COS 451 – PROJECTS, Due 16 January 2018

1 Generation of Simple Polygons

Design, analyze and implement an algorithm for contructing simple polygons. You may use a probabilistic algorithm. The sole input to your program is an integer n. The output should be a "complicated" simple polygon with n vertices. The quality of your program will depend on speed, but more important, on how winding, irregular, and cluster-free your polygons are. You should produce a number of printouts to display your polygons for various values of n.

2 Polygon Triangulation

Implement a sweep-line algorithm for triangulating a simple polygon. The goal is practical efficiency for polygons with fewer than a thousand edges. Try various implementations of the cross-section structure and compare their relative efficiencies. Draw conclusions about the method of choice. Produce printouts of triangulations.

3 Computing Voronoi Diagrams

Learn about Steve Fortune's plane-sweep method for computing the Voronoi diagram of n points in the plane in $O(n \log n)$ time and implement it. Produce graphical evidence that your program works (at least assuming non-degeneracies).

4 Finding the Center

Write a program which, given n points in the plane, finds one of its centers. The cleverness of your solution will be in the implementation of dynamic operations on convex hulls. Give graphical evidence that your program works.

5 Equidecomposability

Implement the equidecomposability algorithm for simple polygons. Provide a clever visualization scheme to make the correspondence between the pieces clear.

6 Point Location

Implement Kirkpatrick's algorithm.

7 Fractional Cascading

Implement and animate fractional cascading for planar catalog graphs. Guess a power-law for avalanche sizes.

8 Low-Cutting Paths

Implement an algorithm that given n points in the plane connects them in a simple path that no line can cut in more than $O(\sqrt{n})$ points. Give graphical evidence that your program works.