The Mathematics Enthusiast

Volume 1 | Number 2 Article 4

10-2004

Voronoi Diagrams

Michael Mumm

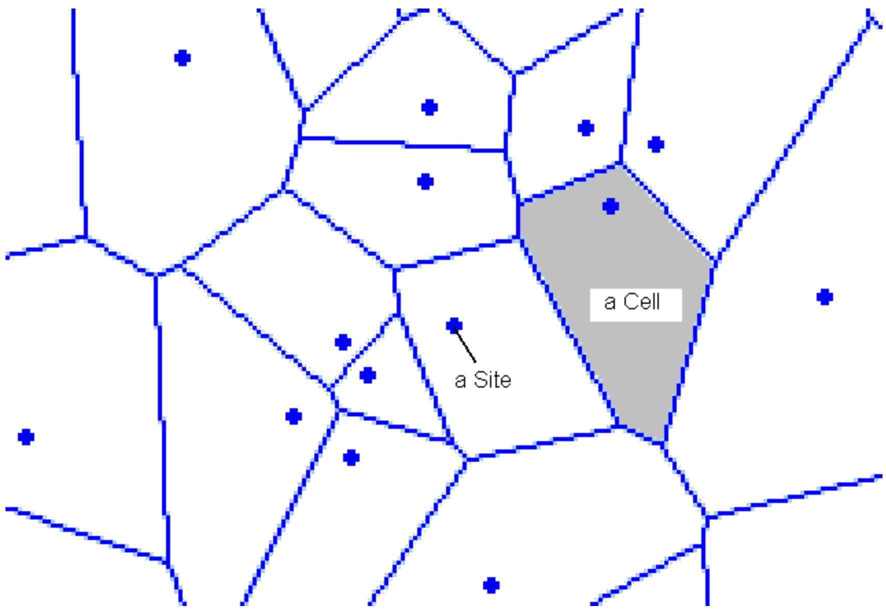
Voronoi Diagrams

Michael Mumm

Statement of the Problem:

Suppose we have a finite number of distinct points in the plane. We refer to these points as *sites*. We wish to partition the plane into disjoint regions called *cells*, each of which contains exactly one site, so that all other points within a cell are closer to that cell's site than to any other site.

An example of a Voronoi diagram:



Stated more formally, suppose P: = { p_1,p_2, …, p_n } is a set of distinct points (sites) in the plane. We subdivide the plane into n cells so that each cell contains exactly one site. An arbitrary point (x,y) is in a cell corresponding to a site p_i with coordinates (x_{pi} , y_{pi}) if and only if $\sqrt{(x - x_{pi})^2 + (y - y_{pi})^2} < \sqrt{(x - x_{pj})^2 + (y - y_{pj})^2}$ for all p_j with $j \neq i$, $1 \leq j,i \leq n$. That is, the Euclidean distance from (x,y) to any other site is greater than the distance from (x,y) to p_i.

It turns out that the boundaries of the cells defined in this way will be composed of straight lines and segments forming convex polygons and will be defined by the perpendicular bisectors of segments joining each pair of sites. This method of partitioning a plane is called a *Voronoi diagram*.

Although this paper deals chiefly with two dimensional diagrams and the Euclidean distance metric, it should be noted that the concept of Voronoi diagrams can be generalized to n dimensions and to an arbitrarily defined distance metric. In addition, general geometric primitives such as line segments or curves may be used as sites instead of ordinary one-dimensional points. In the course of our discussion we will describe techniques of constructing

Voronoi diagrams, provide some historical background on the subject, and discuss the multitude of applications that utilize Voronoi diagrams.

Rationale:

Voronoi diagrams have countless applications in nearly all of the major sciences. Whenever one has a discrete set of data distributed in such a way that the concept of 'distance' has some meaning, a Voronoi diagram may be useful. With a Voronoi diagram as a reference, it is unnecessary to calculate the distance to each site in order to determine which site is closest to a particular point. The site corresponding to the cell that contains the point will always be closest. In applications which have a fixed set of sites, a Voronoi diagram need only be constructed once and then all subsequent distance calculations become unnecessary. Even if more sites are eventually added to a system, the basic structure of the Voronoi diagram remains intact. It is relatively easy to modify an existing diagram to accommodate new sites without reconstructing the entire system. For large scientific projects which use computers, this reduction in basic operations may result in a dramatic increase of algorithmic efficiency.

Voronoi diagrams may be extremely useful in the business world as well. A typical example is a pizza delivery franchise which has a network of restaurants servicing a large city. When an order comes into the central office, the operator, or computer, can use a Voronoi diagram to determine which restaurant will be able to deliver the pizza quickest relative to the location of the caller. It's easy to imagine similar applications arising in a large mail-order company with many distribution warehouses like amazon.com, or in the postal service itself.

