Project 1: Generation of Simple Polygons

Design, analyze and implement an algorithm for constructing 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 $n$.

Project 2: 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.

Project 3: Voronoi Diagrams

Learn about and implement Fortune’s Voronoi diagram algorithm (see lecture notes). Provide graphical evidence that your code works.

Project 4: Minimum Enclosing Disk

Implement the minimum-enclosing disk algorithm and provide graphical evidence that your code works.

Project 5: Linear Programming

Implement the randomized LP algorithm in fixed, arbitrary dimension, and provide graphical evidence that your code works.

Project 6: Random Independent Sets

Consider an undirected, connected graph $G = (V, E)$ and define the following Markov chain: a state is any independent set in $G$ (ie, any subset of nodes with no two of them adjacent to each other); the chain has an edge from $S$ to $S'$ if they differ in exactly one element. The transition probabilities are inferred by the following process. Given the current state $S_t$, pick a random node $v$ uniformly in $V$: (i) if $v \in S_t$, then set $S_{t+1} = S_t \setminus \{v\}$; (ii) if $v$ is neither in $S_t$ nor adjacent to any node in $S_t$,
then set $S_{t+1} = S_t \cup \{v\}$; (iii) in all other cases, set $S_{t+1} = S_t$. Prove that the chain is ergodic (i.e., irreducible and aperiodic). Find its stationary distribution.

It is often desired to achieve a given stationary distribution with a reversible chain. Here is the idea. Pick some $\mu > 0$ and suppose that you want to sample a random independent set $S$ with probability proportional to $\mu^{|S|}$. (The case $\mu = 1$ gives the uniform distribution.) We modify the Markov chain as follows: Given the current state $S_t$, pick a random node $v$ uniformly in $V$: (i) if $v \in S_t$, then set $S_{t+1} = S_t \setminus \{v\}$ with probability $\min\{1, 1/\lambda\}$; (ii) if $v$ is neither in $S_t$ nor adjacent to any node in $S_t$, then set $S_{t+1} = S_t \cup \{v\}$ with probability $\min\{1, \lambda\}$; (iii) in all other cases, set $S_{t+1} = S_t$. Prove that the chain is ergodic and time-reversible. Find its stationary distribution.

Pick a random graph $G$ in $\mathbf{G}(n, p)$, i.e., with $n$ nodes and each pair of them connected at random, independently with probability $p$. Implement the first Markov chain above and estimate the average size of a random independent set as a function of $p$ and $n$. Repeat the experiment with the second chain and study the dependency on $\mu$. 