Hardness results on Voronoi, Laguerre and Apollonius diagrams

Kevin Buchin, Pedro de Castro, Olivier Devillers, Menelaos Karavelas

Hardness results on Voronoi, Laguerre and Apollonius diagrams

Kevin Buchin* Pedro Machado Manhães de Castro† Olivier Devillers‡ Menelaos Karavelas§

Abstract

We show that converting Apollonius and Laguerre diagrams from an already built Delaunay triangulation of a set of n points in 2D requires at least $\Omega(n \log n)$ computation time. We also show that converting an Apollonius diagram of a set of n weighted points in 2D from a Laguerre diagram and vice-versa requires at least $\Omega(n \log n)$ computation time as well. Furthermore, we present a very simple randomized incremental construction algorithm that takes expected $O(n \log n)$ computation time to build an Apollonius diagram of non-overlapping circles in 2D.

1 Introduction

Voronoi diagrams in 2D are one of the most classical objects of computational geometry. Given a set of n points $S = \{p_1, p_2, \ldots, p_n\}$ in the plane, consider n regions $\mathcal{R}_i$ such that $\mathcal{R}_i$ contains all the points *closer* to p_i than any other point $p_j \in S$ with $p_i \neq p_j$. The word *closer* here is crucial. If the distance used is the Euclidean distance in the plane (i.e., $\|p - p'\|$ for points p and p'), each region is a convex (possibly unbounded) polygon and their union is the Voronoi diagram of S; see Figure 1a. The dual of the Voronoi diagram of S is the *Delaunay triangulation* of S.

Now, consider a set of n circles (or weighted points) $\Sigma = \{c_1, c_2, \ldots, c_n\}$ in the plane, with $c_i = (p_i, r_i)$ for $i = 1, \ldots, n$ and the regions $\mathcal{R}_i$, for $i = 1, \ldots, n$, defined as above but with the concept of Euclidean distance replaced by the Power distance between a point and a circle (i.e., $\|p - p'\|^2 - r^2$ for point p' and circle $c = (p, r)$). Then, again, the regions are convex, but their union is the *Laguerre diagram* (or Power diagram) of Σ. Here, input circles may not have a region associated with;[1] we call such circles as *hidden circles*. The dual of the Laguerre diagram of Σ is the *regular triangulation* of Σ; see Figure 1b. The dual, as its name suggest, must be a triangulation. Furthermore, it might not include all centers of the input circles as vertices, since the final construction might have hidden circles.

Finally, consider the same set of circles Σ above, but the distance now is the signed Euclidean distance between a point and a circle (i.e., $\|p - p'\| - r$ for point p' and circle $c = (p, r)$). Then, the regions are no longer convex and their union is the *Apollonius diagram* of Σ; see Figure 1c. Actually, these regions are bounded by segments of lines or hyperbola. As in the Laguerre diagram, some input circles may not have a region associated with, which we call analogously a *hidden point*. The dual of the Apollonius diagram of Σ is the *Apollonius graph* of Σ. Conversely to both structures mentioned above, the Apollonius graph may not be a triangulation.

A lower bound of $\Omega(n \log n)$ in the algebraic computation tree model of computation [1] is known for building any of these diagrams for an input set of size n; this can be proved by a reduction to the problem of sorting n numbers. Also, optimal algorithms achieving a computational complexity of $O(n \log n)$ for building any of these three diagrams (or their duals) are well known [4]. Randomized incremental constructions obtaining an expected cost of $O(n \log n)$ for Voronoi and Power diagrams are also computational geometry classics [5, 3, 2]. However, the situation is not the same for the construction of an Apollonius diagram: to the best of our knowledge there is no randomized incremental construction for Apollonius diagrams with provable $O(n \log n)$ expected computational cost yet.

