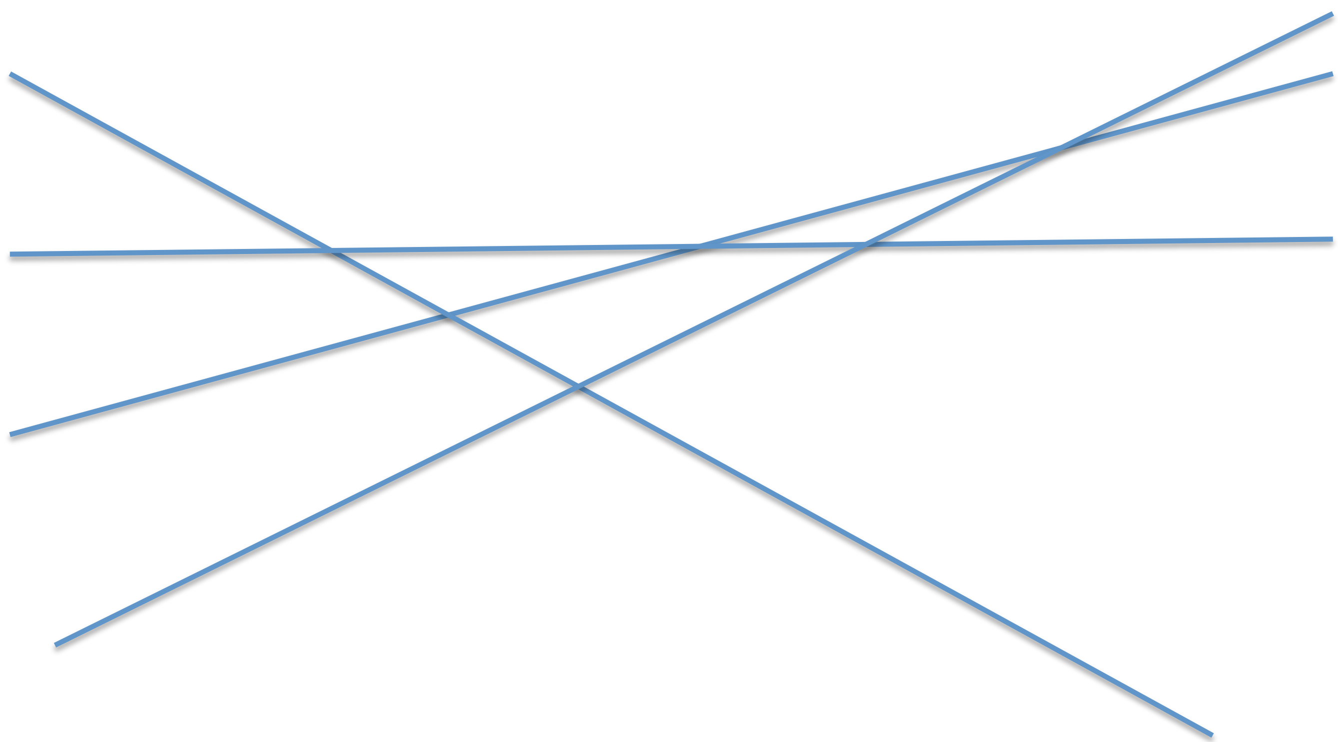
Line Arrangement

Chapter 6

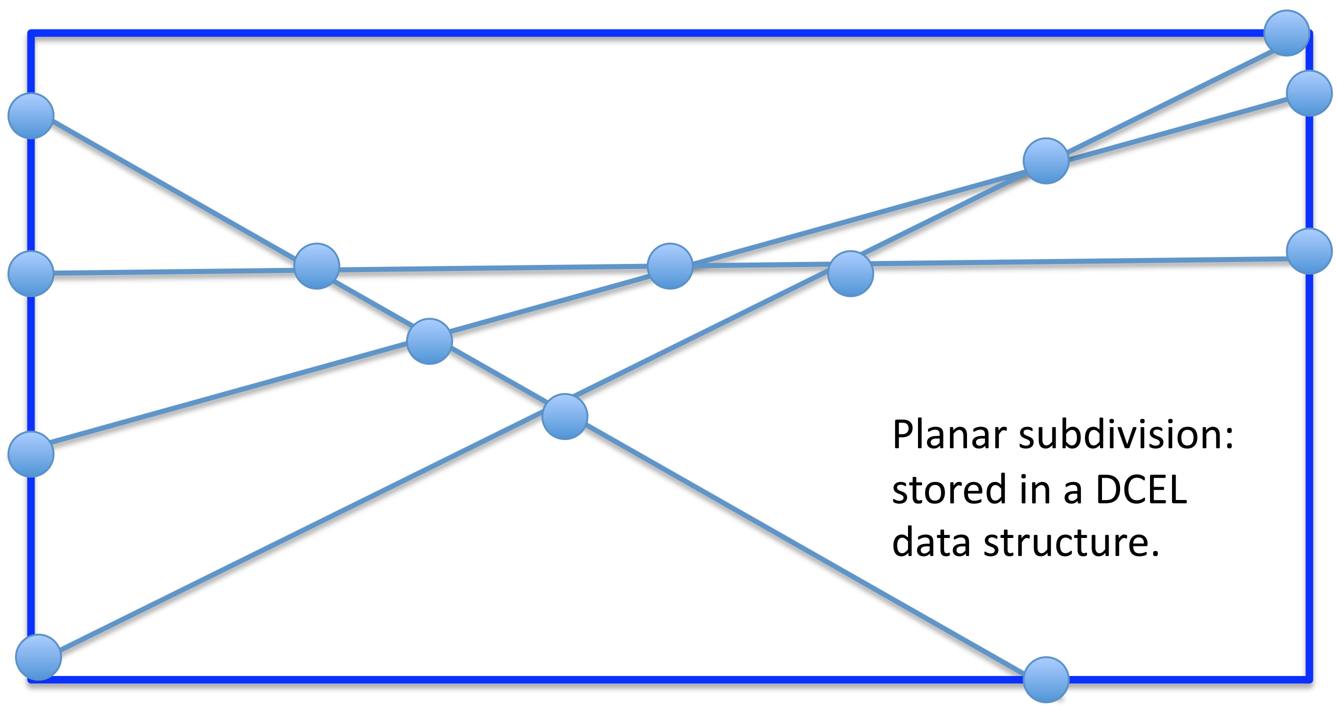
Line Arrangement

Problem: Given a set L of n lines in the plane, compute their arrangement which is a planar subdivision.



Line Arrangements

Problem: Given a set L of n lines in the plane, compute their arrangement which is a planar subdivision.



- Theorem: The complexity of the arrangement of n lines is $\Theta(n^2)$ in the worst case (non-degenerate situation)
  - Number of vertices : $\Theta(n^2)$ (n-1 vertices on each line; total=n(n-1)/2; each vertex is counted twice)
  - Number of edges : n^2 (n edges on each line)
  - Number of faces : $\Theta(n^2)$ (follows from Euler formula : # faces - # edges + # vertices = 2)
- In degenerate situation when all lines pass through a single point (number of vertices = 1), the number of edges and faces are linear in n.

Line Arrangement

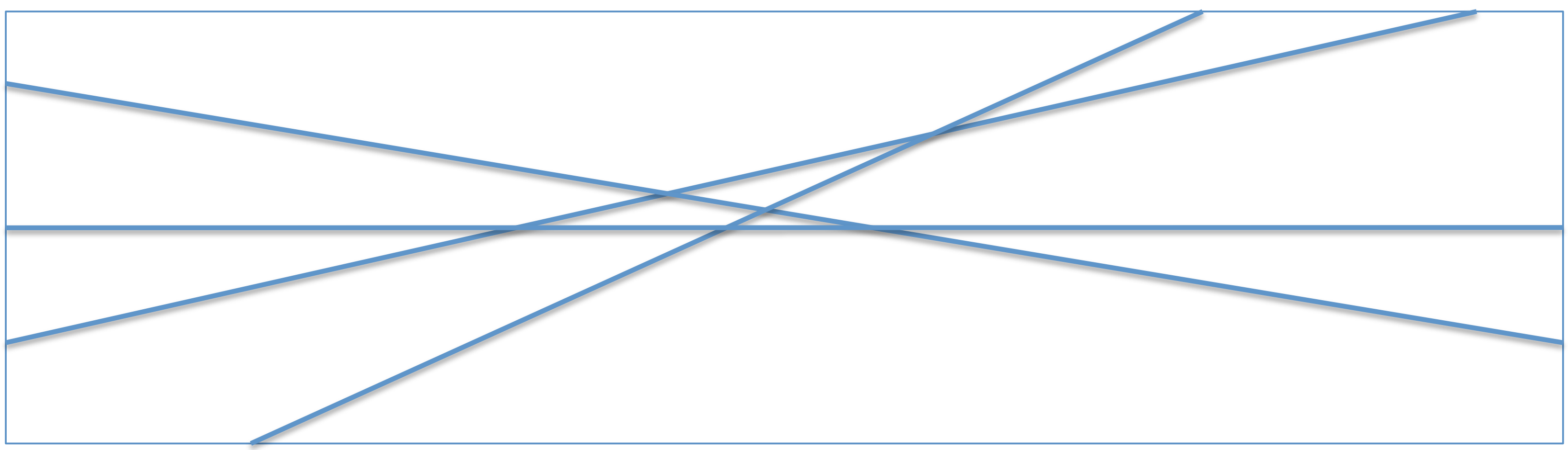
- Goal: compute this planar map (as a DCEL)
- Algorithm: Use an incremental algorithm:

  (add one line at a time and update the DCEL structure)

  We will construct the arrangement inside a rectangular box.

