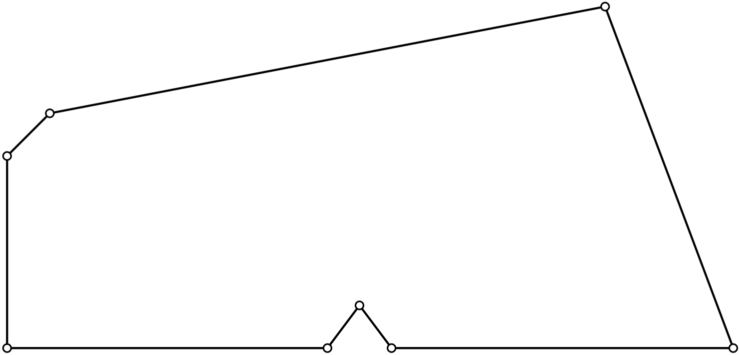
STRAIGHT SKELETONS OF MONOTONE POLYGONS

Therese Biedl   Martin Held   Stefan Huber
Dominik Kaaser   Peter Palfrader

EuroCG 2014, Ein Gedi, Israel

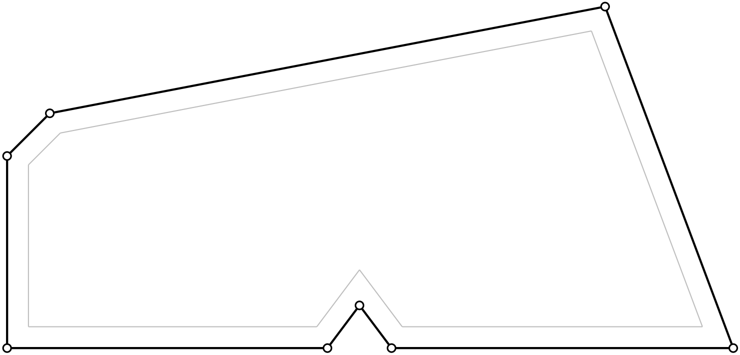
Straight Skeletons

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



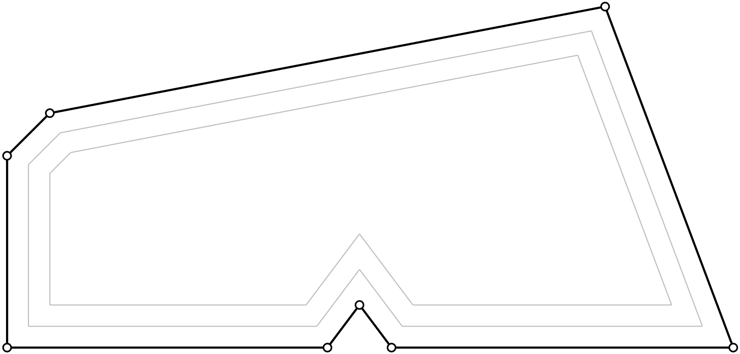
Straight Skeletons

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



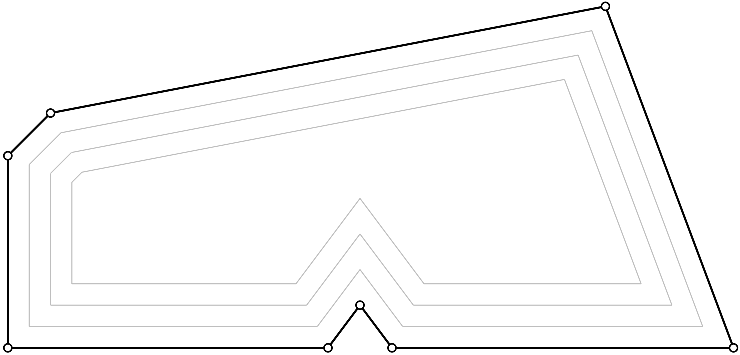
Straight Skeletons

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



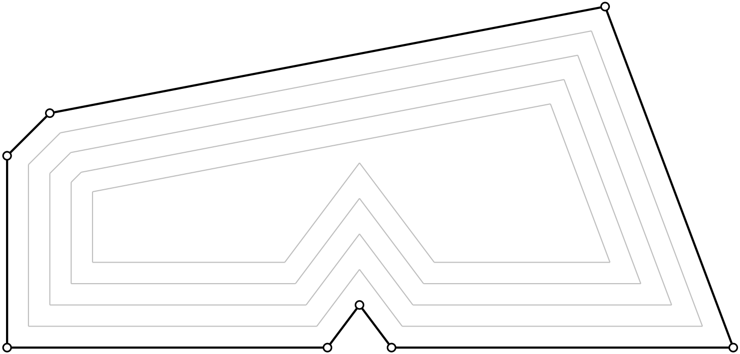
Straight Skeletons

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



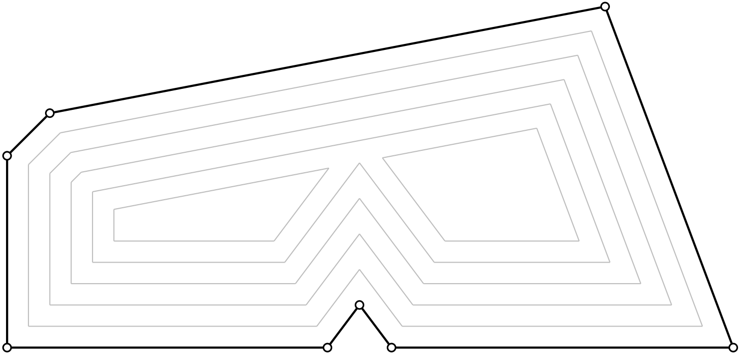
Straight Skeletons

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



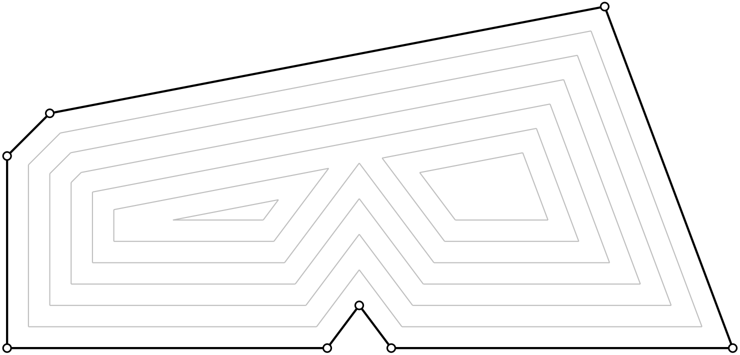
Straight Skeletons

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



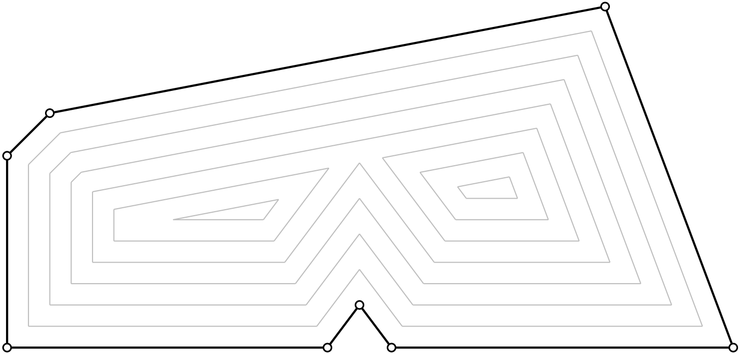
Straight Skeletons

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



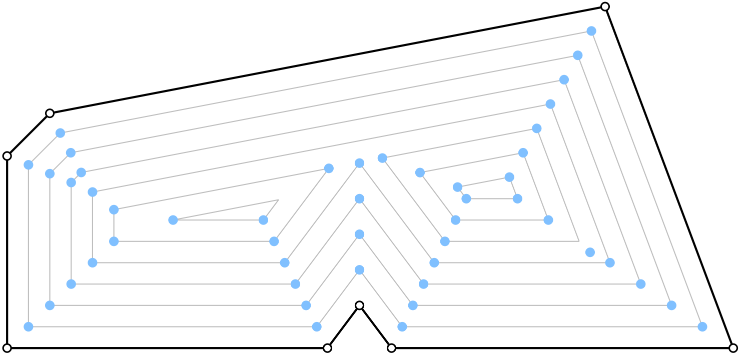
Straight Skeletons

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



