www.cadanda.com
Computer-Aided Design AND APPLICATIONS

On Modeling Coverage Areas of Anisotropic Transmitters by Voronoi-like Structures Based on Star-Shaped Distance Measures

Martin Held[1] and Peter Palfrader[2]

[1]FB Informatik, University of Salzburg, Austria, held@cs.sbg.ac.at
[2]FB Informatik, University of Salzburg, Austria, palfrader@cs.sbg.ac.at

Corresponding author: Martin Held, held@cs.sbg.ac.at

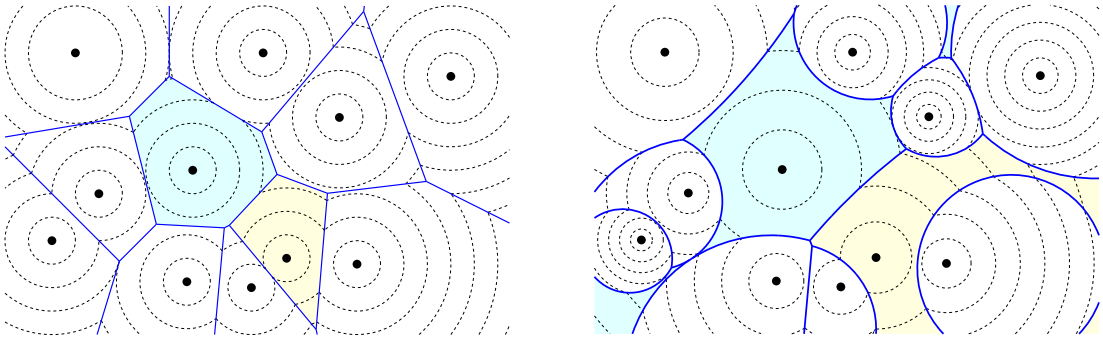
Abstract. The geometric modeling of coverage areas is a well-known problem in the analysis of networks of sensor or transmitters. Prior work often uses Voronoi diagrams of the device locations to obtain estimates of their coverage areas in the plane. In this work we extend these approaches by allowing the signal propagation to be non-uniform both among the devices as well as relative to different directions for an individual device. Depending on whether or not the spreading of an anisotropic signal is stopped once it reaches a point of the plane that has already been covered by some other signal, we get connected or disconnected coverage areas. A proof-of-concept implementation of our algorithms is freely available via GitHub.

Keywords: Coverage area, sensor, transmitter, Voronoi diagram, multiplicative weights, star-shaped polygon, weighted straight skeleton
DOI: https://doi.org/10.14733/cadaps.2022.967-976

1 INTRODUCTION

Consider a set S of points in the plane, called sites, and a signal that is sent out from each site. Now assume that each signal starts at the same time t, say time $t := 0$, and propagates with unit speed uniformly in all directions. The locations at time $t \geq 0$ that are reached by a signal sent out from a site $s \in S$ is given by a circle ("*offset circle*") of radius t centered at s, and the area that has been covered by that signal by time t is the corresponding circular disc ("*offset disk*").

For t sufficiently small, no pair of these discs will intersect. However, as t increases, intersections will occur. Apparently, intersections of two such circles correspond to points of the plane that are reached by two different signals at the same time. Assigning each locus of the plane to the site whose signal reached it first yields a partition of the plane that is well-known as the *Voronoi diagram* of S; cf. Figure 1(a). Adjacent regions of this partition are separated by straight-line segments. (We refer to the textbook by Okabe et al. [19] for more background information on Voronoi diagrams of point sites.) The boundary of the union of all offset disks at time t is commonly called the *wavefront* of S at time t. It is easy to see that every wavefront of S consists of circular arcs whose endpoints lie on the edges of the Voronoi diagram of S.



(a) (b)

Figure 1: (a) The Voronoi diagram of point sites. Wavefronts for some specific points in time are shown by dashed curves. (b) The multiplicatively weighted Voronoi diagram has regions which need not be connected (e.g., the light-blue region). Again, some wavefronts are shown by dashed curves.

Voronoi diagrams can be generalized to settings where the signals are allowed to travel at different speeds. In order to model different speeds, a *weight* $\sigma(s)$ is assigned to each site s that specifies how fast the signal travels: In this modified setting, at time t the signal has reached points that lie at a distance $\sigma(s) \cdot t$ from s. The corresponding Voronoi diagram is known as *multiplicatively weighted Voronoi diagram* [6]. The common boundary of two adjacent regions is no longer a line segment but is a circular arc. Also, the region associated with a specific site s can now be disconnected or multiply-connected; cf. Figure 1(b). In a similar way, one can generalize Voronoi diagrams by allowing the sites to start emitting their signals at different points in time. This leads to the concept of *additively weighted Voronoi diagrams*.

