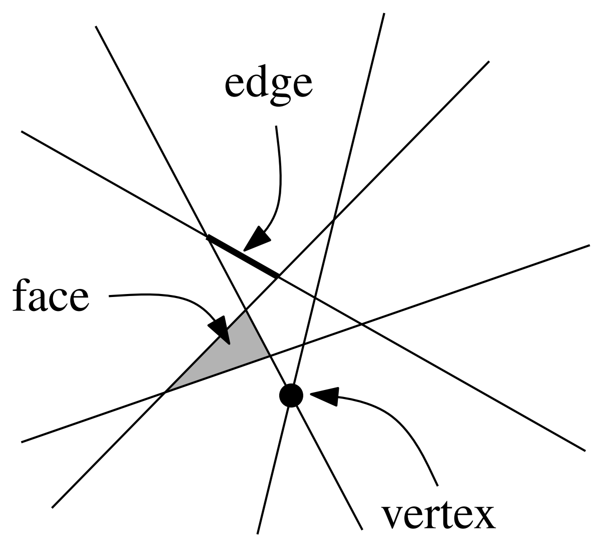
Recall a problem we considered before: given n points, are there 3 (or more) collinear

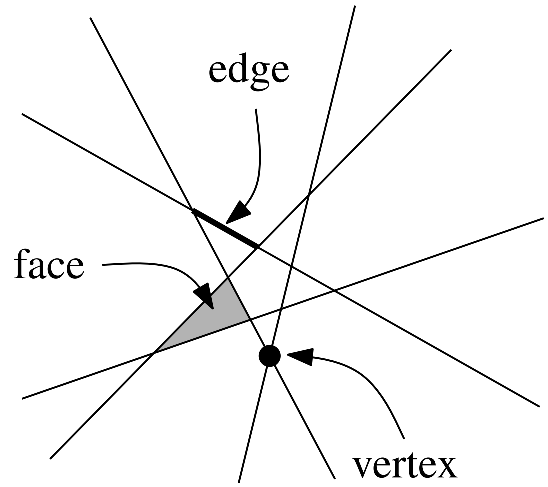
By duality (points <—> lines) this becomes:
given n lines, do 3 of them intersect at a point.

To get an $O(n^2)$ algorithm, we study ***line arrangements***.

A set of n lines in the plane partitions the plane into faces (cells), edges, vertices, called the *arrangement*.



How many vertices, edges, faces for n lines?



A ***degeneracy*** is parallel lines or >2 lines through one point.

Exact bounds if there are no degeneracies (these decrease in case of degeneracy):

vertices: $\binom{n}{2}$

edges: every line is cut by $n-1$ other lines into n edges so n^2 edges.

faces: $f_n = f_{n-1} + n$, $f_0 = 1$

$f_n = n + n-1 \cdots + 1 + f_0 = \binom{n+1}{2} + 1$

Constructing arrangements

input: n lines
output: list of faces, edges, vertices and all incidence relationships (*)
(note: size is Theta(n^2))

Plane sweep would take $O(n^2 \log n)$ (because n^2 events, and log n per update)

There is an $O(n^2)$ time algorithm

Idea: incremental. Add lines one by one.
(not randomized).
Maintain complete info. (*)

How to update after adding line ℓ_i to the line arrangement:

- find intersection point x with ℓ_1
- find edge of ℓ_1 containing x
- find face f containing e

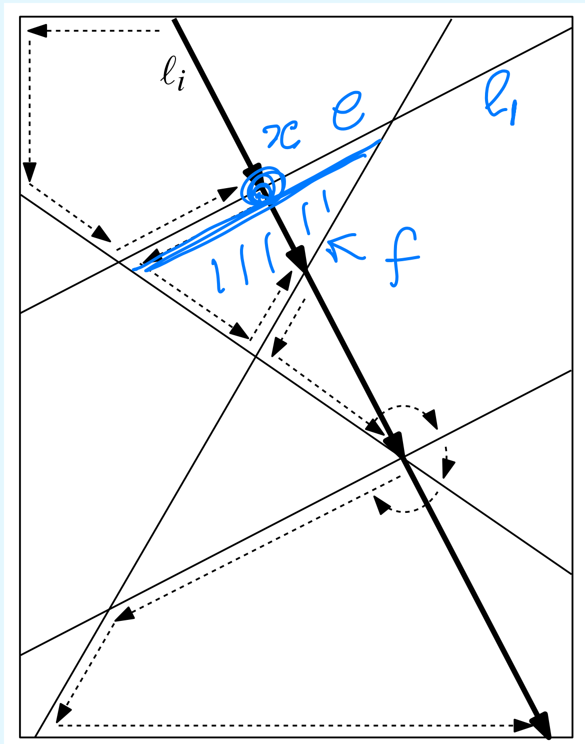
- walk around edges of f (starting from e) to find ℓ_i exits f and enters new face

- continue in new face

... update arrangement as we go.

