An Arrangement of Lines Dividing a Set of Points

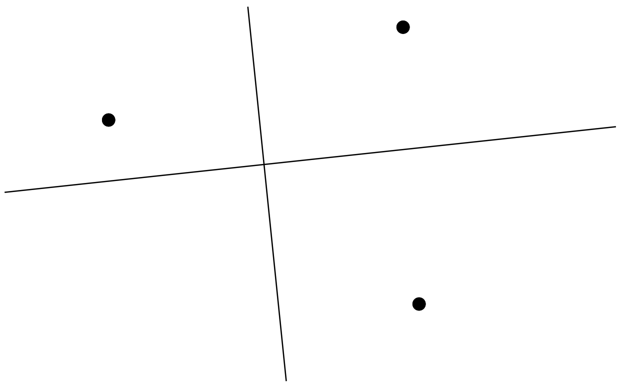
BY CARLOS TORCHIA

(V00717432)
2010-12-03
CSC 426 Paper
University of Victoria

Introduction

Consider the geometric idea of an **arrangement of lines**: a finite set of lines in the plane. Here, the result is a subdivision of the plane, with the lines forming edges, and their intersections forming vertices.

But this idea becomes more interesting when we start throwing points on the plane. Suppose you have a set of n points P, and you want to partition these points by regions of the plane, so that each region has one point of P. Then you could use a line arrangement to partition the points. Here, the arrangement of lines divides the plane into regions, and each of these regions has one of the points. How many lines do you need to do this?



In this paper, I am concerned with finding a way to divide a set of points with an arrangement of lines. In particular, each region of the arrangement of lines contains one point.

To do this, I would like to gain an understanding of line arrangements, so I will discuss relevant ideas about line arrangements too. Complexity of line arrangements and problem transformation become important here.

Thus, this paper is divided into discussions of the following:

1. Counting of the number of regions and k-gons of an arrangement of lines

2. Finding the triangular regions of an arrangement of lines that partition the plane

3. Counting the number of ways of dividing a set of points with one line

4. Finding an arrangement that evenly and optimally partitions a given set of finite points

Counting regions

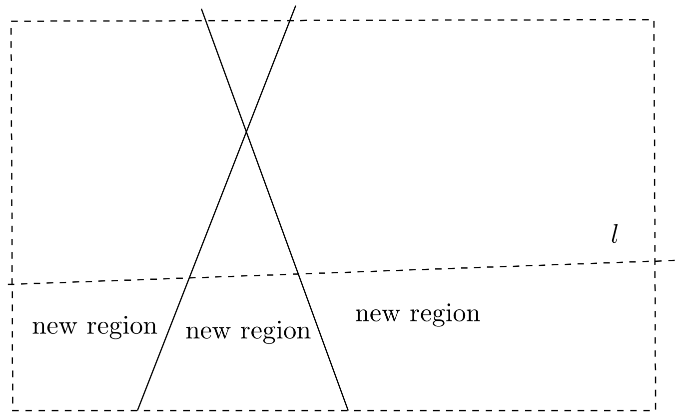
The word **arrangement**, although in this case being used to describe something in space, has combinatorial connotations that suggest a deep connection with counting. Indeed, much of the study done on line arrangements has been to count various quantities of an arrangement of m lines, such as number of edges, regions, triangles, etc.

Number of regions

The regions or faces formed by an arrangement of lines can take all sorts of shapes and sizes. Some regions may be triangles, others may be trapezoids, and still others can reach out to infinity.

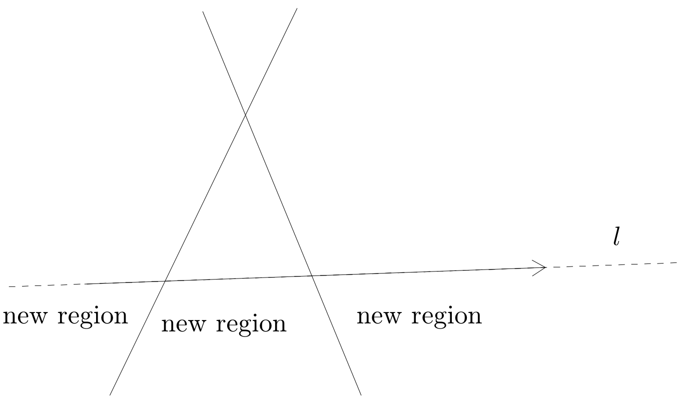
This semester, our computational geometry class briefly discussed the total number of regions that an arrangement of lines can have. Let $A(L)$ be an arrangement of m lines, and let $l \notin L$ be a line not in L. Suppose we add l to $A(L)$ to produce $A(L')$. First we assume that there are no parallel lines.

