Testing Geometric Convexity

FSTTCS 2004, IMSc, Chennai

Luis Rademacher, Santosh Vempala
Mathematics Department and CSAIL, MIT

Outline.

- *Description and motivation of the problem of convexity testing.*
- Our results. A convexity tester with exponential query complexity. A natural “lines-based” algorithm has exponential complexity.

Convexity Testing. Motivation.

- The subject of this talk is another problem that fits the property testing framework: informally, checking algorithmically whether a set in $\mathbb{R}^n$ is convex.
- One of the motivations of this problem: there is hope that approximately convex sets can be sampled efficiently. Sampling makes a series of algorithmic problems tractable, like volume approximation.

Problem Statement.

- Specification of the input set: as a minimum we need a "membership oracle", that is, a black box that takes as input a point $x \in \mathbb{R}^n$ and answers the question "Does x belong to the input set?".
- A membership oracle allows the existence of short proofs of non-convexity: a triple (x, y, z) such that z is in the segment $[x, y]$, and x and y are in the input but z is not.

Problem Statement.

- If the given set is almost convex (say, a convex set missing a point), then it is essentially impossible to detect its non-convexity.
- We relax the requirement of the algorithm to accepting when the set is convex, and rejecting when it is far from convex, with high probability.
- Distance between sets: volume of the symmetric difference.

$$d(S, C) = \mathrm{vol}(S \Delta C)$$

  Analogous to other examples of property testing.
- We say that a set S is ϵ-convex iff there exists a convex set C such that

$$d(S, C) \leq \epsilon \, \mathrm{vol}(S).$$

Problem Statement.

- The problem is still intractable with just a membership oracle: e.g., when some part of the set is far from the rest, how do we find it?
- One solution: the algorithm also has access to a “random oracle”, which gives uniformly and independently sampled points from the input set.
- Having a random oracle seems to be consistent with the chosen metric, which is based on volume.

Problem Statement.

In summary, for us, a tester of convexity is an algorithm that, when given as input a distance $0 < \epsilon < 1$ and a confidence parameter δ and access to the membership and random oracles of a set $S \subseteq \mathbb{R}^n$, accepts with probability at least $1 - \delta$ if S is convex and rejects with probability $1 - \delta$ if the set is not ϵ-convex.
The (total) query complexity of a tester is the number of queries made by it, to any oracle.

Related work.

- Parnas, Ron, Rubinfeld: On testing convexity and submodularity. (2003)
  This paper includes the description and analysis of an algorithm for testing convexity of discrete functions, in one dimension.

Outline.

- Property testing. Description and motivation of the problem of convexity testing.
- *Our results. A convexity tester with exponential query complexity. A natural “lines-based” algorithm has exponential complexity.*

Closest Convex Set Exists.

- Given a compact set, a closest convex set exists:
  Our distance is continuous in the space of convex bodies, which has nice compactness properties (the space of convex bodies contained in a bounded domain with the empty set is compact, by Blaschke's selection theorem).

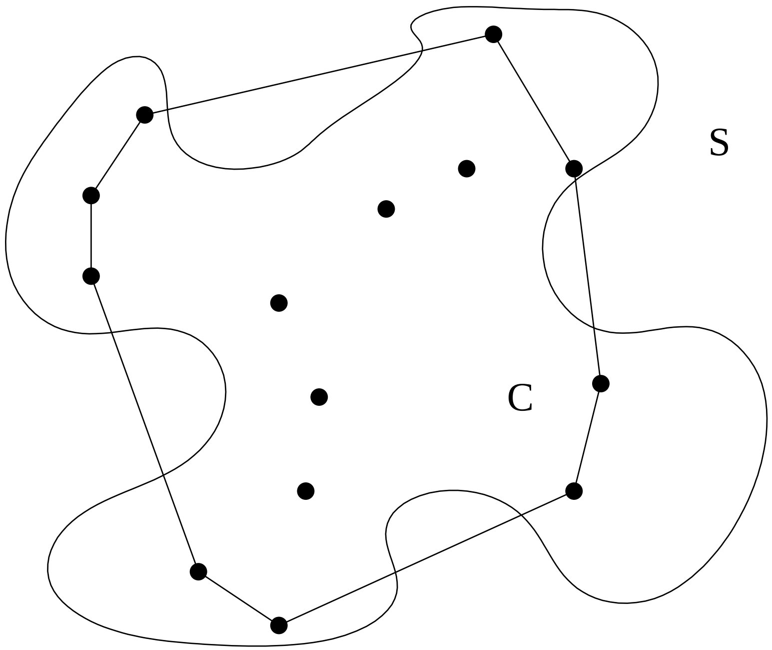
Tester with Exponential Query Complexity.

- There exists a tester with query complexity $\mathrm{poly}(n)\left(\frac{c}{\epsilon}\right)^n$, independent of the set.

Tester with Exponential Query Complexity.

Tester:

- Get $\left(\frac{cn}{\epsilon}\right)^n$ random points from the input set S and consider their convex hull C.
- Check that the volume of the symmetric difference between C and the input set is small by using both oracles.