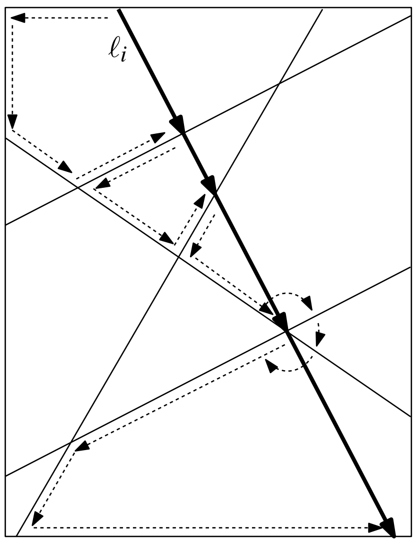
Time = O(# edges in faces intersected by ℓ_i)

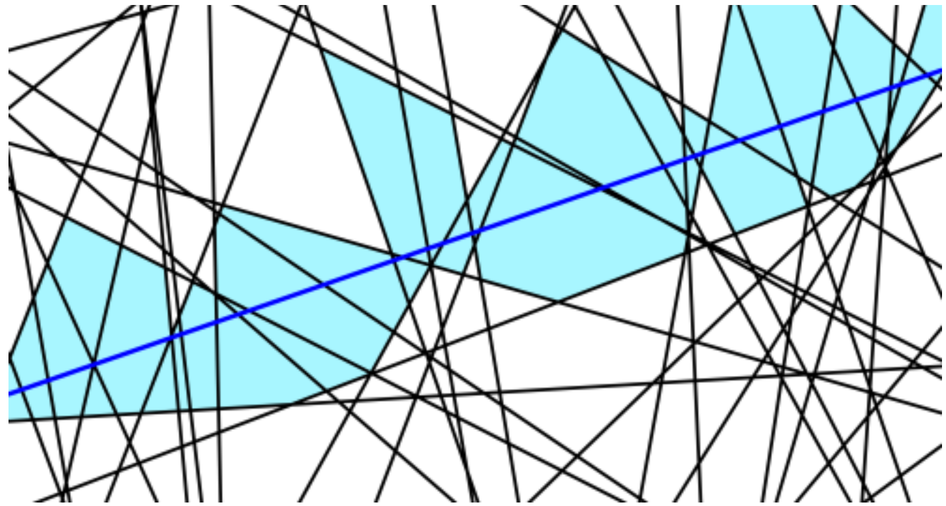
We will show this is $O(i)$



How to update after adding line ℓ_i to the line arrangement:

To bound the run time we need the Zone Theorem





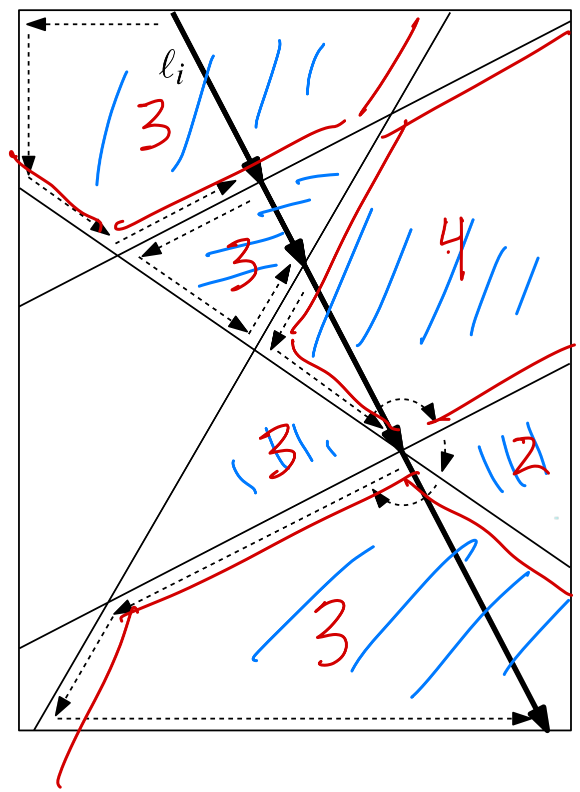
the zone of the blue line

David Dumas

Definitions. Let *A* be an arrangment, and ℓ be a line not in A.
The ***zone*** of ℓ in arrangement *A* is $Z_A(\ell)$ = {faces of A cut by ℓ } .
The ***size*** of the zone is $z_A(\ell) = \sum$ { # edges in face f : f $\in Z_A(\ell)$)
z_n = max{$z_A(\ell)$: over all possible ℓ, A of n lines }

Zone Theorem. z_n is O(n).

