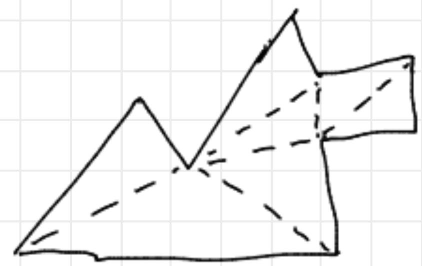
Recall: Every simple polygon can be triangulated.



Following the proof yields an obvious $O(n^4)$ time algorithm, which can be improved to $O(n^2)$.

Today: practical $O(n \log n)$ time algorithm.

History:

- 1978. First $O(n \log n)$ algorithm. Garey, Johnson, Preparata, Tarjan.
- 1984. Simpler. Fournier and Montuno. — this is what we'll study
- . . . $O(n \log \log n)$ . . . $O(n \log^* n)$ . . .
- 1991. $O(n)$ algorithm. Chazelle. But it is too complicated to implement. (Uses polygon-cutting theorem, planar separator theorem, but no fancy data structures.)

We will also study an $O(n \log^* n)$ randomized algorithm of Seidel.

Triangulation Algorithm

Assume points have distinct y coordinates (else imagine tipping slightly).

Note: degeneracy and precision are big topics — we may cover some.

Step 1. Find a *trapezoidization* of the polygon — from each vertex shoot a horizontal line inside the polygon until it hits the boundary.

Step 2. From the trapezoidization, compute a triangulation.

Step 2 takes O(n) time.
Step 1, using plane-sweep, takes O($n \log n$).
Other algorithms make step 1 faster.

Step 2. From trapezoids to triangles.

First join trapezoids to form *unimonotone* polygons.

Triangulating a unimonotone polygon in linear time.

Step 1. Trapezoidization.

Use Plane-Sweep, a basic technique in planar computational geometry.
First used by Shamos and Hoey 1976 to find intersections of line segments in the plane.

Plane Sweep (as a general paradigm)

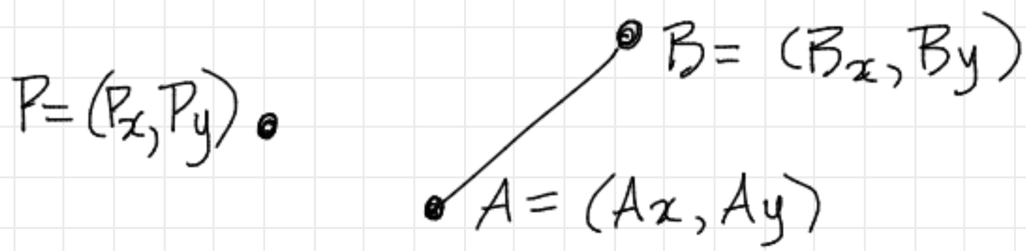
Sweep a horizontal line across the plane, from bottom to top, analyzing the sequence of 1-dimensional cross-sections and the changes to them.

What are cross-sections:

Plane Sweep Algorithm

Elementary test needed by Plane Sweep

Sidedness Test: Given 3 points, A, B, P, is P on/left/right of line through A and B?



How to solve this:

Data structure for Plane Sweep

Timing for Plane Sweep

Details of plane-sweep for trapezoidization.

(above was general plane-sweep)

Faster triangulation algorithms — these improve the trapezoidization step

- O(n) algorithm of Chazelle (as mentioned in previous lecture). Not practical.

- O(n log* n) randomized algorithm of Seidel (faster than O(n log n)).

Idea: Incremental algorithm — add edges one by one in random order, maintaining the trapezoidal decoposition.

as mentioned last day:

optimal algorithm $O(n)$

- Bernard Chazelle 1991

[Triangulating a simple polygon in linear time]
B Chazelle - Discrete & Computational Geometry, 1991 - Springer
Abstract. We give a deterministic algorithm for triangulating a simple polygon in linear time. The basic strategy is to build a coarse approximation of a triangulation in a bottom-up phase and then use the information computed along the way to refine the triangulation in a top- ...
Cited by 840
From: google scholar

But so complicated that there's no implementation!