



WIKIPEDIA  
The Free Encyclopedia

Main page  
Contents  
Featured content  
Current events  
Random article  
Donate to Wikipedia  
Wikipedia store

Interaction

Help  
About Wikipedia  
Community portal  
Recent changes  
Contact page

Tools

What links here  
Related changes  
Upload file  
Special pages  
Permanent link  
Page information  
Wikidata item  
Cite this page

Print/export

Create a book  
Download as PDF  
Printable version

Languages

العربية  
Български  
Català  
Čeština  
Deutsch  
Español  
فارسی  
Français  
한국어  
Հայերեն  
Italiano  
Magyar  
Nederlands  
日本語  
Polski  
Português  
Русский  
Српски / srpski  
ไทย  
Українська  
اردو  
Tiếng Việt  
中文

Edit links

[Create account](#) [Log in](#)

Article [Talk](#)

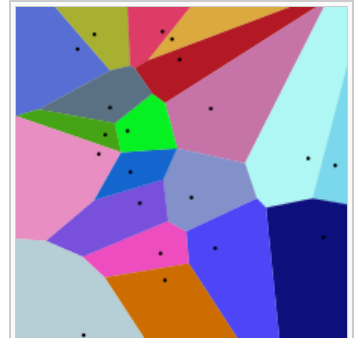
[Read](#) [Edit](#) [View history](#)

# Voronoi diagram

From Wikipedia, the free encyclopedia

In mathematics, a **Voronoi diagram** is a [partitioning](#) of a [plane](#) into regions based on distance to points in a specific subset of the plane. That set of points (called seeds, sites, or generators) is specified beforehand, and for each seed there is a corresponding region consisting of all points [closer](#) to that seed than to any other. These regions are called Voronoi cells. The Voronoi diagram of a set of points is [dual](#) to its [Delaunay triangulation](#). Put simply, it's a diagram created by taking pairs of points that are close together and drawing a line that is equidistant between them and perpendicular to the line connecting them. That is, all points on the lines in the diagram are equidistant to the nearest two (or more) source points.

It is named after [Georgy Voronoy](#), and is also called a **Voronoi tessellation**, a **Voronoi decomposition**, a **Voronoi partition**, or a **Dirichlet tessellation** (after [Peter Gustav Lejeune Dirichlet](#)). Voronoi diagrams have practical and theoretical applications to a large number of fields, mainly in [science](#) and [technology](#) but also including [visual art](#).<sup>[1][2]</sup>



20 points and their Voronoi cells (larger version [below](#)).

## Contents

 [hide]

- The simplest case
- Formal definition
- Illustration
- Properties
- History and research
- Examples
- Higher-order Voronoi diagrams
  - Farthest-point Voronoi diagram
- Generalizations and variations
- Applications
- See also
- Notes
- References
- External links

## The simplest case

 [edit]

In the simplest and most familiar case (shown in the first picture), we are given a finite set of points 



{

p

1


,
…
,

p

n


}


{\displaystyle \{p\_{1},\,\ldots ,p\_{n}\}}

 in the [Euclidean plane](#). In this case each site *p*<sub>*k*</sub> is simply a point, and its corresponding Voronoi cell (also called Voronoi region or Dirichlet cell) *R*<sub>*k*</sub> consists of every point whose distance to *p*<sub>*k*</sub> is less than or equal to its distance to any other *p*<sub>*k*</sub>. Each such cell is obtained from the intersection of [half-spaces](#), and hence it is a [convex polygon](#). The segments of the Voronoi diagram are all the points in the plane that are equidistant to the two nearest sites. The Voronoi vertices ([nodes](#)) are the points equidistant to three (or more) sites.