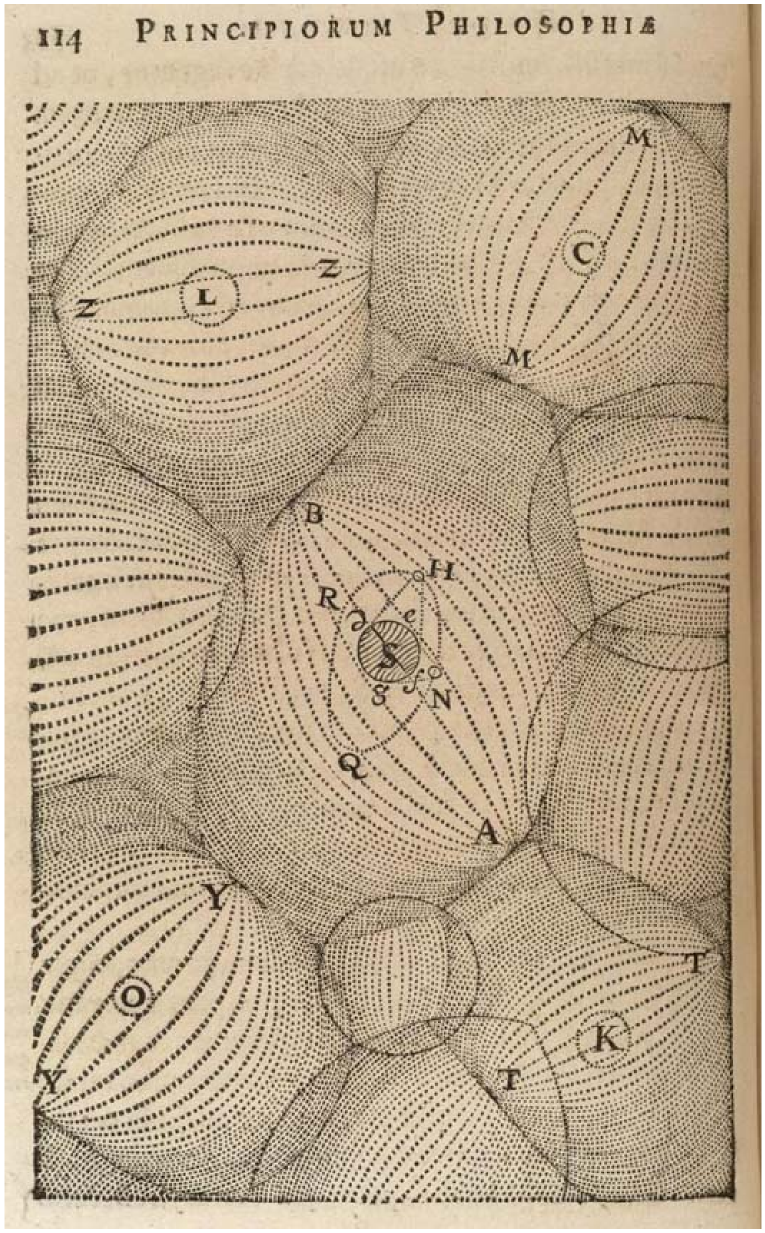
We continue the discussion of applications throughout the paper.

History/Background:

History:

Voronoi diagrams have a long history, dating back as early as the 17^{th} century. Work by Descartes on a partitioning of the universe into 'vortices' is one of the first known references to the subject. Even though Descartes does not explicitly define his vortices in the same way as Voronoi cells, his work is conceptually very similar [3].

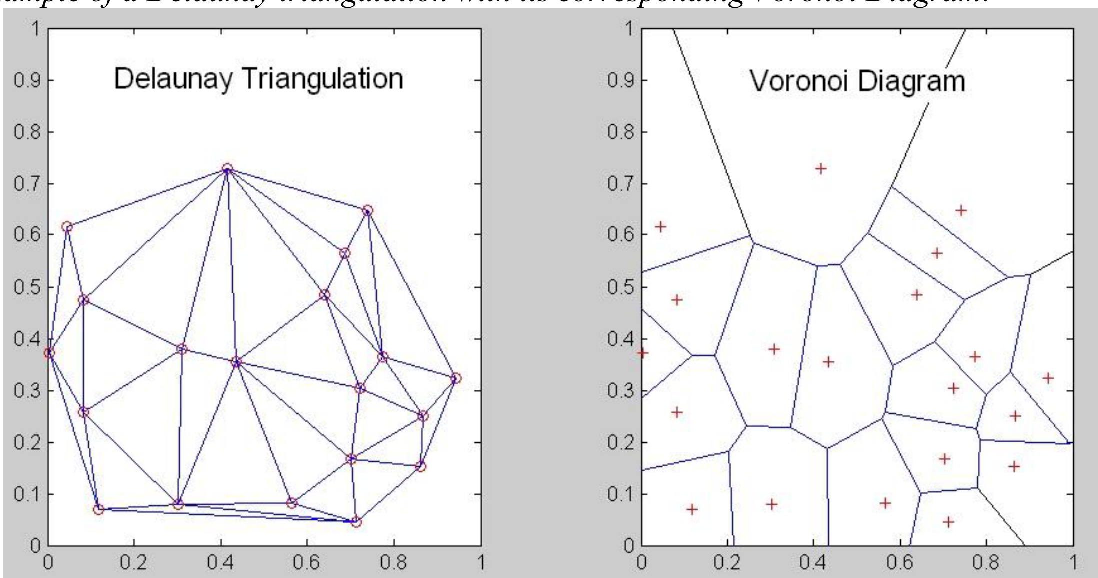
A drawing from Descartes which describes the partitioning of the universe into vortices – Notice that the vortices closely resemble Voronoi cells[11]:



Two German mathematicians, Lejeune Dirichlet and M. G. Voronoi, were credited with formalizing the modern concept of the Voronoi diagram [5]. Dirichlet was born in 1805 and in his work on quadratic forms he made some of the first significant contributions to the field of Voronoi diagrams. Indeed, it is because of him that Voronoi diagrams are also well known as Dirichlet tessellations. Before his death in 1859, Dirichlet had formalized the concept of the Voronoi diagram in the two and three dimensional cases [5]. Work by M. G. Voronoi in 1908 formalized the n dimensional case and gave Voronoi diagrams the name we commonly use today.

The two dimensional dual of the Voronoi diagram in a graph theoretical sense is the Delaunay triangulation [2]. Work on Delaunay triangulations (or, alternately, Delaunay tessellations) was done by French mathematician Charles Delaunay before 1872. In a Delaunay triangulation, any two sites are connected if they share a Voronoi diagram cell boundary, as shown below. An alternate definition, more in accordance with Delaunay's original work, is that two sites are connected if and only if they lie on a circle whose interior contains no other sites [3].

An example of a Delaunay triangulation with its corresponding Voronoi Diagram:



Even before Voronoi diagrams were formalized mathematically, they were developed independently in other sciences. In 1909, BT Boldyrev a Russian scientist, used "area of influence polygons" in his work in Geology [10]. Voronoi diagrams were used in Meteorology by Thiessen in 1911 to help model average rainfall [5]. Influential work in crystallography was done utilizing Voronoi diagrams by a German named Paul Niggli in 1927. In 1933, physicists EP Wigner and F. Seitz did important research using Voronoi diagrams in physics. Voronoi diagrams continued to play a key role in research done in Physics, Ecology, Anatomy, and Astronomy throughout the 1900's.

Applications and algorithms:

As mentioned earlier, applications of Voronoi diagrams are by no means confined strictly to mathematics. They go by many names as they relate to various fields of science. We provide a small glossary below [2,3,16].

Field of science:	**Term used:**
Mathematics:	Voronoi diagram, Dirichlet tessellation
Biology and Physiology:	Plant polygons, Capillary domains, Medial axis transform
Chemistry and Physics:	Wigner-Seitz zones
Crystallography	Domains of action, Wirkungsbereich
Meteorology and Geography:	Thiessen polygons

In the Work/Solutions section of this paper, we describe a way of constructing a Voronoi diagram using a geometric algorithm. It turns out that although this kind of algorithm is very easy to intuitively understand, it is not as computationally efficient as other known techniques. In [3], Aurenhammer and Klein classify some of these algorithms. The geometric technique we describe is an example of an *incremental construction* algorithm and it has a relatively poor algorithmic efficiency of $O(n^2)$. Using this notation, we assume the reader has a basic understanding of algorithmic efficiency classification. In these examples, n is the number of sites in the system.

Another category of algorithm used for Voronoi diagrams is *divide and conquer*. This technique works by recursively dividing the set of sites in order to decrease the problem size. Eventually the subsets of sites are small enough that diagrams are easily constructible. These sub-diagrams then must be merged back together, up the recursive tree, into the complete diagram for the system. Although this merging process is complicated, and care needs to be given as to how the set of sites is split each time, the result is a total algorithmic efficiency of O(n*log(n)). This is a significant improvement over the incremental construction algorithm.

Another algorithm of O (n*log(n)) efficiency is the *sweep* algorithm. This algorithm works by sweeping a vertical line across the plane horizontally. As this line passes through various Voronoi cells, their boundaries are constructed. Although the efficiency of this algorithm is the same as the *divide and conquer*, it can have some additional desirable characteristics if the sites are distributed in a certain way. There is no universally superior algorithm; ultimately one must choose an algorithm based on the particulars of the data and of the application.

Work/Solutions:

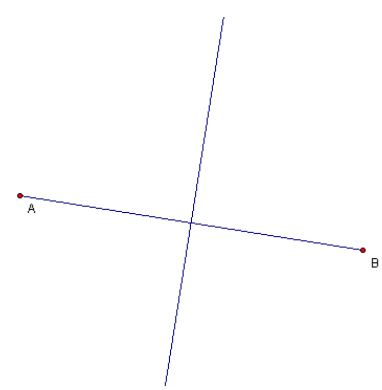
Our first task is to describe a simple geometric algorithm for constructing a Voronoi diagram. We start with a system containing only two sites and build to the four site case. From here it is easy to generalize to any system with a finite number of sites.