Voronoi diagrams have become an important geometric tool for modeling and analyzing coverage areas of sensors and transmitters. We refer to [4, 14, 18, 22] for sample publications on this application. Common to these publications is the fact that the signal propagation is assumed to be uniform both over all sites and over all directions for each site.

2 OUR CONTRIBUTION

The assumption that signals and their spreading are uniform provides only a rather coarse approximation of reality. Rather, different sites should be assumed to emit signals of different strengths. Furthermore, the spreading of a signal should be assumed to be anisotropic, i.e., to vary with the direction. For instance, antennas used by FM broadcast stations tend to have at least some amount of directivity.

In this work we show how concepts of computational geometry can be applied to provide a such refined model for a subsequent coverage analysis: We interpret increasing distance in the Voronoi setting as decreasing signal strength and provide a system where initial signal strength may vary among the sites and its rate of change may vary over different directions for each site. We approximate the azimuth radiation patterns by (star-shaped) polygons. Depending on whether or not the spreading of a signal is stopped once it reaches a point of the plane that has already been covered by some other signal, we get connected or disconnected coverage areas.

3 STAR-INDUCED VORONOI DIAGRAM

In order to model an anisotropic spreading of signals, we consider a variant of point-site Voronoi diagrams: We no longer use an offset circle to model the area covered by one site's signal. Rather, we replace the circle

by a star-shaped polygon that contains the site in its kernel. (Recall that a polygonal area is star-shaped if it contains at least one point from which its entire boundary is visible; the set of all those points is called its kernel.) We call such a polygon an *offset star*. Mitered offsets of an offset star are the appropriate generalization of the expansion of offset circles; cf. Figure 2: If a vertex v of the offset star of s at time t is at distance $t \cdot d$ from s, for some $d \geq 0$, then it will be at distance $t' \cdot d$ from s at time t', thereby also moving on the ray from s through v. Of course, we allow offset stars of different shapes and sizes for different sites. And, in an analogy to additively weighted Voronoi diagrams, we allow the offset stars to start their expansion at different points in time.

One way to interpret this generalization is that each input site is the location of a transmitter whose signal strength decays with distance, but not at the same rate in every direction. Then, offset stars of the same shape but at different sizes can be seen as transmitters with the same anisotropic emission characteristics but whose signals decay at different rates. Furthermore, the wavefronts derived from the expansion of all offset stars are iso-contours of signal strength.

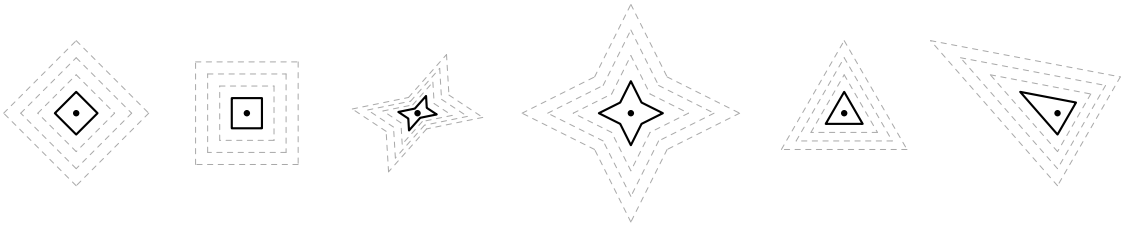


Figure 2: Six sites with different offset stars are shown. Some mitered offsets for distinct points in time show the expansion of the offset stars, i.e., how the signals spread in different directions.

In theory, any star-shaped polygon that contains its site in its kernel can be used as offset star for that site. Of course, the more vertices the polygon has, the finer a direction-dependence of the spreading of the signal can be modeled. Feedback obtained from companies tells us that polygons with 10–20 vertices will be good enough for practical applications. Note that the offset stars shown in this work (cf. Figure 2) were chosen for visual clarity and simplicity rather than genuine practical relevance. In practice, (mostly) convex rather than extremely spiky shapes will prevail as regions that model the (horizontal) transmission characteristics of transmitters and sensors.

