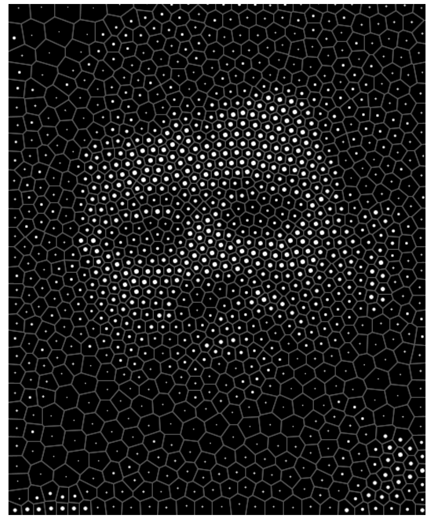
VORONOI DIAGRAM

23-03-2022



- Around 500 years old topic.
- Many applications.
- Also known as *Voronoi tessellation*, *Dirichlet tessellation*.

The figure aside shows an artistic rendition on the image of my face using Voronoi diagram.
Courtesy:
http://www.evilmadscientist.com/2012/stipplegen-weighted-voronoi-stippling-and-tsp-paths-in-processing

Terminology

1. *Point* = any real point on xy-plane.
2. *Site* = special point on xy-plane. Voronoi diagram is defined w.r.t. a set of sites, P.
3. $P = \{p_1, p_2, \ldots, p_n\}$ = set of n sites (input).
4. $\mathrm{VD}(P)$ = Voronoi diagram w.r.t. P (output).
5. $b(p_i, p_j)$ = perpendicular bisector between p_i and p_j.
6. $h(p_i, p_j)$ = half plane induced by $b(p_i, p_j)$ and containing the site p_i.
7. $h(p_j, p_i)$ = half plane induced by $b(p_i, p_j)$ and containing the site p_j.
8. $vc(p_i)$ = Voronoi cell / region of p_i.
9. $d(p,q) = \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2}$ = Euclidean distance between $p = (x_p, x_q)$ and $q = (x_q, x_q)$, where either of p and q is a point or a site, as needed.
10. $d(q, \lambda)$ = distance of a point q from a line λ.
11. n_v = #vertices in $\mathrm{VD}(P)$.
12. n_e = #edges in $\mathrm{VD}(P)$.
13. n_f = #faces in $\mathrm{VD}(P)$.
14. ….

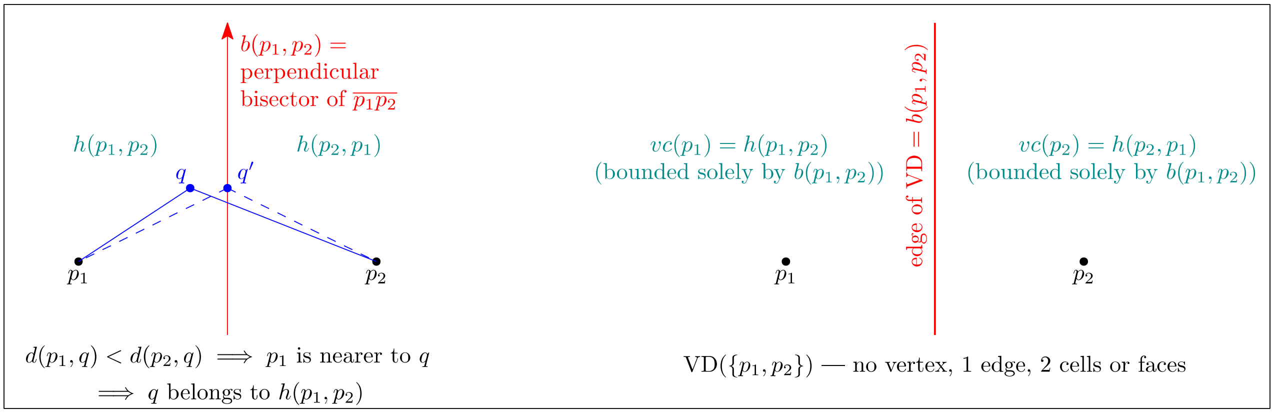


Figure: VD of two sites.