## Formal definition

 [edit]

Let *X* be a [metric space](#) with distance function *d*. Let *K* be a set of indices and let 



(

P

k


)

k
∈

K


{\displaystyle (P\_{k})\_{k\in K}}

 be a [tuple](#) (ordered collection) of nonempty [subsets](#) (the sites) in the space *X*. The Voronoi cell, or Voronoi region, *R*<sub>*k*</sub>, associated with the site *P*<sub>*k*</sub> is the set of all points in *X* whose distance to *P*<sub>*k*</sub> is not greater than their distance to the other sites *P*<sub>*j*</sub>, where *j* is any index different from *k*. In other words, if *d*(*x*, *A*) = inf{*d*(*x*, *a*)|*a* ∈ *A*} denotes the distance between the point *x* and the subset *A*, then

$$R_k = \{x \in X \mid d(x, P_k) \leq d(x, P_j) \text{ for all } j \neq k\}$$

The Voronoi diagram is simply the [tuple](#) of cells 



(

R

k


)

k
∈

K


{\displaystyle (R\_{k})\_{k\in K}}

. In principle some of the sites can intersect and even coincide (an application is described below for sites representing shops), but usually they are assumed to be disjoint. In addition, infinitely many sites are allowed in the definition (this setting has applications in [geometry of numbers](#) and [crystallography](#)), but again, in many cases only finitely many sites are considered.

In the particular case where the space is a [finite-dimensional Euclidean space](#), each site is a point, there are finitely many points and all of them are different, then the Voronoi cells are [convex polytopes](#) and they can be represented in a

combinatorial way using their vertices, sides, 2-dimensional faces, etc. Sometimes the induced combinatorial structure is referred to as the Voronoi diagram. However, in general the Voronoi cells may not be convex or even connected.

In the usual Euclidean space, we can rewrite the formal definition in usual terms. Each Voronoi polygon  $R_k$  is associated with a generator point  $P_k$ . Let  $X$  be the set of all points in the Euclidean space. Let  $P_1$  be a point that generates its Voronoi region  $R_1$ ,  $P_2$  that generates  $R_2$ , and  $P_3$  that generates  $R_3$ , and so on. Then, as expressed by Tran *et al*.<sup>[3]</sup> "all locations in the Voronoi polygon are closer to the generator point of that polygon than any other generator point in the Voronoi diagram in Euclidean plane".