A simple geometric construction algorithm:

As mentioned in the first section, perpendicular bisectors have a fundamental role in the construction of a Voronoi diagram. When our system contains only two sites it is a very easy matter to construct a Voronoi diagram; the perpendicular bisector of the segment joining the two sites is all that is necessary (Figure 2).

Fig 1:

Fig 2:



To build a Voronoi diagram for three sites (points A, B, D below) we first construct a triangle with vertices at the three sites. From here we must rely on the fact that the perpendicular bisectors of the three sides of a triangle meet at a single point. This is illustrated in Figure 3 below. To complete the diagram we simply remove the superfluous rays, line segments, and points which were used in construction (Figure 4).

Fig 3:

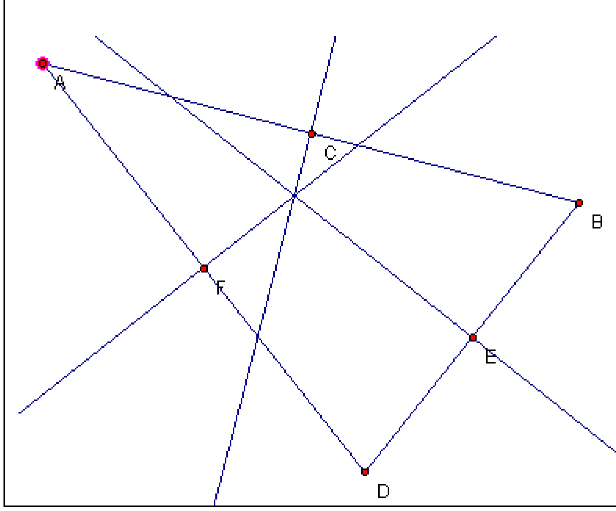
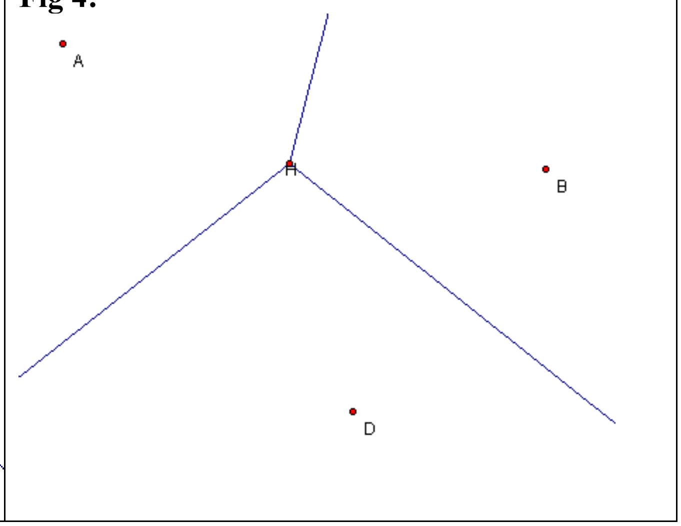


Fig 4:



Things become a bit more complicated when we have four sites. We begin with the existing diagram for three sites, then add another site (point I below) and adjust the diagram accordingly. To do this we must construct perpendicular bisectors between I and each previously existing site. These bisectors allow us to determine which parts of the diagram need to be adjusted.

We begin by constructing the perpendicular bisector between I and D (Figure 5). From this, we see that everything to the left of the line KJ belongs to I's cell. We adjust the diagram accordingly and remove the superfluous parts (Figure 6).

Fig 5:

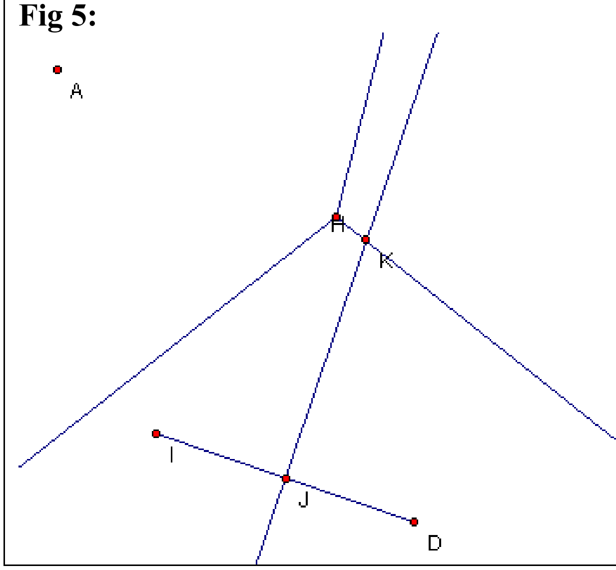
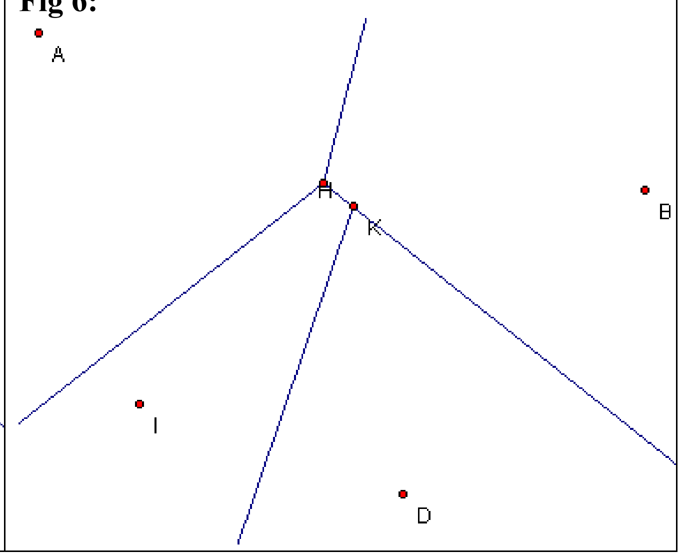


