Chapter 10

Voronoi Diagrams

10.1 Post Office Problem

Suppose there are n post offices $p_1, \ldots p_n$ in a city. Someone who is located at a position q within the city would like to know which post office is closest to him. Modeling the city as a planar region, we think of $p_1, \ldots p_n$ and q as points in the plane. Denote the set of post offices by $P = \{p_1, \ldots p_n\}$.

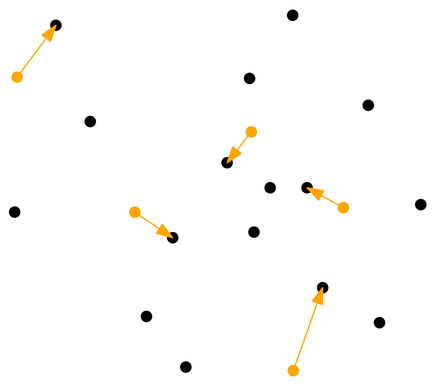


Figure 10.1: *Closest post offices for various query points.*

While the locations of post offices are known and do not change so frequently, we do not know in advance for which—possibly many—query locations the closest post office is to be found. Therefore, our long term goal is to come up with a data structure on top of P that allows to answer any possible query efficiently. The basic idea is to apply a so-called *locus approach*: we partition the query space into regions on which is the answer is the same. In our case, this amounts to partition the plane into regions such that for all points within a region the same point from P is closest (among all points from P).

As a warmup, consider the problem for two post offices $p_i, p_j \in P$. For which query locations is the answer p_i rather than p_j? This region is bounded by the bisector of p_i and p_j, that is, the set of points which have the same distance to both points.

Proposition 10.1 *For any two distinct points in $\mathbb{R}^d$ the bisector is a hyperplane, that is, in $\mathbb{R}^2$ it is a line.*

Proof. Let $p = (p_1, \ldots, p_d)$ and $q = (q_1, \ldots, q_d)$ be two points in $\mathbb{R}^d$. The bisector of p and q consists of those points $x = (x_1, \ldots, x_d)$ for which

$$\|p - x\| = \|q - x\| \iff \|p - x\|^2 = \|q - x\|^2 \iff \|p\|^2 - \|q\|^2 = 2(p - q)^\top x \, .$$

As p and q are distinct, this is the equation of a hyperplane. $\square$

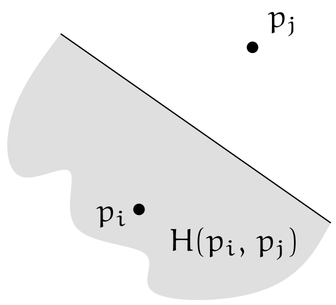


Figure 10.2: *The bisector of two points.*

Denote by $H(p_i, p_j)$ the closed halfspace bounded by the bisector of p_i and p_j that contains p_i. In $\mathbb{R}^2$, $H(p_i, p_j)$ is a halfplane; see Figure 10.2.

Exercise 10.2

a) What is the bisector of a line ℓ and a point $p \in \mathbb{R}^2 \setminus \ell$, that is, the set of all points $x \in \mathbb{R}^2$ with $\|p - x\| = \|p - \ell\|$ $(= \min_{q \in \ell} \|p - q\|)$?

b) For two points $p \neq q \in \mathbb{R}^2$, what is the region that contains all points whose distance to p is exactly twice their distance to q?

10.2 Voronoi Diagram

In the following we work with a set $P = \{p_1, \ldots, p_n\}$ of points in $\mathbb{R}^2$.

