A generator of random convex polygons in a disc

Olivier Devillers[1] Philippe Duchon[2] Rémy Thomasse[1]

[1]INRIA Sophia Antipolis - Méditerranée

[2]Univ. Bordeaux, LaBRI, UMR 5800, F-33400 Talence, France
CNRS, LaBRI, UMR 5800, F-33400 Talence, France

Introduction

- $\mathcal{D}$ a disc in $\mathbb{R}^2$ with radius 1 centered at o
- $(x_1, \dots, x_n)$ a sample of n points uniformly i.i.d. in $\mathcal{D}$
- P_n the convex hull of $(x_1, \dots, x_n)$
- $f_0(P_n)$ its number of vertices.

Then [2] :

$$\mathbb{E} f_0(P_n) = c\, n^{\frac{1}{3}} + o\left(n^{\frac{1}{3}}\right)$$

where $c > 0$ is constant.

- Generation of such a polygon:
  - Explicitly generate n points uniformly in $\mathcal{D}$.
  - Compute the convex hull.
- Faster?

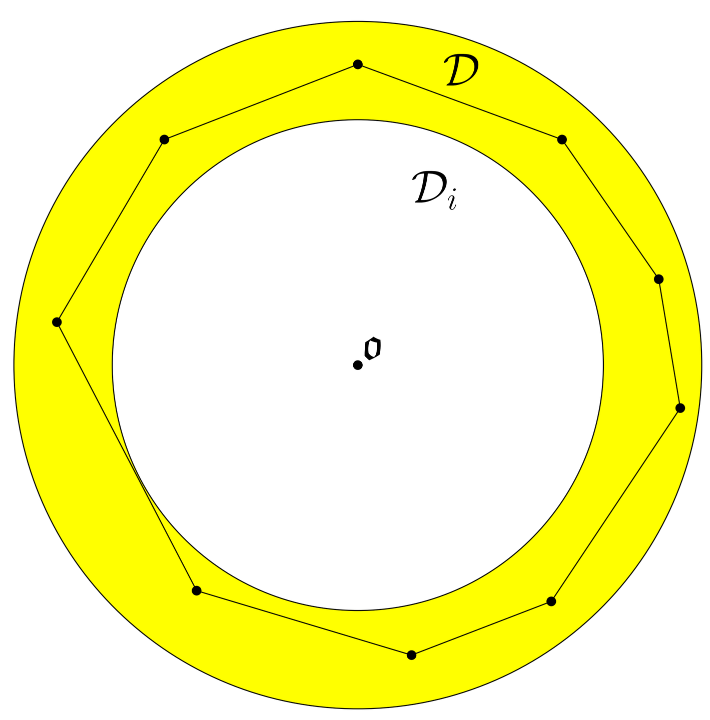
Result

- We propose an algorithm that **reduces the number of points generated**
- The expected number of points generated is only $O(n^{\frac{1}{3}} \log^2 n)$.

Outline of the Algorithm

Idea:

- Given the convex hull of small number of points
- The number of points deeply inside $\mathcal{D}$ can be computed without explicit generation.
- We **generate only the points that are close to the convex hull**.



Algorithm

- First, generate a small number of points in $\mathcal{D}$ and compute its convex hull
- While the number of points is $< n$ do
  - Compute the largest disc centered at o inscribed in the convex hull
  - Choose a number of points to simulate at this step
  - Simulate the number of points that falls in this inscribed disc at this step
  - Generate the rest of the points in the annulus defined by these two discs
  - Update the convex hull

Initialization

- Generate a small number of points in $\mathcal{D}$, such that o is in the convex hull.
- The probability that $o \notin P_n$ decreases exponentially.
- This initialization can be done in constant expected time.

Simulation of Points

- The polygon P_{m_i} is the convex hull of m_i points at step i.
- s_i is the number of new points simulated at step i.

$$s_i = \begin{cases} m_i & \text{if } m_i < n \log^{-2} n \\ n \log^{-2} n & \text{otherwise} \end{cases}$$

- $\mathcal{D}_i$ is be the largest inscribed disc in P_{m_i} centered in o, and r_i its radius.
- The number of points that falls in $\mathcal{D} \setminus \mathcal{D}_i$: Binomial variable $B(s_i, 1 - r_i^2)$.
- The points in $\mathcal{D}_i$ won't change the convex hull: avoid to generate them.
- We generate the rest of the points uniformly in the annulus $\mathcal{D} \setminus \mathcal{D}_i$.
- We update the convex hull using a Graham scan.