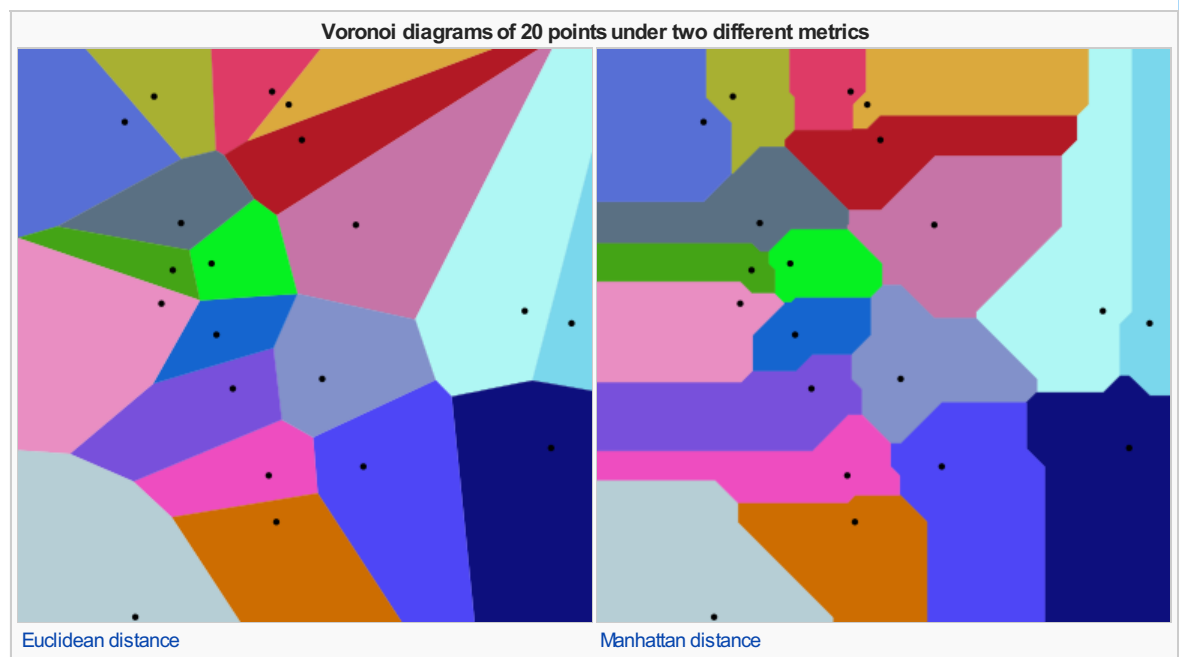
## Illustration <sup>[edit]</sup>

As a simple illustration, consider a group of shops in a city. Suppose we want to estimate the number of customers of a given shop. With all else being equal (price, products, quality of service, etc.), it is reasonable to assume that customers choose their preferred shop simply by distance considerations: they will go to the shop located nearest to them. In this case the Voronoi cell  $R_k$  of a given shop  $P_k$  can be used for giving a rough estimate on the number of potential customers going to this shop (which is modeled by a point in our city).

For most cities, the distance between points can be measured using the familiar [Euclidean distance](#):

$$\ell_2 = d[(a_1, a_2), (b_1, b_2)] = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2} \text{ or the } \text{Manhattan distance:}$$

$$d[(a_1, a_2), (b_1, b_2)] = |a_1 - b_1| + |a_2 - b_2|. \text{ The corresponding Voronoi diagrams look different for different distance metrics.}$$



## Properties <sup>[edit]</sup>

- The [dual graph](#) for a Voronoi diagram (in the case of a [Euclidean space](#) with point sites) corresponds to the [Delaunay triangulation](#) for the same set of points.
- The [closest pair of points](#) corresponds to two adjacent cells in the Voronoi diagram.
- Assume the setting is the [Euclidean plane](#) and a group of different points are given. Then two points are adjacent on the [convex hull](#) if and only if their Voronoi cells share an infinitely long side.
- If the space is a [normed space](#) and the distance to each site is attained (e.g., when a site is a [compact set](#) or a closed ball), then each Voronoi cell can be represented as a union of line segments emanating from the sites.<sup>[4]</sup> As shown there, this property does not necessarily hold when the distance is not attained.
- Under relatively general conditions (the space is a possibly infinite-dimensional [uniformly convex space](#), there can be infinitely many sites of a general form, etc.) Voronoi cells enjoy a certain stability property: a small change in the shapes of the sites, e.g., a change caused by some translation or distortion, yields a small change in the shape of the Voronoi cells. This is the geometric stability of Voronoi diagrams.<sup>[5]</sup> As shown there, this property does not hold in general, even if the space is two-dimensional (but non-uniformly convex, and, in particular, non-Euclidean) and the sites are points.

## History and research <sup>[edit]</sup>

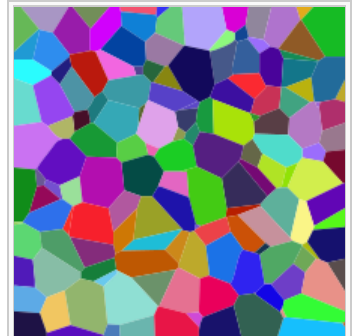
Informal use of Voronoi diagrams can be traced back to [Descartes](#) in 1644. [Peter Gustav Lejeune Dirichlet](#) used 2-dimensional and 3-dimensional Voronoi diagrams in his study of quadratic forms in 1850. British physician [John Snow](#) used a Voronoi diagram in 1854 to illustrate how the majority of people who died in the [Soho cholera epidemic](#) lived closer to the infected [Broad Street pump](#) than to any other water pump.