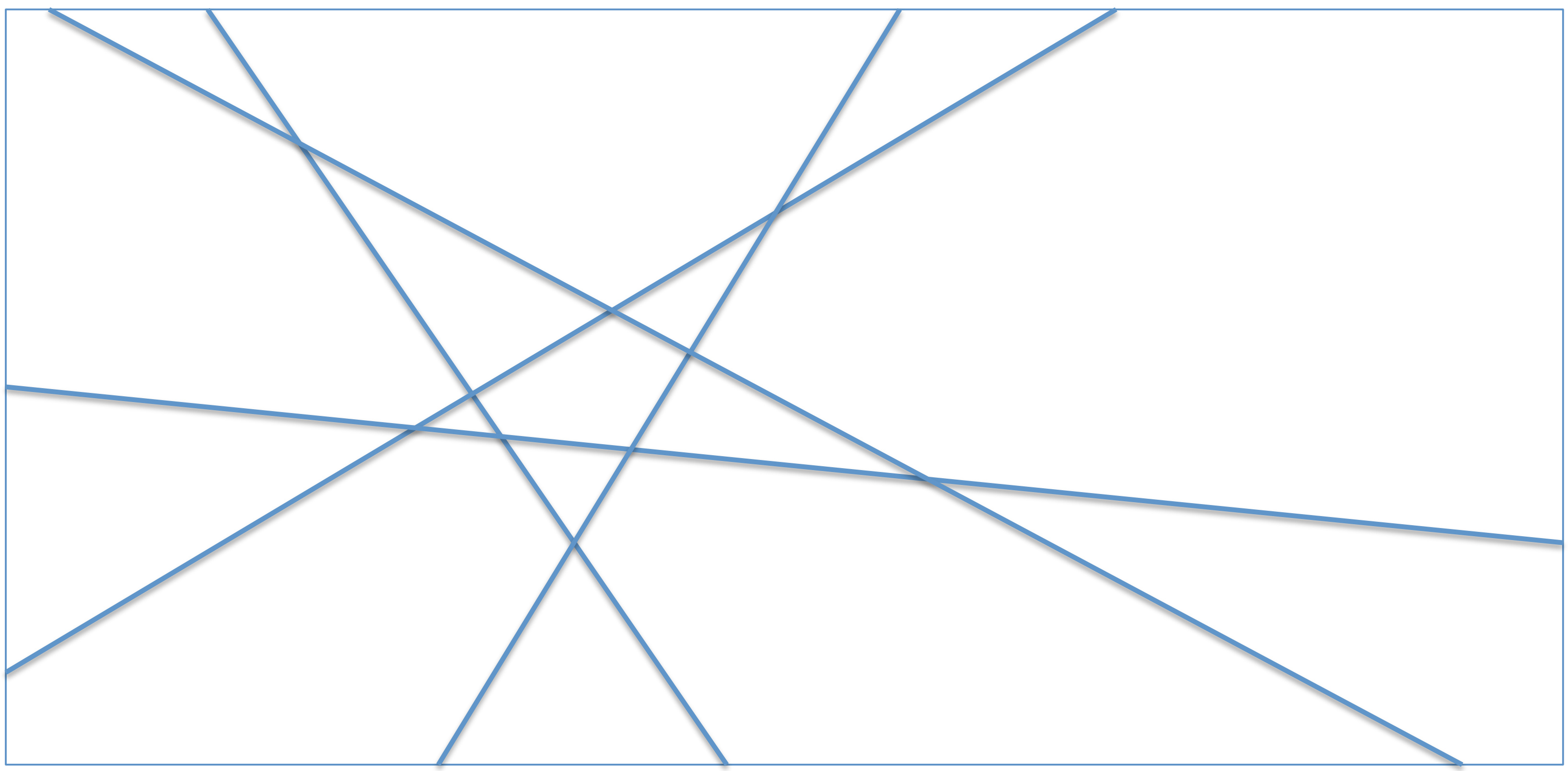
An incremental algorithm:

- Input: A set L of n lines in the plane and a bounding box B.
- Output: The DCEL structure of the arrangement A(L) inside a bounding box.



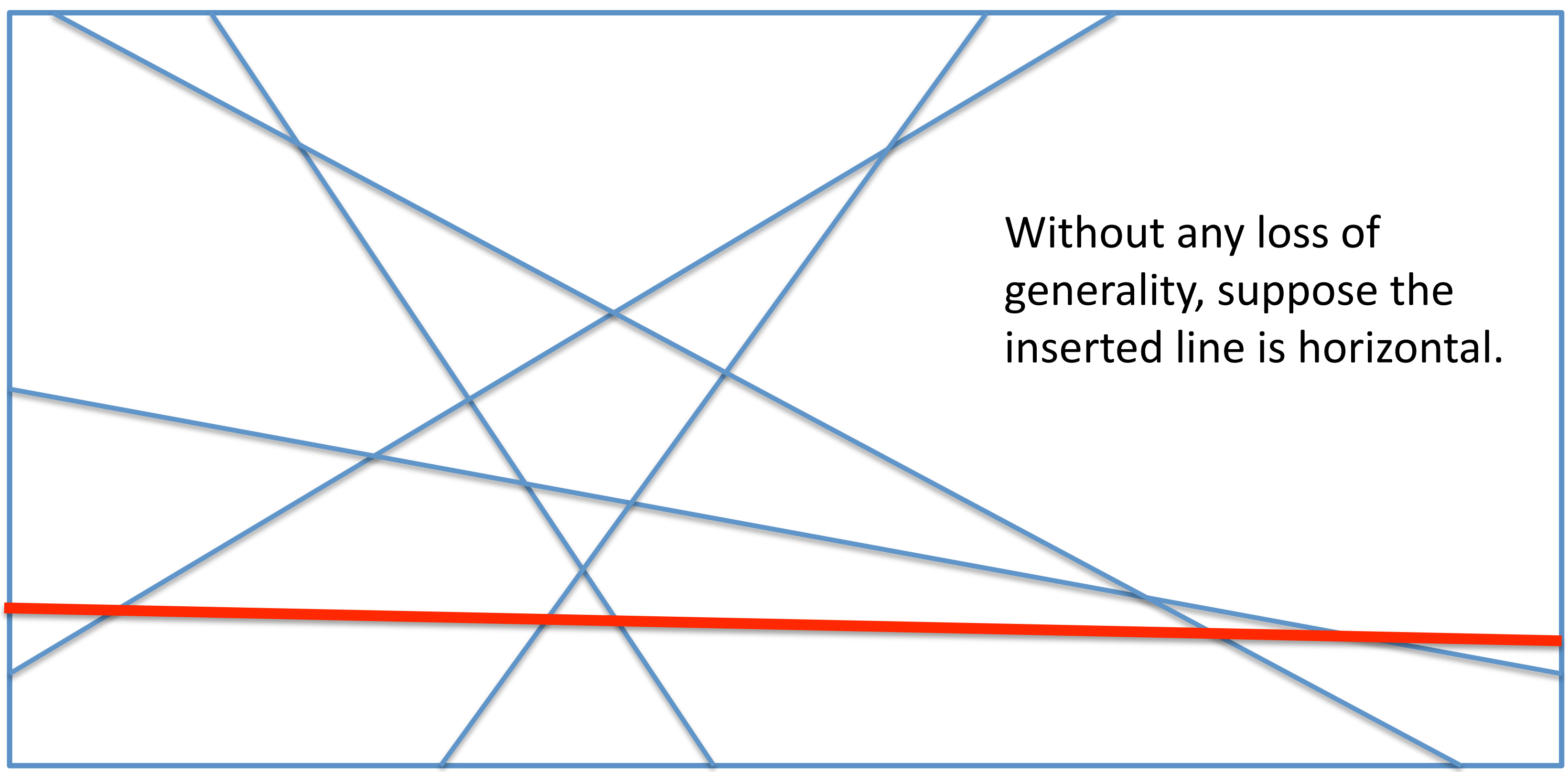
What happens when a line is added?

- Consider the arrangement of i lines



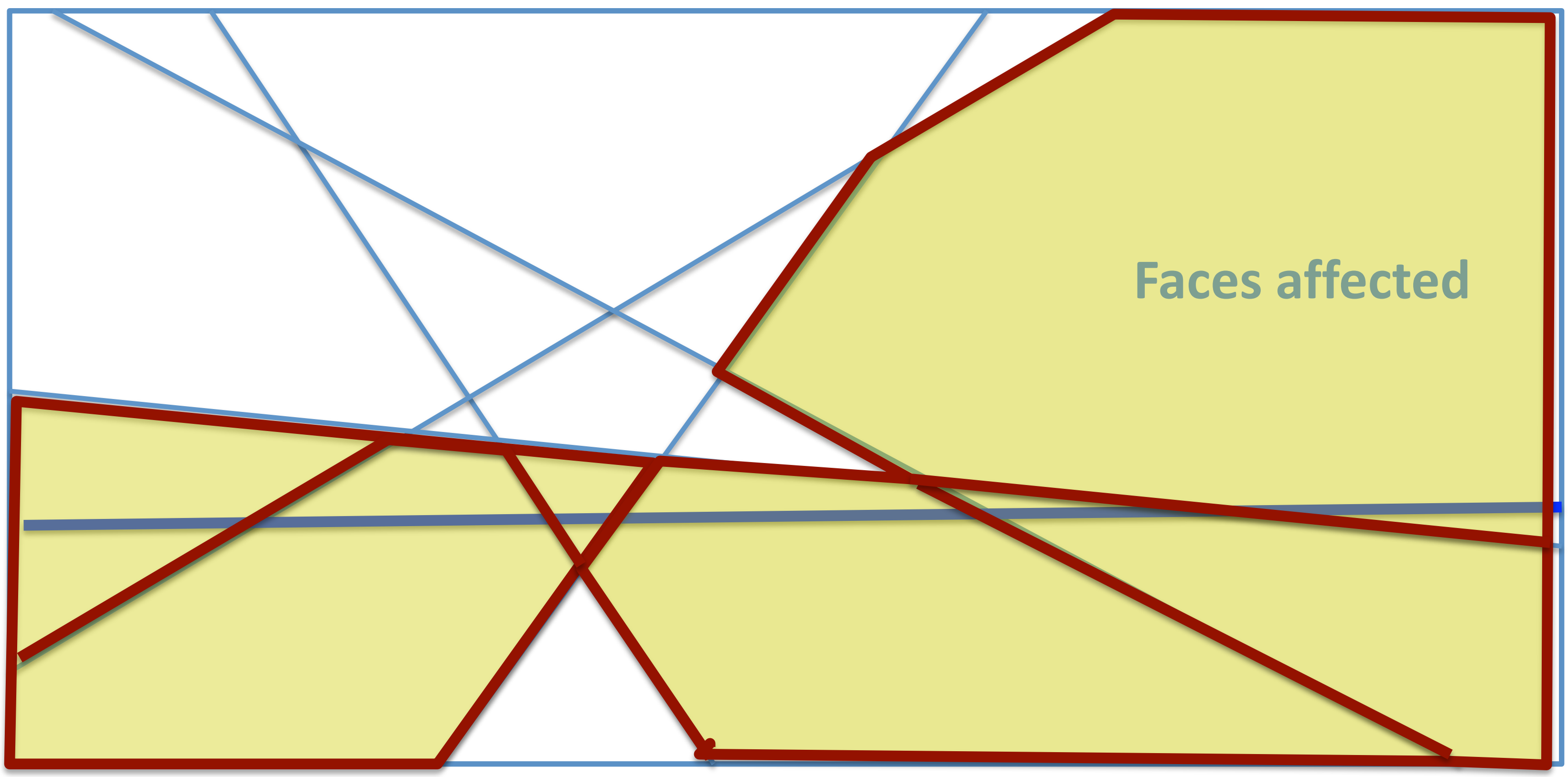
What happens when a line is added?