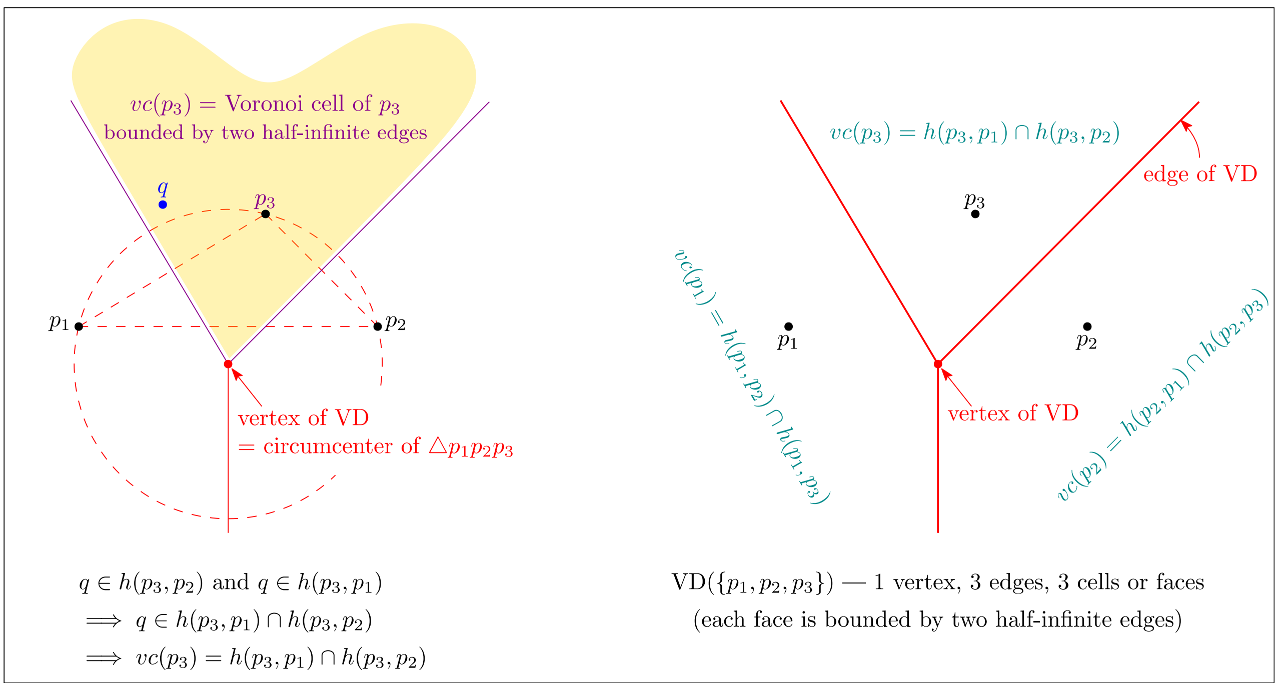


Figure: VD of three sites.

Designing a good algorithm needs good definitions, good characterizations, and good theorems. That is what we shall see today.

Examples of VD

Figure: VD of four and five sites.

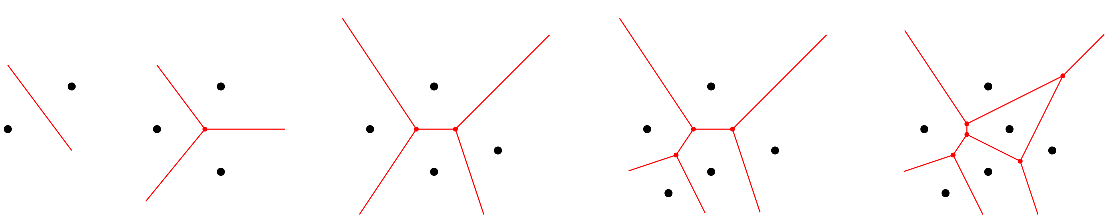


Figure: More examples.