There is an optimal algorithm for Apollonius diagram construction: it is a sweep-line algorithm that has been proposed in the early days of computational geometry [4]. However, this algorithm is complicated, requires high degree predicates, and is not used in practice. The implementation in CGAL [6, 7, 8] is based on randomized incremental construction and more precisely on a generalization of the Delaunay hierarchy [2]. The Delaunay hierarchy allows a logarithmic time point location in a Delaunay triangulation (or a Voronoi diagram). Unfortunately, while generalizing the algorithm to Laguerre or Apollonius diagrams is straightforward, the proof of complexity requires some special properties of Delaunay triangulation and does not generalize so easily. Karavelas and Yvinec [7] propose to go from a site to the next one using a dichotomic search in the neighbors of the

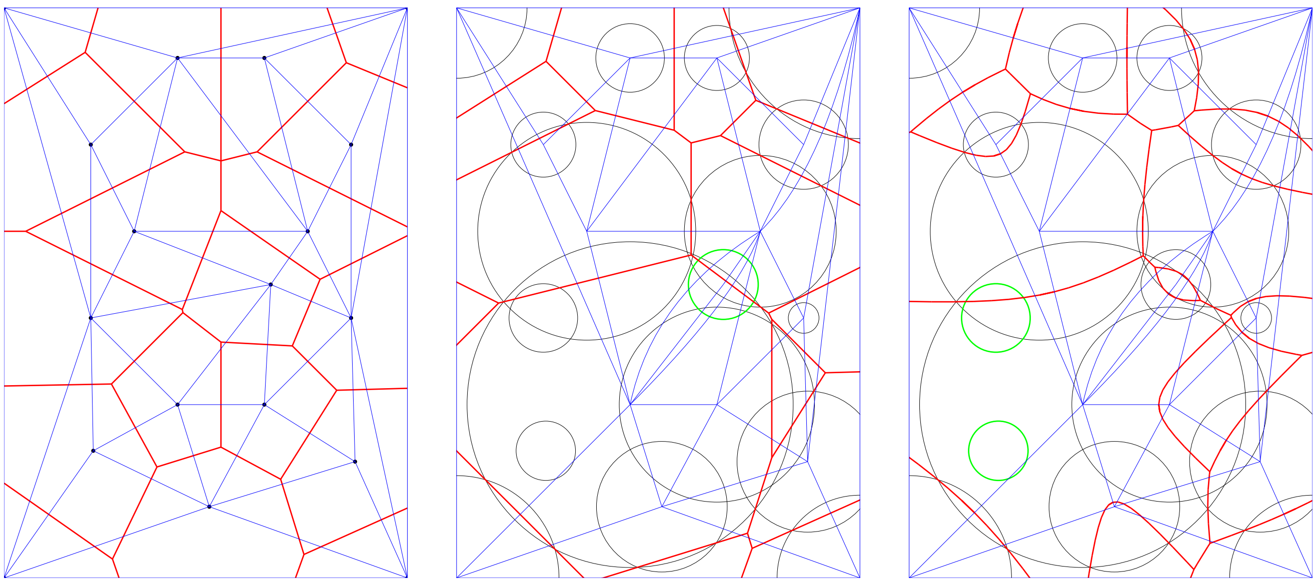
*Technical University Eindhoven, The Netherlands www.win.tue.nl/∼kbuchin

†Universidade Federal de Pernambuco, Brasil cin.ufpe.br/∼pmmc/

‡Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France. members.loria.fr/Olivier.Devillers

§Department of Applied Mathematics, University of Crete & Institute of Applied and Computational Mathematics, Greece tem.uoc.gr/∼mkaravel/

[1] The term *distance* albeit classically used in that case is actually not the most appropriate, since it can be negative when points are inside the circle.



(a) Voronoi (b) Laguerre (c) Apollonius

Figure 1: **Diagrams and their dual.** *In red, we have respectively: the Voronoi diagram, the Laguerre diagram and the Apollonius diagram. In blue we have their dual. Green circles are hidden. For the three figures above, input points in (a) are the same as the centers of input circles in (b) and (c). Moreover, the input circles are the same for (b) and (c). As we can see above these diagrams may have close combinatorial structure, and hence we may ask whether it is cheap to convert from one to another.*

site. This approach yields a provable expected time complexity of $O(n \log^2 n)$ to construct the Apollonius diagram.

With this in mind, we propose the following randomized incremental contruction: computing the Apollonius diagram of a set of circles with zero radius (i.e., the Voronoi diagram of the centers), then increasing the radii of all circles in a random order maintaining the diagram. In this paper, we prove that such approach is in expected $O(n \log n)$ computational cost for n non-overlapping circles.

The idea above is appealing because we already have very efficient algorithms and software for the computation of Voronoi diagram. Also, as shown by the similarity between the different diagrams in Figure 1, one might hope that converting one diagram to another could be done quickly (i.e., linear in the size of the input set). Then, the following question arises:

> *" Is the knowledge of any of the Voronoi, Laguerre (power), or Apollonius diagrams of any help to compute any of the two others? "*

In this paper, we answer negatively any of the six instances of that question.