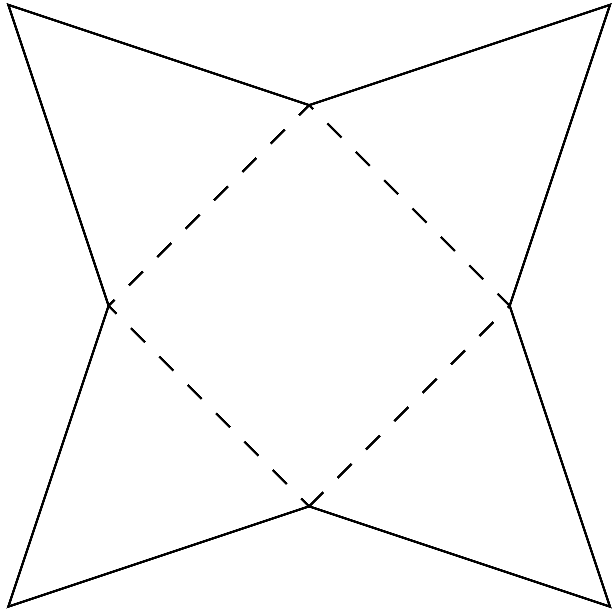


Analysis of a Lines-Based Test.

- What about a test that checks the convexity along some lines? More precisely, random lines determined by pairs of random points from the input set.
- We will see that a tester that checks the convexity of the intersection of the set with random lines may need an exponential number of queries to detect substantial non-convexity, that is, there is a family of bodies that is far from convex (not c/n^2-convex) and such that the intersection of all random lines but an exponentially small fraction with the input set is an interval (i.e., convex).
- Intuition: Lines that only intersect the set near the boundary are unlikely.
- Intuition: However, a substantial fraction of the volume of a set is near the boundary, so there is a lot of room for non-convexity near the boundary.

Description of the Non-Convex Body.

- The cross-polytope is the n-dimensional generalization of the octahedron.
- Our non-convex body, the “cross-polytope with peaks”: on top of every facet of the cross-polytope add the convex hull of the facet and a point above its center, i.e. add a simplex.

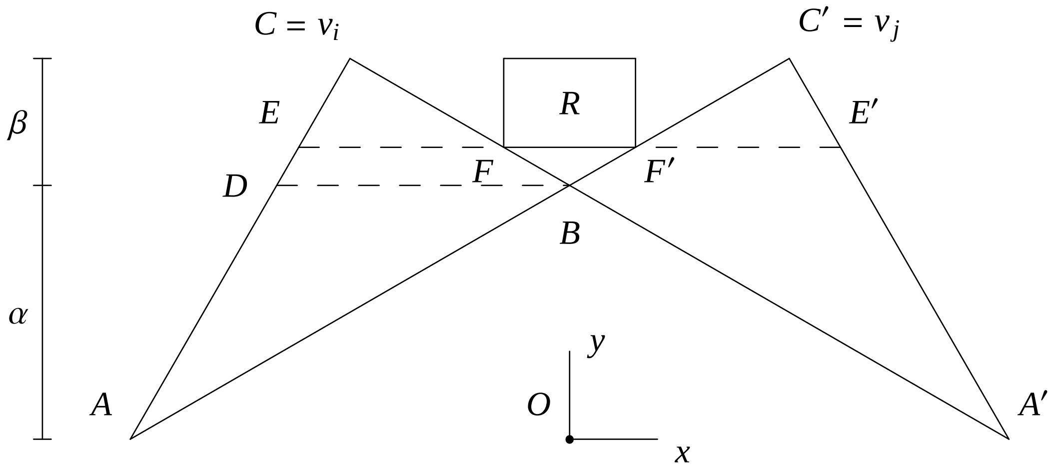


The Lines-Based Algorithm “Fails”

- We choose the height of each peak to be $\Theta(1/n)$ of the distance of any facet to the origin. This ensures that the only lines that show non-convexity are those that intersect the body exactly at 2 adjacent peaks and nowhere else. This event has an exponentially low probability (no more that $(n+1)/2^n$).

The sets are far from convex.

- Proof idea:
  - Any convex set that is close to the cross-polytope with peaks must cover most peaks, substantially. In particular, the convex set must cover substantially most pairs of adjacent peaks.
  - While covering a pair of adjacent peaks substantially, a lot of mass is added between the peaks because it is convex, thereby contributing to the symmetric difference.



Conclusion.

- We showed that convexity can be tested with exponential query complexity.
- We showed that a natural lines-based test has exponential complexity.
- Conjecture: if most (say a $1-\epsilon$ fraction) 2-dimensional random sections through the origin (determined by pairs of random points from the input set) of a set are convex then the set is nearly (say $O(n\epsilon)$) convex.
  Intuition: what happens near the boundary doesn't seem to be hard to detect in this case as in the lines case.
- "Most" and "nearly" in the conjecture are such that it supports a test with polynomial query complexity: test a polynomial number of random planes through the origin.