Definition 10.3 (Voronoi cell) *For $p_i \in P$ denote the* **Voronoi cell** $V_P(i)$ *of p_i by*

$$V_P(i) := \left\{ q \in \mathbb{R}^2 \;\middle|\; \|q - p_i\| \leqslant \|q - p\| \text{ for all } p \in P \right\} .$$

Proposition 10.4

$$V_P(i) = \bigcap_{j \neq i} H(p_i, p_j) \, .$$

Proof. For $j \neq i$ we have $\|q - p_i\| \leqslant \|q - p_j\| \iff q \in H(p_i, p_j)$. $\square$

Corollary 10.5 $V_P(i)$ *is non-empty and convex.*

Proof. According to Proposition 10.4, $V_P(i)$ is the intersection of a finite number of halfplanes and hence convex. $V_P(i) \neq \emptyset$ because $p_i \in V_P(i)$. $\square$

Observe that every point of the plane lies in some Voronoi cell but no point lies in the interior of two Voronoi cells. Therefore these cells form a subdivision of the plane. See Figure 10.3 for an example.

Definition 10.6 (Voronoi Diagram) *The Voronoi Diagram* $VD(P)$ *of a set* $P = \{p_1, \ldots, p_n\}$ *of points in* $\mathbb{R}^2$ *is the subdivision of the plane induced by the Voronoi cells* $V_P(i)$, *for* $i = 1, \ldots, n$.
Denote by $VV(P)$ *the set of vertices, by* $VE(P)$ *the set of edges, and by* $VR(P)$ *the set of regions (faces) of* $VD(P)$.

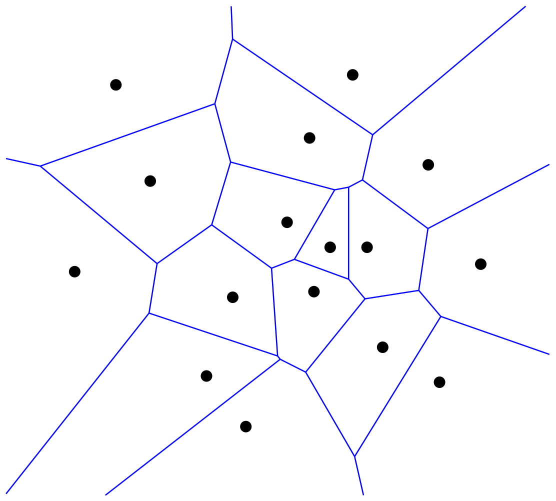


Figure 10.3: *Example: The Voronoi diagram of a point set.*

Lemma 10.7 *For every vertex* $v \in VV(P)$ *the following statements hold.*

a) v *is the common intersection of at least three edges from* $VE(P)$;

b) v *is incident to at least three regions from* $VR(P)$;

c) v *is the center of a circle* $C(v)$ *through at least three points from* P *such that*

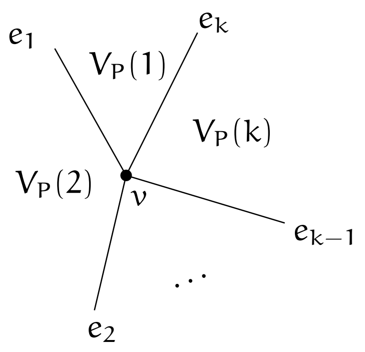
d) $C(v)^\circ \cap P = \emptyset$.

Proof. Consider a vertex $v \in VV(P)$. As all Voronoi cells are convex, $k \geqslant 3$ of them must be incident to v. This proves Part a) and b).

Without loss of generality let these cells be $V_P(i)$, for $1 \leqslant i \leqslant k$. Denote by e_i, $1 \leqslant i \leqslant k$, the edge incident to v that bounds $V_P(i)$ and $V_P((i \bmod k)+1)$.

For any $i = 1, \dots, k$ we have $v \in e_i \Rightarrow \|v - p_i\| = \|v - p_{(i \bmod k)+1}\|$. In other words, $p_1, p_2, \dots, p_k$ are cocircular, which proves Part c).