2 Lower bounds

In this section, we present the hardness results on any conversion between the diagrams mentioned above. More precisely, we show that such a conversion has a $\Omega(n \log n)$ computational cost in the algebraic computation tree model of computation [1]. When more convenient, we consider the dual of these structures, respectively: Delaunay triangulation, regular triangulation and Apollonius graph (converting primals to their duals and vice-versa is of course in $\Theta(n)$).

2.1 Knowing the Voronoi diagram does not help to compute the Laguerre diagram

Theorem 1 *Computing the regular triangulation of a set of n weighted points knowing the Delaunay triangulation of the unweighted points has $\Omega(n \log n)$ complexity lower bound.*

Proof. Consider a set of points $p_i = \{(x_i, y_i)\}_{0 \le i < n}$ with $y_i > 0$. We first remark that the Laguerre diagram allows to sort numbers, actually assigning weights $w_i = y_i^2$ to points p_i (i.e. radius y_i) ensure that points with consecutive x coordinates are neighbors in the regular triangulation. Actually for a point $(x, 0)$ its weighted distance to p_i is $(x - x_i)^2$ and a moving point on the x-axis gets as closest site all the sites in the order of their x-coordinates (see Figure 2). If the Laguerre di-

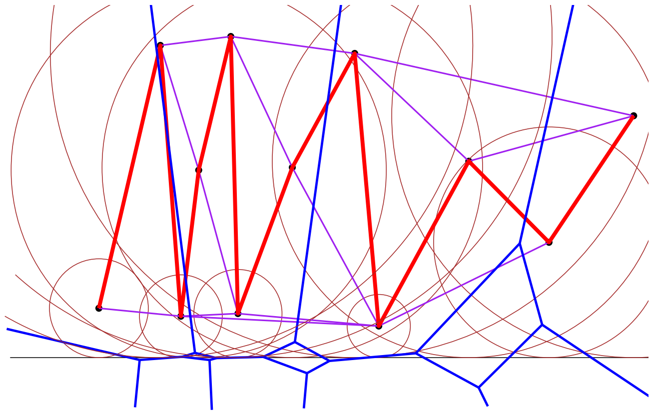


Figure 2: **Laguerre diagram allows to sort the centers by x-coordinate.** *The red x monotone curve is a subset of the dual: the regular triangulation.*

agram is known, the x-order can be retrieved looking for the x-successor of a site in its neighbors in the regular triangulation. Since the sum of the degrees of all sites is less than $6n$ this operation can be done in linear time.

It is known that having the Delaunay triangulation already built does not help to sort points by x coordinates (in the sense that it is still $\Omega(n \log n)$). More precisely, Seidel [9] proposed a construction that, given n numbers, presents a set of points having these numbers as abscissa and their Delaunay triangulation in linear time. If Delaunay would help to sort vertices by x coordinates it would contradict the sorting lower bound.

Combining the two constructions, it is possible to sort n numbers by first building Seidel's Delaunay triangulation in linear time, then building the regular triangulation from the Delaunay triangulation with the above weights and finally extracting the x-order of the sites. If the Delaunay to regular transform used $o(n \log n)$ computation time, we would get a contradiction on the lower bound result for sorting. □

2.2 Knowing the Voronoi diagram does not help to compute the Apollonius diagram

Theorem 2 *Computing the Apollonius diagram of a set of n circles knowing the Delaunay triangulation of the centers has $\Omega(n \log n)$ complexity lower bound.*

Proof. The construction is almost the same as the one in the proof of Theorem 1. We use the same circles and the same moving point on the line $x = 0$, the distance from $(x, 0)$ to the weighted points p_i is $\sqrt{(x - x_i)^2 + y_i(y_i - 1)}$. The distance to the closest site is always positive, being zero for all points in turn according to their x-order. The rest of the proof is identical. □

