Lecture 1
Introduction to Computational Geometry

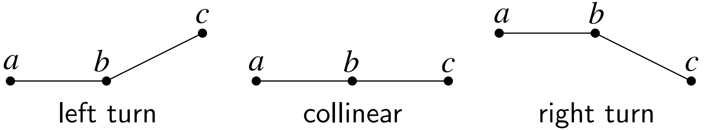
Elisha Sacks

Computational Geometry

Algorithmic study of combinatorial geometry.

- Many simple elements (points, lines, circles, triangles).
- Mostly 2D and 3D; some higher dimensions.
- Queries and constructions.
- Optimal algorithms and lower bounds.

Parameters and Predicates

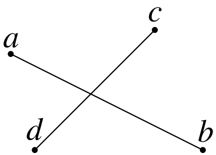


- Geometry is modeled with numerical parameters.
- The most common case is points with Cartesian coordinates.
- Predicates are signs of polynomials in these parameters.
- Geometric properties are expressed as predicates.
- Example: a path *abc* is a left or right turn if

$$\mathrm{LT}(a, b, c) = (c_x - b_x)(a_y - b_y) - (c_y - b_y)(a_x - b_x)$$

  is positive or negative. We will see why later.
- The zero (degenerate) case is rare but important.

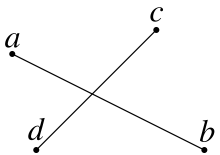
Line Segment Intersection



When do segments *ab* and *cd* intersect?

- Can test if the line intersection point is on both segments.

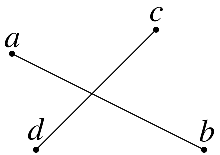
Line Segment Intersection



When do segments *ab* and *cd* intersect?

- Can test if the line intersection point is on both segments.
- But the intersection point is not needed.

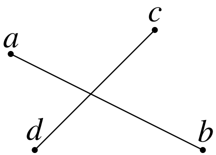
Line Segment Intersection



When do segments *ab* and *cd* intersect?

- Can test if the line intersection point is on both segments.
- But the intersection point is not needed.
- Check if *c* and *d* are on opposite sides of the *ab* line, and if *a* and *b* are on opposite sides of the *cd* line.

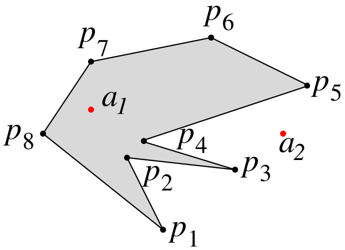
Line Segment Intersection



When do segments *ab* and *cd* intersect?

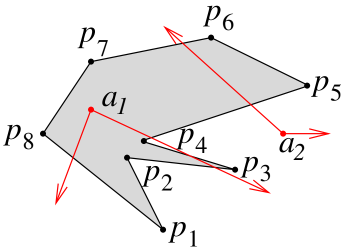
- Can test if the line intersection point is on both segments.
- But the intersection point is not needed.
- Check if *c* and *d* are on opposite sides of the *ab* line, and if *a* and *b* are on opposite sides of the *cd* line.
- Intersection test:
  $\mathrm{LT}(a, b, c)\mathrm{LT}(a, b, d) < 0$ and $\mathrm{LT}(c, d, a)\mathrm{LT}(c, d, b) < 0$.

Point in Polygon



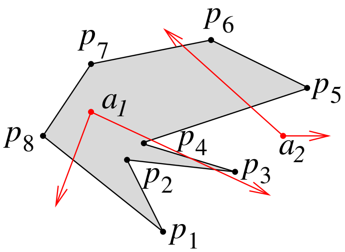
- A polygon is represented by its vertices in boundary order.
- What is an algorithm for testing if a point is inside a polygon?

Point in Polygon



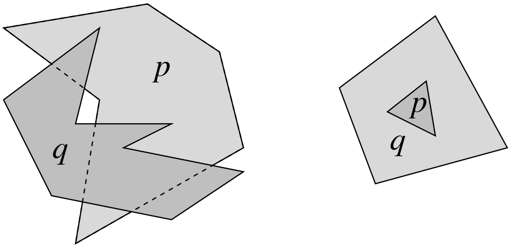
- A polygon is represented by its vertices in boundary order.
- What is an algorithm for testing if a point is inside a polygon?
- Any ray based at a intersects p an odd number of times.

Point in Polygon



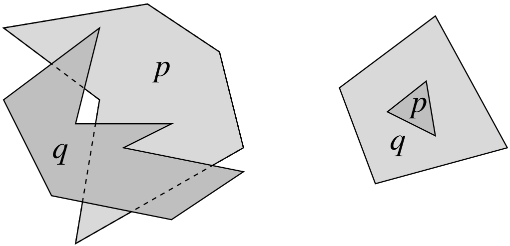
- A polygon is represented by its vertices in boundary order.
- What is an algorithm for testing if a point is inside a polygon?
- Any ray based at a intersects p an odd number of times.
- What about special cases?

Polygon Intersection



When do polygons p and q intersect?