Example



zone of ℓ_i has 6 faces

$z(\ell_i) = 3+3+4+3+2+3$
$= 18$

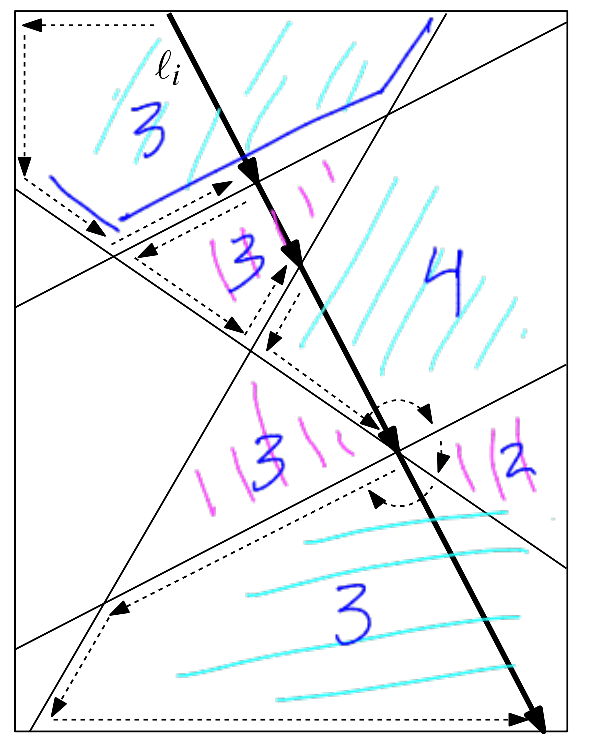
Zone Theorem. z_n is O(n). (For non-degenerate case, $z_n \le 6n$.)

[Chazelle, Guibas, Lee, '85, Edelsbrunner, O'Rourke, Seidel, '86, correct proof for dimensions ≥ 3, Edelsbrunner, Seidel, Sharir, '93]

Consequence: the incremental algorithm takes time $O(n^2)$.

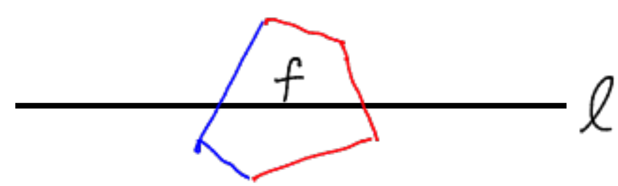
Proof

We will bound $z_A(\ell) = \sum \{ \#\text{ edges in face } f : f \in Z_A(\ell) \}$



Rotate so ℓ is horizontal. Perturb so no other line is horizontal (this only increases the zone size).

Any face f in $Z_A(\ell)$ has left and right boundary edges.



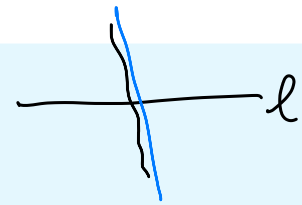
$z_A(\ell) = z(\ell) = z^L(\ell) + z^R(\ell)$

Claim. $z^L(\ell) \le 5n$ [$\le 3n$ for non-degenerate case]

This will prove the Zone Theorem.

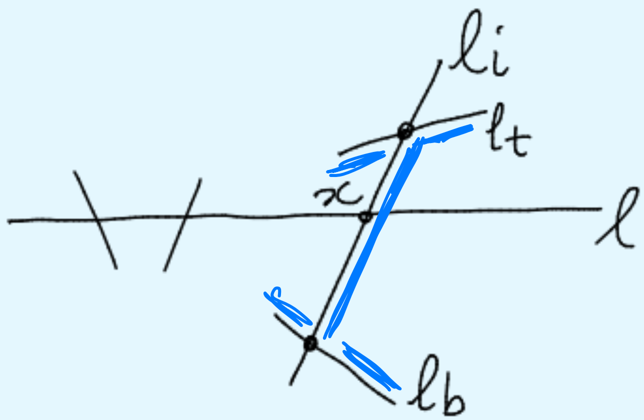
Claim. $z^L(\ell) \leq 5n$ [$\leq 3n$ for non-degenerate case]

Proof By induction. Basis $n=1$.



Zone has 2 faces. 1 left edge

General case:



Find rightmost intersection x on line ℓ — say line ℓ_i

We will apply ind. to lines $-\ell_i$

Find first intersection on ℓ_i above /below x — call ℓ_t, ℓ_b

Note: ℓ_t has + slope, ℓ_b has − slope because x was rightmost.