Fig 6:



Now we construct the bisector between I and A (Figure 7). From this we see that everything below the line LN belongs to I's cell. Now we remove the superfluous parts (Figure 8).

Fig 7:

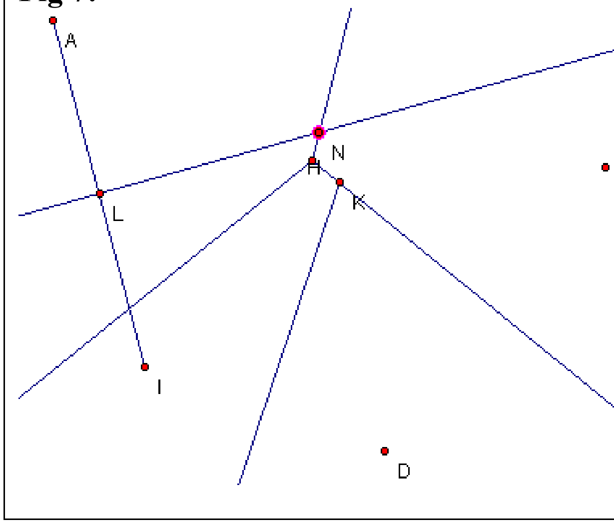
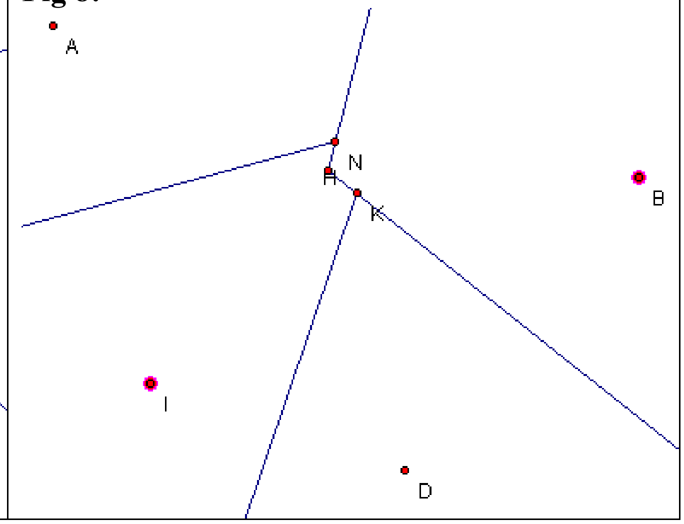


Fig 8:



Finally, we construct the bisector between I and B, our last pair of sites (Figure 9). From this we see that everything to the left of segment NK belong to I's cell. After removing the remaining superfluous parts, we have our completed Voronoi diagram for four sites (Figure 10).

Fig 9:

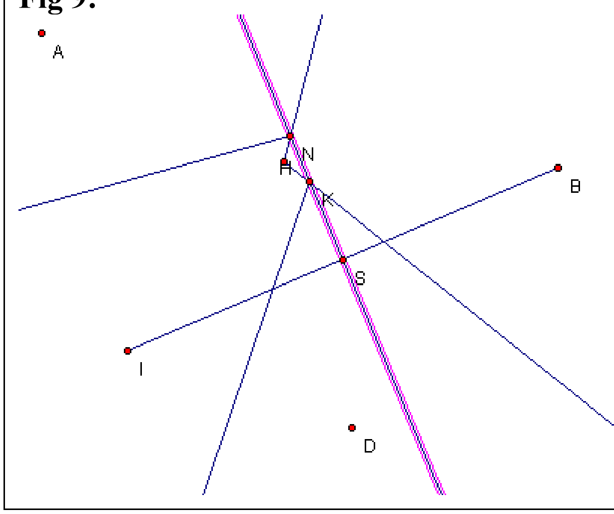
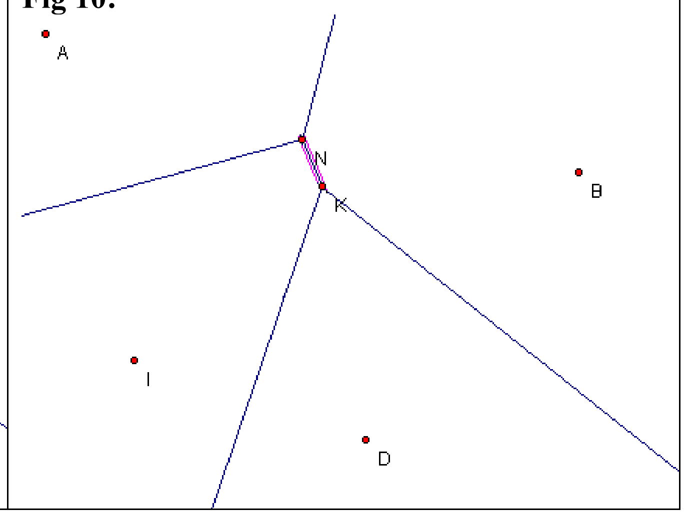