- Consider the arrangement of first i lines.
- We now insert the $(i+1)^{th}$ line.



What happens when a line is added?

- Consider the arrangement of first i lines.
- We now insert the $(i+1)^{th}$ line.

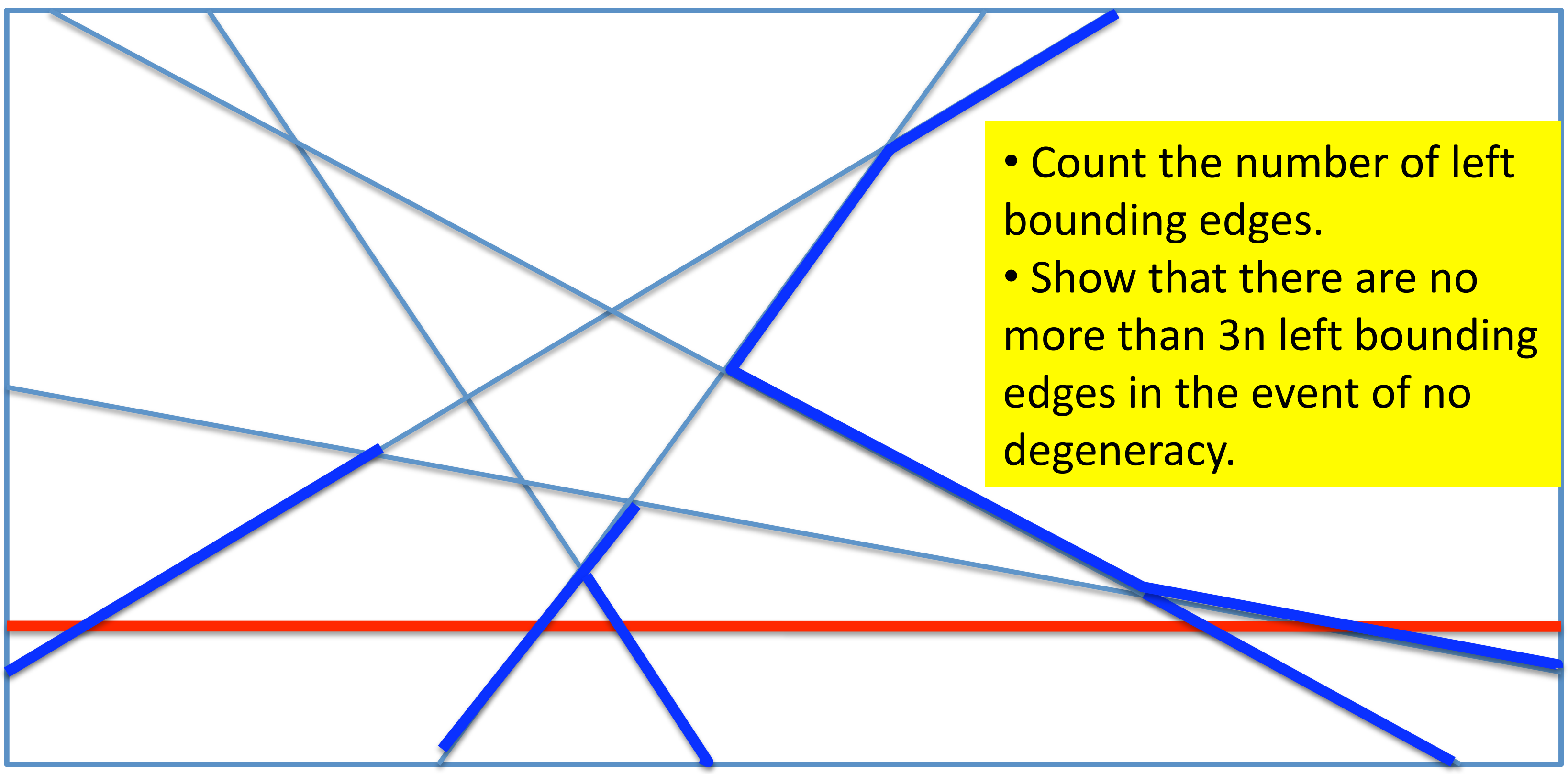


Zone Theorem

- Zone of a line l: The zone of a line l in an arrangement A(L) is the set of faces of A(L) whose closure intersects l .
- The complexity of a zone (z_n) of A(L) is the total complexity of all the faces: the total sum of edges (or vertices) of these faces.
- Theorem: $z_n \leq 6n$ where $|L| = n$.

What happens when a line is added?

- Consider the arrangement of first i lines
- We now insert the $(i+1)^{th}$ line.



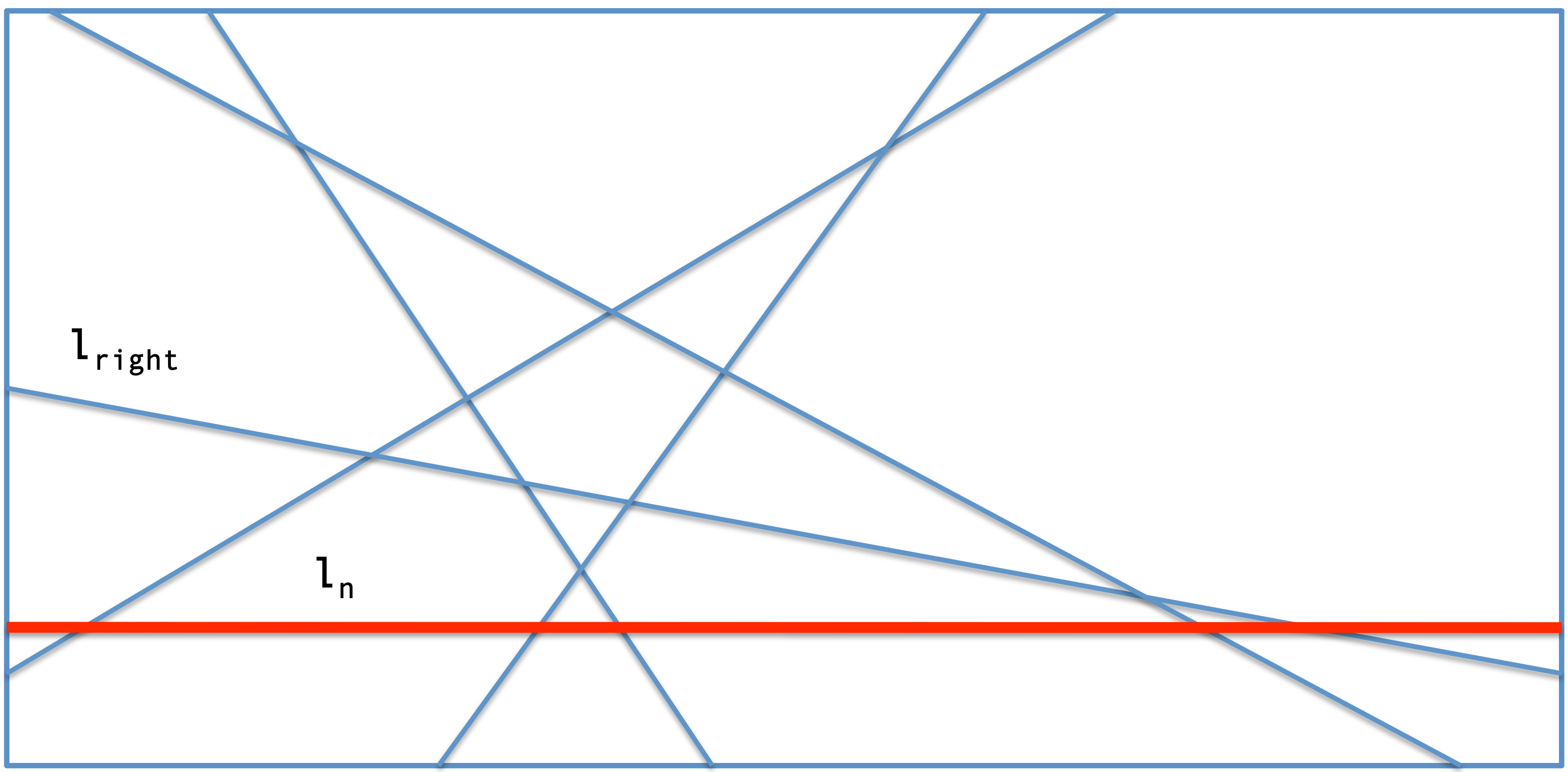
Zone Theorem (left bounding edges):

Theorem: The number of left bounding edges in the zone of a line in A(L) is at most 3n.