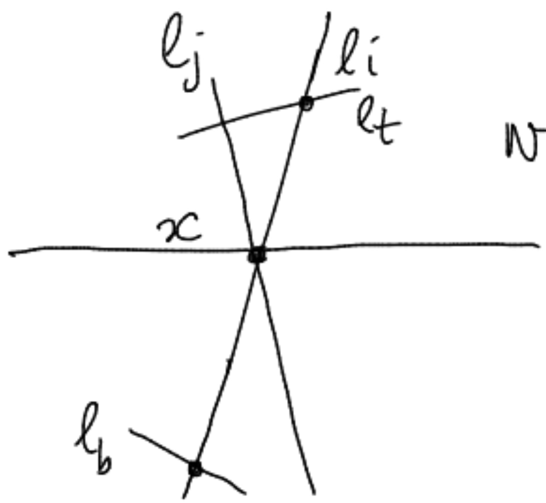
How many new left edges by adding ℓ_i?

- one edge on ℓ_i
- left edge on ℓ_t gets split in 2
- " " " " ℓ_b " " " "

Total 3 new left edges. By ind. $\leq 3(n-1)+3 = 3n$

ignoring degeneracy.

What happens in case there is degeneracy?



what if another line ℓ_j goes though x?

- still split ℓ_t and ℓ_b.
- but ℓ_i gets 2 left edges
- and ℓ_j gets split.

change in # left edges: $+5$ *

by induction # left edges $\leq 5(n-1)+5 = 5n$. $\boxtimes$

* and note that increase is less if more lines go through x.

Arrangements in higher dimensions

For an arrangement of n hyperplanes in R^d

- the number of cells is $O(n^d)$
- Zone Theorem. The zone of a hyperplane has complexity $O(n^{d-1})$

In 3D, for n planes, there are $O(n^3)$ cells, and a zone has complexity $O(n^2)$.

Application: Aspect Graph

What are all the combinatorially distinct viewpoints of an object?

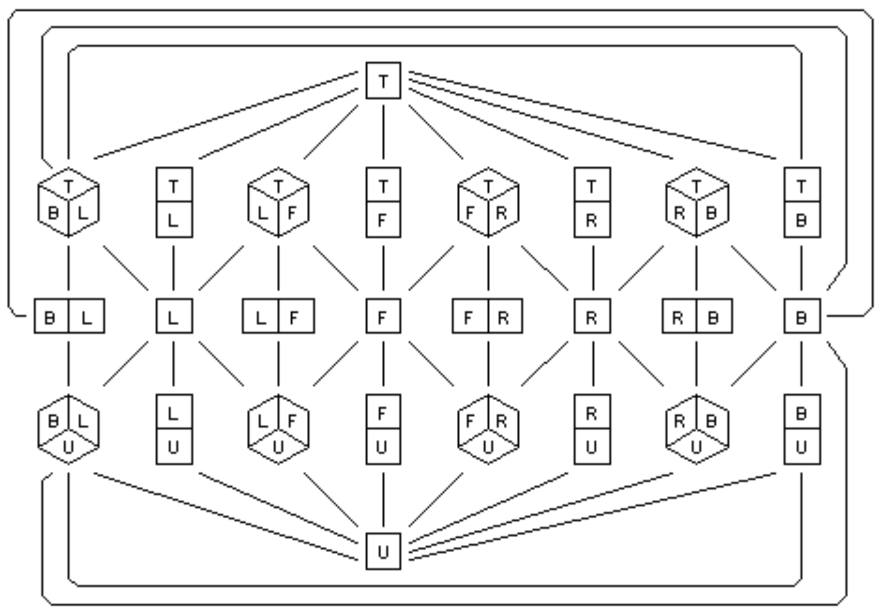
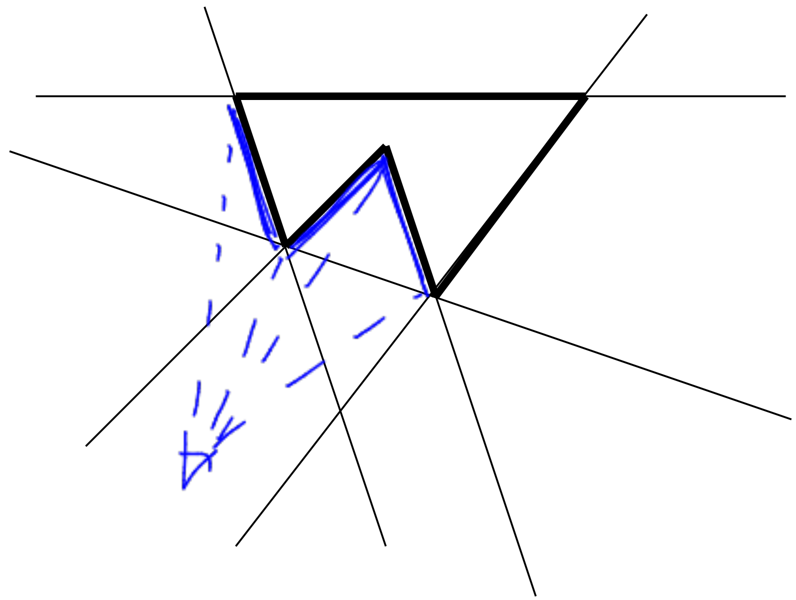


