COMPOSITION AND DECOMPOSITION OF THE WEIGHTED VORONOI DIAGRAM

Mu, Lan and Radke, John

Abstract:

Weighted Voronoi diagrams enhance the strictly location criteria of ordinary planar Voronoi diagrams by considering weights associated with sites under study. These weights are most often derived from what is commonly referred to as attributes, data stored in tables within a Geographic Information System. Based on different underlying processes, weighted Voronoi diagrams can be classified into: multiplicatively weighted, additively weighted, compoundly weighted, power, and sectional diagrams (Okabe, Boots, Sugihara and Chiu, 2000). The purpose of this paper is to present and demonstrate algorithms that compose and decompose the multiplicatively weighted Voronoi diagram, constructed from point data defined as:

\[ \text{region}(p) = \{ x | d_w(x, p) \leq d_w(x, q), q \in S \}, \]

where the weighted distance \( d_w(x, p) \) is the Euclidean distance \( d_e(x, p) \) divided by the weight: \( d_w(x, p) = d_e(x, p) / w(p) \) (Aurenhammer and Edelsbrunner 1984).

Our motivation and purpose to compose and decompose is to characterize landscape, record and detect environmental change over time. We develop a program “\text{WVD}” in Visual Basic, which distinguishes itself from other approaches found in the literature. The concepts and algorithms we propose are based in the field of Geographic Information Science, the implementation process is straightforward for the transition from a model environment (virtual space) to the real world (physical space), it supports an easy exchange with common GIS software, and it serves as a teaching tool helping to vision the abstract model.

We target and overcome obstacles to using weighted Voronoi diagrams which include weak links between theory and operational models, and the lack of off-the-shelf tools in GIS to generate such diagrams. We develop two methods to construct the multiplicatively weighted Voronoi diagram: 1) a growth simulation model, and 2) a vertex calculation method.
Figure 1 illustrates the graphic user interface (GUI) of the program **WVD**: 

![WVD Interface](image)

The main features of **WVD**:

1) inputs of point locations and weights can be either interactive using a mouse (Figure1) (freehand or specify x, y coordinates) or read from a text file in a common format (weight, x, y);
2) the points can be easily added, moved or removed, and the weights can be interactively modified which allows continuous tuning of the model;
3) features such as weighted Voronoi, ordinary Voronoi, Appolonius circles, weights, and points can be easily switched on or off;
4) the extent of growth can be controlled (for growth model only);
5) the points and weights can be output as text files, and the weighted Voronoi and ordinary Voronoi (vector) diagrams can be output as an AML file for easy input into an Arc/Info coverage;
6) raster images can be loaded into the background to facilitate the composing and decomposing process;
7) vector data (in ESRI ungenerate format) can be input to approximate polygons as circular arcs and line segments after which they can be decomposed with points and weights following the weighted Voronoi process;
8) the decomposed results can be output as text tables and provide a highly compressed and alternative method for representing polygons as points and weights;
9) batch processing is accommodated for multiple inputs, composing the weighted Voronoi and outputting the results as AMLs.
The **growth simulation method** dynamically simulates the weighted Voronoi diagram on the user’s computer screen. It visually allows one to observe how points compete with one another and partitions the space following the weighted Voronoi model. The **vertex calculation method** uses the geometric properties and relations in the Apollonius circles, their generator point weights and locations, their interaction and vertex points to calculate and determine valid vertices and edges (arcs and lines) in the final diagram. Although this method is mathematically more complicated it is computationally less intensive, is faster, can be monitored during the process, which provides an instructional visualization aid, and is easily modified.

The decomposition feature is another powerful tool in this program. Given the boundary shape of a 2D object, we decompose it to a set of points with weights associated with each segment. The properties and rules obtained from the composition process play a key role in this feature. The result of this process helps one understand the spatial relationships in terms of form and process.

**Reference:**