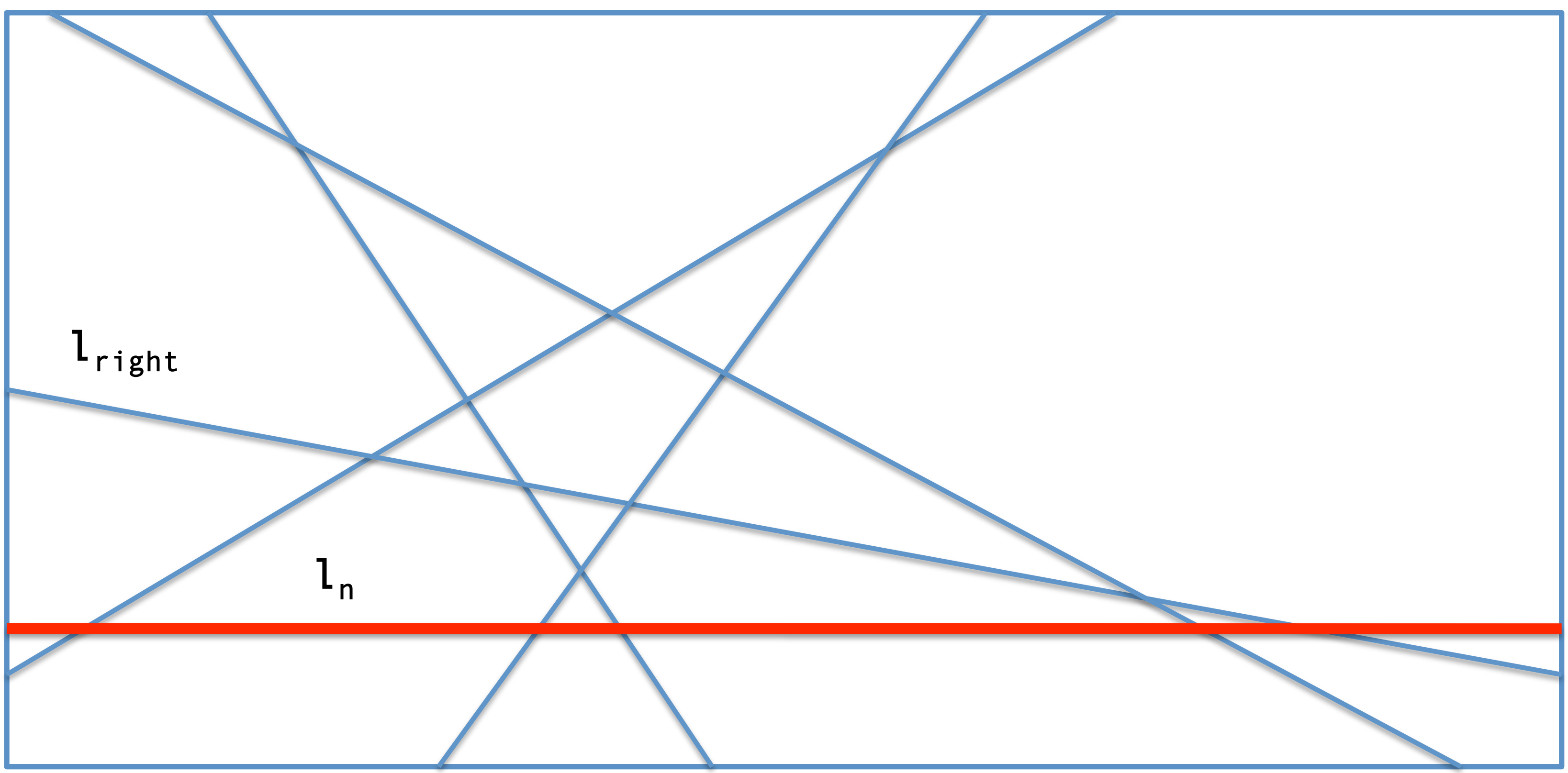
Zone Complexity: Proof (no degeneracy is assumed, i.e. no three lines are concurrent)

- By induction on n; for n=1, it is trivial.
- Suppose the zone complexity is true for any arrangement of m lines, m < n.
- For any n > 1:
  - Let l_{right} be the rightmost line intersecting l_n, the line being inserted. Without any loss of generality we assume that l_n is horizontal. We now remove the line l_{right}.
  - By the induction hypothesis, the zone of l_n in A(L-$\{l_{right}\}$) has at most 3(n-1) left bounding edges.
  - When adding l_{right} back, the number of left bounding edges in the zone of l_n increases as follows:
    - One new left bounding edge on l_{right}.
    - At most two old left bounding edges get split by l_{right}.
  - The zone complexity of l_n is at most $3(n-1)+3 \leq 3n$.
  - The theorem follows from the principle of mathematical induction.

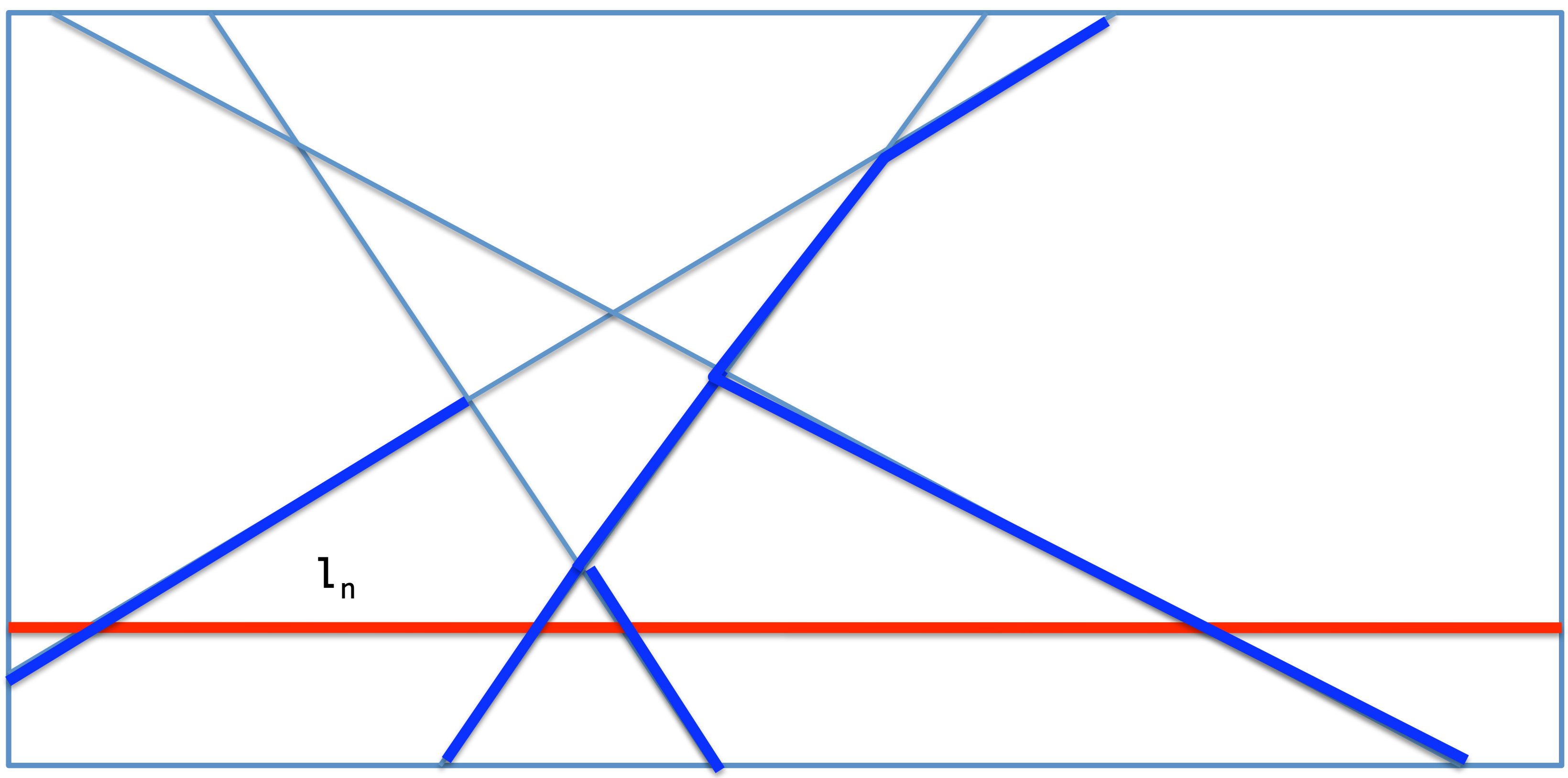
l_{right} is the line with the rightmost intersection with l_n