Part d): Suppose there exists a point $p_\ell \in C(v)^\circ$. Then the vertex v is closer to p_ℓ than it is to any of $p_1, \dots, p_k$, in contradiction to the fact that v is contained in all of $V_P(1), \dots, V_P(k)$. □



Corollary 10.8 *If P is in general position (no four points from P are cocircular), then for every vertex $v \in VV(P)$ the following statements hold.*

a) v is the common intersection of exactly three edges from $VE(P)$;

b) v is incident to exactly three regions from $VR(P)$;

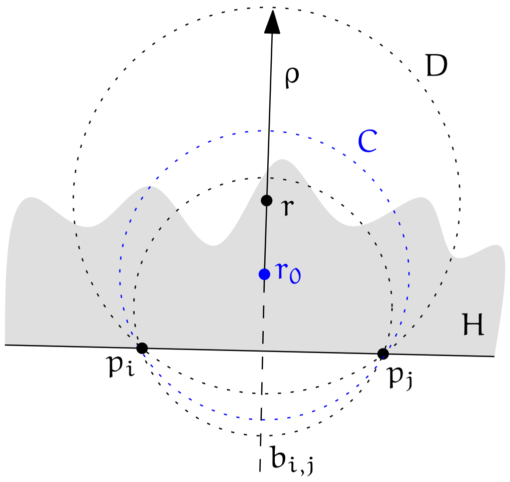
c) v is the center of a circle $C(v)$ through exactly three points from P such that

d) $C(v)^\circ \cap P = \emptyset$. □

Lemma 10.9 *There is an unbounded Voronoi edge bounding $V_P(i)$ and $V_P(j)$ $\iff$ $\overline{p_i p_j} \cap P = \{p_i, p_j\}$ and $\overline{p_i p_j} \subseteq \partial\text{conv}(P)$, where the latter denotes the boundary of the convex hull of P.*

Proof.

Denote by $b_{i,j}$ the bisector of p_i and p_j, and let $\mathcal{C}$ denote the family of circles centered at some point on $b_{i,j}$ and passing through p_i (and p_j). There is an unbounded Voronoi edge bounding $V_P(i)$ and $V_P(j)$ $\iff$ there is a ray $\rho \subset b_{i,j}$ such that $\|r - p_k\| > \|r - p_i\|$ $(= \|r - p_j\|)$, for every $r \in \rho$ and every $p_k \in P$ with $k \notin \{i, j\}$. Equivalently, there is a ray $\rho \subset b_{i,j}$ such that for every point $r \in \rho$ the circle $C \in \mathcal{C}$ centered at r does not contain any point from P in its interior.



The latter statement implies that the open halfplane H, whose bounding line passes through p_i and p_j and such that H contains the infinite part of ρ, contains no point from P in its interior. Therefore, $\overline{p_i p_j}$ appears on $\partial\text{conv}(P)$ and $\overline{p_i p_j}$ does not contain any $p_k \in P$, for $k \neq i, j$.

Conversely, suppose that $\overline{p_i p_j}$ appears on $\partial\text{conv}(P)$ and $\overline{p_i p_j} \cap P = \{p_i, p_j\}$. Then some halfplane H whose bounding line passes through p_i and p_j contains no point from

P in its interior. In particular, the existence of H together with $\overline{p_i p_j} \cap P = \{p_i, p_j\}$ implies that there is some circle $C \in \mathcal{C}$ such that $C \cap P = \{p_i, p_j\}$. Denote by r_0 the center of C and let ρ denote the ray starting from r_0 along $b_{i,j}$ such that the infinite part of ρ is contained in H. Consider any circle $D \in \mathcal{C}$ centered at a point $r \in \rho$ and observe that $D \setminus H \subseteq C \setminus H$. As neither H nor C contain any point from P in their respective interior, neither does D. This holds for every D, and we have seen above that this statement is equivalent to the existence of an unbounded Voronoi edge bounding $V_P(i)$ and $V_P(j)$. □

10.3 Duality

A *straight-line dual* of a plane graph G is a graph G' defined as follows: Choose a point in the interior of each face of G and connect any two such points by a straight edge, if the corresponding faces share an edge of G. Observe that this notion depends on the embedding; that is why the straight-line dual is defined for a plane graph rather than for an abstract graph. In general, G' may have edge crossings, which may also depend on the choice of representative points within the faces. However, for Voronoi diagrams is a particularly natural choice of representative points such that G' is plane: the points from P.