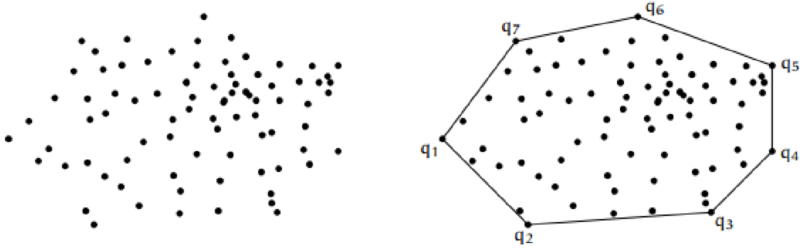
Polygon Intersection



When do polygons p and q intersect?

- Two edges intersect, p is inside q, or q is inside p.
- We know how to test for edge intersection.
- There is a faster algorithm called a line sweep.
- If no edges intersect, p is inside q if any vertex of p is inside q.

Convex Hull

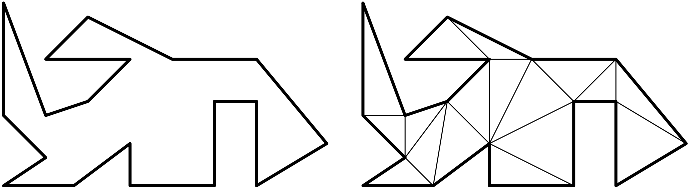


(a) Input. (b) Output.

Smallest convex region containing n points.

- 2D: $n \log n$.
- 3D: $n \log n$.

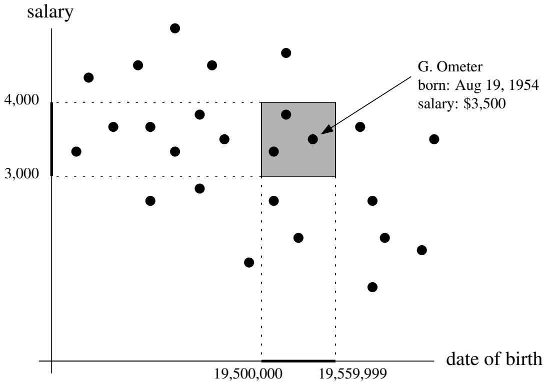
Polygon Triangulation



Decompose a polygon with n vertices into triangles.

- 2D: $n \log n$.
- 3D: decompose a polyhedron into tetrahedrons.
- Need to add vertices (Steiner points) for some inputs.
- $nr + r^2 \log r$ for $r = \mathrm{O}(n^2)$ reflex edges.

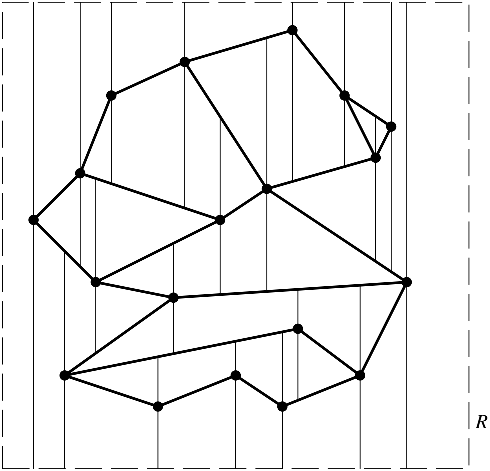
Range Search



Find points in axis-aligned box.

- Input size is n; output size is k.
- 2D: $k + \log n$ query; $n \log n$ preprocessing.
- 3D: $k + \log^2 n$ query; $n \log^2 n$ preprocessing.
- Octrees: fast in practice.

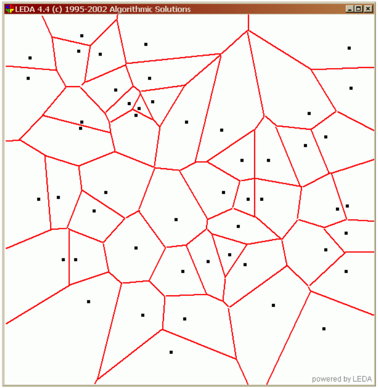
Point Location



Locate the cell that contains a point in a mesh with n vertices.

- 2D: $\log n$ query; $n \log n$ preprocessing.
- 3D: open problem!

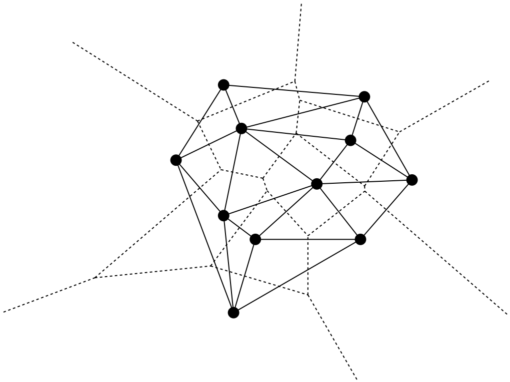
Voronoi Diagram



Compute the region that is closest to each of n sites.

- 2D: $n \log n$.
- 3D: $k + n \log n$ for output size $k = O(n^2)$.

Delaunay Triangulation



Triangulate n vertices with a maximal minimum angle.

- Equivalent to Voronoi diagram.
- convex hull in dimension d gives Delaunay triangulation in dimension $d - 1$.

Other Topics

- Robustness and ACP
- Linear programming
- Persistent binary trees
- Duality
- Binary space partitions
- Robot motion planning
- Meshing
- Euclidean shortest paths
- Differential and projective geometry
- Mechanism kinematics