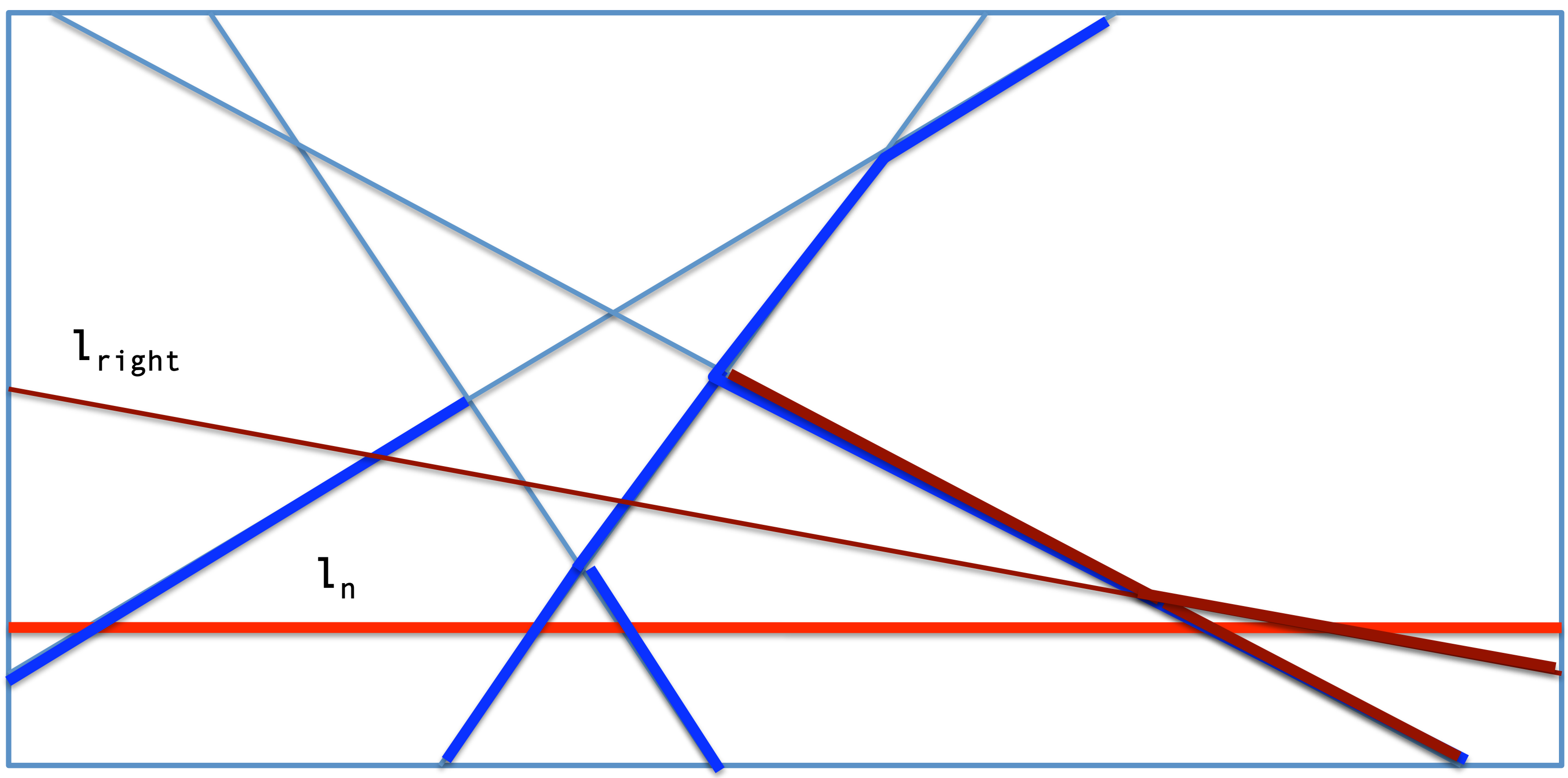
Remove l_{right}



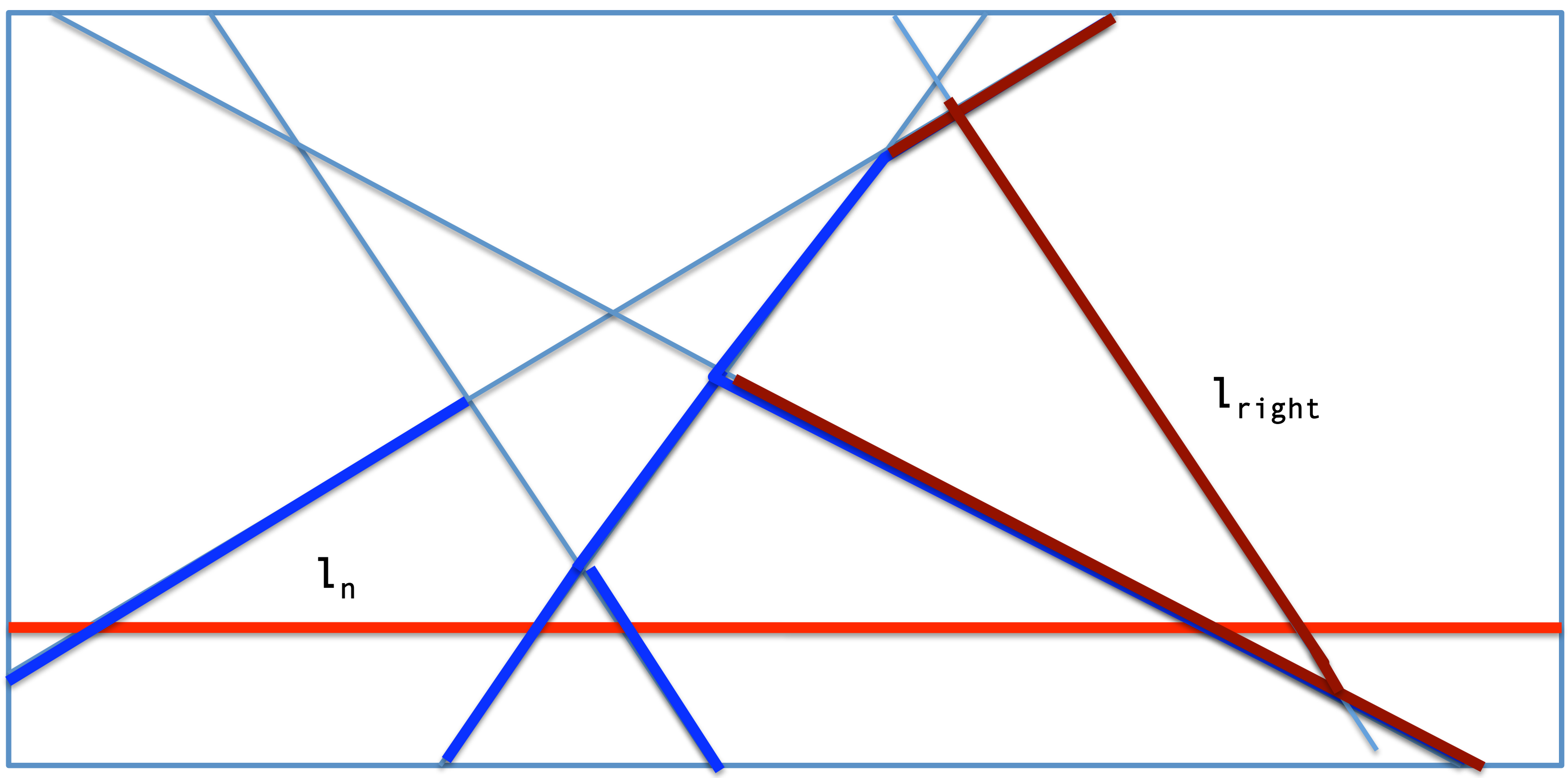
All left bounding edges in the zone of l_n in A (L-{l_{right}}) is highlighted



Adding l_{right} introduces two extra left bounding edges in this case



Adding l_{right} introduces ~~two~~ (three) extra left bounding edges



Zone Theorem (right bounding edges):

- Similarly we can show that

Theorem: The number of right bounding edges in the zone of a line in A(L) is at most 3n.

Constructing the Arrangement

- The time insert the (i+1)th line is linear in the complexity of the zone, which is linear in the number of existing lines (i.e. i). Therefore, the total running time of the incremental algorithm is

$$O(n^2) \quad + \quad \sum_{i=1}^{n}(O(\log i) + O(i)) \quad = \quad O(n^2)$$

Finding a bounding box	Finding the left entry point	According to the zone theorem

Note: Bound doesn't depend on the insertion order.

Applications of Arrangement

- **Sweeping Arrangements**
  - We know the arrangement of n lines can be constructed in $O(n^2)$ time, which is optimal.
  - For some instances, any graph traversal would suffice.
  - For other problems, we need traversal in some order, for example as a plane sweep from left to right.
    - Requires the sweep to stop at every intersection point of the lines. This implies $O(n^2 \log n)$ time algorithm requiring $O(n)$ space.
  - There is more sophisticated version of the sweep, called topological plane sweep. The plane is swept by a thread, instead of a line. This can be done in $O(n^2)$ time and $O(n)$ space. It is applicable in almost all instances where ordinary plane sweep works.

Applications of Arrangements and Duality

- Duality concept and arrangements allow a large number of problems to be solved. Unless otherwise stated, all problems can be solved in $O(n^2)$ time and $O(n^2)$ space by constructing a line arrangement, or in $O(n^2 \log n)$ time and $O(n)$ space through plane sweep. (In all instances, the extra logn factor can be removed through the use of the topological plane sweep.

Applications of Arrangements and Duality

- **General position test**: Given a set of n points in the plane, determine whether any three are collinear.
- **Minimum area triangle:** Given a set of n points in the plane, determine the minimum area triangle whose vertices are selected from these points.
- **Minimum k-corridor:** Given a set of n points, and integer k, determine the narrowest pair of parallel lines that enclose at least k points of the set. The distance between the lines can be defined either as the vertical distance between the lines or the perpendicular distance between the lines.
- **Visibility graphs non-intersecting line segments:** The vertices are the end points of the segments and two endpoints are visible if the interior of the line segment joining them intersects none of the segments.

Applications of Arrangements and Duality

- **Maximum stabbing line:** Given a set of line segments in the plane, compute the line that stabs (intersects) the maximum number of segments.
- **Hidden surface removal:** Given a set of n non-intersecting polygons in 3-space, imagine projecting these polygons onto a plane (either orthogonally or using perspective). Determine which portions of the polygons are visible from a viewpoint under this projection. In practice, the projected scene is rarely quadratic. $O(n^2)$ algorithm is really of theoretical values.
- **Ham Sandwich Cut:** Given n red points and m blue points, find a single line that simultaneously bisects these point sets. If the point sets are linearly separable, this can be done in $O(n+m)$ time and space.

Sorting all angular sequences.

- Consider a set of n points in the plane. For each point p in this set, we want to perform an angular sweep, say in counterclockwise order, visiting the other n-1 points of the set. For each point p, we can order the angles of the points around p in O(nlogn) time per point and $O(n^2 \log n)$ overall.
  - With arrangements we can speed this up to $O(n^2)$ total time.
  - A point $p=(p_x,p_y)$ and line l: (y=ax-b) in the primal plane are respectively mapped to a dual point p^* and dual line l^* where $l^* = (a,b)$ and p^*: $(b=p_x a - p_y)$.
  - Suppose p is the point around which we want to sort. Let $p_1, p_2, \ldots, p_n$ be the points in the final angular order.

Sorting all angular sequences.

- Suppose p is the point around which we want to sort. Let $p_1, p_2, \ldots, p_n$ be the points in the final angular order.
- Since the a-axis in the dual plane is the slope-axis, the intersection points of p_i^*, i= 1, 2, ...,n in increasing order along p^* realize the slope ordering around p. This ordering can be determined in O(n) time from the DCEL structure of the arrangement.

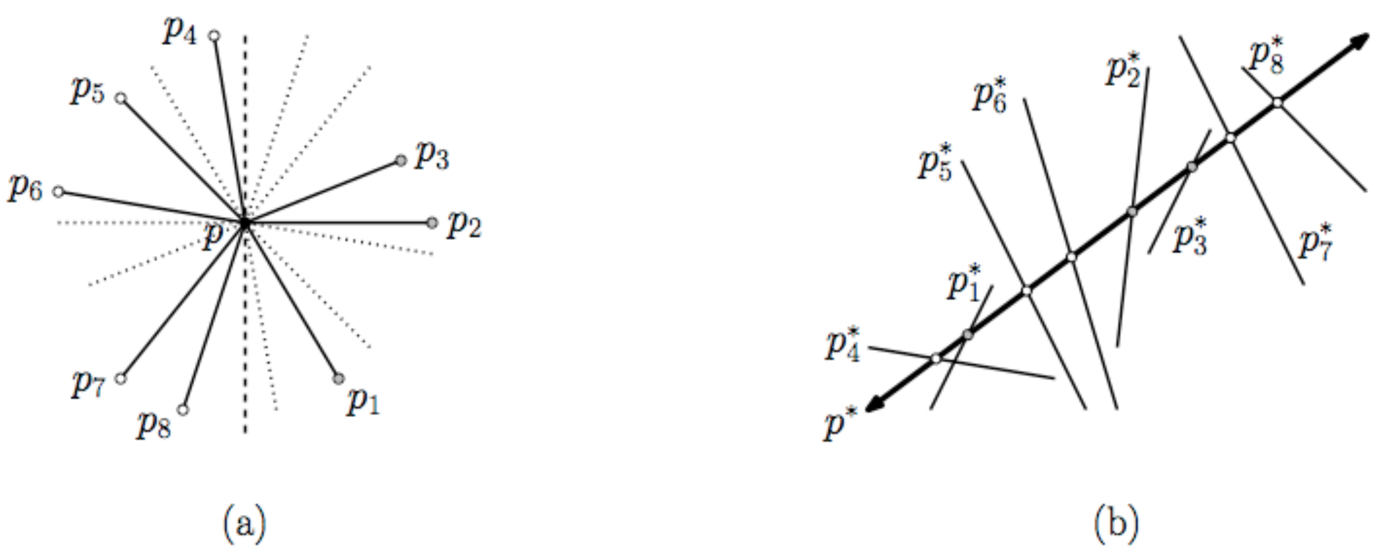


Figure 1: Arrangements and angular sequences.