Voronoi diagrams are named after Ukrainian mathematician [Georgy Fedosievych Voronoi](#) (or *Voronoy*) who defined and studied the general  $n$ -dimensional case in 1908. Voronoi diagrams that are used in [geophysics](#) and [meteorology](#) to analyse spatially distributed data (such as rainfall measurements) are called Thiessen polygons after American meteorologist [Alfred H. Thiessen](#). In [condensed matter physics](#), such tessellations are also known as [Wigner–Seitz unit cells](#). Voronoi tessellations of the [reciprocal lattice](#) of [momenta](#) are called [Brillouin zones](#). For general lattices in [Lie groups](#), the cells are simply called [fundamental domains](#). In the case of general [metric spaces](#), the cells are often called metric [fundamental polygons](#). Other equivalent names for this concept (or particular important cases of it): [Voronoi polyhedra](#), [Voronoi polygons](#), domain(s) of influence, Voronoi decomposition, Voronoi tessellation(s), [Dirichlet tessellation](#)(s).

## Examples [\[edit\]](#)

Voronoi tessellations of regular [lattices](#) of points in two or three dimensions give rise to many familiar tessellations.

- A 2D lattice gives an irregular honeycomb tessellation, with equal hexagons with point symmetry; in the case of a regular triangular lattice it is regular; in the case of a rectangular lattice the hexagons reduce to rectangles in rows and columns; a [square](#) lattice gives the regular tessellation of squares; note that the rectangles and the squares can also be generated by other lattices (for example the lattice defined by the vectors  $(1,0)$  and  $(1/2,1/2)$  gives squares). See [here](#) [↗](#) for a dynamic visual example.
- A simple [cubic lattice](#) gives the [cubic honeycomb](#).
- A [hexagonal close-packed](#) lattice gives a tessellation of space with [trapezoidal dodecahedra](#).
- A [face-centred cubic](#) lattice gives a tessellation of space with [rhombic dodecahedra](#).
- A [body-centred cubic](#) lattice gives a tessellation of space with [truncated octahedra](#).
- Parallel planes with regular triangular lattices aligned with each other's centers give the [hexagonal prismatic honeycomb](#).
- Certain body centered tetragonal lattices give a tessellation of space with [rhombo-hexagonal dodecahedra](#).



This is a slice of the Voronoi diagram of a random set of points in a 3D box. In general a cross section of a 3D Voronoi tessellation is not a 2D Voronoi tessellation itself. (The cells are all [convex polyhedra](#).)

For the set of points  $(x, y)$  with  $x$  in a discrete set  $X$  and  $y$  in a discrete set  $Y$ , we get rectangular tiles with the points not necessarily at their centers.

## Higher-order Voronoi diagrams [\[edit\]](#)

Although a normal Voronoi cell is defined as the set of points closest to a single point in  $S$ , an  $n$ th-order Voronoi cell is defined as the set of points having a particular set of  $n$  points in  $S$  as its  $n$  nearest neighbors. Higher-order Voronoi diagrams also subdivide space.

Higher-order Voronoi diagrams can be generated recursively. To generate the  $n^{\text{th}}$ -order Voronoi diagram from set  $S$ , start with the  $(n - 1)^{\text{th}}$ -order diagram and replace each cell generated by  $X = \{x_1, x_2, \dots, x_{n-1}\}$  with a Voronoi diagram generated on the set  $S - X$ .

### Farthest-point Voronoi diagram [\[edit\]](#)

For a set of  $n$  points the  $(n - 1)^{\text{th}}$ -order Voronoi diagram is called a farthest-point Voronoi diagram.

For a given set of points  $S = \{p_1, p_2, \dots, p_n\}$  the farthest-point Voronoi diagram divides the plane into cells in which the same point of  $P$  is the farthest point. Note that a point of  $P$  has a cell in the farthest-point Voronoi diagram if and only if it is a vertex of the [convex hull](#) of  $P$ . Thus, let  $H = \{h_1, h_2, \dots, h_k\}$  be the convex hull of  $P$  we define the farthest-point Voronoi diagram as the subdivision of the plane into  $k$  cells, one for each point in  $H$ , with the property that a point  $q$  lies in the cell corresponding to a site  $h_i$  if and only if  $\text{dist}(q, h_i) > \text{dist}(q, p_j)$  for each  $p_j \in S$  with  $h_i \neq p_j$ . Where  $\text{dist}(p, q)$  is the [Euclidean distance](#) between two points  $p$  and  $q$ .<sup>[\[6\]](#)[\[7\]](#)</sup>

## Generalizations and variations [\[edit\]](#)

As implied by the definition, Voronoi cells can be defined for metrics other than Euclidean (such as the [Mahalanobis](#) or [Manhattan](#)) distances. However, in these cases the boundaries of the Voronoi cells may be more complicated than in the Euclidean case, since the equidistant locus for two points may fail to be subspace of codimension 1, even in the 2-dimensional case.