Figure 7: Aspect graph of a cube. The front, left, right, back, top and under sides of the cube are denoted by the letters F, L, R, B, T and U respectively.

http://im-possible.info/english/articles/animation/animation.html

Application: Aspect Graph



aspect graph of a polygon

area with same viewpoint = cell in arrangement of lines through pairs of visible points

n^2 lines, so n^4 cells

Application: Aspect Graph

Aspect graph of convex polyhedron with n vertices

O(n) faces $\Rightarrow$ $O(n^3)$ cells

for non-convex polyhedron Theta(n^9) cells

we need planes through every 3 points (in worst case)
so $O(n^3)$ planes $\Rightarrow$ $O(n^9)$ cells

The aspect graph can be used to:

- find the viewpoint seeing the maximum number of faces

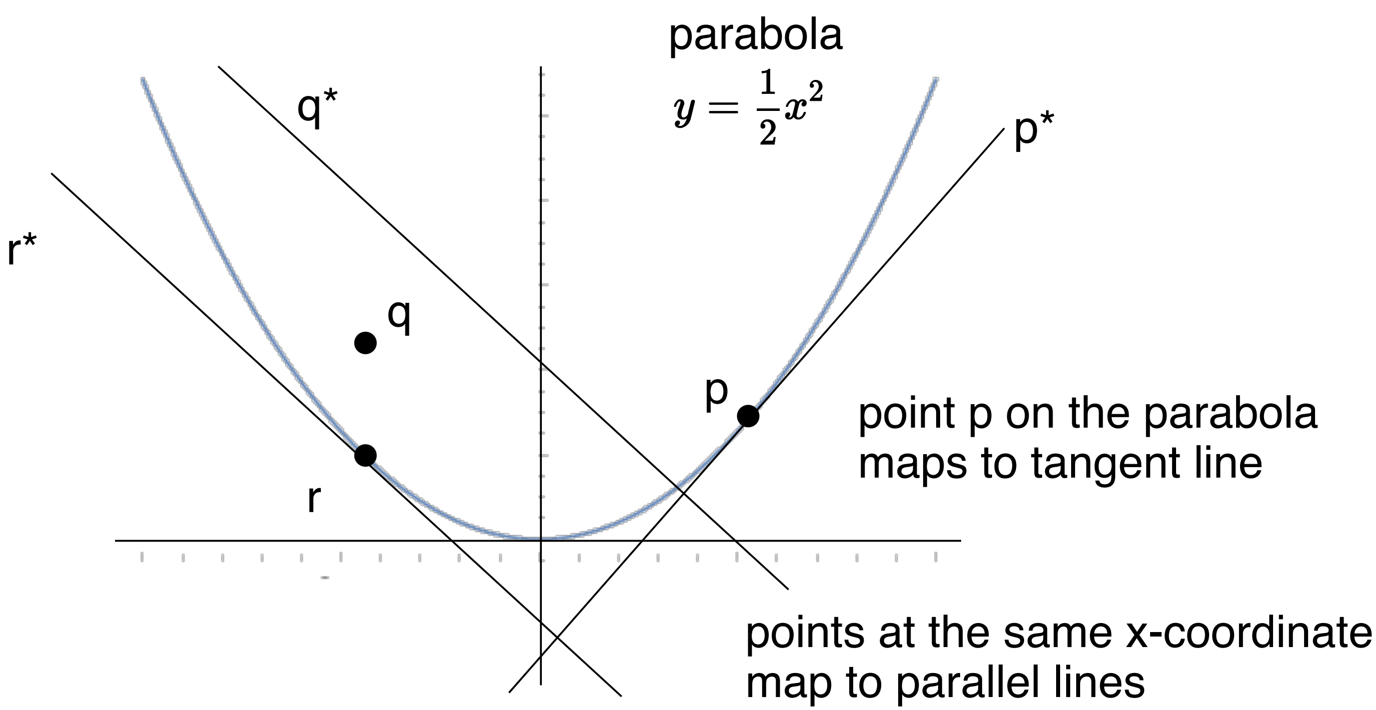
- find a “nice” projection

- figure out where a robot is, based on what it sees.