Maximum Discrepancy

- we consider a problem derived from computer graphics and sampling.
- Suppose that we are given a collection of n points S lying in a unit square $U=[0,1]^2$. We want to use these points for random sampling purposes.
- In particular, the property that we would like these points to have is that for any halfplane h, we would like the size of the fraction of points of P that lie within h should be roughly equal to the area of intersection of h with U .
- If we define $\mu(h)$ to be the area of $h \cap U$, and $\mu_S(h)=|S \cap h|/|S|$ then we would like $\mu(h) = \mu_S(h)$.
- This property is important when point sets are used for things like sampling and Monte-Carlo integration.

Maximum Discrepancy

- We define the discrepancy of S with respect to a halfplane h to be $\Delta_S(h) = |\mu(h) - \mu_S(h)|$.
- In the following figure, $\mu_S(h) = 7/13 = 0.538$; $\mu(h) = 0.625$
- Halfplane discrepancy of S is defined to be the maximum (least upper bound) of $\Delta_S(*)$ over all halfplanes. This is denoted by $\Delta(S)$.

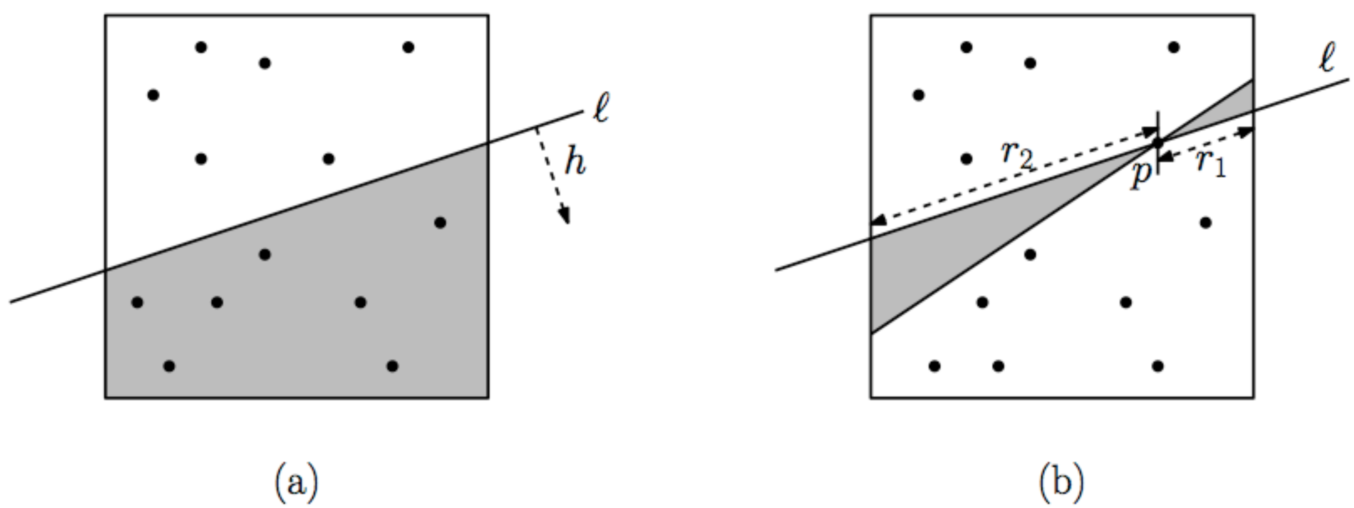


Figure 4: Discrepancy of a point set.

Maximum Discrepancy

- Finiteness criterion

  Lemma Let h denote the halfplane that generates the maximum discrepancy with respect to S , and let denote the line l that bounds h. Then either (i) l passes through at least two points of S , or (ii) l passes through one point of S , and this point is the midpoint of the line segment l ∩ U.

  – **Remark:** If a line passes through one or more points of S , then should this point be included in μ_S (h)? For the purposes of computing the maximum discrepancy, the answer is to either include or omit the point, whichever will generate the larger discrepancy. The justification is that it is possible to perturb h infinitesimally so that it includes none or all of these points without altering μ(h).

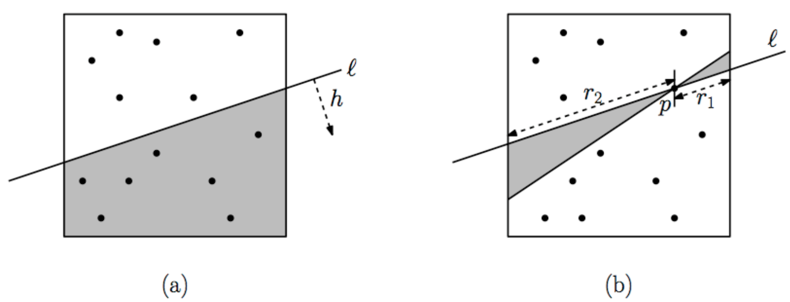


Figure 4: Discrepancy of a point set.

Maximum Discrepancy

- h is determined by two points.
  - There are $O(n^2)$ such pairs determining halfplanes. These can be picked from the arrangement easily. Moreover, the number of points lying on both halfplanes can be determined from the arrangment of the dual lines as well. This can be done in $O(n^2 \log n)$ time and $O(n)$ space using the sweep method, or $O(n^2)$ time and linear space using the topological sweep.
- h is determined by one point.
  - Every point p realizes at most two placements. There are $O(n)$ such halfplanes, and the discrepancy of each such halfplane can be determined in $O(n)$ time.

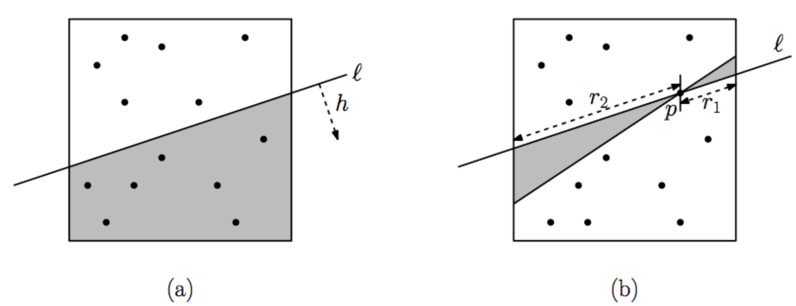
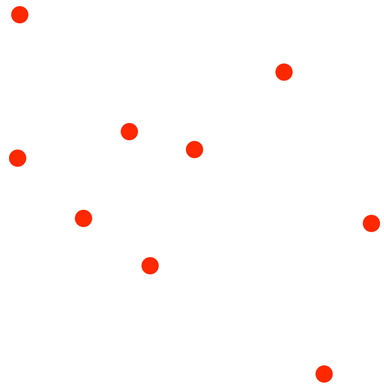


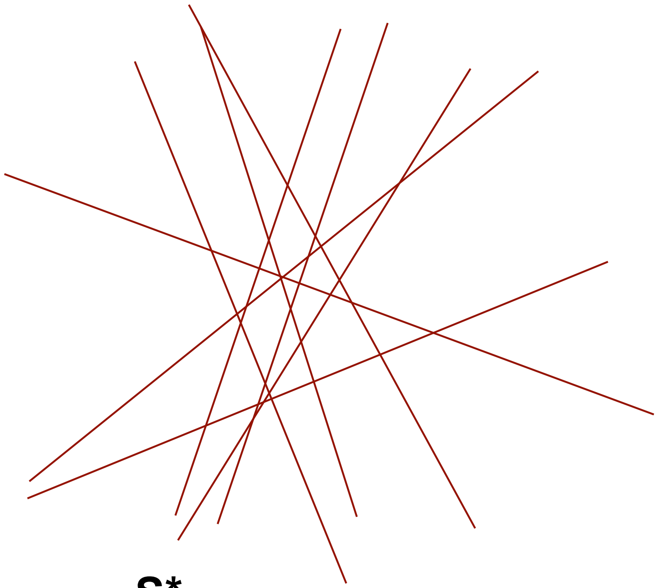
Figure 4: Discrepancy of a point set.