In class, we found that if one puts a bounding box around the intersections of $A(L')$, then the number of new regions formed is $m + 1$ if l does not intersect any pre-existing intersections in $A(L)$.



But this still works too without the bounding box. We think of travelling along l in a certain direction. In general, before each intersection of l with a line $l' \in L$, the addition of l produces a new region. After the last intersection, a final new region is created. So the number of new regions we get by adding l is $|l \cap A(L)| + 1$. Here, it doesn't even matter if any of the lines are parallel: what matters is the intersections, because that's exactly when new regions are created.

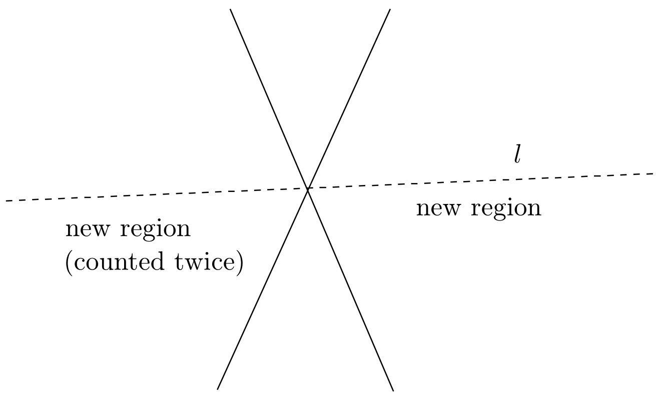
Suppose that $A(L')$ is simple, i.e. at most two lines are concurrent. Then the number of regions created by adding l to L is $m + 1$: one new region before each line of L, and then one after the last line that l intersects.



A line l can intersect another line l' at at most one point. If l did intersect any already-present intersections, then the number of new regions added is:

$$m - 0\,i_1 - 1\,i_2 - 2\,i_3 - \ldots + 1$$

where i_k is the number of intersections of $A(L)$ with l that are incident on k lines in $A(L)$.



Basically, to get the total number of new regions, we count the total number of intersections with l. If an intersection has k lines, we must subtract off $k-1$ from the total because there is only one new region, not k new regions.

But no matter what, we get at most $m+1$ new regions by adding l to $A(L)$. Also if there are no lines, we already have one region. This means that for an arrangements of m lines, the upper bound for the number regions $R(m)$ is:

$$R(m) = R(m-1) + m = 1 + 1 + 2 + \ldots + (m-1) + m \text{ (by induction)}$$

$$= m\,(m+1)/2 + 1$$

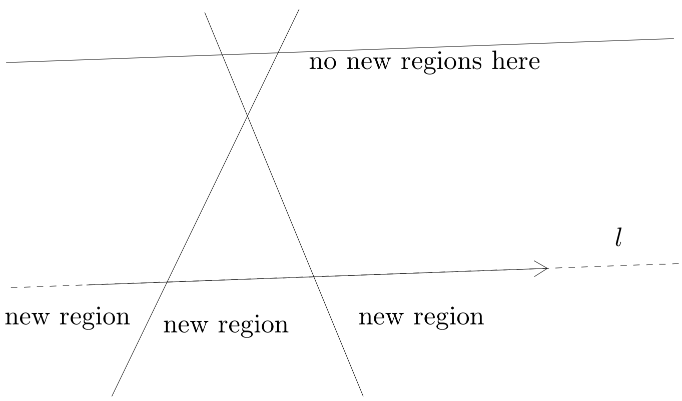
and this is what it is exactly if l intersects only one line at a time.

However, let the exact number of regions of an arrangement of $A(L)$ be $r(L)$. Write $L = \{l_1, l_2, \ldots, l_m\}$. Then:

$$r(L) = \sum_{i=1}^{m} \left| l_i \cap A(\{l_1, l_2, \ldots, l_{i-1}\}) \right| + 1$$

which we get by considering each l_i being added to the arrangement incrementally.