Recall Duality Map

point in $\mathbb{R}^2$ $\overset{*}{\longleftrightarrow}$ line in $\mathbb{R}^2$

$p = (p_1, p_2)$ $\longleftrightarrow$ $y = p_1 x - p_2$ (not defined for vertical lines)



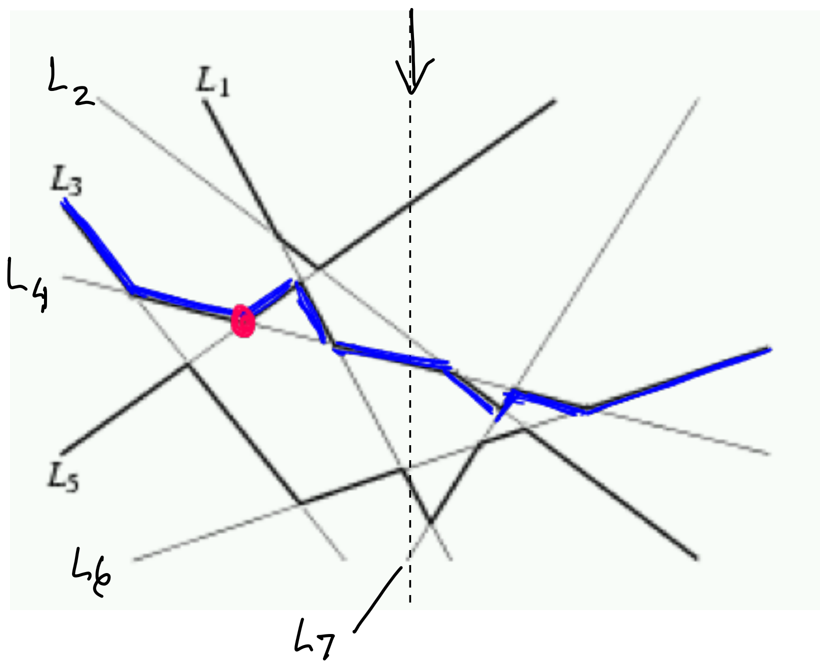
Lemma. point p lies on/above/below line q* iff point q lies on/above/below line p*.
Corollary. Some points lie on a line iff their dual lines go through a point.

Application: Collinear points.

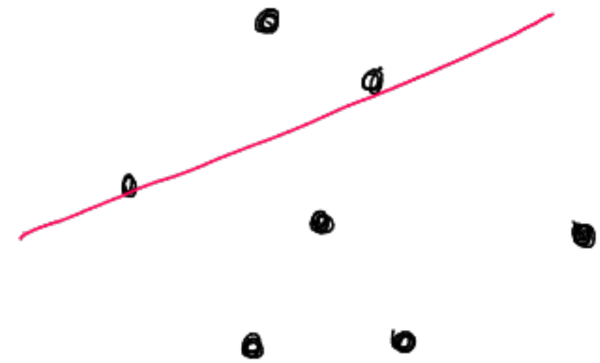
Given n points, are there 3 (or more) collinear?

Solution. Apply duality. There are 3 collinear points iff the dual has 3 lines through a point. Construct the arrangement, and check for this. $O(n^2)$

Levels in an arrangement



dual



Any vertical line not through vertices orders the edges top to bottom.

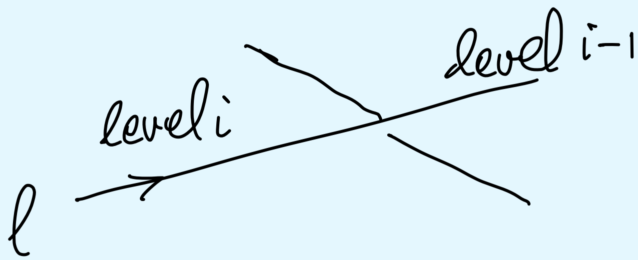
Level L_1 = all edges that appear first (topmost) along such a vertical line L_1 L_2 L_5 L_7
Level L_i = all edges that appear in i-th place along such a line

Claim. Levels can be constructed in $O(n^2)$ time.

Claim. Levels can be constructed in $O(n^2)$ time.

Sort lines by slope

Consider line ℓ — we have level of ℓ at far left (= rank of slope)



Trace ℓ through the arrangement, updating level at each vertex.

Levels in an arrangement

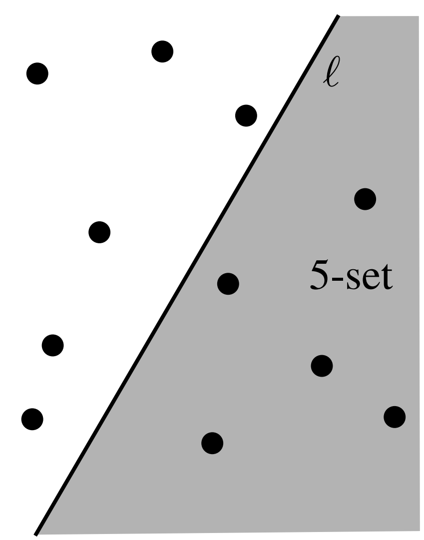
Open problem: what is the complexity of level L_k?
i.e. what is the worst case number of edges in level L_k?