Median Level of an Arrangement

- Consider a set S of n points
- Without any loss of generality suppose n is odd



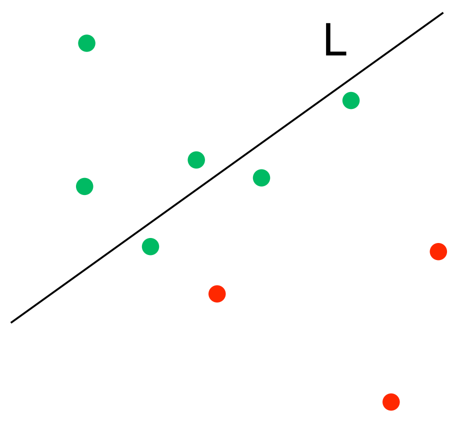
S



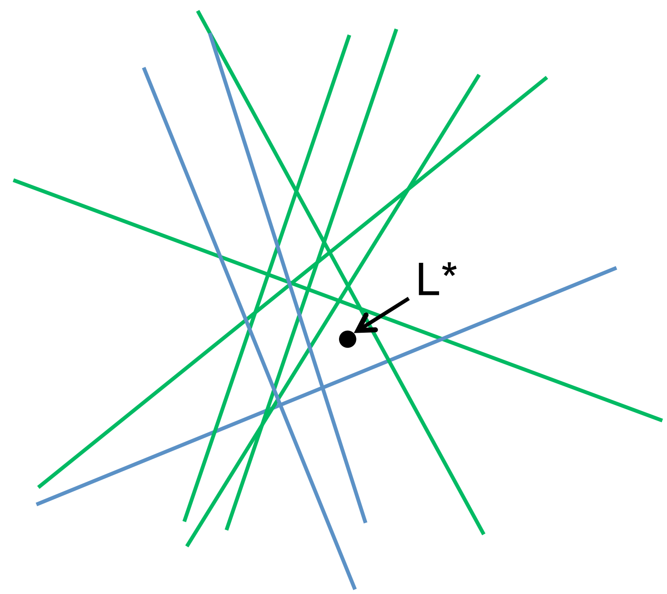
S*

Median Level of an Arrangement

- Consider a set S of n points
- Without any loss of generality suppose n is odd



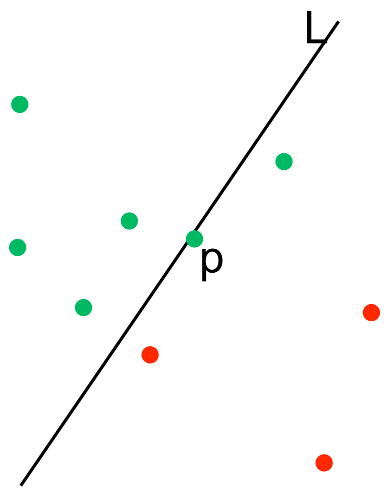
3points lie above L



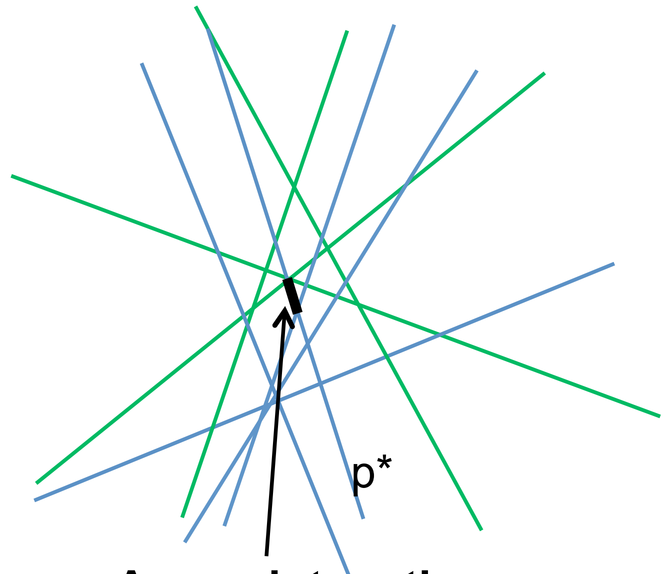
3 lines lie below L*

Median Level of an Arrangement

- Consider a set S of n points
- Without any loss of generality suppose n is odd



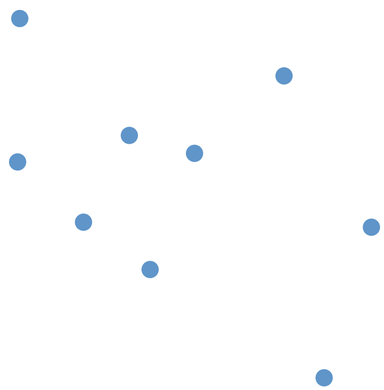
4 points lie above and below and p lies on L

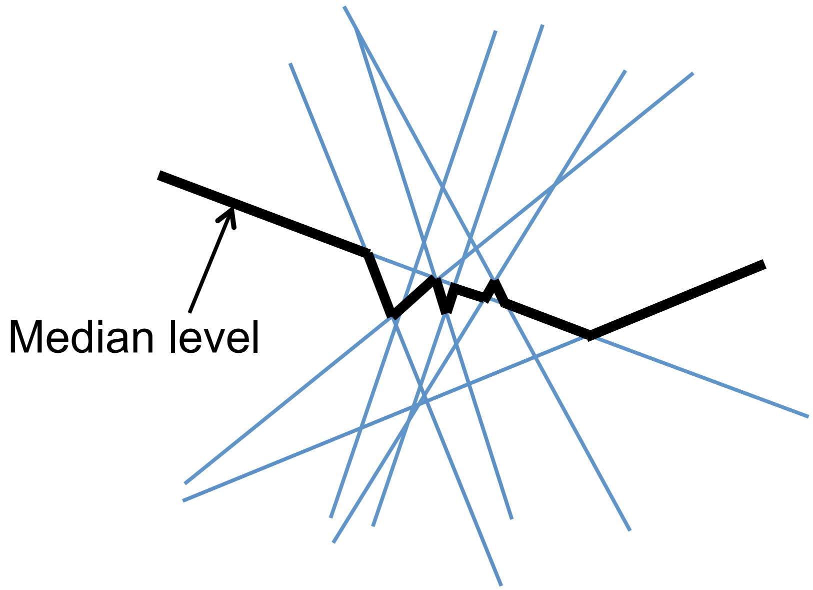


Any point on the segment has 4 points above and 4 points below

Median Level of an Arrangement

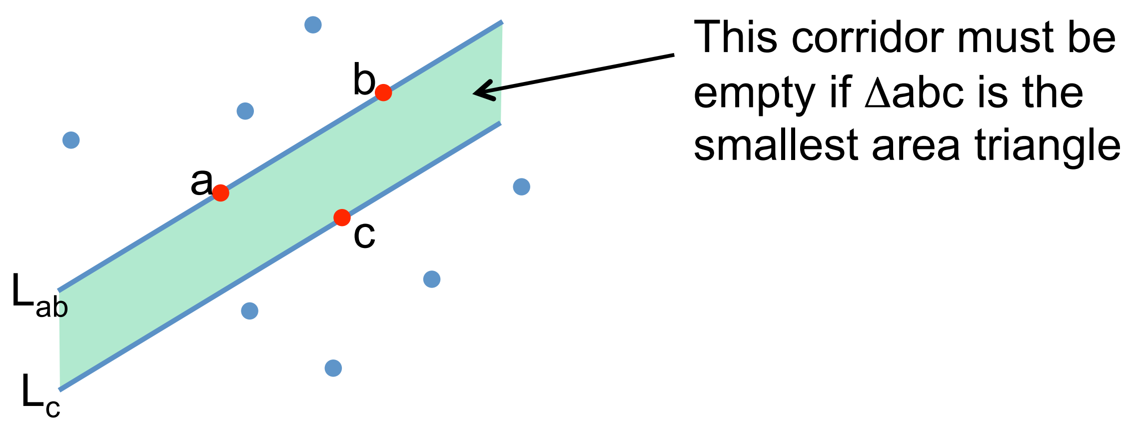
- Any point on the median level => the corresponding line in the primal plane bisects S





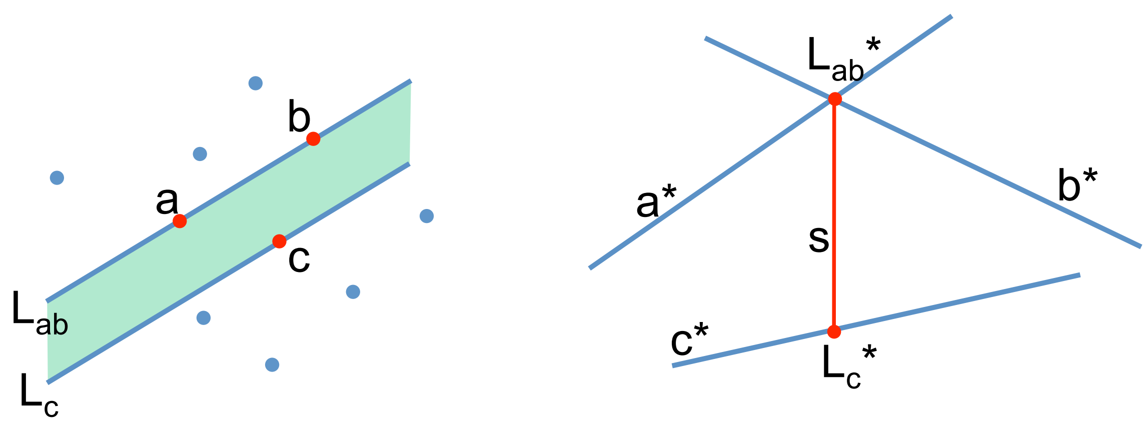
Minimum-Area Triangle

- If points {a, b, c} achieve a minimum area triangle among the points in a given point set P, c is the closest point among P - {a,b} to the line L_{ab} containing ab, where the distance is measured orthogonal to L_{ab}



Minimum-Area Triangle

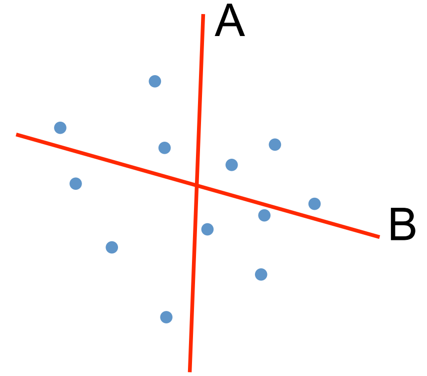
- Interpret the relationship in the dual arrangement (assuming order preserving dual)



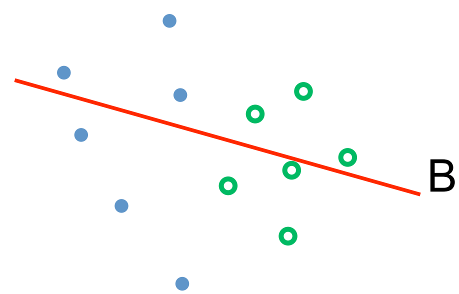
The line segment s must be contained in a face (i.e. no dual line can intersect s, otherwise the corridor is not empty

Problem 4: 219

- **4 Sections**: A point set P in the plane can be partitioned into 4 open wedges by two lines such that each wedge contains no more than $\left\lfloor \frac{n}{4} \right\rfloor$ points



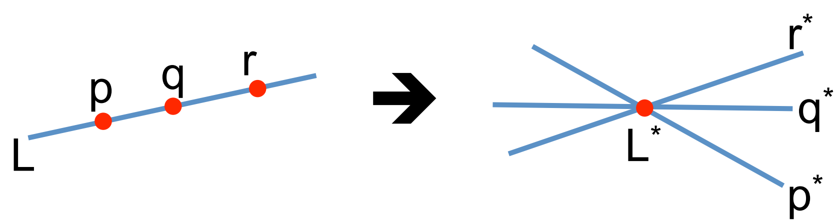
- Find A that splits P into P_1, P_2 ;
  - $|P_1| = |P_2|$
- B is a ham-sandwich cut of P_1 and P_2



Other forms of duality

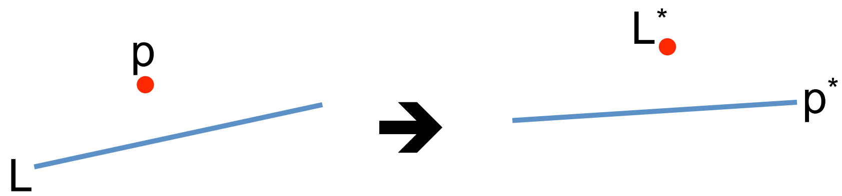
Duality

- p : (p_x,p_y) => p^*: $b = p_x a - p_y$
- L : y = ax -b => L^* : (a,b)
- **<u>Properties:</u>**
  - $(p^*)^* = p$
  - $(L^*)^* = L$
  - **<u>Incidence:</u>**
    - If p, q, r lies on L, p^*, q^*, and r^* contain L^*