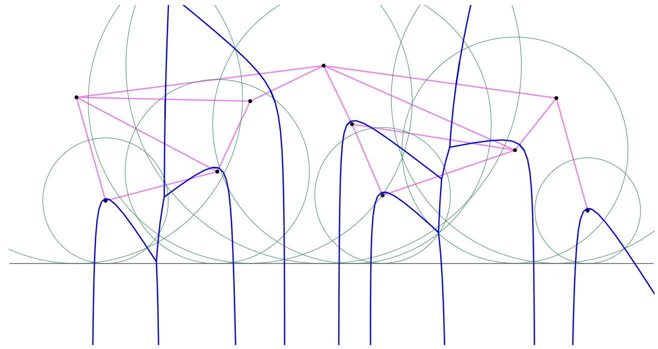


Figure 3: **Apollonius diagram allows to sort the centers by x-coordinate.** *The lower part of the dual (in pink) enumerates all points in x-order.*

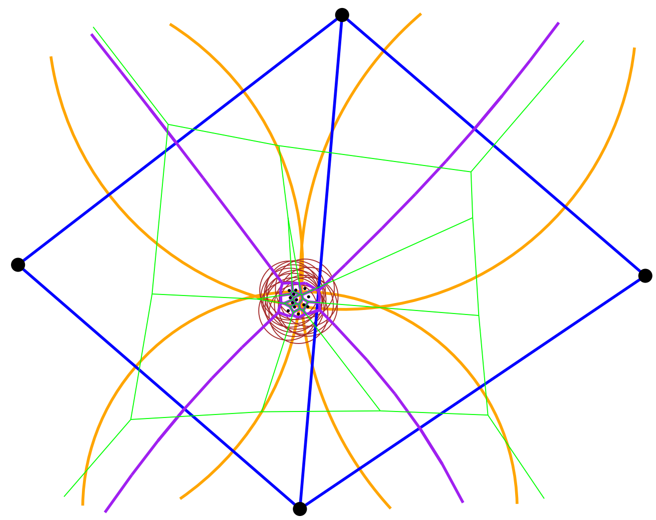


Figure 4: **Regular triangulation does not help to compute Apollonius diagram nor the Voronoi diagram.** *Voronoi diagram in green, Apollonius diagram in purple, regular triangulation in blue.*

2.3 Other lower bounds

Computing Apollonius or Laguerre diagram is not so much helpful when many points are hidden. For instance, the conversion of either regular or Apollonius to Delaunay triangulation is hopeless. This is because, for any set of centers, by adjusting the radii, essentially all points but one can be hidden, thus the Delaunay would need to be built from scratch. The hardness results for converting Apollonius to Regular and vice-versa are presented in the sequel.

Theorem 3 *Computing the Apollonius diagram of a set of n circles or the Voronoi diagram of their centers knowing the regular triangulation has $\Omega(n \log n)$ complexity lower bound.*

Proof. As described in Figure 4, consider four big circles that pass close to the origin, and a set of small cir-

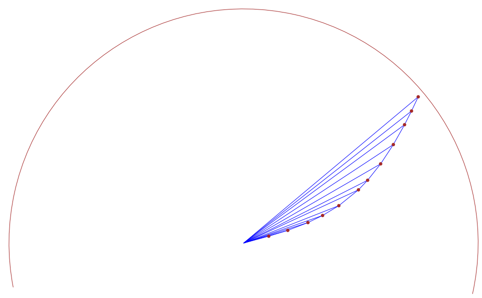


Figure 5: **Apollonius diagram does not help to compute the regular triangulation nor the Delaunay triangulation.** *Delaunay triangulation in blue. Note the big red circle hidden every point inside it.*

cles of same radii centered close to the origin. Radii can be tuned so that all centers of small circles are hidden in the regular triangulation while their Voronoi diagram, up to the convex hull, is present in the Apollonius or Voronoi diagram. □

Theorem 4 *Computing the regular triangulation of a set of n circles or the Delaunay triangulation of their centers knowing the Apollonius diagram has $\Omega(n \log n)$ complexity lower bound.*

Proof. Consider a set of circles with centers $p_i = (x_i, x_i^2)$ for $0 \leq i < n$ with $x_i > 0$ and zero radius plus one circle center at the origin with radius big enough to contain all p_i's. Then the Delaunay and the regular triangulation are equal and allow to sort the points by x-coordinate while the Apollonius diagram does not give any hint since the big circle is the only one with non empty region (see Figure 5). □

3 Building the Apollonius diagram of non-overlapping circles quickly

3.1 Static randomized incremental construction

Let $AG(\Sigma)$ be the Apollonius graph of $\Sigma = \{c_1, c_2, \ldots, c_i\}$ and $c_i = (p_i, w_i)$, $AG_i(\Sigma)$ be the Apollonius graph of $\{c_1, c_2, \ldots, c_i, (p_{i+1}, 0), \ldots, (p_n, 0)\}$ and $DT(S)$ be the Delaunay triangulation of $S = \{p_1, p_2, \ldots, p_n\}$. The Apollonius graph can be computed as described in Algorithm 1.