If we remove the assumption that not any lines are parallel, then we may not intersect all lines when we are adding l to $A(L)$ to form $A(L')$. Here, we produce one less region for each line we do not intersect. Thus, the upper bound $R(m)$ still holds.



Number of k-gons

You could go further and ask how many regions of an arrangement of lines there are with a certain number of sides.

G.B. Purdy showed that the number of triangles formed by an arrangement of m nonconcurrent lines is at most $\frac{5}{12}\,m\,(m-1)$ [1]. Using this, Purdy proves several other bounds regarding the number of polygons with certain sides. In general, the total number of quadrilaterals, pentagons, and hexagons is $O(m^2)$ [1].

Another interesting result due to Canham is that the sum of the numbers of sides of each of r polygonal regions of $A(L)$ is no more than $m + 4\binom{r}{2}$ [4]. Note that this implies that traversing each face's sides of $A(L)$ would require at least $O(m^2)$ operations. This makes sense because, since Euler's formula implies there are only $O(m^2)$ edges, since there are $O(m^2)$ intersection points and $A(L)$ is planar. Of course, one wonders if you could not do much better, since it is just an upper bound. However, Canham suggests this is the best you can do if there are sufficiently many lines [4].

Partitioning space with triangles

Consider the problem of partitioning space by a set of triangles formed by an arrangement of m lines. This is useful for various algorithms that operate on sets of lines, as Agarwal discusses [2]. Here, the idea is that if you partition the plane into triangles, then you can break up a geometric problem into subproblems on those triangles.

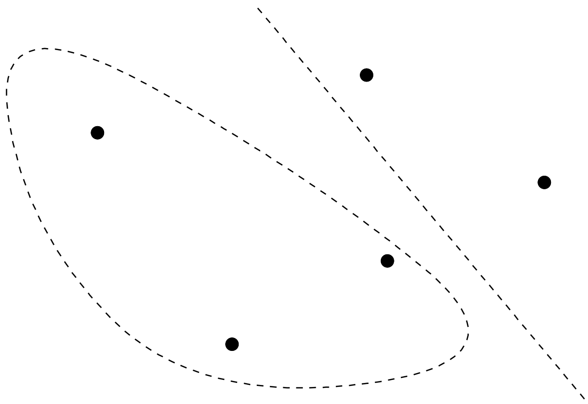
According to Agarwal, this can be done in $O(m\,r \log m \log^{\omega} r)$ time, where $\omega < 3.33$ [2]. Here, the plane is partitioned into $O(r^2)$ triangles, where $r < m$.

Agarwal goes into many applications of this algorithm [2]. One such application is as follows. Given a set of m lines and a set of n points, find the number of lines that intersect each point. I will interpret the approach that Agarwal basically takes. Here, we divide the plane into triangles formed by the set of lines. This allows us to efficiently find the points in each triangle. Then, for each triangle, find the points that intersect the triangle.

Agarwal is mainly concerned with efficiency. You could also find how many intersections for each point there are by testing each point with each line. This would run in $O(m \cdot n)$ time.

Semispaces

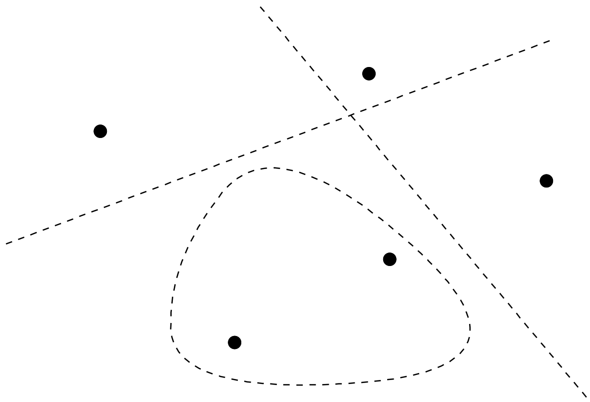
A **semispace** of a finite set of points S is defined as a subset of S that lies on a half-plane. Since a half-plane can be visualized as one side of a line, this equates to the set of points on one side of a certain line.



Alon and Györi showed that the number of semispaces of a set of n points with at most k elements is at most $k \cdot n$ [3]. It seems abstract, but you can see it is true for $k = 1$. Here, a line divides one point from the rest, for each point: the number of 1-sets is at most n.