Theorem 10.10 (Delaunay [2]) *The straight-line dual of* $VD(P)$ *for a set* $P \subset \mathbb{R}^2$ *of* $n \geqslant 3$ *points in general position (no three points from* P *are collinear and no four points from* P *are cocircular) is a triangulation: the unique Delaunay triangulation of* P.

Proof. By Lemma 10.9, the convex hull edges appear in the straight-line dual T of VD(P) and they correspond exactly to the unbounded edges of VD(P). All remaining edges of VD(P) are bounded, that is, both endpoints are Voronoi vertices. Consider some $v \in VV(P)$. According to Corollary 10.8(b), v is incident to exactly three Voronoi regions, which, therefore, form a triangle $\triangle(v)$ in T. By Corollary 10.8(d), the circumcircle of $\triangle(v)$ does not contain any point from P in its interior. Hence $\triangle(v)$ appears in the (unique by Corollary 6.18) Delaunay triangulation of P.

Conversely, for any triangle $p_i p_j p_k$ in the Delaunay triangulation of P, by the empty circle property the circumcenter c of $p_i p_j p_k$ has p_i, p_j, and p_k as its closest points from P. Therefore, $c \in VV(P)$ and—as above—the triangle $p_i p_j p_k$ appears in T. □

Corollary 10.11 $|VE(P)| \leqslant 3n - 6$ *and* $|VV(P)| \leqslant 2n - 5$.

Proof. Every edge in VE(P) corresponds to an edge in the dual Delaunay triangulation. The latter is a plane graph on n vertices and thus has at most $3n-6$ edges and at most $2n-4$ faces by Lemma 5.1. Only the bounded faces correspond to a vertex in VD(P). □

Corollary 10.12 *For a set* $P \subset \mathbb{R}^2$ *of* n *points, the Voronoi diagram of* P *can be constructed in expected* $O(n \log n)$ *time and space.*

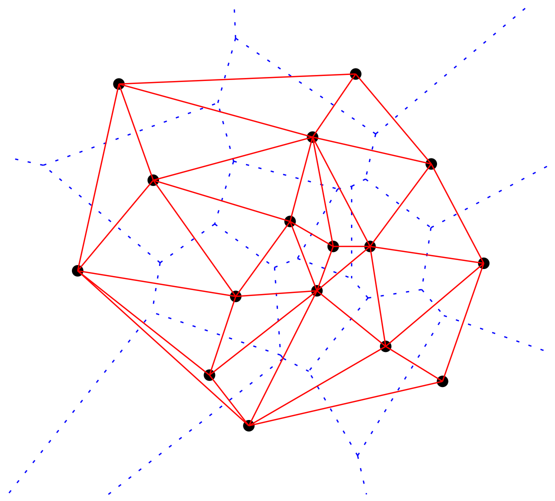


Figure 10.4: *The Voronoi diagram of a point set and its dual Delaunay triangulation.*

Proof. We have seen that a Delaunay triangulation T for P can be obtained using randomized incremental construction in the given time and space bounds. As T is a plane graph, its number of vertices, edges, and faces all are linear in n. Therefore, the straight-line dual of T—which by Theorem 10.10 is the desired Voronoi diagram—can be computed in $O(n)$ additional time and space. □

Exercise 10.13 *Consider the Delaunay triangulation* T *for a set* $P \subset \mathbb{R}^2$ *of* $n \geqslant 3$ *points in general position. Prove or disprove:*

a) Every edge of T *intersects its dual Voronoi edge.*

b) Every vertex of $VD(P)$ *is contained in its dual Delaunay triangle.*

10.4 Lifting Map

Recall the lifting map that we used in Section 6.3 to prove that the Lawson Flip Algorithm terminates. Denote by $\mathcal{U} : z = x^2 + y^2$ the unit paraboloid in $\mathbb{R}^3$. The lifting map $\ell : \mathbb{R}^2 \to \mathcal{U}$ with $\ell : p = (p_x, p_y) \mapsto (p_x, p_y, {p_x}^2 + {p_y}^2)$ is the projection of the x/y-plane onto $\mathcal{U}$ in direction of the z-axis.