First, the Delaunay triangulation of the centers of the circles is computed, which is equivalent to the Apollonius graph of circles with radius zero. Then the circles with their true radii are incrementally added in a random order updating the Apollonius graph. Notice that the insertion of a circle hides its center, thus the final result is just the Apollonius graph of the circles. We obtain the Apollonius diagram by extracting the dual from the primal.

```
Data: A set Σ of n circles (or weighted points)
      c_i = (p_i, w_i), i = 1...n.
Result: AG(Σ), which is the Apollonius graph
        of Σ.
Let S = {p_i, i = 1...n};
Build DT(S) the Delaunay Triangulation of S;
Let AG_0(Σ) be DT(S);
drop DT(S);
Shuffle indices of circles in Σ;
for i = 1...n do
    get AG_i(Σ) by inserting c_i into AG_{i-1}(Σ)
      using p_i as hint;
end
return AG_n(Σ);
```

Algorithm 1: Algorithm for building the Apollonius graph of non-overlapping circles.

Theorem 5 *Algorithm 1 constructs the Apollonius diagram of n disjoint circles in $O(n \log n)$ expected time.*

Proof. Let $d^o_{AG_i(\Sigma)}(c)$ the degree of c in $AG_i(\Sigma)$. Consider the diagram at step i, its total complexity is linear, thus the expected complexity of the cell of the last (the ith) circle $\mathcal{E}_i$ is bounded as follows:

$$\begin{aligned}\mathcal{E}_i &= \frac{1}{i} \sum_{c \in \{c_1, \ldots, c_i\}} d^o_{AG_i(\Sigma)}(c) \\ &\leq \frac{1}{i} \sum_{c \in \{c_1, \ldots, c_i\} \cup \{p_{i+1}, \ldots, p_n\}} d^o_{AG_i(\Sigma)}(c) \\ &\leq \frac{6n}{i}.\end{aligned}$$

When summing $\mathcal{E}_i$, for $i = 1, \ldots, n$, the total structural change is $O(n \log n)$. □

3.2 Lower bound on the number of structural changes overall

For usual randomized incremental construction in the context of Voronoi diagrams of points, the total complexity of the structural change has $O(n)$ size and the usual expected $O(n \log n)$ computation time arises because of point locations, which is actually the algorithm's bottleneck. In the second part of our algorithm (when converting from Voronoi to Apollonius), the point location is avoided, but the size of structural change becomes $\Omega(n \log n)$ since the total size of the diagram is linear from the beginning of that second part. Figure 6 shows an example where the structural change has actually $\Theta(n \log n)$ size.

3.3 Issue with overlapping circles

When a circle is inserted, if its center is not hidden in the diagram just before the insertion, this center is a