Similar to how classic Voronoi diagrams tessellate the plane into regions such that all points within the same region have been reached first by the same offset circle, we now want to partition the plane into regions such that all points of a region are reached first by the same offset star. We call this structure the *star-induced (weighted) Voronoi diagram*; cf. Figure 3. As for standard multiplicatively weighted Voronoi diagrams, some regions may be disconnected and consist of more than one face. For instance the purple region of the site close to the bottom-right corner of Figure 3 consists of two faces. (Several more disconnected components show up outside of the image frame.)

3.1 Computing a Star-Induced Voronoi Diagram

For the sake of descriptional simplicity, we start with assuming that all additive weights are zero, i.e., that all signals start to spread at the same time. All offset stars are scaled uniformly such that no pair of offset stars overlaps at time $t := 1$. (This can be done easily based on the standard Voronoi diagram of the sites.)

We will now take a look at how that portion inside of the wavefront that belongs to a site's region changes as time progresses. At time $t := 0$, the signal of each transmitter has not yet spread at all. Hence, at this time, each site's region within the wavefront only consists of the site itself. By our assumption, at time $t := 1$ no two signals have yet interfered, and each site's signal has covered its corresponding offset star.

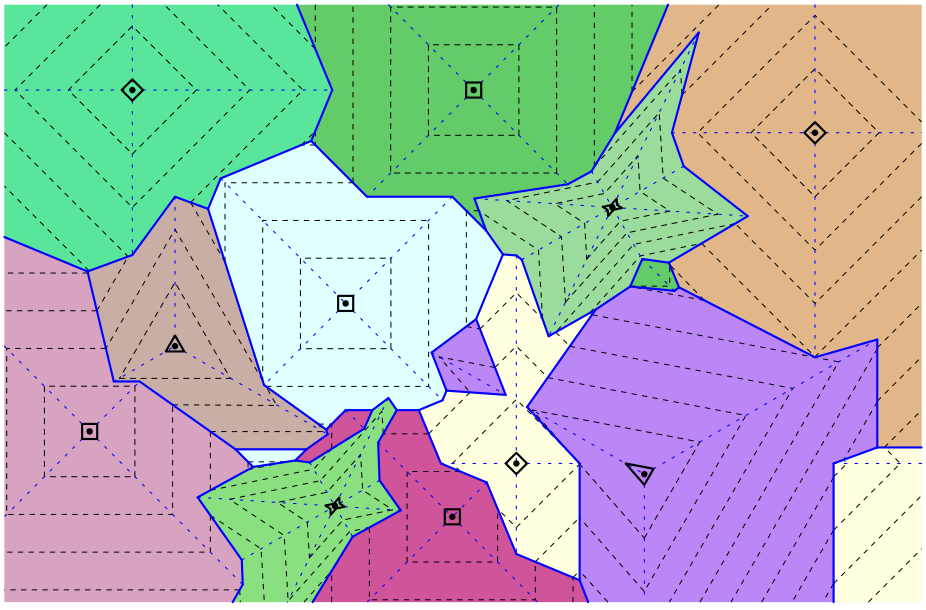


Figure 3: A sample star-induced Voronoi diagram for the sites shown in Figure 1. Each region is assigned a unique color and its offset star is shown. Black dashed lines show a family of wavefronts.

Of course, intersections of the offset stars will occur as time progresses. To get a hand on these intersections and obtain an actual algorithm for constructing a star-induced Voronoi diagram, we lift the entire problem into $\mathbb{R}^3$, where the input plane (of $\mathbb{R}^2$) is the xy-plane and the third coordinate represents the time t. Each site's signal is represented by one upside-down, infinitely tall, right pyramid whose apex coincides with the site's location. The inclinations of the lateral faces of the pyramid are chosen such that the intersection of the pyramid with the plane that is parallel to the xy-plane and whose t-value equals 1 matches exactly the boundary of the offset star. We call such a pyramid an *offset pyramid*. The intercept theorem implies that the intersection of the offset pyramid with some other plane parallel to the xy-plane yields a polygon that is a mitered (inward or outward) offset of the offset star that defines the pyramid.

Theory knows of a fairly general relation between Voronoi diagrams in $\mathbb{R}^2$ and lower envelopes of suitable distance functions in $\mathbb{R}^3$, see [9]. This relation is also applicable to our problem: The lower envelope of all offset pyramids projected to the xy-plane or, equivalently, their so-called minimization diagram, yields our star-induced Voronoi diagram.