Dual: given a set of points, how many subsets of size k can be cut away with a line?
For k = n/2, how many *halving lines* can there be?

Best known bounds:

$\Omega(n \log k)$ 1973, Erdös et al., raised a bit by Toth, 2001

$O(nk^{1/3})$ Tamal Dey, 1997

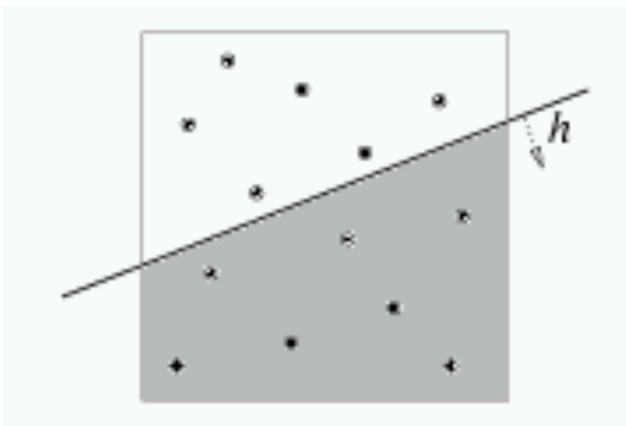


Also: find level L_k (without constructing whole arrangement)

Application: Discrepancy problem.

Given n points in a unit square, do they provide a reasonable random sample?

discrepancy of half-plane h = | area of square below h – fraction of points below h |



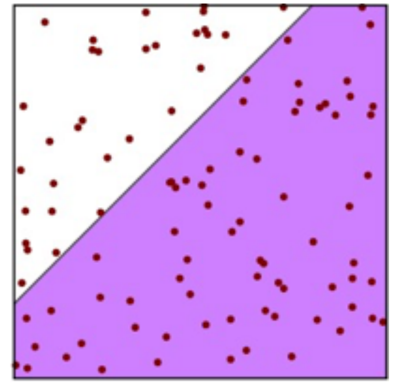
example:
7 points in shaded area; 13 points total
so fraction of points below h is 7/13

Given n points, find the maximum discrepancy of any half-plane.

Arrangements give an $O(n^2)$ time algorithm for this.

nice presentation: http://www.ams.org/samplings/feature-column/fc-2011-12

Application: Discrepancy problem.

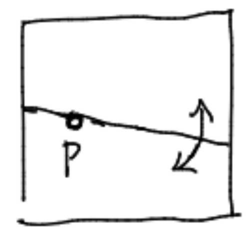


Lemma. Maximum discrepancy occurs

1. at line h through 2 points, or
2. at line h through 1 point and the point is the midpoint of the segment h ∩ unit square

Proof.
If h goes through no points, we can slide it up or down to increase discrepancy.

If h goes through 1 point p, we can rotate it to increase discrepancy unless p is midpoint.



Solving the discrepancy problem via arrangements.

Lines of type 2 can be checked brute force — each point can be the midpoint of only 2 segments; for each one, just count the points below. $O(n^2)$

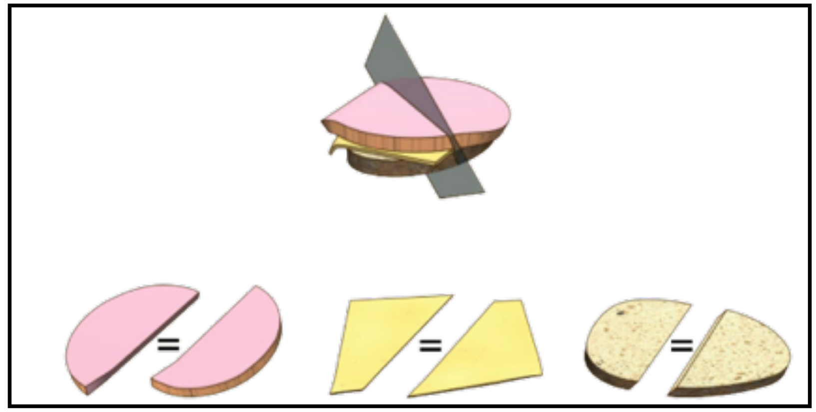
For lines of type 1, use the dual arrangement.
Line h through 2 points corresponds to point h* on 2 lines
points below h corresponds to # lines below h* = n – level of h*

So test all n^2 points of the dual arrangement, O(1) per test.
Total is $O(n^2)$.

Application: Ham sandwich theorem.

Theorem. Given a set of red points and a set of blue points in the plane, there exists a line that cuts both sets in half.

General Ham-Sandwich Theorem
(from ’30’s - 40’s).
In R^d, any d measurable objects can be cut in half by one (d-1) dimensional hyperplane.