For $p \in \mathbb{R}^2$ let H_p denote the plane of tangency to $\mathcal{U}$ in $\ell(p)$. Denote by $h_p : \mathbb{R}^2 \to H_p$ the projection of the x/y-plane onto H_p in direction of the z-axis (see Figure 10.5).

Lemma 10.14 $\|\ell(q) - h_p(q)\| = \|p - q\|^2$, *for any points* $p, q \in \mathbb{R}^2$.

Exercise 10.15 *Prove Lemma 10.14.* Hint: *First determine the equation of the tangent plane* H_p *to* $\mathcal{U}$ *in* $\ell(p)$.

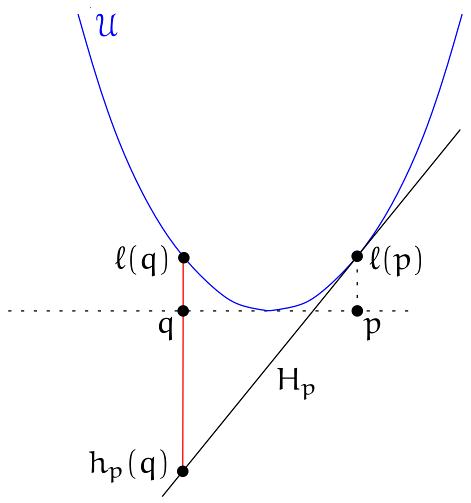


Figure 10.5: *Lifting map interpretation of the Voronoi diagram in a two-dimensional projection.*

Theorem 10.16 *For $p = (p_x, p_y) \in \mathbb{R}^2$ denote by H_p the plane of tangency to the unit paraboloid $\mathcal{U} = \{(x, y, z) : z = x^2 + y^2\} \subset \mathbb{R}^3$ in $\ell(p) = (p_x, p_y, {p_x}^2 + {p_y}^2)$. Let $\mathcal{H}(P) := \bigcap_{p \in P} H_p^+$ the intersection of all halfspaces above the planes H_p, for $p \in P$. Then the vertical projection of $\partial\mathcal{H}(P)$ onto the x/y-plane forms the Voronoi Diagram of P (the faces of $\partial\mathcal{H}(P)$ correspond to Voronoi regions, the edges to Voronoi edges, and the vertices to Voronoi vertices).*

Proof. For any point $q \in \mathbb{R}^2$, the vertical line through q intersects every plane H_p, $p \in P$. By Lemma 10.14 the topmost plane intersected belongs to the point from P that is closest to q. □

10.5 Point location in a Voronoi Diagram

One last bit is still missing in order to solve the post office problem optimally.

Theorem 10.17 *Given a triangulation T for a set $P \subset \mathbb{R}^2$ of n points, one can build in $O(n)$ time an $O(n)$ size data structure that allows for any query point $q \in \text{conv}(P)$ to find in $O(\log n)$ time a triangle from T containing q.*

The data structure we will employ is known as *Kirkpatrick's hierarchy*. But before discussing it in detail, let us put things together in terms of the post office problem.