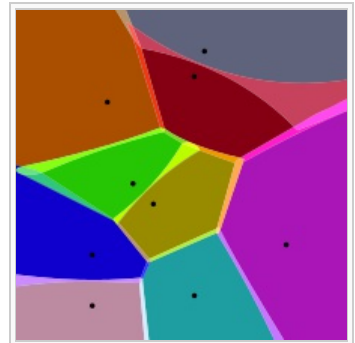
A [weighted Voronoi diagram](#) is the one in which the function of a pair of points to define a Voronoi cell is a distance function modified by multiplicative or additive weights assigned to generator points. In contrast to the case of Voronoi cells defined using a distance which is a [metric](#), in this case some of the Voronoi cells may be empty. A [power diagram](#) is a type of Voronoi diagram defined from a set of circles using the [power distance](#); it can also be thought of as a weighted

Voronoi diagram in which a weight defined from the radius of each circle is added to the [squared distance](#) from the circle's center.<sup>[8]</sup>

The Voronoi diagram of  $n$  points in  $d$ -dimensional space requires  $O\left(n^{\lceil \frac{1}{2}d \rceil}\right)$

storage space. Therefore, Voronoi diagrams are often not feasible for  $d > 2$ . An alternative is to use [approximate Voronoi diagrams](#), where the Voronoi cells have a fuzzy boundary, which can be approximated.<sup>[9]</sup> Another alternative is when any site is a fuzzy circle and as a result the cells become fuzzy too.<sup>[10]</sup>

Voronoi diagrams are also related to other geometric structures such as the [medial axis](#) (which has found applications in image segmentation, [optical character recognition](#), and other computational applications), [straight skeleton](#), and [zone diagrams](#). Besides points, such diagrams use lines and polygons as seeds. By augmenting the diagram with line segments that connect to nearest points on the seeds, a planar subdivision of the environment is obtained.<sup>[11]</sup> This structure can be used as a [navigation mesh](#) for path-finding through large spaces. The navigation mesh has been generalized to support 3D multi-layered environments, such as an airport or a multi-storey building.<sup>[12]</sup>



Approximate Voronoi diagram of a set of points. Notice the blended colors in the fuzzy boundary of the Voronoi cells.