One can show that a delay in the start time of the signal of site s can be handled by lifting the pyramid of s vertically upwards above s: If it is to start at time t', then its apex lies (vertically above s) in the plane $t = t'$. No other modification of this general scheme is needed.

The relation between Voronoi diagrams in $\mathbb{R}^2$ and lower envelopes in $\mathbb{R}^3$ also provides further insights: If each offset star has only a constant number of edges then results by Edelsbrunner [8] tell us that the maximum combinatorial complexity of the star-induced Voronoi diagram of n sites is $\Theta(n^2\alpha(n))$, where α is the extremely slow growing inverse Ackermann function. Furthermore, the divide&conquer algorithm presented by Agarwal et al. [1] allows computing the lower envelope and, thus, also the star-induced Voronoi diagram of n sites in worst-case time $O(n^{2+\varepsilon})$, for any fixed $\varepsilon > 0$. These complexity bounds can be adapted in a straightforward way if the n offset stars would have more than $O(n)$ edges in total.

3.2 Implementation

We have developed a proof-of-concept implementation of this approach using exact arithmetic, based on CGAL and, in particular, based on CGAL's 3D Envelopes package [12] that implements the algorithm described by Agarwal et al. [1]. To make it easier to use existing CGAL code, we decided to use finite triangles as the lateral faces of our pyramids instead of infinitely large ones. Of course, this raised a new problem: How tall is "sufficiently tall" to get the correct lower envelope and, thus, also the star-induced Voronoi diagram?

To obtain an upper bound we proceed as follows. First, we consider the supporting lines of all edges of the offset stars as they move away from their respective stars. At some point in time each supporting line will have the entire bounding box B of the input sites on the same side. If we make all pyramids at least this tall, then the projection of every pyramid onto the xy-plane will cover all of B, and the union of the pyramids' lateral faces projected to the xy-plane will form a star-shaped polygon whose kernel contains all of B. This implies that the minimization diagram restricted to B is guaranteed to represent the Voronoi diagram restricted to B. However, using pyramids of that size does not guarantee that the minimization diagram correctly represents the Voronoi diagram outside of B, even in places where it is defined.

Therefore, as a second step, we also attempt to find the latest point in time when a reflex vertex of the mitered offset of an offset star pierces the supporting plane of any other pyramid face. After this time we know that the boundary of the minimization diagram will not see new vertices appear even if we proceed further in time. There still might be changes as edges of the boundary shrink to length zero but if need be, these changes could be dealt with easily. (In a nutshell, we store the times of these events in a priority queue and proceed similar to the standard theory of wavefront propagation, see, e.g., [16].)

We emphasize that the question of how tall the pyramids need to be, respectively how far in time one has to go with the wavefront propagation, is mostly academic. In practice, we are almost certainly given a region R of interest for which we are to compute the diagram. And this is easily achieved, since it suffices to ensure that R is covered by the projection of each pyramid. Then we can compute the minimization diagram. Seen from a purely theoretical point of view, the algorithms to compute lower envelopes [1] work just fine with infinitely large surfaces, and implementational convenience is of no concern, anyway.

4 STAR-INDUCED SKELETON

If the star-induced Voronoi diagram described in the previous section is used to model coverage areas, signals are sent out from sites, and then they spread across the plane $\mathbb{R}^2$ without affecting each other: Recall that the star-induced Voronoi diagram may contain disconnected regions, which implies that one signal had to travel over an area already covered by some other signal. Now we consider a second variant in which the signal paths do not overlap with each other. It is based on straight skeletons and allows modeling coverage areas that are connected.

4.1 Preliminaries

The straight skeleton of a polygon was introduced to computational geometry by Aichholzer et al. [3]. It is an angular bisector graph that is the result of a wavefront propagation process: The edges of the polygon start moving inwards in parallel and at the same speed while, initially at least, maintaining their neighborhood relations. Similarly, to our expanding offset stars, these moving edges, the so-called wavefront edges, form the wavefront polygon. Updates to the wavefront happen, when edges collapse to a length of zero and are removed in so-called *edge events*, or when a vertex of the wavefront polygon moves into a previously non-incident edge, thereby splitting a wavefront polygon into two subpolygon at *split events*. The process is shown for a polygon in Figure 4(a).