Corollary 10.18 (Nearest Neighbor Search) *Given a set $P \subset \mathbb{R}^2$ of n points, one can build in expected $O(n \log n)$ time an $O(n)$ size data structure that allows for any query point $q \in \text{conv}(P)$ to find in $O(\log n)$ time a nearest neighbor of q among the points from P.*

Proof. First construct the Voronoi Diagram V of P in expected $O(n \log n)$ time. It has exactly n convex faces. Every unbounded face can be cut by the convex hull boundary into a bounded and an unbounded part. As we are concerned with query points within $\text{conv}(P)$ only, we can restrict our attention to the bounded parts.[1] Any convex polygon can easily be triangulated in time linear in its number of edges (= number of vertices). As V has at most $3n - 6$ edges and every edge appears in exactly two faces, V can be triangulated in $O(n)$ time overall. Label each of the resulting triangles with the point from p, whose Voronoi region contains it, and apply the data structure from Theorem 10.17. □

10.5.1 Kirkpatrick's Hierarchy

We will now the develop the data structure for point location in a triangulation, as described in Theorem 10.17. The main idea is to construct a hierarchy $T_0, \ldots, T_h$ of triangulations, such that

- $T_0 = T$,
- the vertices of T_i are a subset of the vertices of T_{i-1}, for $i = 1, \ldots, h$, and
- T_h comprises a single triangle only.

Search. For a query point x the triangle from T containing x can be found as follows.

```
Search($x \in \mathbb{R}^2$)
1. For $i = h, h-1, \ldots, 0$: Find a triangle $t_i$ from $T_i$ that contains $x$.
2. return $t_0$.
```

This search is efficient under the following conditions.

(C1) Every triangle from T_i intersects only few ($\leqslant c$) triangles from T_{i-1}. (These will then be connected via the data structure.)

(C2) h is small ($\leqslant d \log n$).

Proposition 10.19 *The search procedure described above needs $\leqslant 3cd \log n = O(\log n)$ orientation tests.*

Proof. For every T_i, $0 \leqslant i < h$, at most c triangles are tested as to whether or not they contain x. □

[1] We even know how to decide in $O(\log n)$ time whether or not a given point lies within $\text{conv}(P)$, see Exercise 3.20.

Thinning. Removing a vertex v and all its incident edges from a triangulation creates a non-triangulated hole that forms a star-shaped polygon since all points are visible from v (the star-point).

Lemma 10.20 *A star-shaped polygon, given as a sequence of $n \geqslant 3$ vertices and a star-point, can be triangulated in $O(n)$ time.*

Exercise 10.21 *Prove Lemma 10.20.*

As a side remark, the *kernel* of a simple polygon, that is, the set of all star-points, can be constructed in linear time as well. We will get back to this question later in the course...

Our working plan is to obtain T_i from T_{i-1} by removing several *independent* (pairwise non-adjacent) vertices and re-triangulating. These vertices should

a) have small degree (otherwise the degree within the hierarchy gets too large, that is, we need to test too many triangles on the next level) and

b) be many (otherwise the height h of the hierarchy gets too large).

The following lemma asserts the existence of a sufficiently large set of independent small-degree vertices in every triangulation.

Lemma 10.22 *In every triangulation of n points in $\mathbb{R}^2$ there exists an independent set of at least $\lceil n/18 \rceil$ vertices of maximum degree 8. Moreover, such a set can be found in $O(n)$ time.*

Proof. Let $T = (V, E)$ denote the graph of the triangulation, which we consider as an abstract graph in the following. We may suppose that T is maximally planar, that is, all faces are triangles. (Otherwise triangulate the exterior face arbitrarily. An independent set in the resulting graph is also independent in T.) For $n = 3$ the statement is true. Let $n \geqslant 4$.

By the Euler formula we have $|E| = 3n - 6$, that is,

$$\sum_{v \in V} \deg_T(v) = 2|E| = 6n - 12 < 6n.$$

Let $W \subseteq V$ denote the set of vertices of degree at most 8. Claim: $|W| > n/2$. Suppose $|W| \leqslant n/2$. As every vertex has degree at least three, we have

$$\begin{aligned}\sum_{v \in V} \deg_T(v) &= \sum_{v \in W} \deg_T(v) + \sum_{v \in V \setminus W} \deg_T(v) \geqslant 3|W| + 9|V \setminus W| \\ &= 3|W| + 9(n - |W|) = 9n - 6|W| \geqslant 9n - 3n = 6n,\end{aligned}$$

in contradiction to the above.