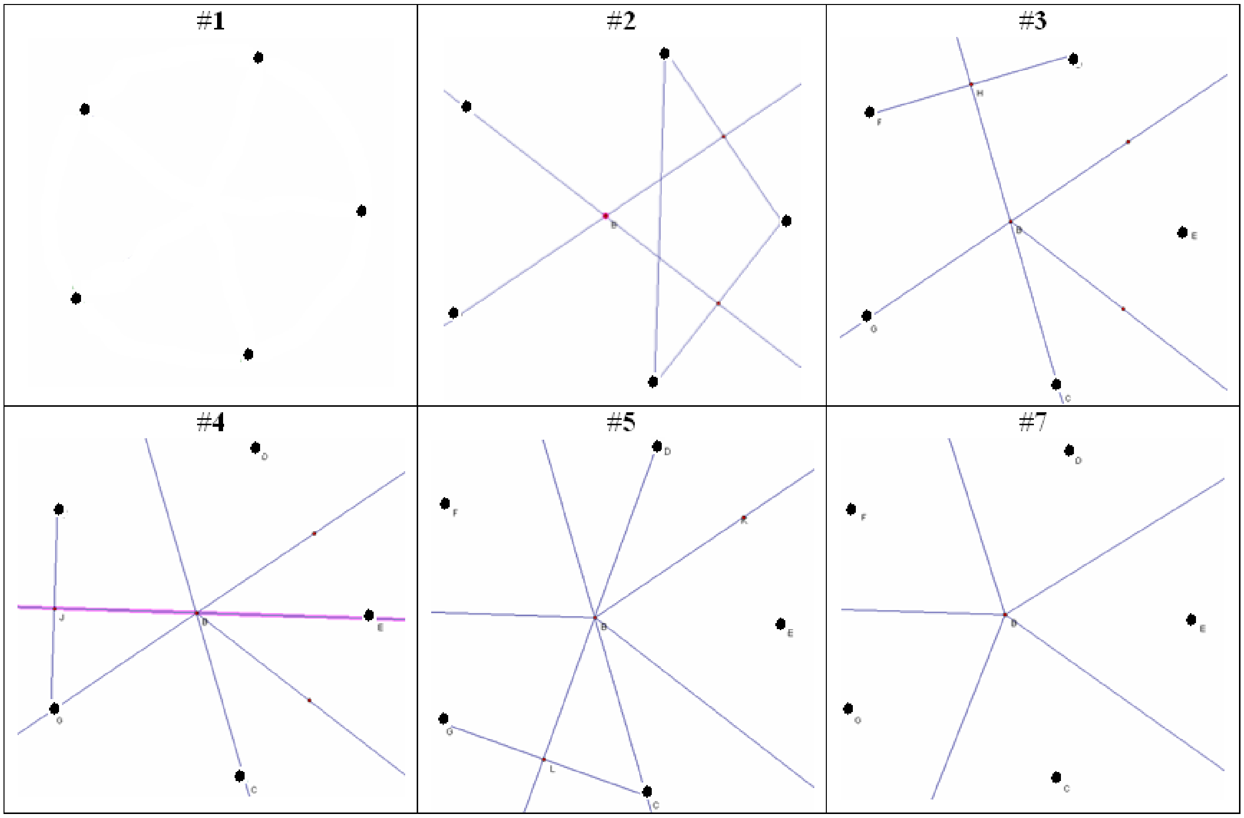
Fig 10:



This kind of approach is sufficient for any number of sites. From the four site system above we can construct a five site system by adding a site and considering which boundaries need to be adjusted for each new pair of sites. Continuing in this way we can build our system as large as we wish.