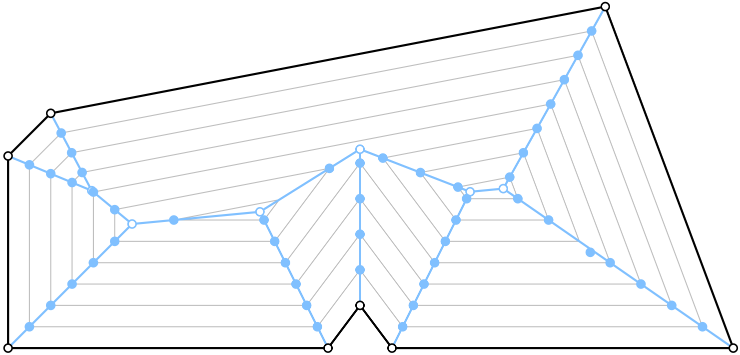
Straight Skeletons

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



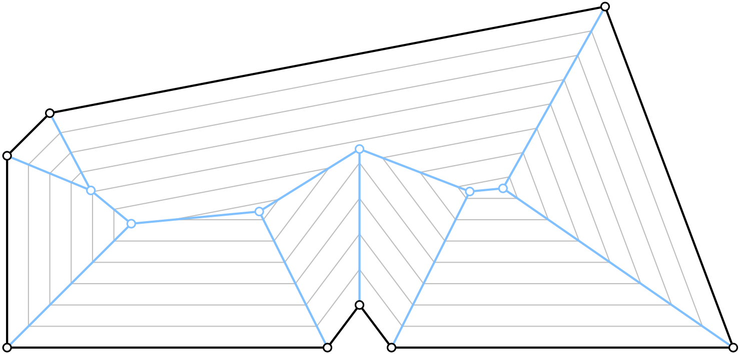
STRAIGHT SKELETONS

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



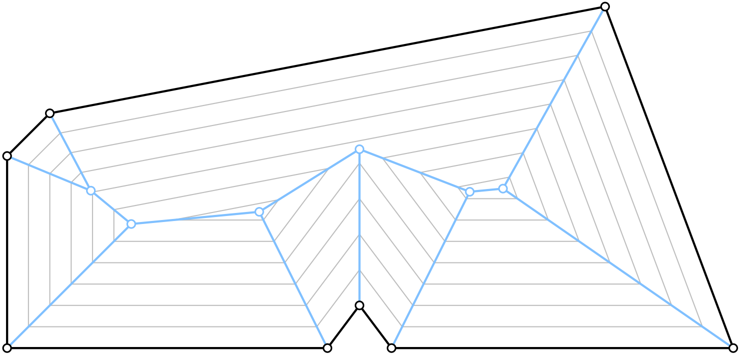
Straight Skeletons

- Aichholzer, Aurenhammer, Alberts, Gärtner 1995[2].
- *Wavefront propagation*: shrinking, mitered offset curves of polygon $\mathcal{P}$.
- Traces of wavefront vertices are the edges of the straight skeleton $\mathcal{S}(\mathcal{P})$.



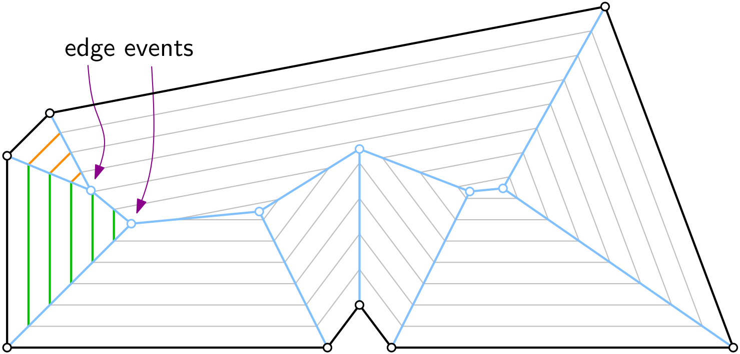
Topology Changes – Edge and Split Events

- Wavefront topology changes over time.
- *edge event*: an edge of the wavefront vanishes.
- *split event*: wavefront splits into two parts.
- In $\mathcal{S}(\mathcal{P})$, events (topology changes) are witnessed by *nodes*.



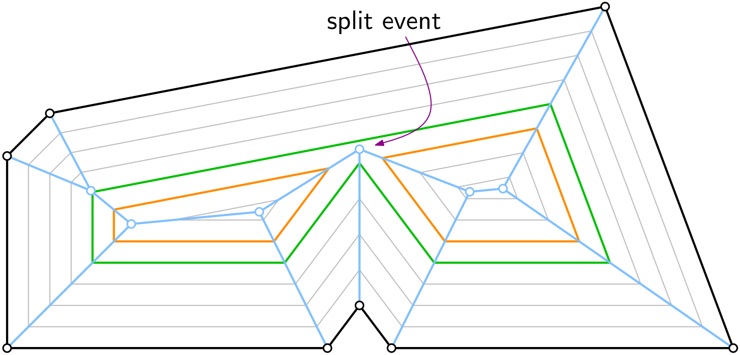
Topology Changes – Edge and Split Events

- Wavefront topology changes over time.
- *edge event*: an edge of the wavefront vanishes.
- *split event*: wavefront splits into two parts.
- In $\mathcal{S}(\mathcal{P})$, events (topology changes) are witnessed by *nodes*.