For discrete version in plane, there is an O(n) time algorithm to find the halving line.

Can be viewed in terms of arrangements.

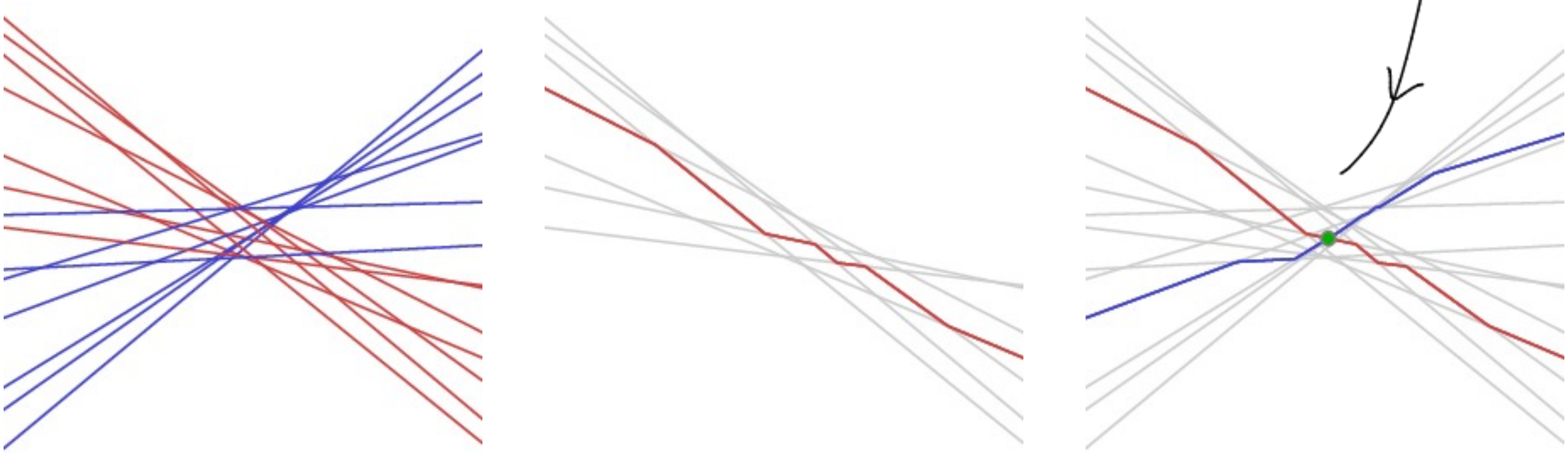
Application: Ham sandwich theorem.

Theorem. Given a set of red points and a set of blue points in the plane, there exists a line that cuts both sets in half.

Standard proof idea uses a rotating line.

In terms of arrangements:

the two $\frac{n}{2}$-levels intersect at a point $\longleftrightarrow$ halving line.



the n/2 level of red lines

David Austin

3-SUM hardness https://en.wikipedia.org/wiki/3SUM

Can we test for 3 collinear points (for point set in the plane) faster than $O(n^2)$?

It is a "3-SUM-hard" problem, one of a large class of "equivalent" problems that all seem(ed) to need $O(n^2)$ time.

3-SUM problem: Given n numbers, are there 3 that sum to 0? (repetition is allowed)

Exercise. Find an $O(n^2)$ time algorithm for 3-SUM.
[This is not too hard. Start by sorting the points.]

Lemma. If we could test for 3 collinear points in $o(n^2)$, then we could solve 3-SUM in $o(n^2)$.

Proof. Given n numbers as input to 3-SUM, map each number x to the point (x,x^3).
Claim. 3 numbers a,b,c sum to 0 iff the corresponding points are collinear.

points are collinear iff slope (a,a^3) to (b,b^3) = slope (b,b^3) to (c,c^3)

iff $\dfrac{b^3-a^3}{b-a} = \dfrac{c^3-b^3}{c-b}$ iff $b^2 + a^2 + ab = c^2 + b^2 + cb$

iff $b(a-c) = c^2 - a^2$ iff $b = \dfrac{c^2-a^2}{a-c}$ iff $b = -(a+c)$ iff $a+b+c=0$

recent breakthrough on 3-SUM:

an algorithm with run time $O(n^2/(\log n/\log\log n)^{2/3})$

Grønlund, Allan, and Seth Pettie. "Threesomes, degenerates, and love triangles." Journal of the ACM, 2018. (conference version 2014)

d https://doi.org/10.1145/3185378

improved by Timothy Chan

Summary

- arrangements
- size of parts of arrangements and the zone theorem
- applications of arrangements
- testing collinearity and the 3-SUM problem.

References

- [CGAA] Chapter 8
- [Zurich notes] Chapter 8