The straight skeleton can be generalized by allowing edges to move at different speeds, resulting in the (multiplicatively) weighted straight skeleton, first mentioned by Aichholzer and Aurenhammer [2] and Eppstein

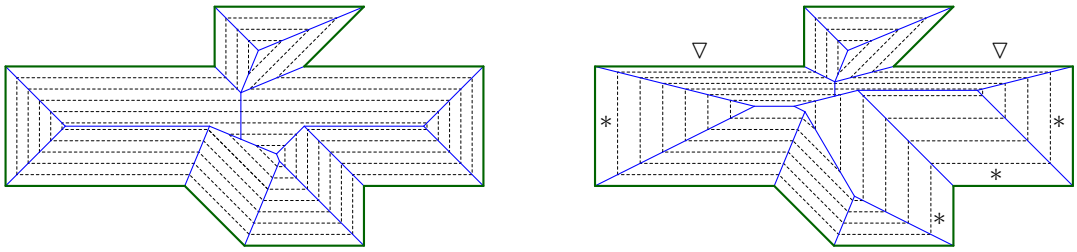


Figure 4: (a) The (unweighted) straight skeleton (in blue) plus a family of wavefronts (dashed) for the green polygon. (b) The weighted straight skeleton for the case that edges marked with $*$ have twice the weight and edges marked with $\triangledown$ have half the weight of the unmarked edges.

and Erickson [11]. Many, though not all of the properties of unweighted straight skeletons carry over to positively weighted straight skeletons. For instance, the straight skeleton of an n-vertex polygon is still a tree with $O(n)$ edges and has no crossings. However, the faces induced by the straight skeleton are no longer necessarily monotone with respect to their defining input edges [5]. Figure 4(b) depicts a weighted straight skeleton. Further, *additive* weights can be employed where wavefront edges start moving at different times [16].

The concept can be further generalized by considering arbitrary planar straight-line graphs (PSLGs) as input instead of just one polygon [2]. The edges of a planar straight-line graphs are straight-line segments that do not intersect except at common endpoints which form the vertices of the graph. Vertices of degree one need some special setup of the wavefront, but other than that the wavefront propagation proceeds just like for the straight skeleton in the interior of one polygon; cf. Figure 5.

The currently fastest straight-skeleton algorithms with the best worst-case bounds are by Eppstein and Erickson [11] and Vigneron et al. [7, 21]. These algorithms seem difficult to implement, though. The straight-skeleton code Bone by Huber and Held [17] handles PSLGs as input and runs in $O(n \log n)$ time and $O(n)$ space in practice. However, it is not capable of handling weighted skeletons.

Recently, a straight-skeleton library ("Surfer2") has been developed by the authors' group at Salzburg [10]. It implements an extended version of a straight-skeleton algorithm by Aichholzer and Aurenhammer [2]; it accepts weighted PSLGs as input and computes their weighted straight skeleton. Surfer2 is available under a free license at https://github.com/cgalab/surfer2.

4.2 Star-Induced Skeleton

Now consider a set of sites as in Section 3.1, again with their star-shaped offset stars scaled such that at time $t := 1$ no two offset stars overlap or intersect. We know that the wavefront consists of wavefront vertices and edges that move away from their sites at different but constant speeds as time progresses. Furthermore, every wavefront edge also moves away at a constant speed from the corresponding edge of its initial offset star (at time $t := 1$). Thus, this propagation mirrors exactly the wavefront propagation employed in the definition of straight skeletons, where edges move in a self-parallel manner at constant speeds.

Hence, we can view the offset stars at time $t := 1$ as a PSLG. Each edge e of that PSLG is weighted according to the orthogonal distance of its corresponding site from e. The resulting (multiplicatively) PSLG can be taken as the input for a straight skeleton algorithm. Inside of each offset star, the wavefront propagation will cause the offset star to collapse exactly at the site in a (multi-)edge event. On the outside of the offset stars, the wavefront propagation will observe split and edge events as it progresses. Once the wavefront