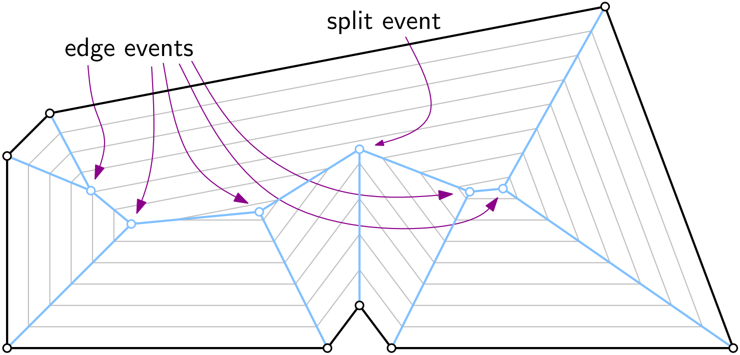
Topology Changes – Edge and Split Events

- Wavefront topology changes over time.
- *edge event*: an edge of the wavefront vanishes.
- *split event*: wavefront splits into two parts.
- In $\mathcal{S}(\mathcal{P})$, events (topology changes) are witnessed by *nodes*.



TOPOLOGY CHANGES – EDGE AND SPLIT EVENTS

- Wavefront topology changes over time.
- *edge event*: an edge of the wavefront vanishes.
- *split event*: wavefront splits into two parts.
- In $\mathcal{S}(\mathcal{P})$, events (topology changes) are witnessed by *nodes*.



CONSTRUCTING THE STRAIGHT SKELETON

Algorithm	time	space
Aichholzer, Aurenhammer '98 [1]	$\mathcal{O}(n^3 \log n)$	$\mathcal{O}(n)$
Eppstein, Erickson '99 [5] $\star$	$\mathcal{O}(n^{17/11+\epsilon})$	
Cacciola '04 [3]	$\mathcal{O}(n^2 \log n)$	$\mathcal{O}(n^2)$
Huber and Held '10 [7]	$\mathcal{O}(n^2 \log n)$	$\mathcal{O}(n)$
Vigneron and Yan '13 [8] $\star\star$	$\mathcal{O}(n^{4/3+\epsilon})$	$\mathcal{O}(n)$

- Popular approach: Simulate the wavefront propagation.
- Main Problem: Identify next event.
- Edge events are cheap. Split events are expensive.

Can we do better for specific input classes?

YES, FOR (STRICTLY) MONOTONE POLYGONS.

CONSTRUCTING THE STRAIGHT SKELETON

Algorithm	time	space
Aichholzer, Aurenhammer '98 [1]	$\mathcal{O}(n^3 \log n)$	$\mathcal{O}(n)$
Eppstein, Erickson '99 [5] $\star$	$\mathcal{O}(n^{17/11+\epsilon})$	
Cacciola '04 [3]	$\mathcal{O}(n^2 \log n)$	$\mathcal{O}(n^2)$
Huber and Held '10 [7]	$\mathcal{O}(n^2 \log n)$	$\mathcal{O}(n)$
Vigneron and Yan '13 [8] $\star\star$	$\mathcal{O}(n^{4/3+\epsilon})$	$\mathcal{O}(n)$

- Popular approach: Simulate the wavefront propagation.
- Main Problem: Identify next event.
- Edge events are cheap. Split events are expensive.

Can we do better for specific input classes?

YES, FOR (STRICTLY) MONOTONE POLYGONS.

Constructing the Straight Skeleton

Algorithm	time	space
Aichholzer, Aurenhammer '98[1]	$\mathcal{O}(n^3 \log n)$	$\mathcal{O}(n)$
Eppstein, Erickson '99[5] ⋆	$\mathcal{O}(n^{17/11+\epsilon})$	
Cacciola '04[3]	$\mathcal{O}(n^2 \log n)$	$\mathcal{O}(n^2)$
Huber and Held '10[7]	$\mathcal{O}(n^2 \log n)$	$\mathcal{O}(n)$
Vigneron and Yan '13[8] ⋆⋆	$\mathcal{O}(n^{4/3+\epsilon})$	$\mathcal{O}(n)$

- Popular approach: Simulate the wavefront propagation.
- Main Problem: Identify next event.
- Edge events are cheap. Split events are expensive.

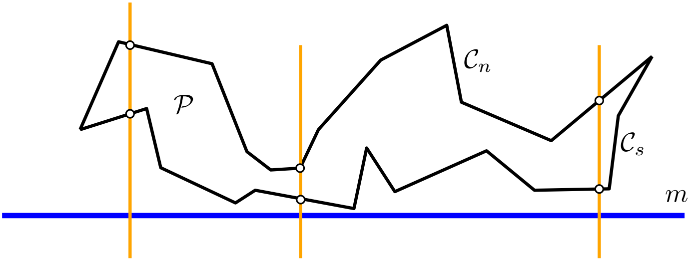
Can we do better for specific input classes?

Yes, for (strictly) monotone polygons.

MONOTONE POLYGONS

- Strictly monotone chain $\mathcal{C}$ (monotone w.r.t. to a line *m*):
  Polygonal chain that intersects normals of *m* in at most one point.

- Strictly monotone polygon $\mathcal{P}$ (monotone w.r.t. to a line *m*):
  Simple polygon that can be split into two strictly monotone chains.



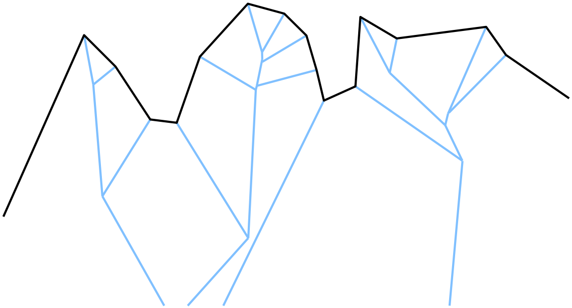
Monotone Polygons - Prior Work

Das et al. claim $\mathcal{O}(n \log n)$ time algorithm[4]:

- Requires general position.
- Correctness?

