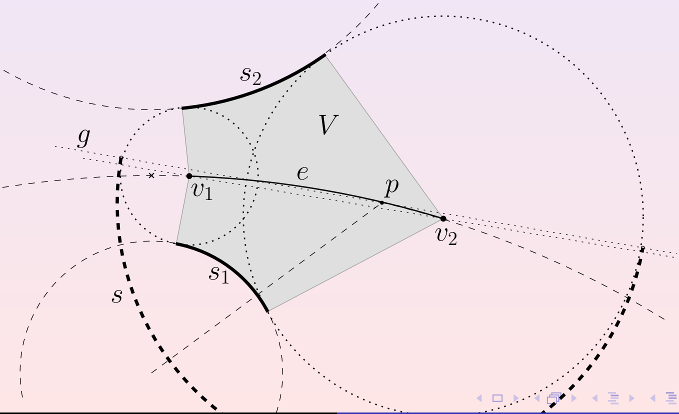
Breaking up a cycle

It turns out that apex splitting is not sufficient when considering arcs. We distinguish the following cases:

1. An edge has been marked and reaches outside of $\mathcal{CI}(s)$.
2. An edge has been marked which is completely in $\mathcal{CI}(s)$, but partly remains in $\mathcal{VD}(S^+)$.

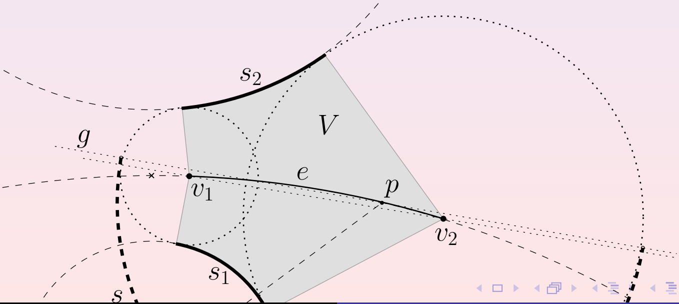
Breaking up a cycle outside of $\mathcal{CI}$

- Suppose that two arcs s_1, s_2 define a hyperbolic edge e, as illustrated.
- v_1, v_2 are the end nodes of e.



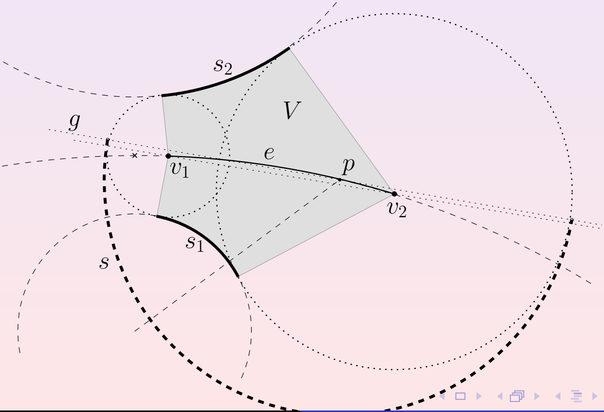
Breaking up a cycle outside of $\mathcal{CI}$

- g is the supporting line of a secant of e.
- V is the union of projection segments of e on s_1 and s_2.
- we choose a point on each of the two parts of $g \setminus (V \cup \mathcal{CD}(v_1, S) \cup \mathcal{CD}(v_2, S))$,
- such that their normals onto g intersect $\mathcal{CD}(v_1, S)$ resp. $\mathcal{CD}(v_2, S)$.



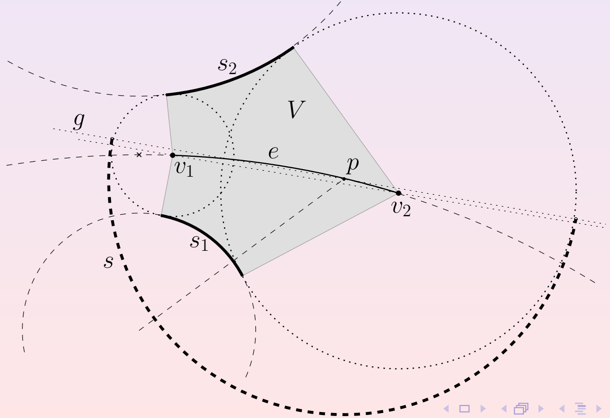
Breaking up a cycle outside of $\mathcal{CI}$

- Now we can insert a semi-circle between the two chosen points as illustrated.
- e is marked, but reaches outside of $\mathcal{CI}(s)$.



Breaking up a cycle outside of $\mathcal{CI}$

Solution: Let us denote with p the intersection of e and the line through the centers of s_1 and s. It can be shown that $p \notin \mathcal{VC}(s, S^+)$, hence p is a proper split point.