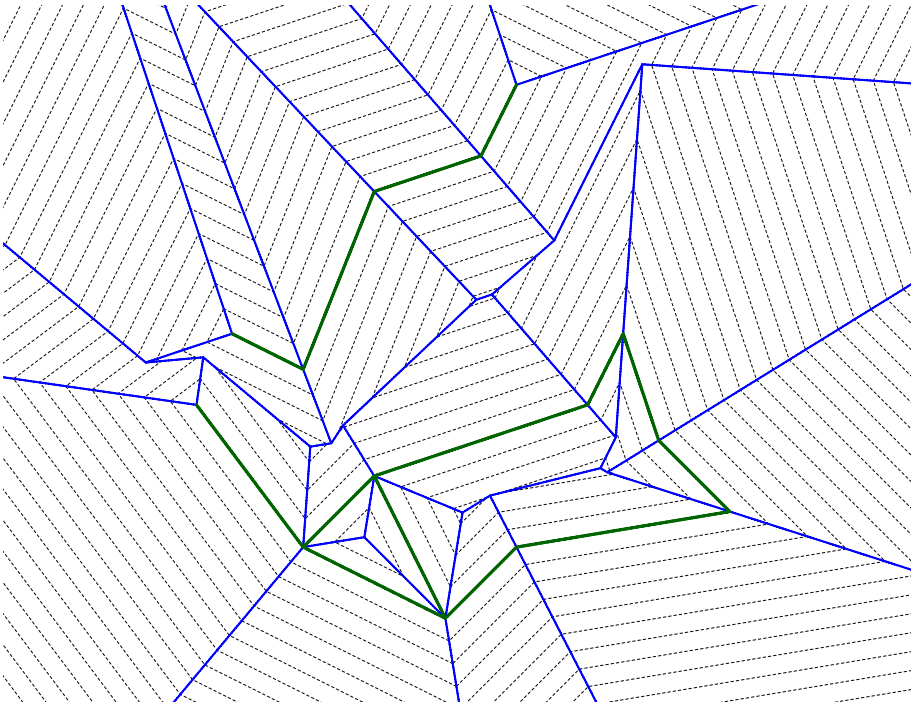


Figure 5: Straight skeleton of a planar straight-line graph with families of offset curves.

propagation has finished, the entire plane has been tessellated by the straight skeleton. We call the resulting structure the *star-induced skeleton*; cf. Figure 6.

As a result of how the straight skeleton propagation works, a site's signal stops to spread once it reaches a point that had already been covered by some other signal. Also, as straight skeleton faces are connected, this yields a partition of the plane into regions where no region is disconnected. In order to model the actual coverage areas we apply a post-processing step and merge faces that were traced out by edges that belong to the same offset star and, thus, obtain one connected region per input site, i.e., per signal.

Implementation Our implementation for computing the star-induced skeleton is based on CGAL and our own SURFER2 library. We first compute appropriately scaled offset stars and appropriate weights for all edges of the weighted PSLG. Then input the resulting weighted PSLG into the straight skeleton library to compute the straight skeleton of all offset stars. In a post-processing step we then merge straight-skeleton faces that come from different edges of the same offset star, thus obtaining our star-induced skeleton that models the coverage areas.

5 DISCCUSSION AND CONCLUSION

We present two models of coverage areas of anisotropic transmitters in the plane. Both models rely on Voronoi-like structures induced by star-shaped distance measures, but they differ in how the signals are assumed to interact: star-induced Voronoi diagrams model disconnected coverage areas while star-induced skeletons model connected areas.

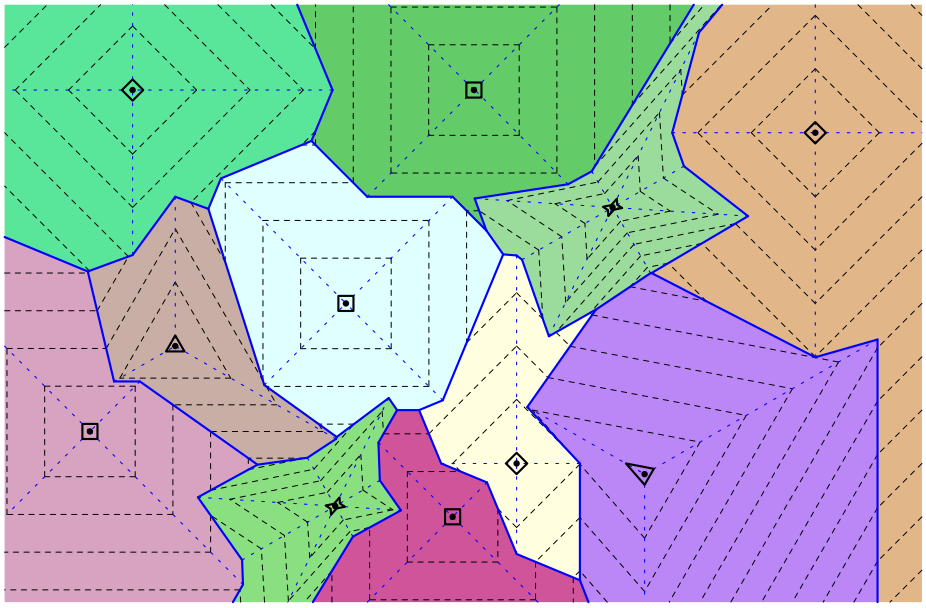


Figure 6: A sample star-induced skeleton for the same weighted sites as in Figures 1 and 3. Dashed blue lines are straight-skeleton arcs that are removed in the post-processing step, solid blue lines are the final edges tessellating the planes into distinct regions, with one connected region per input site. A family of wavefronts is shown for the same points in time as in Figure 3.