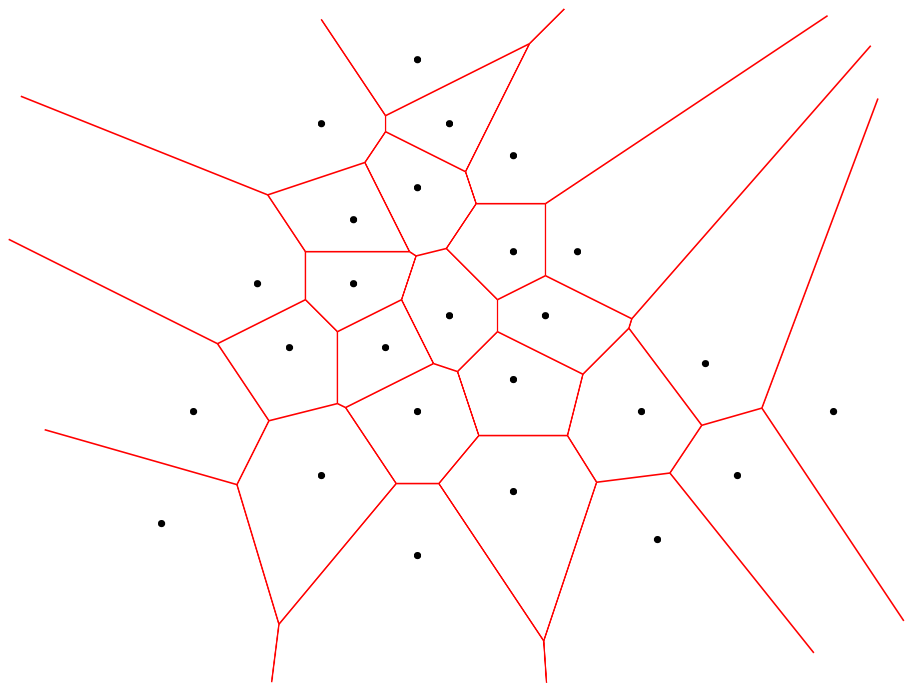


Figure: An example of VD for many sites.

Definition [Voronoi Diagram (VD)] It subdivides the xy-plane into n cells or regions, called *Voronoi cells*, such that for each Voronoi cell $vc(p_i)$, for any point q in $vc(p_i)$, p_i is a/the nearest site.

Definition [Voronoi cell] For every site p_i, the corresponding Voronoi cell $vc(p_i)$ is the set of all points for which p_i is a/the nearest site. For n sites, we have n Voronoi cells, which are pairwise disjoint by their interiors, and their union is VD(P).

Theorem [$vc(p_i)$ characterization] $vc(p_i) = \bigcap_{j=1,2,\ldots,n, j\neq i} h(p_i, p_j)$.

There are $n-1$ half-planes for each p_i. So, computing the Voronoi cell for p_i using the above equation will take at least $O(n)$ time. So, for all cells, we need $n \cdot O(n) = O(n^2)$ time. We have to dive deeper to get a better algorithm.

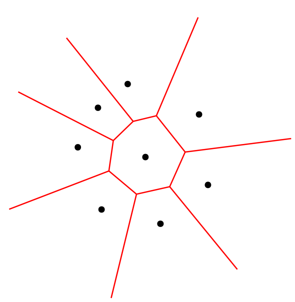


Figure: An example of VD in which a single Voronoi cell has $O(n)$ vertices. That implies a VD may have $O(n^2)$ vertices and $O(n^2)$ edges. But they will be $O(n)$, as stated in the next theorem.

Theorem [upper bounds on VD size] $\mathrm{VD}(P)$ has $O(n)$ vertices and $O(n)$ edges.

Proof – Euler's formula for planar graph: #vertices $-$ #edges $+$ #faces $= 2$.

The extended VD is a planar graph, call it G. Its #vertices $= n_v + 1$, #edges $= n_e$, #faces $= n_f = n$.

So, $(n_v + 1) - n_e + n_f = 2$, or, $n_v + 1 - n_e + n = 2$, or,

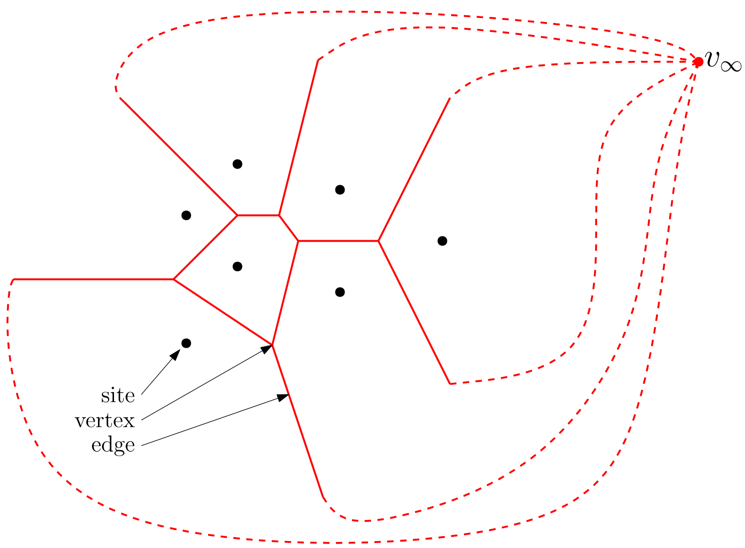
$$n_v - n_e + n = 1. \qquad (\clubsuit)$$