Construct an independent set U in T as follows (greedily): As long as $W \neq \emptyset$, add an arbitrary vertex $v \in W$ to U and remove v and all its neighbors from W.

Obviously U is independent and all vertices in U have degree at most 8. At each selection step at most 9 vertices are removed from W. Therefore $|U| \geqslant \lceil (n/2)/9 \rceil = \lceil n/18 \rceil$. □

Proof. (of Theorem 10.17)
Construct the hierarchy $T_0, \ldots T_h$ with $T_0 = T$ as follows. Obtain T_i from T_{i-1} by removing an independent set U as in Lemma 10.22 and re-triangulating the resulting holes. By Lemma 10.20 and Lemma 10.22 every step is linear in the number $|T_i|$ of vertices in T_i. The total cost for building the data structure is thus

$$\sum_{i=0}^{h} \alpha|T_i| \leqslant \sum_{i=0}^{h} \alpha n (17/18)^i < \alpha n \sum_{i=0}^{\infty} (17/18)^i = 18\alpha n \in O(n),$$

for some constant α. Similarly the space consumption is linear.

The number of levels amounts to $h = \log_{18/17} n < 12.2 \log n$. Thus by Proposition 10.19 the search needs at most $3 \cdot 8 \cdot \log_{18/17} n < 292 \log n$ orientation tests. □

Improvements. As the name suggests, the hierarchical approach discussed above is due to David Kirkpatrick [5]. The constant 292 that appears in the search time is somewhat large. There has been a whole line of research trying to improve it using different techniques.

- Sarnak and Tarjan [6]: $4 \log n$.
- Edelsbrunner, Guibas, and Stolfi [3]: $3 \log n$.
- Goodrich, Orletsky, and Ramaier [4]: $2 \log n$.
- Adamy and Seidel [1]: $1 \log n + 2\sqrt{\log n} + O(\sqrt[4]{\log n})$.

Exercise 10.23 *Let* $\{p_1, p_2, \ldots, p_n\}$ *be a set of points in the plane, which we call* obstacles*. Imagine there is a disk of radius* r *centered at the origin which can be moved around the obstacles but is not allowed to intersect them (touching the boundary is ok). Is it possible to move the disk out of these obstacles? See the example depicted in Figure 10.6 below.*

More formally, the question is whether there is a (continuous) path $\gamma : [0,1] \longrightarrow \mathbb{R}^2$ *with* $\gamma(0) = (0,0)$ *and* $\|\gamma(1)\| \geqslant \max\{\|p_1\|, \ldots, \|p_n\|\}$*, such that at any time* $t \in [0,1]$ *and* $\|\gamma(t) - p_i\| \geqslant r$*, for any* $1 \leqslant i \leqslant n$*. Describe an algorithm to decide this question and to construct such a path—if one exists—given arbitrary points* $\{p_1, p_2, \ldots, p_n\}$ *and a radius* $r > 0$*. Argue why your algorithm is correct and analyze its running time.*

Exercise 10.24 *This exercise is about an application from* Computational Biology*:*
You are given a set of disks $P = \{a_1, .., a_n\}$ *in* $\mathbb{R}^2$*, all with the same radius* $r_a > 0$*.*