Our current implementation makes use of exact arithmetic, as provided by CGAL. Not having to worry about numerical stability problems caused by the use of a standard floating-point arithmetic allowed us to save time devoted to the implementational work. Experimental tests documented in [10] show that SURFER2 achieves $O(n \log n)$ runtime for the computation of straight skeletons of n-vertex weighted PSLGs. But those tests made it also apparent that there is a significant price to pay for the use of exact arithmetic.

Hence, it seems natural to extend the algorithm discussed in [20] to positive weights, in an attempt to provide an alternative implementation that runs on a standard floating-point arithmetic. Work on the computation of lower envelopes that does not rely on CGAL and exact arithmetic has already been started at the authors' group. This can be expected to facilitate an actual use of our work for real-world applications.

Another interesting avenue for further work is research on the combinatorial complexity bounds for star-induced Voronoi diagrams. Preliminary experiments indicate that the current bounds are far too pessimistic for practical input. Indeed, theoretical and practical analysis shows that multiplicatively weighted Voronoi diagrams of point sites can be assumed to have only a slightly super-linear combinatorial complexity [13, 15]. It is a fascinating question whether and under which conditions these results carry over to our star-induced Voronoi diagrams.

Finally, there is another avenue for both practical as well as theoretical work: It is a law of physics that transmission distance attenuates the signal strength. Similar to prior work on the Voronoi-based modelling of coverage areas we also assume signal strength to decay linearly with distance. However, field strength of a radio signal decays with the square of the distance. Hence, a refined modeling of coverage areas for long-distance signal propagation will benefit from resorting to another distance measure.

6 SOURCE CODE

Our proof-of-concept code is available via GitHub and can be used freely under the GPL(v3) license: See https://github.com/cgalab/stardist/.

Acknowledgements

This research was supported by the Austrian Science Fund (FWF): Grant P31013-N31.

Martin Held, http://orcid.org/0000-0003-0728-7545
Peter Palfrader, http://orcid.org/0000-0002-5796-6362

REFERENCES

[1] Agarwal, P.K.; Schwarzkopf, O.; Sharir, M.: The Overlay of Lower Envelopes and Its Applications. Discrete Comput. Geom., 15(1), 1–13, 1996. http://doi.org/10.1007/BF02716576.

[2] Aichholzer, O.; Aurenhammer, F.: Straight Skeletons for General Polygonal Figures in the Plane. In Voronoi's Impact on Modern Sciences II, vol. 21, 7–21. Institute of Mathematics of the National Academy of Sciences of Ukraine, Kiev, 1998.

[3] Aichholzer, O.; Aurenhammer, F.; Alberts, D.; Gärtner, B.: A Novel Type of Skeleton for Polygons. J. Univ. Comp. Sci, 1(12), 752–761, 1995. http://doi.org/10.1007/978-3-642-80350-5_65.

[4] Argany, M.; Mostafavi, M.A.; Karimipour, F.: Voronoi-Based Approaches for Geosensor Networks Coverage Determination and Optimisation: A Survey. In 7th Int. Symp. Voronoi Diagrams in Science & Engineering, 115–123, 2010. http://doi.org/10.1109/ISVD.2010.36.

[5] Biedl, T.; Held, M.; Huber, S.; Kaaser, D.; Palfrader, P.: Weighted Straight Skeletons in the Plane. Comp. Geom.: Theory and Appl., 48(2), 120–133, 2015. http://doi.org/10.1016/j.comgeo.2014.08.006.

[6] Boots, B.N.: Weighting Thiessen Polygons. Economic Geography, 56(3), 248–259, 1980. http://doi.org/10.2307/142716.

[7] Cheng, S.W.; Mencel, L.; Vigneron, A.: A Faster Algorithm for Computing Straight Skeletons. ACM Trans. Algorithms, 12(3), 44:1–44:21, 2016. http://doi.org/10.1145/2898961.