LEMMATA I

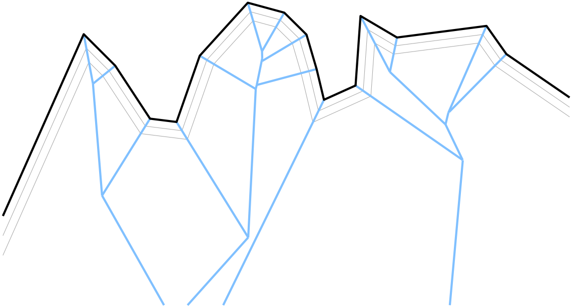
- The wavefront propagation of strictly monotone polygonal chain $\mathcal{C}$ changes only when edges collapse.
- In particular, the wavefront never splits into parts.



- Consequence: We can construct $\mathcal{S}(\mathcal{C})$ in $\mathcal{O}(n \log n)$ time.

Lemmata I

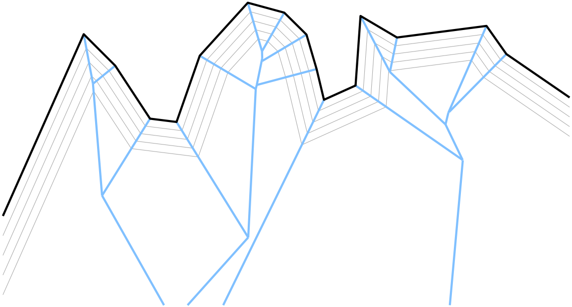
- The wavefront propagation of strictly monotone polygonal chain $\mathcal{C}$ changes only when edges collapse.
- In particular, the wavefront never splits into parts.



- Consequence: We can construct $\mathcal{S}(\mathcal{C})$ in $\mathcal{O}(n \log n)$ time.

Lemmata I

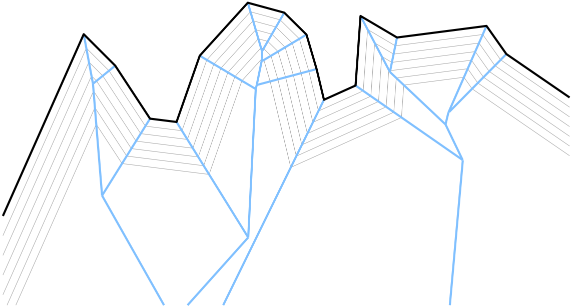
- The wavefront propagation of strictly monotone polygonal chain $\mathcal{C}$ changes only when edges collapse.
- In particular, the wavefront never splits into parts.



- Consequence: We can construct $\mathcal{S}(\mathcal{C})$ in $\mathcal{O}(n \log n)$ time.

Lemmata I

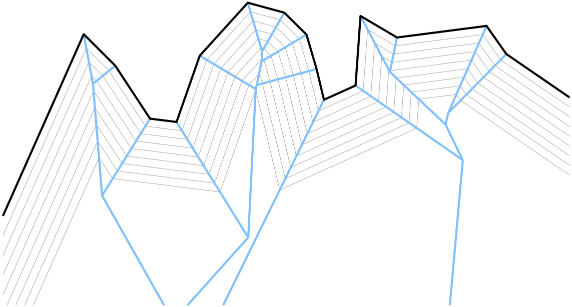
- The wavefront propagation of strictly monotone polygonal chain $\mathcal{C}$ changes only when edges collapse.
- In particular, the wavefront never splits into parts.



- Consequence: We can construct $\mathcal{S}(\mathcal{C})$ in $\mathcal{O}(n \log n)$ time.

LEMMATA I

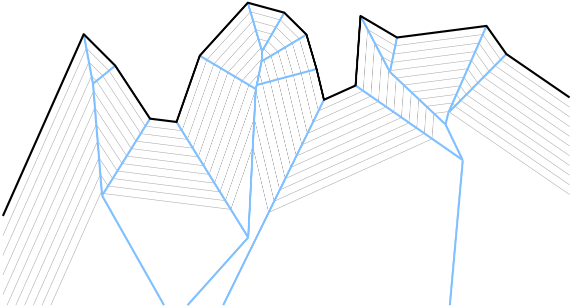
- The wavefront propagation of strictly monotone polygonal chain $\mathcal{C}$ changes only when edges collapse.
- In particular, the wavefront never splits into parts.



- Consequence: We can construct $\mathcal{S}(\mathcal{C})$ in $\mathcal{O}(n \log n)$ time.

Lemmata I

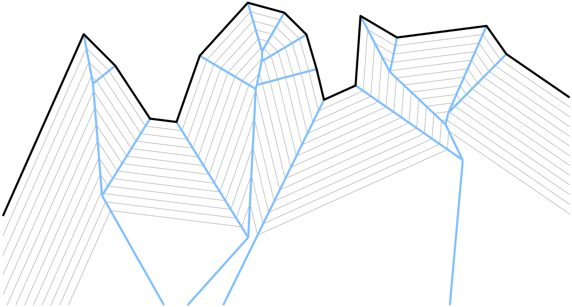
- The wavefront propagation of strictly monotone polygonal chain $\mathcal{C}$ changes only when edges collapse.
- In particular, the wavefront never splits into parts.



- Consequence: We can construct $\mathcal{S}(\mathcal{C})$ in $\mathcal{O}(n \log n)$ time.

LEMMATA I

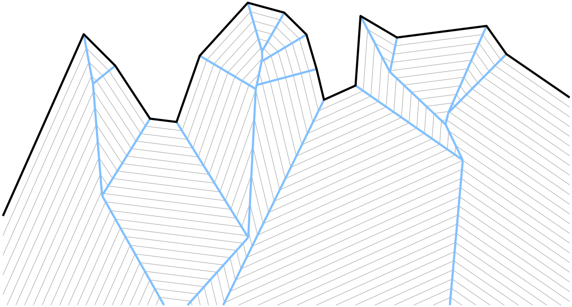
- The wavefront propagation of strictly monotone polygonal chain $\mathcal{C}$ changes only when edges collapse.
- In particular, the wavefront never splits into parts.



- Consequence: We can construct $\mathcal{S}(\mathcal{C})$ in $\mathcal{O}(n \log n)$ time.

LEMMATA I

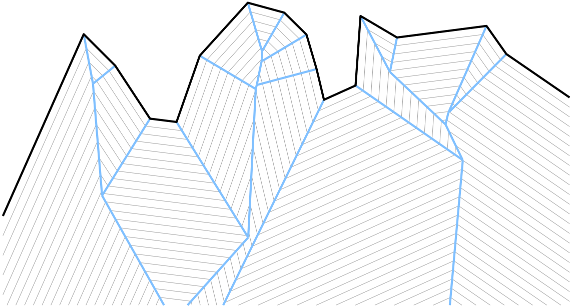
- The wavefront propagation of strictly monotone polygonal chain $\mathcal{C}$ changes only when edges collapse.
- In particular, the wavefront never splits into parts.



- Consequence: We can construct $\mathcal{S}(\mathcal{C})$ in $\mathcal{O}(n \log n)$ time.