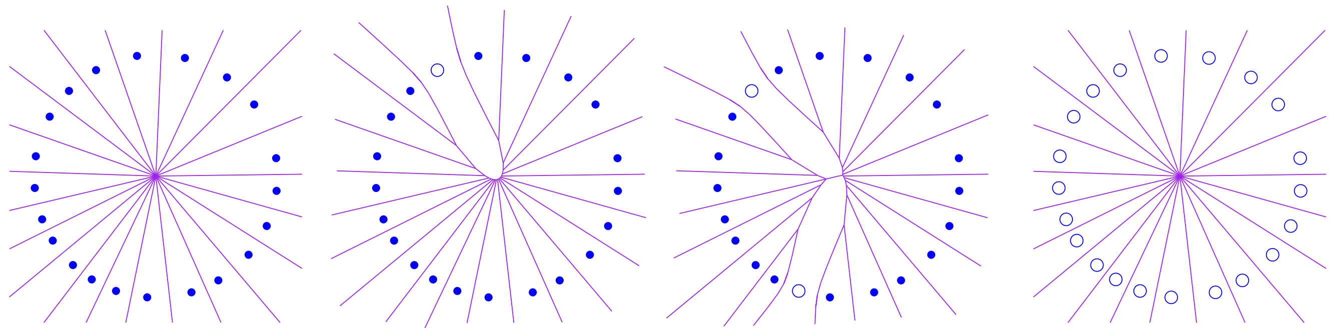
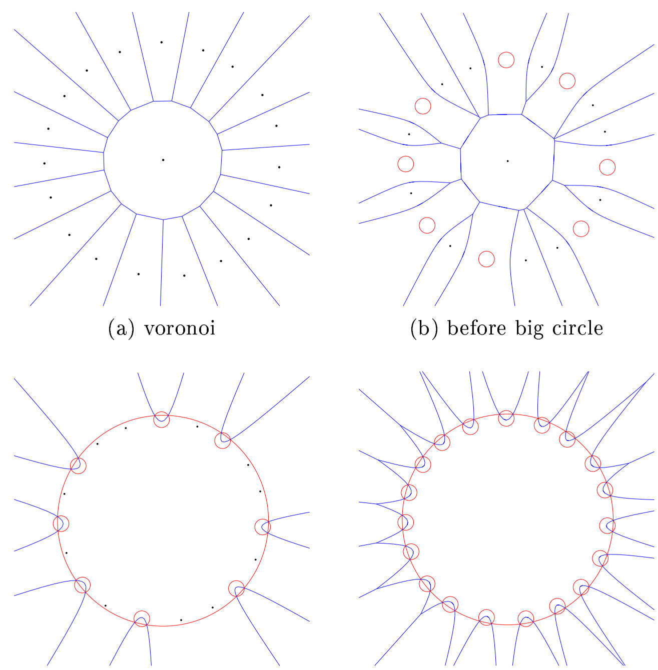


Figure 6: **The number of structural changes is not linear.** *Going from Voronoi to Apollonius incrementally may requires* $\Omega(n \log n)$ *expected structural changes: the first insertion requires* $\Omega(n)$*, the second* $\Omega(\frac{n}{2})$ *expected, and so on.*

perfect hint to locate the circle. The point location part is completely avoided since a conflict with the new circle is known, which is the case for disjoint circles. However, if the center is already hidden, which can happen when allowing overlapping circles as input, point location is still needed.

Consider one big circle, and n small disjoint circles intersecting the big circle whose centers are inside the big circle. A typical increase of radius for a small circle after the insertion of the big circle is problematic. To avoid point location, we need a good hint to insert the circle, but its center is hidden and no longer present in the Apollonius graph and its nearest neighbor is the big circle and has a high degree in the Apollonius graph. Thus defining an hint allowing fast point location seems difficult in such a case.



(a) voronoi (b) before big circle (c) after big circle (d) apollonius

Figure 7: **Handling hidden circles may be not trivial.** *Black points are circles yet to be grown whereas red circles are already inserted. In blue, we have the corresponding Apollonius diagram of the set of points and circles. Note that after inserting the big circle, each remaining circle to be grown has as its nearest neighbor the big circle, which has a high degree.*

References

[1] M. Ben-Or Lower Bounds for Algebraic Computation Trees *Proc. 15th ACM Symposium on Theory of Computing*, pages 80–86, 1983.

[2] O. Devillers The Delaunay hierarchy *Internat. J. Found. Comput. Sci.*, 13:163–180, 2002.

[3] H. Edelsbrunner and N. R. Shah Incremental topological flipping works for regular triangulations *Algorithmica*, 15(3):223–241, 1996.

[4] S. J. Fortune A sweepline algorithm for Voronoi diagrams *Algorithmica*, 2:153–174, 1987.

[5] L. J. Guibas, D. E. Knuth, and M. Sharir Randomized incremental construction of Delaunay and Voronoi diagrams *Algorithmica*, 7(1):381–413, 1992.

[6] M. I. Karavelas and M. Yvinec 2D Apollonius graphs (Delaunay graphs of disks) In *CGAL User and Reference Manual*. CGAL Editorial Board, 4.2 edition, 2013.

[7] M. I. Karavelas and M. Yvinec Dynamic additively weighted Voronoi diagrams in 2D In *Proceedings of the 10th European Symposium on Algorithms*, volume 2461 of *LNCS*, pages 568–598, 2002.

[8] M. I. Karavelas and M. Yvinec The Voronoi diagram of planar convex objects In *Proceedings of the 11th European Symposium on Algorithms*, volume 2832 of *LNCS*, pages 337–348, 2003.

[9] R. Seidel A method for proving lower bounds for certain geometric problems In *Computational Geometry*, G. T. Toussaint, North-Holland, pages 319–334, 1985.