Computational Geometry and Geometric Computing
Eric Berberich
Kurt Mehlhorn
Michael Sagraloff

Winter 2009/2010
Discussion on November 25th.

Exercise 5

Motivation

We practise planar subdivisions.

Bisectors

The distance of a point p to a point set A is defined as follows: $\delta(p, A) := \inf\{|p-a| : a \in A\}$. The bisector of two point sets A_1, A_2 is again the set of points having equal distance to A_1 and A_2:

$$\mathcal{B}(A_1, A_2) := \{x : \delta(x, A_1) = \delta(x, A_2)\}$$

- Determine the bisector of two lines.
- Determine the bisector of a line and a circle.
- Determine the bisector of two circles (of different radius).

Note that objects can intersect.

Point in polygon

Design an algorithm that decides whether a point is contained in the bounded region induced by a weakly simple chain. You may first exclude some degenerate cases, but your overall solution should be able to work for any input (that is, to be *complete*).

DCEL

- Give two (significantly different) examples of doubly-connected edge list representations, in which a pair of twin half-edges e and e' have the same incident face.
- Give pseudo-code (using basic dcel-pointers) to
  - list all vertices adjacent to a given vertex
  - list all edges of a bounded face in a not necessarily connected subdivision

Point location

Let S be a subdivision of $\mathbb{R}^2$ with complexity n. Give a sweep-like algorithm that computes for every point $p \in P$ ($|P| =: m > 1$) in which face of S it is contained. Show that your algorithm runs in time $O((n+m)\log(n+m))$.

Circles

Let C be a set of n circles (maybe of different radius) in the plane. Describe a plane sweep algorithm to compute all the circle's intersection points. Note that a circle does not intersect a second, if it is completely contained in that, but circles can intersect tangentially. Your algorithm should run in time $O((n + k) \log n)$ where k is the number of intersection points.

Have fun with the solution!