Lemmata I

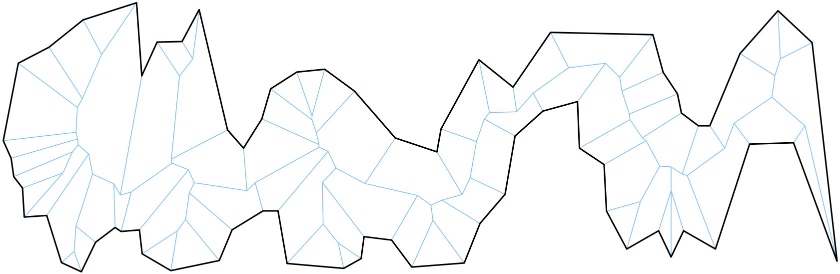
- The wavefront propagation of strictly monotone polygonal chain $\mathcal{C}$ changes only when edges collapse.
- In particular, the wavefront never splits into parts.



- Consequence: We can construct $\mathcal{S}(\mathcal{C})$ in $\mathcal{O}(n \log n)$ time.

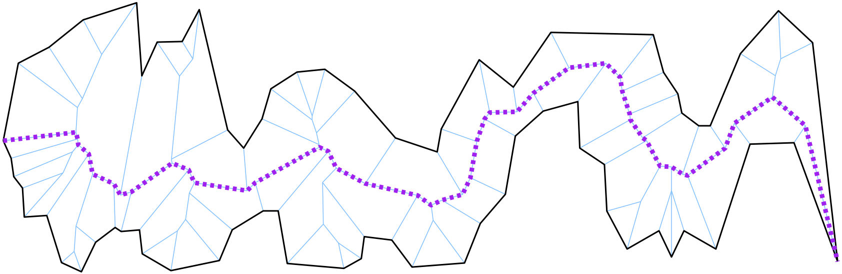
Lemmata II

- In $\mathcal{S}(\mathcal{P})$, a unique chain $\mathcal{M}$ of arcs connects west to east.
- $\mathcal{S}(\mathcal{P})$ north of $\mathcal{M}$ is not influenced by the south chain.
- $\mathcal{S}(\mathcal{P})$ **north of** $\mathcal{M}$ **is identical to** $\mathcal{S}(\mathcal{C}_n)$ **between** $\mathcal{C}_n$ **and** $\mathcal{M}$.
- Given $\mathcal{C}_n$, $\mathcal{C}_s$, $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$, we can find $\mathcal{M}$ in time $\mathcal{O}(n \log n)$.
- $\mathcal{S}(\mathcal{P})$ comprises $\mathcal{M}$ and parts of $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$



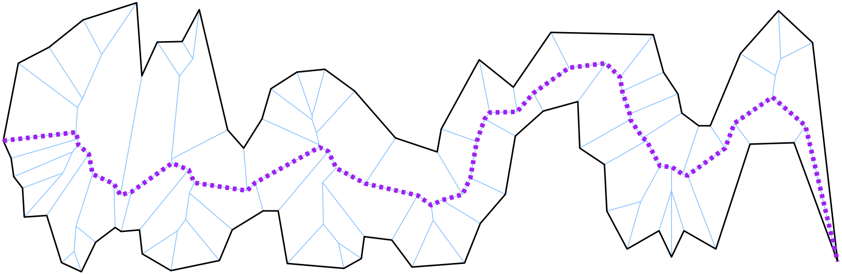
LEMMATA II

- In $\mathcal{S}(\mathcal{P})$, a unique chain $\mathcal{M}$ of arcs connects west to east.
- $\mathcal{S}(\mathcal{P})$ north of $\mathcal{M}$ is not influenced by the south chain.
- **$\mathcal{S}(\mathcal{P})$ north of $\mathcal{M}$ is identical to $\mathcal{S}(\mathcal{C}_n)$ between $\mathcal{C}_n$ and $\mathcal{M}$.**
- Given $\mathcal{C}_n$, $\mathcal{C}_s$, $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$, we can find $\mathcal{M}$ in time $\mathcal{O}(n \log n)$.
- $\mathcal{S}(\mathcal{P})$ comprises $\mathcal{M}$ and parts of $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$



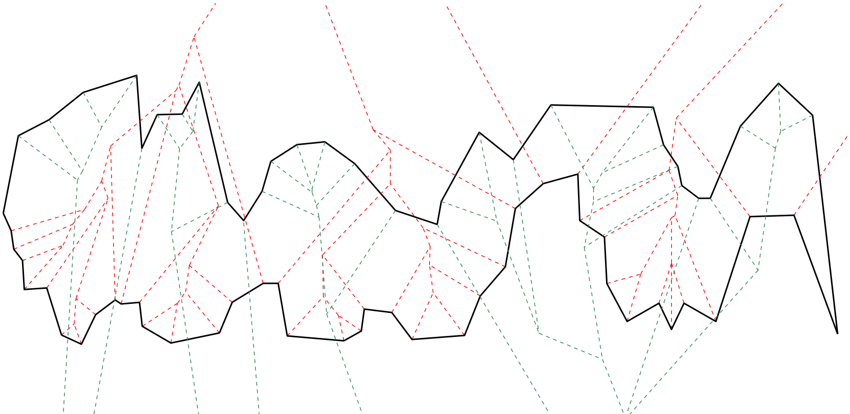
Lemmata II

- In $\mathcal{S}(\mathcal{P})$, a unique chain $\mathcal{M}$ of arcs connects west to east.
- $\mathcal{S}(\mathcal{P})$ north of $\mathcal{M}$ is not influenced by the south chain.
- **$\mathcal{S}(\mathcal{P})$ north of $\mathcal{M}$ is identical to $\mathcal{S}(\mathcal{C}_n)$ between $\mathcal{C}_n$ and $\mathcal{M}$.**
- Given $\mathcal{C}_n$, $\mathcal{C}_s$, $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$, we can find $\mathcal{M}$ in time $\mathcal{O}(n \log n)$.
- $\mathcal{S}(\mathcal{P})$ comprises $\mathcal{M}$ and parts of $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$