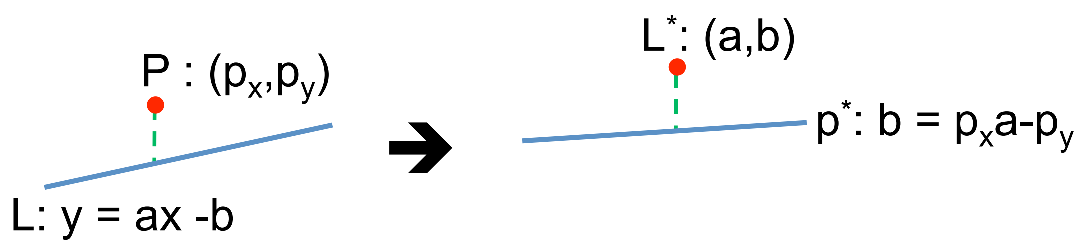


Duality (properties)

- **<u>Above-Below Relationship:</u>**
  - If p lies above L, then L^* lies above p^*



- **<u>Y-distance is Preserved:</u>**

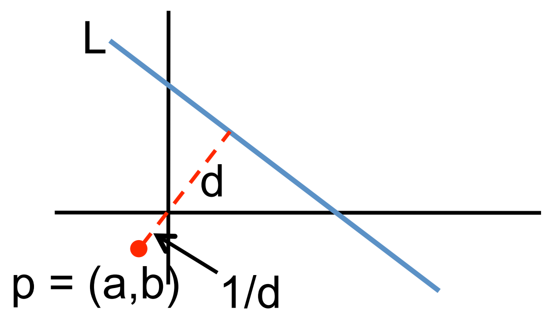


- **Y-distance of p to L =p_y- (ap_x-b)**
- **Y-distance of L* to p* = b-** (p_xa-p_y)

Duality (properties)

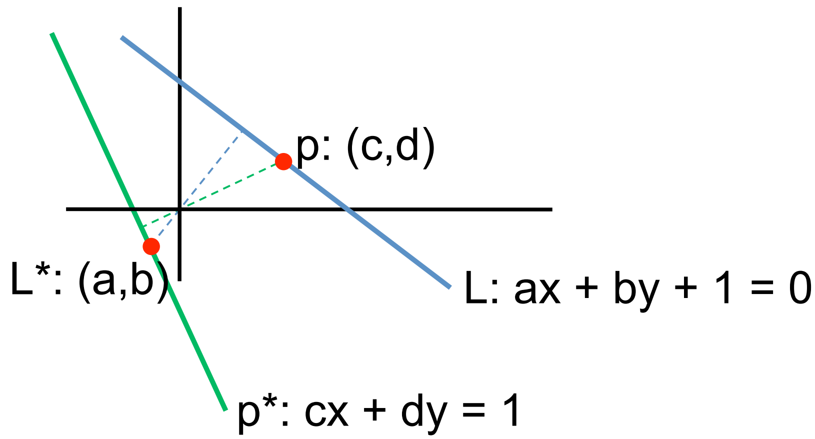
- Vertical line does not dualize => singular
- (0,0) does not dualize either

- **Polar Duality**
  - p: (a,b) => p^* : ax + by + 1 = 0
  - L : ax +by + 1 = 0 => L^* : (a,b)



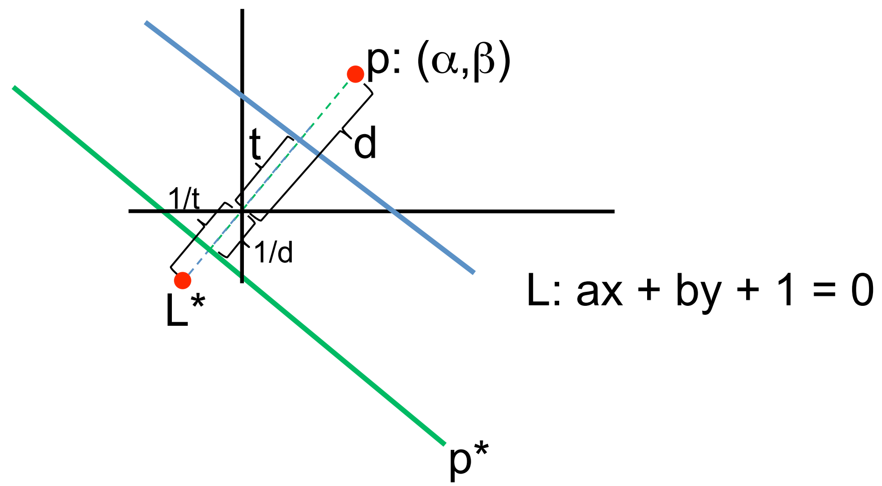
Polar Duality

- **Incidence Property is Preserved:**



Polar Duality

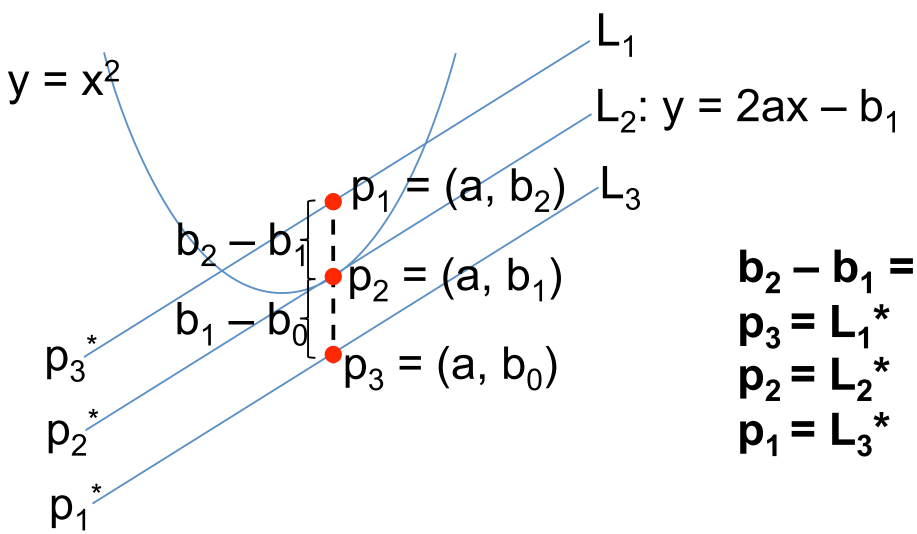
- **Above-Below Relationship**



- If p lies above the line L, p* lies above L*. (order does not change
- **Singularity**
  - Any line through the origin
  - Point (0,0)

Duality (from the text)

- p: (a,b) => p*: y = 2ax – b
- L: y = 2ax – b => L* : (a,b)

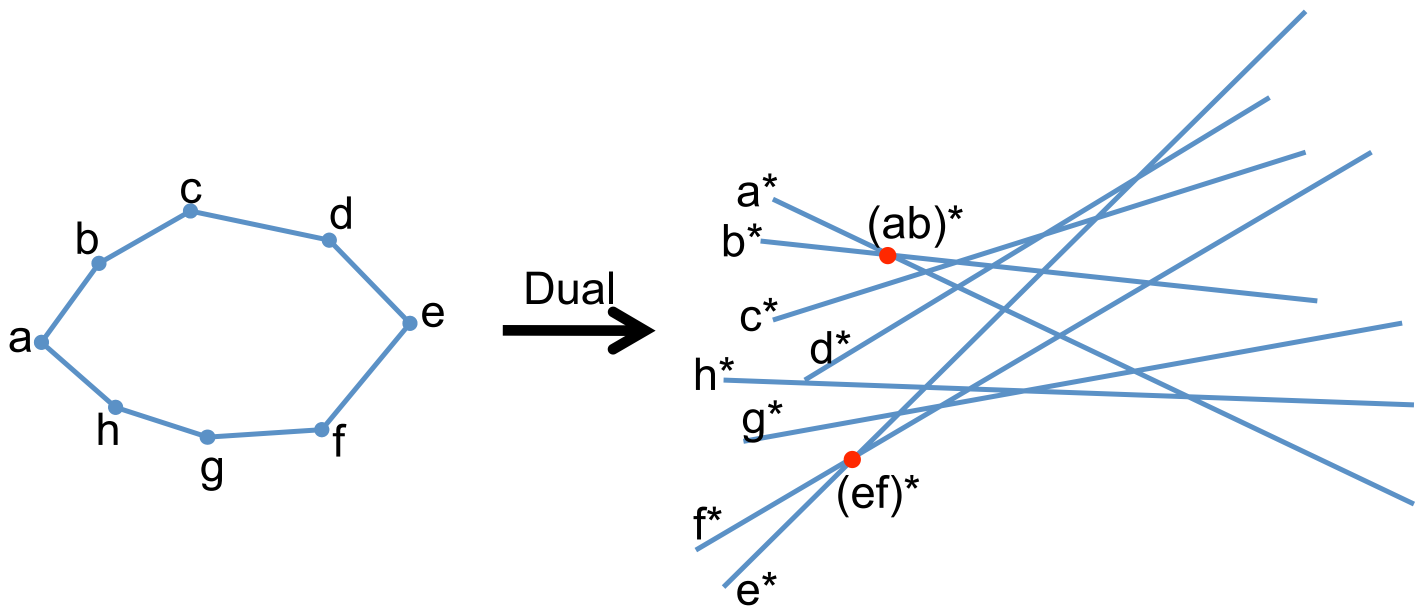


$b_2 - b_1 = b_1 - b_0$
$p_3 = L_1^*$
$p_2 = L_2^*$
$p_1 = L_3^*$

- Can be shown that if p is above L, then L* is above p*

Order Preserving Duality

- p : (a,b) => p* : ax + by + 1 = 0
- L : ax + by + 1 = 0 => L* : (a,b)



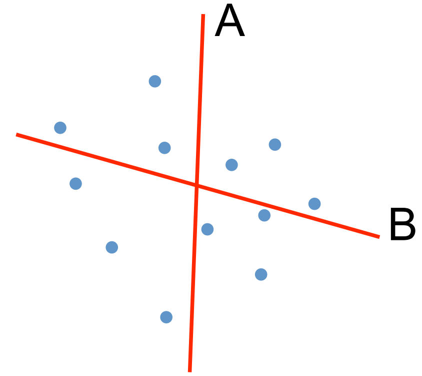
Duality

- **Order Preserving Duality**
  - **Envelope Theorem:** Computing the lower (upper) envelope of a set L of lines is equivalent to computing the lower (upper) convex hull of the set L* of points

- **Non-order Preserving Duality**
  - **Envelope Theorem:** Computing the lower (upper) envelope of a set L of lines is equivalent to computing the upper (lower) convex hull of the set L* of points

Problem 4: 219

- **<u>4 Sections</u>**: A point set P in the plane can be partitioned into 4 open wedges by two lines such that each wedge contains no more than $\left\lfloor \frac{n}{4} \right\rfloor$ points



- Find A that splits P into P_1, P_2 ;
  - $|P_1| = |P_2|$
- B is a <u>ham-sandwich cut</u> of P_1 and P_2