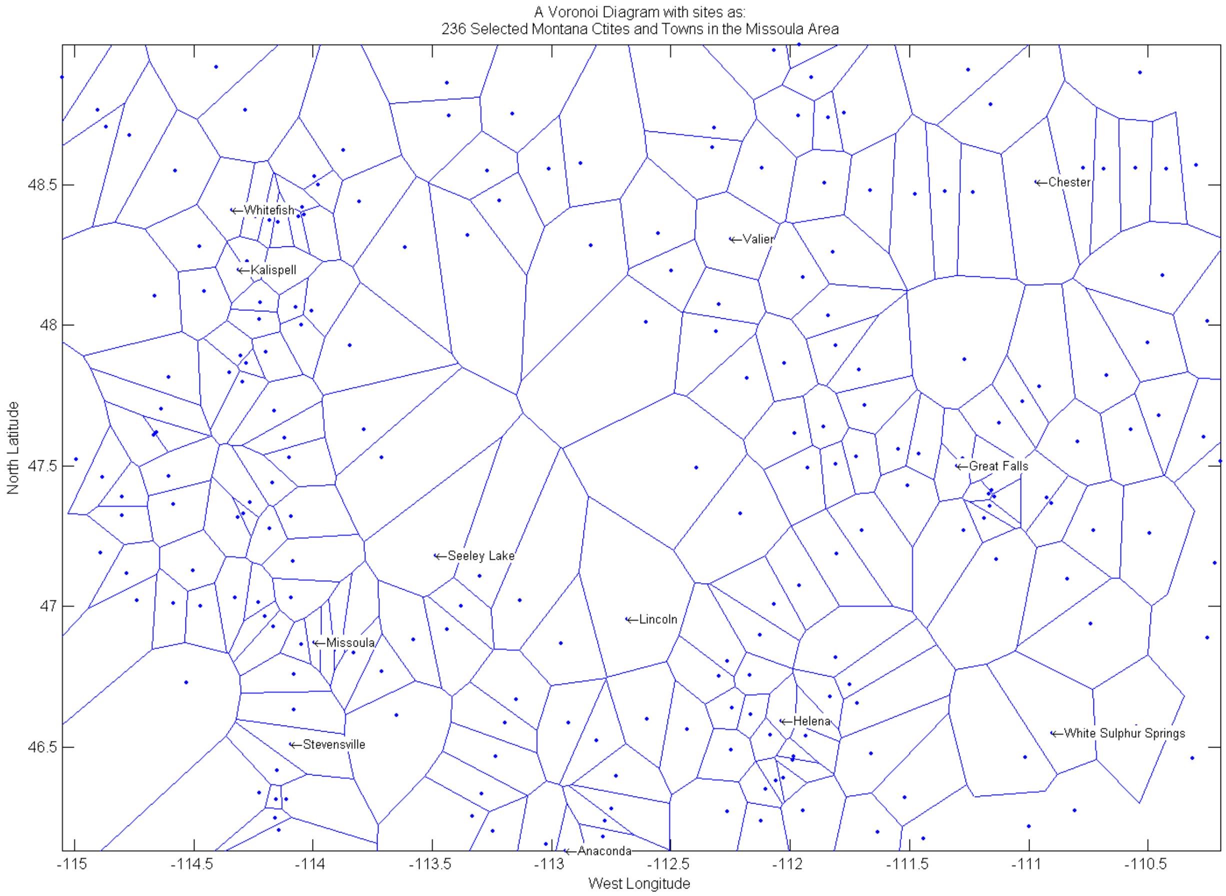
An example problem:

Construct a Voronoi diagram using the vertices of a regular pentagon as sites.



Voronoi of the Missoula area:

I decided it would be interesting to construct a Voronoi diagram of sites with local geographical relevance. After a good deal of searching, I was able to track down some data through the Montana Natural Resource Information System's GIS web service [7]. Using their service I was able to produce a list of the longitude and latitude of some 236 cities and towns in the Missoula, Montana area. This data then needed to be formatted so that it could be plotted in a Euclidean space. After that, MATLAB's Voronoi package was used to produce the actual diagram. Finally, I labeled a few key communities based on size of population and range of disbursement, so that one could get a good geographic impression of how the data was distributed. The result would have been too cluttered if I had labeled every site.



One thing the diagram provides is a sense of how remote certain sites are. If a city or town has few neighboring communities, its Voronoi cell is generally large. For example, Seeley Lake has a large Voronoi region corresponding to its relative remoteness. Even though the diagram gives us no true indication of population, we can make a rough extrapolation based on the fact that, in general, urban areas of high population densities have numerous nearby neighbors. Thus, we would expect larger cities to occur in areas of the diagram with smaller cells. This is more or less consistent with what the diagram indicates.

A more practical application of a diagram like this might be related to school districting. If each site on our diagram corresponded to a community with a school, the boundaries of the cells would be effective districting lines. If I child lived in a particular cell, they would go to school in the community that corresponded to that cell because it would be the shortest distance away.

Conclusions and Implications:

As we have seen, Voronoi diagrams have been around a long time and have undergone a good deal of study. In fact, Aurenhammer and Klein claim that about one out of 16 papers in computational geometry have been on research concerning Voronoi diagrams! More than 600 papers on the subject are listed in [8]. Ever since Descartes partitioned the universe into vortices in 1644, research in Voronoi diagrams has been done by some of the brightest minds in science and mathematics.

Because of the importance of Voronoi diagrams, the efficiency of computer algorithms used in diagram construction is equally important. There has been a good deal of progress made in algorithm design already and popular modern algorithms are extremely efficient. Even so, there is room for improvement by adapting algorithms to take advantage of known characteristics (about distribution etc.) in a set of data or a type of application.