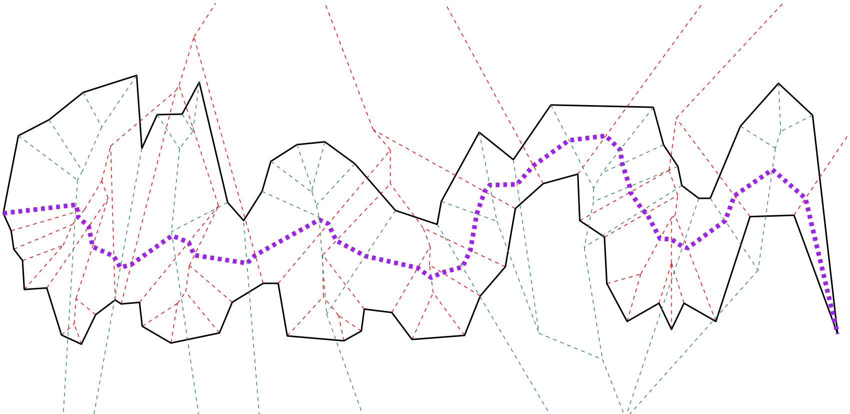
Lemmata II

- In $\mathcal{S}(\mathcal{P})$, a unique chain $\mathcal{M}$ of arcs connects west to east.
- $\mathcal{S}(\mathcal{P})$ north of $\mathcal{M}$ is not influenced by the south chain.
- **$\mathcal{S}(\mathcal{P})$ north of $\mathcal{M}$ is identical to $\mathcal{S}(\mathcal{C}_n)$ between $\mathcal{C}_n$ and $\mathcal{M}$.**
- Given $\mathcal{C}_n$, $\mathcal{C}_s$, $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$, we can find $\mathcal{M}$ in time $\mathcal{O}(n \log n)$.
- $\mathcal{S}(\mathcal{P})$ comprises $\mathcal{M}$ and parts of $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$



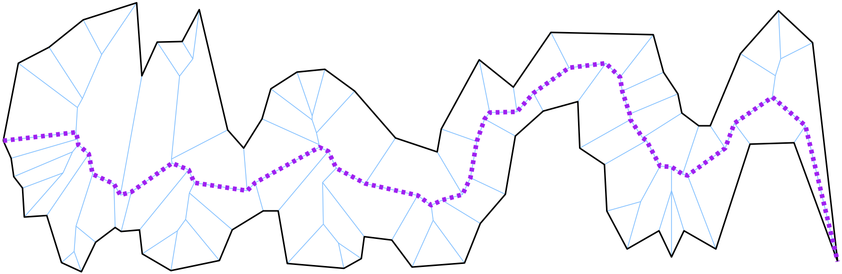
LEMMATA II

- In $\mathcal{S}(\mathcal{P})$, a unique chain $\mathcal{M}$ of arcs connects west to east.
- $\mathcal{S}(\mathcal{P})$ north of $\mathcal{M}$ is not influenced by the south chain.
- **$\mathcal{S}(\mathcal{P})$ north of $\mathcal{M}$ is identical to $\mathcal{S}(\mathcal{C}_n)$ between $\mathcal{C}_n$ and $\mathcal{M}$.**
- Given $\mathcal{C}_n$, $\mathcal{C}_s$, $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$, we can find $\mathcal{M}$ in time $\mathcal{O}(n \log n)$.
- $\mathcal{S}(\mathcal{P})$ comprises $\mathcal{M}$ and parts of $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$



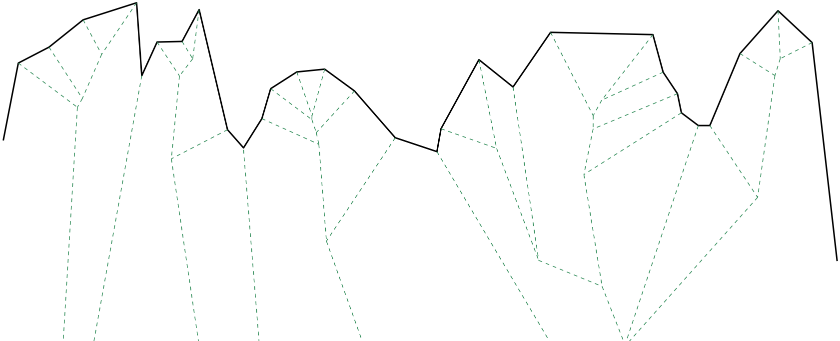
LEMMATA II

- In $\mathcal{S}(\mathcal{P})$, a unique chain $\mathcal{M}$ of arcs connects west to east.
- $\mathcal{S}(\mathcal{P})$ north of $\mathcal{M}$ is not influenced by the south chain.
- $\mathcal{S}(\mathcal{P})$ **north of** $\mathcal{M}$ **is identical to** $\mathcal{S}(\mathcal{C}_n)$ **between** $\mathcal{C}_n$ **and** $\mathcal{M}$.
- Given $\mathcal{C}_n$, $\mathcal{C}_s$, $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$, we can find $\mathcal{M}$ in time $\mathcal{O}(n \log n)$.
- $\mathcal{S}(\mathcal{P})$ comprises $\mathcal{M}$ and parts of $\mathcal{S}(\mathcal{C}_n)$ and $\mathcal{S}(\mathcal{C}_s)$



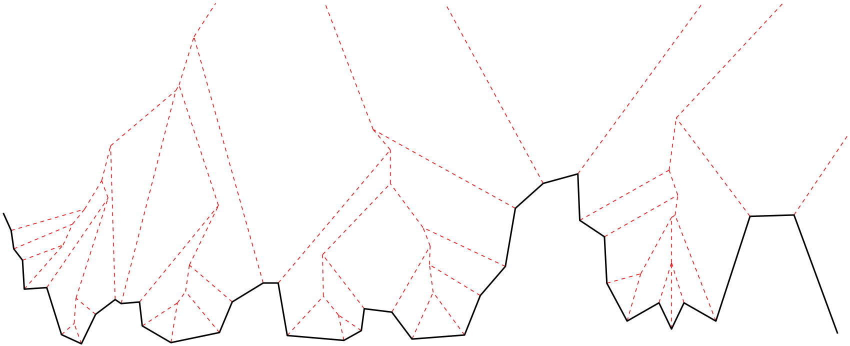
ALGORITHM

- Independently construct the north and south skeletons.
- Construct $\mathcal{M}$ and clip the north and south skeletons.
- Assemble $\mathcal{S}(\mathcal{P})$ out of these parts.



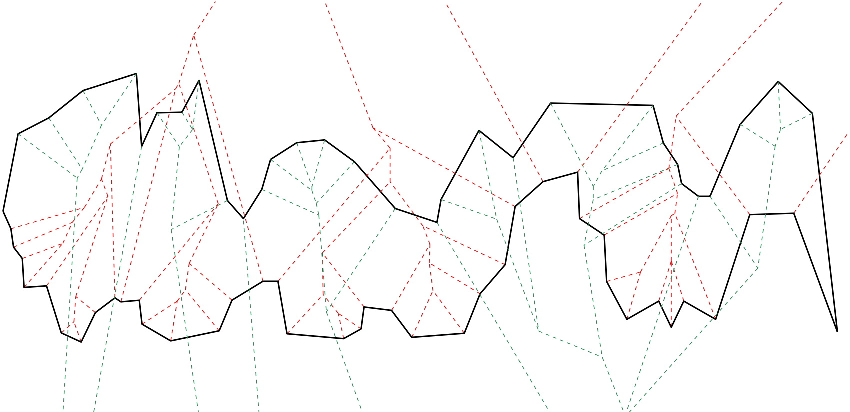
ALGORITHM

- Independently construct the north and south skeletons.
- Construct $\mathcal{M}$ and clip the north and south skeletons.
- Assemble $\mathcal{S}(\mathcal{P})$ out of these parts.