## Applications [\[edit\]](#)

- In [astrophysics](#), Voronoi diagrams are used to generate adaptative smoothing zones on images, adding signal fluxes on each one. The main objective for these procedures is to maintain a relatively constant [signal-to-noise ratio](#) on all the image.
- In [epidemiology](#), Voronoi diagrams can be used to correlate sources of infections in epidemics. One of the early applications of Voronoi diagrams was implemented by [John Snow](#) to study the [1854 Broad Street cholera outbreak](#) in Soho, England. He showed the correlation between residential areas on the map of Central London whose residents had been using a specific water pump, and the areas with most deaths due to the outbreak.
- A [point location](#) data structure can be built on top of the Voronoi diagram in order to answer [nearest neighbor](#) queries, where one wants to find the object that is closest to a given query point. Nearest neighbor queries have numerous applications. For example, one might want to find the nearest hospital, or the most similar object in a [database](#). A large application is [vector quantization](#), commonly used in [data compression](#).
- In [geometry](#), Voronoi diagrams can be used to find the [largest empty circle](#) amid a set of points, and in an enclosing polygon; e.g. to build a new supermarket as far as possible from all the existing ones, lying in a certain city.
- Voronoi diagrams together with farthest-point Voronoi diagrams are used for efficient algorithms to compute the [roundness](#) of a set of points.<sup>[6]</sup>
- The Voronoi approach is also put to good use in the evaluation of circularity/[roundness](#) while assessing the dataset from a [coordinate-measuring machine](#).
- In aviation, Voronoi diagrams are superimposed on oceanic plotting charts to identify the nearest airfield for in-flight diversion, as an aircraft progresses through its flight plan.
- In networking, Voronoi diagrams can be used in derivations of the capacity of a [wireless network](#).
- In [hydrology](#), Voronoi diagrams are used to calculate the rainfall of an area, based on a series of point measurements. In this usage, they are generally referred to as Thiessen polygons.
- In [ecology](#), Voronoi diagrams are used to study the growth patterns of forests and forest canopies, and may also be helpful in developing predictive models for forest fires.
- In [architecture](#), Voronoi patterns were the basis for the winning entry for redevelopment of [The Arts Centre Gold Coast](#).<sup>[13]</sup>
- In [computational chemistry](#), Voronoi cells defined by the positions of the nuclei in a molecule are used to compute [atomic charges](#). This is done using the [Voronoi deformation density](#) method.
- In polymer physics, Voronoi diagrams can be used to represent free volumes of polymers.
- In [materials science](#), polycrystalline microstructures in metallic alloys are commonly represented using Voronoi tessellations. In solid state physics, the [Wigner-Seitz cell](#) is the Voronoi tessellation of a solid, and the [Brillouin zone](#) is the Voronoi tessellation of reciprocal (wave number) space of crystals which have the symmetry of a space group.
- In [mining](#), Voronoi polygons are used to estimate the reserves of valuable materials, minerals, or other resources. Exploratory drillholes are used as the set of points in the Voronoi polygons.
- In computer graphics, Voronoi diagrams are used to calculate 3D shattering / fracturing geometry patterns. It is also used to procedurally generate organic or lava-looking textures.
- In autonomous [robot navigation](#), Voronoi diagrams are used to find clear routes. If the points are obstacles, then the edges of the graph will be the routes furthest from obstacles (and theoretically any collisions).



John Snow's original diagram

- In [machine learning](#), Voronoi diagrams are used to do [1-NN](#) classifications.<sup>[14]</sup>
- In [biology](#), Voronoi diagrams are used to model a number of different biological structures, including [cells](#)<sup>[15]</sup> and [bone microarchitecture](#).<sup>[16]</sup>
- In [user interface](#) development, Voronoi patterns can be used to compute the best hover state for a given point.<sup>[17]</sup>
- In [computational fluid dynamics](#), the Voronoi tessellation of a set of points can be used to define the computational domains used in [finite volume](#) methods, e.g. as in the moving-mesh cosmology code AREPO.<sup>[18]</sup>

## See also <sup>[[edit](#)]</sup>

### Algorithms

Direct algorithms:

- [Fortune's algorithm](#), an  $O(n \log(n))$  algorithm for generating a Voronoi diagram from a set of points in a plane.
- [Lloyd's algorithm](#), aka [k-means](#) clustering, produces a Voronoi tessellation in a space of arbitrary dimensions

Starting with a [Delaunay triangulation](#) (obtain the dual):

- [Bowyer–Watson algorithm](#), an  $O(n \log(n))$  to  $O(n^2)$  algorithm for generating a Delaunay triangulation in any number of dimensions, from which the Voronoi diagram can be obtained.

### Related subjects

- [Centroidal Voronoi tessellation](#)
- [Computational geometry](#)
- [Delaunay triangulation](#)
- [Mathematical diagram](#)
- [Natural neighbor interpolation](#)
- [Nearest neighbor search](#)
- [Nearest-neighbor interpolation](#)
- [Voronoi pole](#)
- [Power diagram](#)