Let $s =$ sum of degrees over all vertices of G. Its every edge contributes degree 2 to s. So, $s = 2n_e$. Now, every vertex of G is incident on at least 3 edges of G. So, $s \geq 3(n_v + 1)$. So, we have

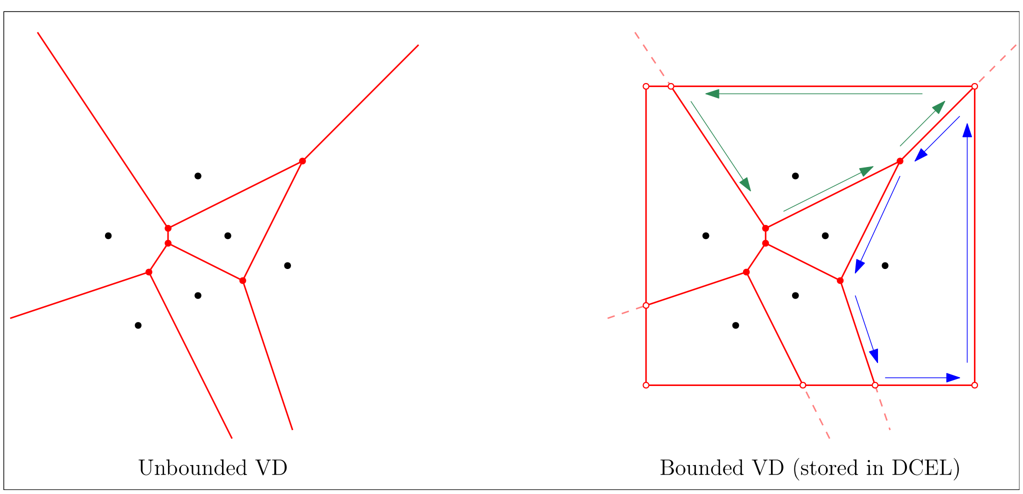
$$2n_e \geq 3n_v + 3. \qquad (\spadesuit)$$

From $\clubsuit$ and $\spadesuit$, $2n_e = 2n_v + 2n - 2 \geq 3n_v + 3 \implies n_v \leq 2n - 5$.

Similarly, we get $n_e \leq 3n - 6$. And hence the proof.



Representation of VD: An axis-parallel rectangle is used to bound the VD so that all edges and faces become bounded. The bounded VD is stored in a Doubly Connected Edge List (DCEL). Figure below.



Definition [Largest empty circle] $C(q)$ centered at any point q contains no site in its interior and contains at least one site on its boundary.

Theorem [Vertex and Edge characterization] q is a ***vertex*** of $\mathrm{VD}(P)$ if and only if $C(q)$ contains three or more sites on its boundary. q is a point on an ***edge*** of $\mathrm{VD}(P)$ if and only if $C(q)$ contains exactly two sites on its boundary. So, q lies in the ***interior*** of some cell/face of $\mathrm{VD}(P)$ if and only if $C(q)$ contains exactly one site on its boundary.

Question 1: Let λ be a horizontal line. Which Voronoi cells will not be controlled by any site below λ?

Answer: Those for which *any* point q in the Voronoi cell is farther off from λ compared to the distance of q from its nearest site.