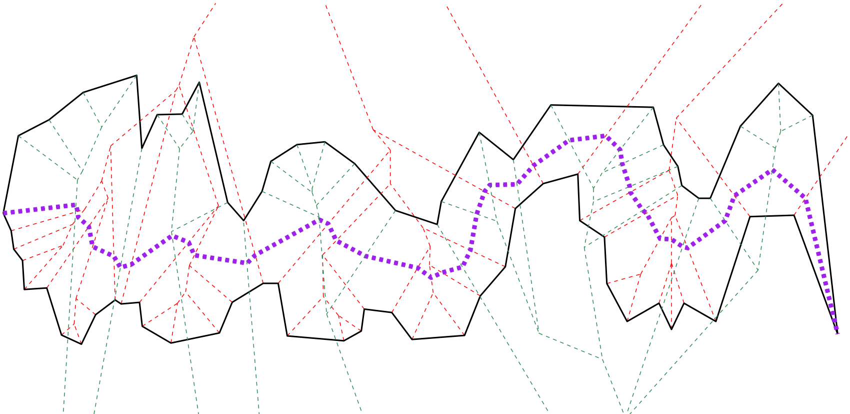
ALGORITHM

- Independently construct the north and south skeletons.
- Construct $\mathcal{M}$ and clip the north and south skeletons.
- Assemble $\mathcal{S}(\mathcal{P})$ out of these parts.



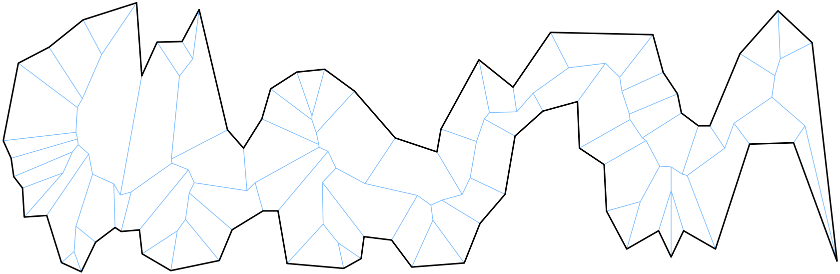
ALGORITHM

- Independently construct the north and south skeletons.
- Construct $\mathcal{M}$ and clip the north and south skeletons.
- Assemble $\mathcal{S}(\mathcal{P})$ out of these parts.



ALGORITHM

- Independently construct the north and south skeletons.
- Construct $\mathcal{M}$ and clip the north and south skeletons.
- Assemble $\mathcal{S}(\mathcal{P})$ out of these parts.



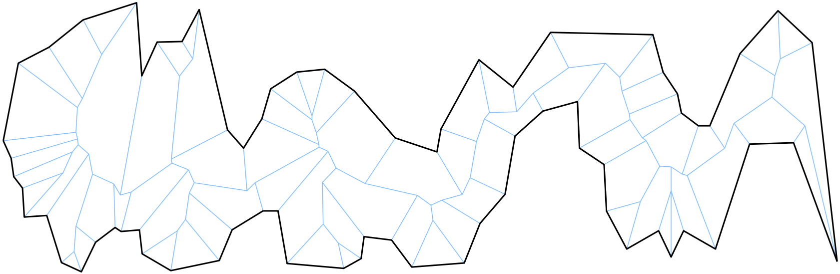
ALGORITHM

- Independently construct the north and south skeletons.
- Construct $\mathcal{M}$ and clip the north and south skeletons.
- Assemble $\mathcal{S}(\mathcal{P})$ out of these parts.

THEOREM:

We can construct $\mathcal{S}(\mathcal{P})$ of strictly monotone polygons $\mathcal{P}$ in time $\mathcal{O}(n \log n)$.

- This also works for the positively weighted straight skeleton $\mathcal{S}_\sigma(\mathcal{P})$.



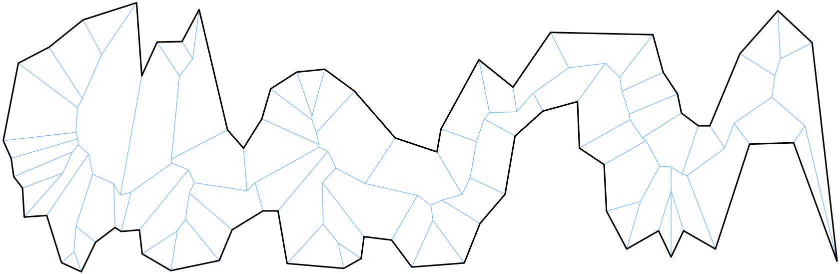
Algorithm

- Independently construct the north and south skeletons.
- Construct $\mathcal{M}$ and clip the north and south skeletons.
- Assemble $\mathcal{S}(\mathcal{P})$ out of these parts.

Theorem:

We can construct $\mathcal{S}(\mathcal{P})$ of strictly monotone polygons $\mathcal{P}$ in time $\mathcal{O}(n \log n)$.

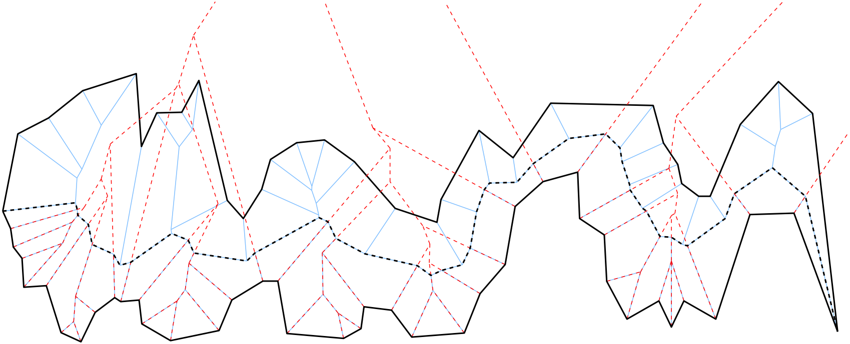
- This also works for the positively weighted straight skeleton $\mathcal{S}_\sigma(\mathcal{P})$.



Straight Skeletons of Monotone Polygons

Theorem:

We can construct $\mathcal{S}(\mathcal{P})$ of strictly monotone polygons $\mathcal{P}$ in time $\mathcal{O}(n \log n)$.