[8] Edelsbrunner, H.: Edelsbrunner - The Upper Envelope of Piecewise Linear Functions: Tight Bounds on the Number of Faces. Discrete Comput. Geom., 4(1), 337–343, 1989. http://doi.org/10.1007/BF02187734.

[9] Edelsbrunner, H.; Seidel, R.: Voronoi Diagrams and Arrangements. Discrete Comput. Geom., 1(1), 25–44, 1986. http://doi.org/10.1007/BF02187681.

[10] Eder, G.; Held, M.; Palfrader, P.: Implementing Straight Skeletons with Exact Arithmetic: Challenges and Experiences. Comp. Geom.: Theory and Appl., 96, 101760, 2021. http://doi.org/10.1016/j.comgeo.2021.101760.

[11] Eppstein, D.; Erickson, J.: Raising Roofs, Crashing Cycles, and Playing Pool: Applications of a Data Structure for Finding Pairwise Interactions. Discrete Comput. Geom., 22(4), 569–592, 1999. http://doi.org/10.1007/PL00009479.

[12] Halperin, D.; Meyerovitch, M.; Wein, R.; Zukerman, B.: 3D Envelopes. In CGAL User and Reference Manual. CGAL Editorial Board, 5.2 ed., 2020. https://doc.cgal.org/5.2/Manual/packages.html#PkgEnvelope3.

[13] Har-Peled, S.; Raichel, B.: On the Complexity of Randomly Weighted Multiplicative Voronoi Diagrams. Discrete Comput. Geom., 53(3), 547–568, 2015. http://doi.org/10.1007/s00454-015-9675-0.

[14] Hasegawa, G.; Takemori, S.; Taniguchi, Y.; Nakano, H.: Determining Coverage Area Using Voronoi Diagram Based on Local Information for Wireless Mesh Networks. In 9th International Conference on Information Technology: New Generations, 71–76, 2012. http://doi.org/10.1109/ITNG.2012.19.

[15] Held, M.; de Lorenzo, S.: An Efficient, Practical Algorithm and Implementation for Computing Multiplicatively Weighted Voronoi Diagrams. In Proc.28thAnnu. Europ. Symp. Alg. (ESA'20), 56:1–56:15, 2020. http://doi.org/10.4230/LIPIcs.ESA.2020.56.

[16] Held, M.; Palfrader, P.: Straight Skeletons with Additive and Multiplicative Weights and Their Application to the Algorithmic Generation of Roofs and Terrains. Comp.-Aided Design, 92, 33–41, 2017. ISSN 0010-4485. http://doi.org/10.1016/j.cad.2017.07.003.

[17] Huber, S.; Held, M.: Theoretical and Practical Results on Straight Skeletons of Planar Straight-Line Graphs. In SoCG '11:Proc. twenty-seventh Ann. Sympos. Comp. Geom., 171–178. Paris, France, 2011. http://doi.org/10.1145/1998196.1998223.

[18] Megerian, S.; Koushanfar, F.; Potkonjak, M.; Srivastava, M.B.: Worst and Best-case Coverage in Sensor Networks. IEEE Trans. Mobile Comput., 4(1), 84–92, 2005. http://doi.org/10.1109/TMC.2005.15.

[19] Okabe, A.; Boots, B.; Sugihara, K.; Chiu, S.N.: Spatial Tesselations: Concepts and Applications of Voronoi Diagrams. Wiley, 2nd ed., 2000. ISBN 978-0471986355.

[20] Palfrader, P.; Held, M.; Huber, S.: On Computing Straight Skeletons by Means of Kinetic Triangulations. In L. Epstein; P. Ferragina, eds., Proc. 20th Annu. Europ. Symp. Alg. (ESA 2012), vol. 7501 of LNCS, 766–777. Springer, Ljubljana, Slovenia, 2012. http://doi.org/10.1007/978-3-642-33090-2_66.

[21] Vigneron, A.; Yan, L.: A Faster Algorithm for Computing Motorcycle Graphs. Discrete Comput. Geom., 52(3), 492–514, 2014. http://doi.org/10.1007/s00454-014-9625-2.

[22] Wang, Q.; Guo, G.; Cao, L.; Xing, X.: Voronoi Coverage Algorithm Based on Connectivity for Wireless Sensor Networks. In 34th Chinese Control Conference, 7833–7837, 2015. http://doi.org/10.1109/ChiCC.2015.7260884.