Size and Time Complexities

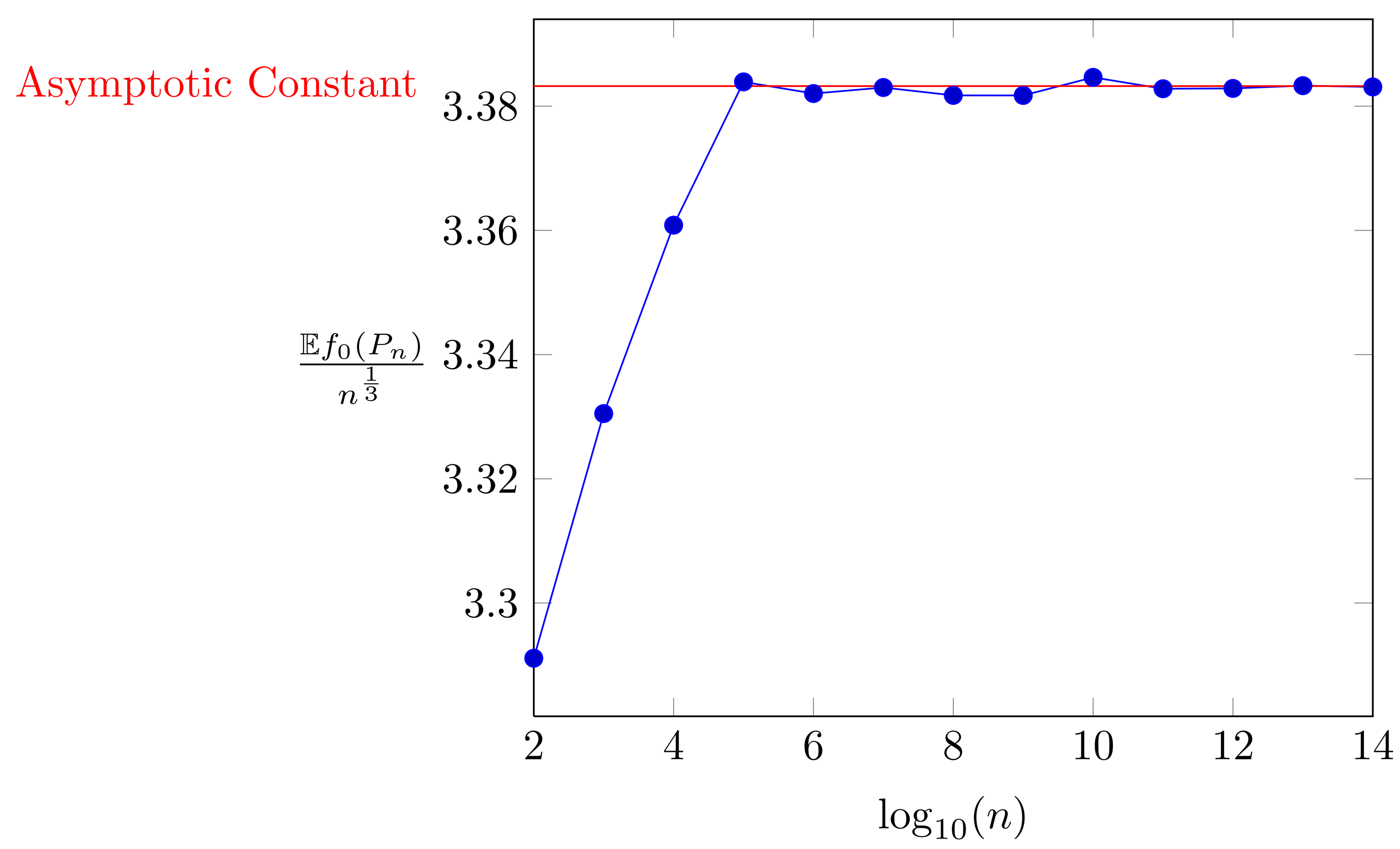
- At a given step i, the expected area of $\mathcal{D} \setminus \mathcal{D}_i$ is $O(m_i^{-\frac{2}{3}} \log^{\frac{2}{3}} m_i)$.
- The number of points generated is the size of the hull up to a factor $\log^2 n$.
- The expected memory needed is $O(n^{\frac{1}{3}})$ [1].

Trade-off:

- Increase the expected memory by a logarithmic factor.
- The expected time complexity is reduced from $O(n^{\frac{1}{3}} \log^2 n)$ to $O(n^{\frac{1}{3}} \log^{\frac{2}{3}} n)$.

Experiments

- Implementation in C++.
- Submitted for integration in the CGAL library.



Bibliography

[1] Olivier Devillers, Philippe Duchon, and Rémy Thomasse.
A generator of random convex polygon in a disc.
Research report, INRIA, January 2014.

[2] Matthias Reitzner.
Random polytopes.
In *New perspectives in stochastic geometry*, pages 45–76. Oxford Univ. Press, Oxford, 2010.

Acknowledgements

The authors wish to thank the participants of 2013 Presage workshop in Valberg and particularly Pierre Calka for stimulating discussions.

Supported by ANR blanc PRESAGE (ANR-11-BS02-003) and Région PACA.