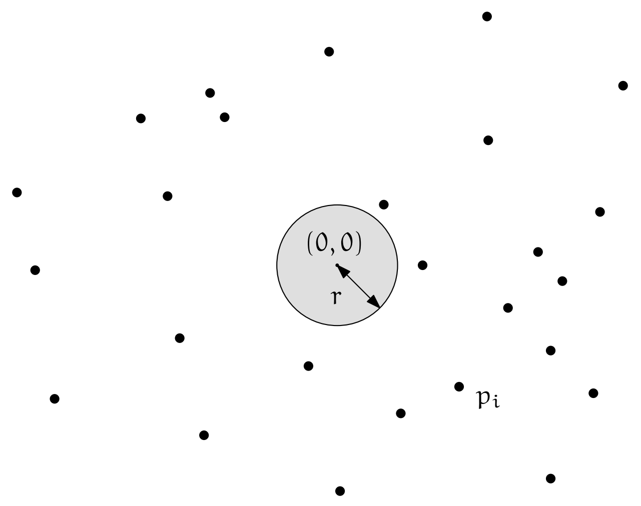


Figure 10.6: *Motion planning: Illustration for Exercise 10.23.*

Each of these disks represents an atom of a protein. A water molecule is represented by a disc with radius $r_w > r_a$. A water molecule cannot intersect the interior of any protein atom, but it can be tangent to one. We say that an atom $a_i \in P$ is accessible *if there exists a placement of a water molecule such that it is tangent to a_i and does not intersect the interior of any other atom in P. Given P, find an $O(n \log n)$ time algorithm which determines all atoms of P that are inaccessible.*

Exercise 10.25 *Let $P \subset \mathbb{R}^2$ be a set of n points. Describe a data structure to find in $O(\log n)$ time a point in P that is furthest from a given query point q among all points in P.*

Exercise 10.26 *Show that the bounds given in Theorem 10.17 are optimal in the algebraic computation tree model.*

Questions

45. *What is the Voronoi diagram of a set of points in $\mathbb{R}^2$?* Give a precise definition and explain/prove the basic properties: convexity of cells, why is it a subdivision of the plane?, Lemma 10.7, Lemma 10.9.

46. *What is the correspondence between the Voronoi diagram and the Delaunay triangulation for a set of points in $\mathbb{R}^2$?* Prove duality (Theorem 10.10) and explain where general position is needed.

47. *How to construct the Voronoi diagram of a set of points in $\mathbb{R}^2$?* Describe an $O(n \log n)$ time algorithm, for instance, via Delaunay triangulation.

48. *How can the Voronoi diagram be interpreted in context of the lifting map?* Describe the transformation and prove its properties to obtain a formulation of the Voronoi diagram as an intersection of halfspaces one dimension higher.

49. *What is the Post-Office Problem and how can it be solved optimally?* Describe the problem and a solution using linear space, $O(n \log n)$ preprocessing, and $O(\log n)$ query time.

50. *How does Kirkpatrick's hierarchical data structure for planar point location work exactly?* Describe how to build it and how the search works, and prove the runtime bounds. In particular, you should be able to state and prove Lemma 10.20, Lemma 10.22, and Theorem 10.17.

References

[1] Udo Adamy and Raimund Seidel, On the exaxt worst case query complexity of planar point location. *J. Algorithms*, **37**, (2000), 189–217, URL `http://dx.doi.org/10.1006/jagm.2000.1101`.

[2] Boris Delaunay, Sur la sphère vide. A la memoire de Georges Voronoi. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskih i Estestvennyh Nauk*, **7**, (1934), 793–800.

[3] Herbert Edelsbrunner, Leonidas J. Guibas, and Jorge Stolfi, Optimal point location in a monotone subdivision. *SIAM J. Comput.*, **15**, 2, (1986), 317–340, URL `http://dx.doi.org/10.1137/0215023`.

[4] Michael T. Goodrich, Mark W. Orletsky, and Kumar Ramaiyer, Methods for achieving fast query times in point location data structures. In *Proc. 8th ACM-SIAM Sympos. Discrete Algorithms*, pp. 757–766, 1997, URL `http://doi.acm.org/10.1145/314161.314438`.

[5] David G. Kirkpatrick, Optimal search in planar subdivisions. *SIAM J. Comput.*, **12**, 1, (1983), 28–35, URL `http://dx.doi.org/10.1137/0212002`.

[6] Neil Sarnak and Robert E. Tarjan, Planar point location using persistent search trees. *Commun. ACM*, **29**, 7, (1986), 669–679, URL `http://dx.doi.org/10.1145/6138.6151`.