REFERENCES I

[1] Oswin Aichholzer and Franz Aurenhammer. Straight Skeletons for General Polygonal Figures in the Plane. In A.M. Samoilenko, editor, *Voronoi's Impact on Modern Sciences II*, pages 7–21. Institute of Mathematics of the National Academy of Sciences of Ukraine, Kiev, Ukraine, 1998.

[2] Oswin Aichholzer, Franz Aurenhammer, David Alberts, and Bernd Gärtner. A Novel Type of Skeleton for Polygons. *Journal of Universal Computer Science*, 1(12):752–761, 1995.

[3] Fernando Cacciola. A CGAL Implementation of the Straight Skeleton of a Simple 2D Polygon with Holes. In *2nd CGAL User Workshop*, Polytechnic Univ., Brooklyn, New York, USA, June 2004.

[4] Gautam K. Das, Asish Mukhopadhyay, Subhas C. Nandy, Sangameswar Patil, and S. V. Rao. Computing the Straight Skeleton of a Monotone Polygon in $O(n \log n)$ Time. In *Proceedings of the 22nd Canadian Conference on Computational Geometry (CCCG 2010)*, pages 207–210, August 2010.

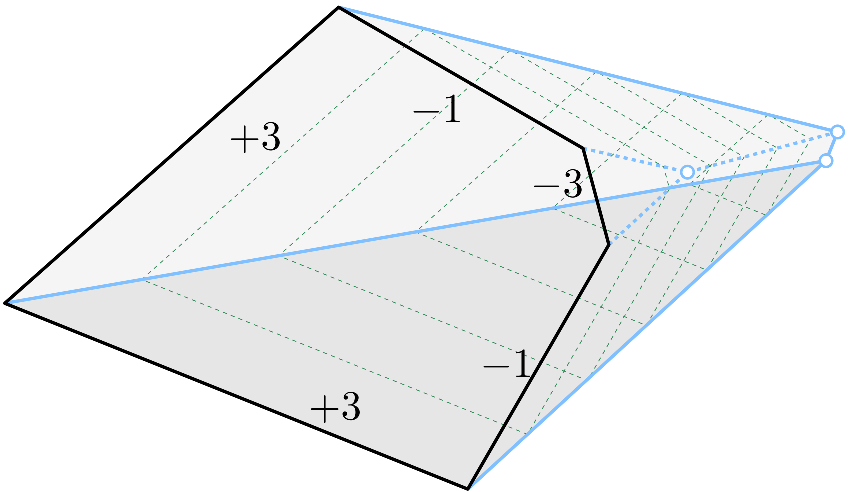
[5] David Eppstein and Jeff Erickson. Raising Roofs, Crashing Cycles, and Playing Pool: Applications of a Data Structure for Finding Pairwise Interactions. *Discrete & Computational Geometry*, 22(4):569–592, 1999.

[6] Petr Felkel and Štěpán Obdržálek. Straight Skeleton Implementation. In *Proceedings of the 14th Spring Conference on Computer Graphics (SCCG 1998)*, pages 210–218, Budmerice, Slovakia, April 1998.

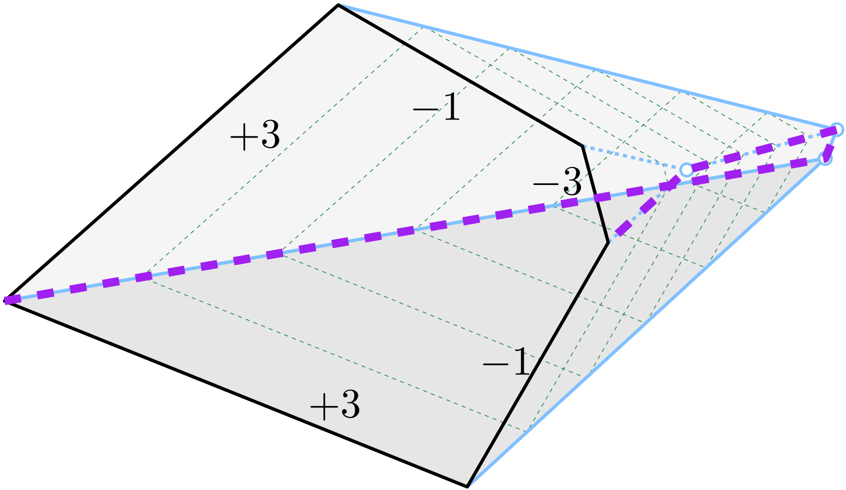
[7] Stefan Huber and Martin Held. A Fast Straight-Skeleton Algorithm Based On Generalized Motorcycle Graphs. *International Journal of Computational Geometry*, 22(5):471–498, October 2012.

[8] Antoine Vigneron and Lie Yan. A Faster Algorithm for Computing Motorcycle Graphs. In *Proceedings of the 29th Symposium on Computational Geometry (SoCG 2013)*, pages 17–26, Rio de Janeiro, Brazil, 2013.

Negative weights

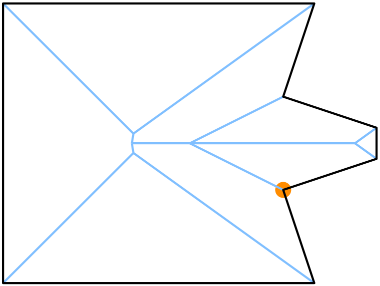


Negative weights



Das et al.[4]: claim $\mathcal{O}(n \log n)$ time algorithm:

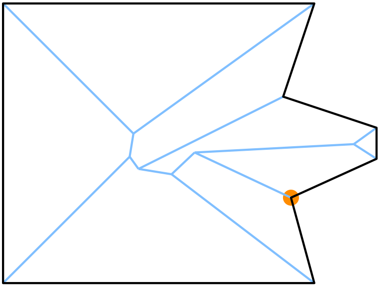
- Cannot handle vertex events (perturbation cannot work).
- Wrongly assumes that split event nodes are located at an offset that is the distance to their closest supporting line (similar to Felkel[6]).



MONOTONE POLYGONS - DAS ET AL.

Das et al.[4]: claim $\mathcal{O}(n \log n)$ time algorithm:

- Cannot handle vertex events (perturbation cannot work).
- Wrongly assumes that split event nodes are located at an offset that is the distance to their closest supporting line (similar to Felkel[6]).



MONOTONE POLYGONS - DAS ET AL.

Das et al.[4]: claim $\mathcal{O}(n \log n)$ time algorithm:

- Cannot handle vertex events (perturbation cannot work).
- Wrongly assumes that split event nodes are located at an offset that is the distance to their closest supporting line (similar to Felkel[6]).