## Notes <sup>[[edit](#)]</sup>

- <sup>^</sup> Franz Aurenhammer (1991). *Voronoi Diagrams – A Survey of a Fundamental Geometric Data Structure*. ACM Computing Surveys, 23(3):345–405, 1991
- <sup>^</sup> Atsuyuki Okabe, Barry Boots, Kokichi Sugihara & Sung Nok Chiu (2000). *Spatial Tessellations – Concepts and Applications of Voronoi Diagrams*. 2nd edition. John Wiley, 2000, 671 pages [ISBN 0-471-98635-6](#)
- <sup>^</sup> Q.T.Tran, D.Tainar and M.Safar (2009) "Transactions on Large-Scale Data- and Knowledge-Centered Systems", pag357. [ISBN 9783642037214](#).
- <sup>^</sup> Daniel Reem, *An algorithm for computing Voronoi diagrams of general generators in general normed spaces*, *In Proceedings of the sixth International Symposium on Voronoi Diagrams in science and engineering (ISVD 2009)*, 2009, pp. 144–152 [↗](#)
- <sup>^</sup> Daniel Reem, *The geometric stability of Voronoi diagrams with respect to small changes of the sites*, Full version: [arXiv 1103.4125 \(2011\)](#) [↗](#), Extended abstract in *Proceedings of the 27th Annual ACM Symposium on Computational Geometry (SoCG 2011)*, pp. 254–263 [↗](#)
- <sup>^</sup> <sup>a</sup> <sup>b</sup> Mark de Berg; Marc van Kreveld; Mark Overmars; Otfried Schwarzkopf (2008). *Computational Geometry* (Third ed.). Springer-Verlag. 7.4 Farthest-Point Voronoi Diagrams. Includes a description of the algorithm.
- <sup>^</sup> Skyum, Sven (18 February 1991). "A simple algorithm for computing the smallest enclosing circle". *Information Processing Letters* **37** (3): 121–125. doi:10.1016/0020-0190(91)90030-L [↗](#)., contains a simple algorithm to compute the farthest-point Voronoi diagram.
- <sup>^</sup> Edelsbrunner, Herbert (1987), "13.6 Power Diagrams", *Algorithms in Combinatorial Geometry*, EATCS Monographs on Theoretical Computer Science **10**, Springer-Verlag, pp. 327–328.
- <sup>^</sup> S. Arya, T. Malamatos, and D. M. Mount, [Space-Efficient Approximate Voronoi Diagrams](#) [↗](#) (PDF), Proc. 34th ACM Symp. on Theory of Computing (STOC 2002), pp. 721–730.
- <sup>^</sup> Jooyandeh, Mohammadreza; Mohades, Ali; Mirzakhah, Maryam (2009). "Uncertain Voronoi Diagram" [↗](#) (PDF). *Information Processing Letters* (Elsevier) **109** (13): 709–712. doi:10.1016/j.ipl.2009.03.007 [↗](#).
- <sup>^</sup> Geraerts, Roland (2010), [Planning Short Paths with Clearance using Explicit Corridors](#) [↗](#) (PDF), International Conference on Robotics and Automation, IEEE, pp. 1997–2004.
- <sup>^</sup> van Toll, Wouter G.; Cook IV, Atlas F.; Geraerts, Roland (2011), [Navigation Meshes for Realistic Multi-Layered Environments](#) [↗](#) (PDF), International Conference on Intelligent Robots and Systems, IEEE/RSJ, pp. 3526–3532.
- <sup>^</sup> "GOLD COAST CULTURAL PRECINCT" [↗](#). ARM Architecture.
- <sup>^</sup> Tom M. Mitchell (1997). *Machine Learning* (International Edition 1997 ed.). McGraw-Hill. p. 233. [ISBN 0-07-042807-7](#).
- <sup>^</sup> Martin Bock (2009). "Generalized Voronoi Tessellation as a Model of Two-dimensional Cell Tissue Dynamics".
- <sup>^</sup> Hui Li (2012). "Spatial Modeling of Bone Microarchitecture".
- <sup>^</sup> "User Interface Algorithms" [↗](#).
- <sup>^</sup> Springel, Volker (2010). "E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh". *MNRAS* **401** (2): 791–851. doi:10.1111/j.1365-2966.2009.15715.x [↗](#).