We have hinted at a number of general applications of the Voronoi technique, but it should be made clear that there are vastly more that have gone without mention. Because of their generality, Voronoi diagrams are useful in an extremely diverse array of situations, in many different aspects of science and of everyday life. Here are a just few other applications, listed in [9]:

- Voronoi diagrams may be used in computer GUI applications that need to determine which link is closest to the mouse cursor when a user clicks on the screen.
- Voronoi diagrams have been used to correspond distribution of Bark Beetle attacks on trees to the beetle's known territorial behavior. The beetles tend to feed in isolated Voronoi regions, in a way that discourages intra-species competition for the same resources [12].
- Voronoi diagrams are used in some graphics software to create "Non-Photorealistic" images. The software distorts the original image around Voronoi cells of selected points.
- Voronoi diagrams have been investigated as a means of grouping words and multi-part symbols in documents [13].
- Voronoi diagrams have been used to increase the accuracy in the "river-mile" positioning system often employed along rivers and in reservoirs. Given an arbitrary position on the water, Thiessen polygons are used to determine which known river mile position is closest [14].
- Voronoi diagrams of a collection of randomly distributed points are being used to model some kinds of steel[15].

It is possible to generalize Voronoi diagrams to three or more dimensions and to define our distance metric however we wish. For instance, we could use a taxicab distance metric to produce a more useful diagram of sites in a city. Since it is seldom possible to travel directly to a destination in a city, a Voronoi diagram using the taxicab metric would more accurately reflect the amount of time it takes to travel along city streets to a particular site. In addition, we could weigh the 'distance' of certain busy streets more than streets in other less trafficked areas so that our diagram would even more accurately reflect travel time. The exceptional adaptability of the Voronoi diagram concept is what makes it so versatile and widely used throughout the sciences.

Works Cited:

1. Index of Biographies. 12 Nov. 2003. School of Mathematics and Statistics, University of St. Andrews, 1996 <http://www-gap.dcs.st-and.ac.uk/~history/BiogIndex.html>

2. Voronoi Diagram. 12 Nov. 2003. Wolfram Research, 2003. <http://mathworld.wolfram.com/VoronoiDiagram.html>

3. Aurenhammer, Franz and Klein, Rolf. "Voronoi Diagrams." <wwwpi6.fernuni-hagen.de/publ/tr198.pdf> FernUniversität in Hagen

4. Meyer, Walter. Geometry and Its Applications. New York: Harcourt Academic Press, 1999

5. Voronoi Diagrams in Biology. 15 Nov. 2003. Beloit College, 5/15/98 <http://biology.beloit.edu/students/zdravko/vor_history.html>

6. Voronoi Diagrams: The Post Office Problem. 15 Nov. 2003. <www.comp.lancs.ac.uk/~kristof/research/ notes/voronoi/voronoi.pdf>

7. The Montana Natural Resource Information System Geographic Information System (GIS). 22 Nov. 2003. Montana Natural Resource Information System, 2003, <http://nris.state.mt.us/gis/default2.htm>

8. A. Okabe, B. Boots, and K. Sugihara. Spatial Tessellations: Concepts and Applications of Voronoi Diagrams. John Wiley & Sons, Chichester, UK, 1992.

9. Geometry in Action. 22 Nov. 2003. David Eppstein, ICS, UC Irvine, 2003, <http://www.ics.uci.edu/~eppstein/gina/voronoi.html>

10. Fast Generalizations of Voronoi Diagrams Using Graphics Hardware. 23 Nov. 2003. Kenneth E. Hoff III, University of North Carolina. 10/6/99 <http://www.cs.unc.edu/~geom/voronoi/>

11. Heavens (World Treasures of the Library of Congress: Beginnings). 12 Mar. 2004. <http://www.loc.gov/exhibits/world/heavens.html>

12. Dirichlet tessellation of bark beetle spatial attack points. 15 Mar. 2004. Byers, J.A., 1992. <http://www.wcrl.ars.usda.gov/cec/dir-abs.htm>

13. Using the Voronoi Tessellation for Grouping Words and Multi-part Symbols in Documents. 12 Mar. 2004. M. Burge and G. Monagan, Department of Computer Science, Swiss Federal Institute of Technology. 3 Jul. 1997 <http://www.computing.armstrong.edu/FacNStaff/burge/publications/papers/visgeo-95/>

14. Dynamic Segmentation and Thiessen Polygons: A Solution to the River Mile Problem. 17 Mar. 2004. William W. Hargrove, Richard F. Winterfield, Daniel A. Levine, Office of

Environmental Restoration and Waste Management.
<http://research.esd.ornl.gov/CRERP/DOCS/RIVERMI/P114.HTM>

15. Modeling of Microstructures by Voronoi cells. 17 Mar. 2004. Mikael Nygårds and Peter Gudmundson. Department of Solid Mechanics KTH, Stockholm, Sweden.
<http://www.brinell.kth.se/CONF99/Mikael_Nygards.html>

16. Voronoi Diagrams: Applications from Archaology to Zoology. 18 Mar. 2004. Scot Drysdale, Dartmouth College. <http://www.ics.uci.edu/~eppstein/gina/scot.drysdale.html>