

Random Polygons  
2. interim

Freie Universität Berlin

6th December 2011

- ▶ accomplish 6.12.-12.12.
- ▶ all polygon generation algorithms
- ▶ shortest path algorithm
- ▶ GUI basic version

## Geometry Framework:

- ▶ needed for some polygon generation algorithms
- ▶ existing:
  - ▶ needed accuracy not given
  - ▶ too big
  - ▶ partly inconsistent
  - ▶
- ▶ therefore small own implementation

## 2nd thing i forgot:

- ▶ sthg

## Permute and Reject by Auer and Held

input: size  $n$  or  $n$  points

- i. generate points, if necessary
  - ii. permute points
  - iii. test if polygon simple, if continue with ii.
- 
- ▶ simple to implement
  - ▶ generates all simple polygons with uniform distribution
  - ▶ runtime depends on polygons needed to be generated to encounter simple polygon (max.  $n=15$  for fast results)
  - ▶ not suited for practical use

input: size  $n$  or  $n$  points

- ▶ needs special treatment for polygons not in general position
- ▶ generates all polygons, but not with uniform distribution

input: set  $S$  of  $n$  points

- ▶ generates not every possible simple polygon

## Incremental Construction & Backtracking by Auer and Held

input:  $n$  points

- i. set of all possible edges, all unmarked, randomly choose current point  $s$
  - ii. recursively for current point:
    - a. add next possible unmarked edge to polygon, mark all intersecting edges
    - b. backtrack if no unmarked edges left and incompleted polygon
- 
- ▶ speed depends on backtracking
  - ▶ complex to optimize backtracking
  - ▶ currently not suited for practical use

input: size n, radius circle, max. speed, steps t

generate random points on circle  $\rightarrow$  regular polygon

- a. move each vertex with random speed and direction

- ▶ polygon is simple
- ▶ vertices in bounding region

- ▶ still not accessible from GUI
- ▶ seems to generate specific class of polygons
- ▶ interesting for statistical analysis



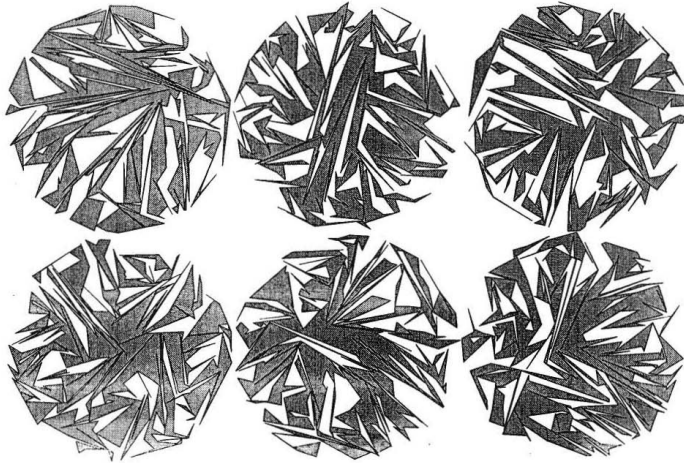


Figure 6:  $n = 360$ ,  $t = 1000$ , six different random seeds.

8

## Random Polygon Algorithm by Dailey and Whitfield

input: size  $n$

- i. generate 3 random points  $\rightarrow$  random  $n$ -gon  $P$  size 3
- ii. randomly choose and discard edge  $\overline{ab}$
- iii. determine region  $P'$  in polygon visible from  $\overline{ab}$
- iv. randomly choose point  $c$  in  $P'$
- v. add edges  $\overline{ac}$ ,  $\overline{bc}$  to polygon  $P$

- ▶ main reason for geometry framework
- ▶ nontrivial problem to determine visible region  $P'$
- ▶ most komplex of all generation algorithms

- ▶ order of points representing polygon: cc-wise
- ▶ triangularization
- ▶ random point in polygon
- ▶ intersection line with polygon
- ▶ surface area of polygon

- ▶ basic version
- ▶ next important step: better way to pass parameters

## This Milestone:

- ▶ 2 polygon generation algorithms missing
- ▶ shortest path generator missing
- ▶ possible in 1 week

## Next Milestone:

- ▶ history objects
- ▶ step-by-step visualization
- ▶ statistic backend