## References [[edit](#)]

- G. Lejeune Dirichlet (1850). "Über die Reduktion der positiven quadratischen Formen mit drei unbestimmten ganzen Zahlen". *Journal für die Reine und Angewandte Mathematik* **40**: 209–227.
- Voronoi, Georgy (1908). "Nouvelles applications des paramètres continus à la théorie des formes quadratiques". *Journal für die Reine und Angewandte Mathematik* **133** (133): 97–178. doi:10.1515/crll.1908.133.97 .
- Atsuyuki Okabe, Barry Boots, Kokichi Sugihara & Sung Nok Chiu (2000). *Spatial Tessellations – Concepts and Applications of Voronoi Diagrams*. 2nd edition. John Wiley, 2000, 671 pages, ISBN 0-471-98635-6
- Franz Aurenhammer, Rolf Klein & Der-Tsai Lee (2013) "Voronoi Diagrams and Delaunay Triangulations". World Scientific, 2013, 337 pages, ISBN 978-9814447638
- Bowyer, Adrian (1981). "Computing Dirichlet tessellations". *Comput. J.* **24** (2): 162–166. doi:10.1093/comjnl/24.2.162 .
- Reem, Daniel (2009). "An algorithm for computing Voronoi diagrams of general generators in general normed spaces". *Proceedings of the sixth International Symposium on Voronoi Diagrams in science and engineering (ISVD 2009)*. pp. 144–152. doi:10.1109/ISVD.2009.23 .
- Daniel Reem (2011). *The geometric stability of Voronoi diagrams with respect to small changes of the sites*. Full version: [arXiv 1103.4125 \(2011\)](#) , Extended abstract: in *Proceedings of the 27th Annual ACM Symposium on Computational Geometry (SoCG 2011)*, pp. 254–263 .
- Watson, David F. (1981). "Computing the *n*-dimensional Delaunay tessellation with application to Voronoi polytopes". *Comput. J.* **24** (2): 167–172. doi:10.1093/comjnl/24.2.167 .
- Mark de Berg; Marc van Kreveld; Mark Overmars; Otfried Schwarzkopf (2000). *Computational Geometry* (2nd revised ed.). Springer-Verlag. ISBN 3-540-65620-0. Chapter 7: Voronoi Diagrams: pp. 147–163. Includes a description of Fortune's algorithm.
- Rolf Klein (1989). *Abstract voronoi diagrams and their applications*. *Lecture Notes in Computer Science* **333**. Springer-Verlag. pp. 148–157. doi:10.1007/3-540-50335-8\_31 . ISBN 3-540-52055-4.

## External links [[edit](#)]

- Real time interactive Voronoi / Delaunay diagrams with draggable points, Java 1.0.2, 1996–1997
- Real time interactive Voronoi and Delaunay diagrams with source code
- Interactive Voronoi diagrams with Natural Neighbor Interpolation visualization (in WebGL)
- Demo for various metrics
- Mathworld on Voronoi diagrams
- Qhull for computing the Voronoi diagram in 2-d, 3-d, etc.
- Voronoi Diagrams: Applications from Archaeology to Zoology
- Voronoi Diagrams  in CGAL, the Computational Geometry Algorithms Library
- Voronoi Web Site : using Voronoi diagrams for spatial analysis
- More discussions and picture gallery on centroidal Voronoi tessellations
- Voronoi Diagrams  by Ed Pegg, Jr., Jeff Bryant, and Theodore Gray, Wolfram Demonstrations Project.
- Nice explanation of voronoi diagram and visual implementation of fortune's algorithm
- A Voronoi diagram on a sphere
- Plot a Voronoi diagram with Mathematica
- Voronoi software for shattering 3D geometry
- Hand-drawing Voronoi diagrams
- Overlaid Voronoi diagram of the United States based on state capitals
- Overlaid Voronoi diagram of the world based on national capitals



Wikimedia Commons has media related to **Voronoi diagrams**.



This page was last modified on 5 September 2015, at 05:10.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Mobile view](#)