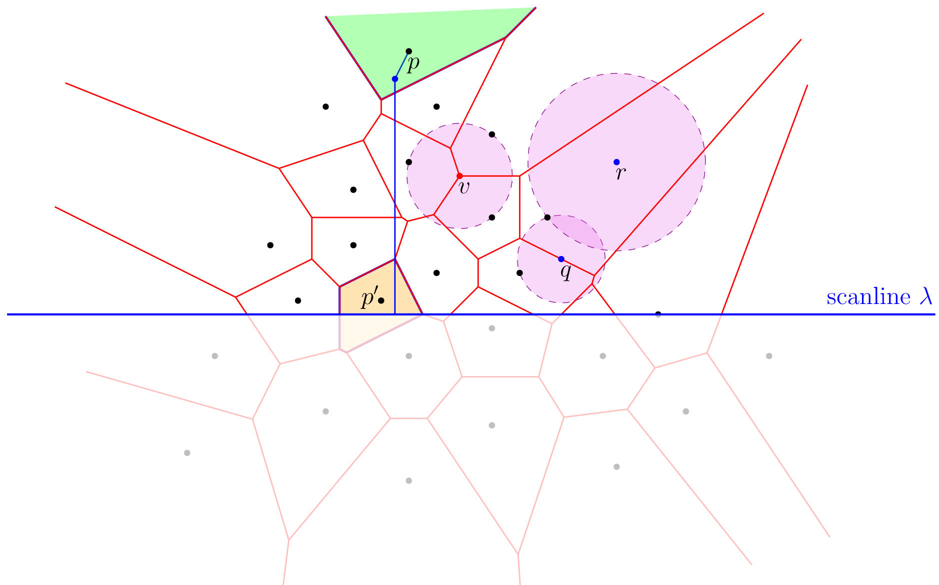
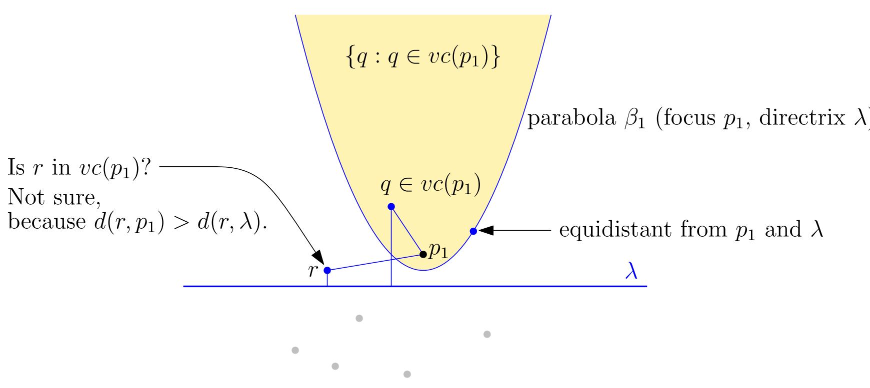
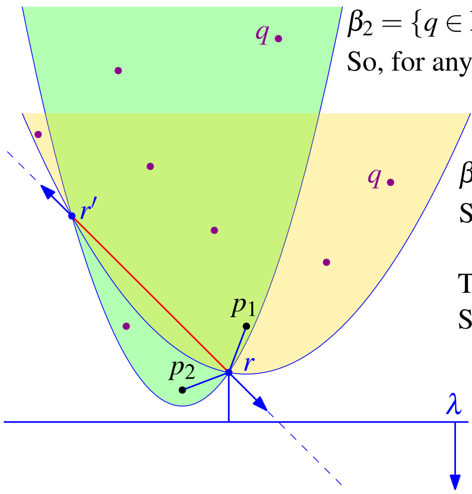


Figure: Vertex and edge characterization using *largest empty circles*, and characterization of VD *above* λ. For any point in $vc(p)$, its distance from p is less than that from λ, and so the shape of $vc(p)$ is determined only by the sites above λ. But for $vc(p')$, it is not so, i.e., its shape may vary with the positions of sites below λ.

Question 2: Let λ be a horizontal line. Which portion of VD is fixed for ever and will not be changed by any site below λ?

Answer: Let $p_1, p_2, \ldots, p_i$ be the sites above λ. Consider the parabolic region β_j whose focus is p_j and directrix is λ. The union of the parabolic regions $\beta_1, \beta_2, \ldots, \beta_i$ contains the portion of VD whose structure is determined only by $p_1, p_2, \ldots, p_i$.





$\beta_2 = \{q \in \mathbb{R}^2 : d(q, p_2) \leq d(q, \lambda)\}$
So, for any $q \in \beta_2$, a site below λ cannot be the nearest site of q.

$\beta_1 = \{q \in \mathbb{R}^2 : d(q, p_1) \leq d(q, \lambda)\}$
So, for any $q \in \beta_1$, a site below λ cannot be the nearest site of q.

The *break point* r (and r' as well) is equidistant from p_1 and p_2.
So, it will trace $b(p_1, p_2)$ as λ moves down.

Algorithm for Voronoi Diagram

See the other PDF (demo on 10 sites).

Time and space complexities

Number of nodes in B and Q are bounded by $O(n)$. So any query / insertion / deletion will be $O(\log n)$ time. As there will be $O(n)$ event points in total [think why], total time for operations on B and Q will be $O(n \log n)$.

Vertex or edge creation/update in DCEL takes $O(1)$ time. So, for all vertices, edges, and faces, DCEL creation will take $O(n)$ time [think why].

Hence, time complexity is $O(n \log n)$.

Q always stores the sites that are yet to be handled, and it also stores the circle events resulting from only the triplets of consecutive arcs in β, which has at most $O(n)$ parabolic arcs in sequence. So, B stores $O(n)$ sites as foci of parabolic arcs in β, and the breakpoints for each pair of consecutive arcs in β. Hence, space complexity is $O(n)$.