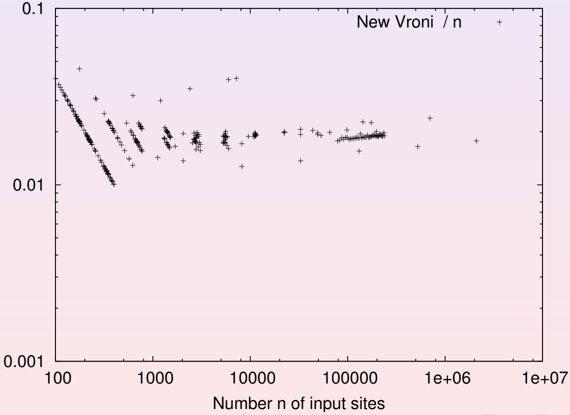


Implementation

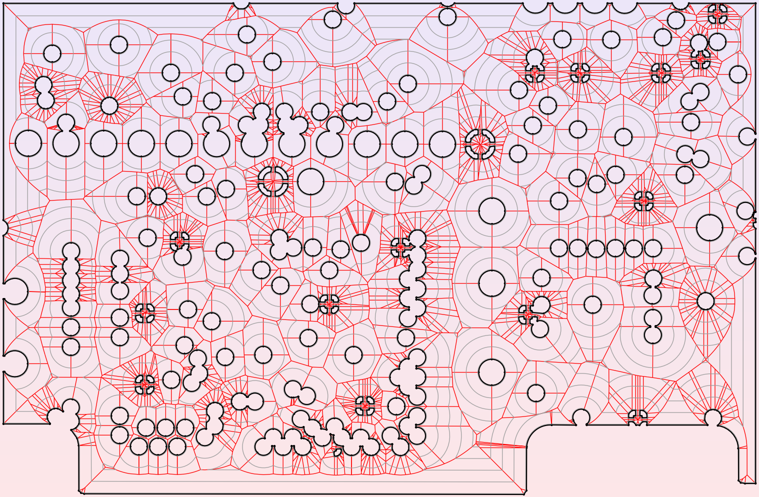
- extended Held's VRONI to circular arcs
- ANSI C
- double-precision floating-point arithmetic
- tested on several hundred synthetic and real-world data sets

Complexity

- Randomized insertion results in $O(n \log n)$ expected runtime
- Experimental evaluation "yields a close to linear" behaviour:



Finish



Bibliography I

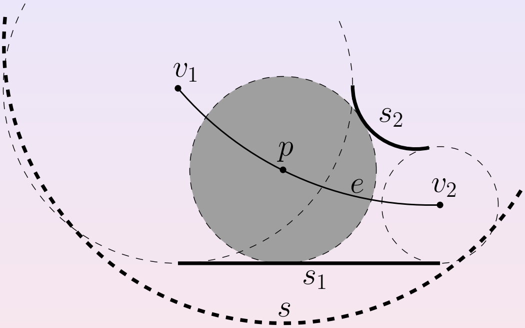
Held, M. 2001.
VRONI: An Engineering Approach to the Reliable and Efficient Computation of Voronoi Diagrams of Points and Line Segments.
Comput. geom. theory and appl., **18**(2), 95–123.

Imai, T. 1996.
A Topology Oriented Algorithm for the Voronoi Diagram of Polygons.
Pages 107–112 of: Proc. canad. conf. comput. geom. (cccg'96).
Ottawa, Canada: Carleton University Press.

Bibliography II

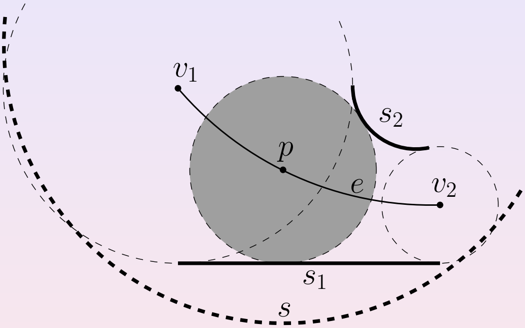
Sugihara, K., & Iri, M. 1992.
Construction of the Voronoi Diagram for 'One Million' Generators in Single-Precision Arithmetic.
Proc. of the ieee, **80**(9), 1471–1484.

Breaking up a cycle inside of $\mathcal{CI}$



- e is an edge completely contained in $\mathcal{CI}(s)$.
- v_1 and v_2 are marked, but e contains points not in $\mathcal{VC}(s, S^+)$.

Breaking up a cycle inside of $\mathcal{CI}$



Solution: Let us denote with p the intersection of e and the projection line of the center of s on s_1. It can be shown that $p \notin \mathcal{VC}(s, S^+)$. Hence p is a proper split point.