Alon and Györi also point out that the number semispaces with k elements is equal to the number of semispaces with $n - k$ elements [3]. You can see this is true because if there are k elements on one side of a line, there must be $n - k$ elements on the other side, so a semispace of k elements uniquely describes another with $n - k$.

The idea of semispaces begins to pay off when you realize that the intersection of semispaces defines the points of S that are in a particular region of an arrangement of lines. Here, we take the half planes supported by the lines of $A(L)$. A subset of S that is in a region of $A(L)$ is equal to the intersection of semispaces supported by the lines bounding the region.



Can you use this idea to find lines that divide the points evenly?

Partitioning points with lines

We now may discuss the problem of dividing points with lines. Suppose we are given a set of n points P, and we want to divide the plane with a set of lines so that each point is in its own region. How do you find the arrangement that has the least number of lines?

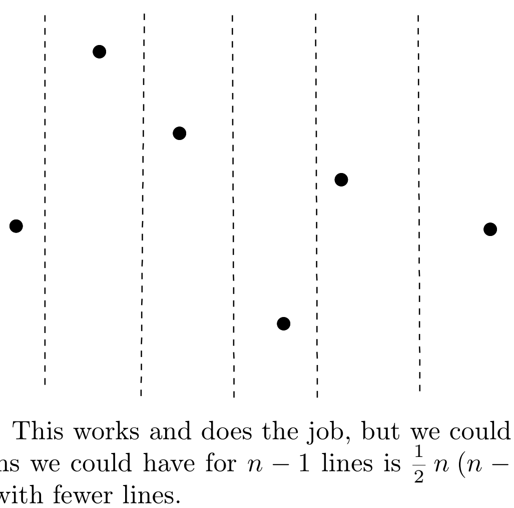
I did not find any other sources that discuss this problem. But sure enough, given such variation in school curricula, some Grade 10 student probably has this as a question on a take-home practice quiz as I am writing this very sentence. Regardless, it is an interesting problem, and I consider it anyway.

Points in regions

If we want to partition a set of points by an arrangement of lines, what we want to do, precisely, is have each point in one region of the arrangement of lines. It follows, then, that the number of lines you need to partition a set of points depends on the number of regions you need. The bottom line is that if you have n points, then you need at least n regions: one for each point.

Naive approach

Suppose we are given a set of n points. Assume none of the points have the same x coordinate. Then a quick, simple way to divide each point into its own region is to sort the points by x coordinate $p_1, p_2, ..., p_n$, and then calculate the midpoints q_i between p_i and p_{i+1} for i from 1 to $n-1$. Then the lines $\{\{(x, y)\,|\,x = q_{i_x}\}\,|\,1 \leqslant i \leqslant n-1\}$ partition the set of points evenly, i.e. one point in each region.



Here, we have $n-1$ lines. This works and does the job, but we could do it with fewer lines. The maximal number of regions we could have for $n-1$ lines is $\frac{1}{2}\,n\,(n-1)+1$. Therefore we could get just as many regions with fewer lines.

How few lines can we get away with? As we saw earlier, the maximal number of regions is attained when no three lines are concurrent and no two lines are parallel. Here, the number of regions is $r = \frac{1}{2} m (m+1) + 1$, where m is the number of lines.

Suppose we have at least $\sqrt{2n} - 1$ lines. Then:

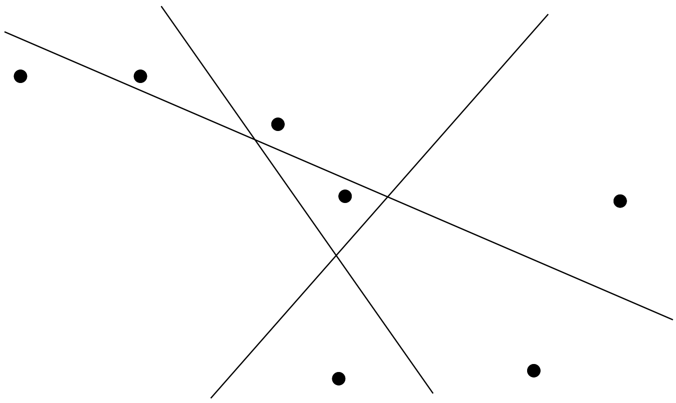
$$r = \frac{1}{2} m (m+1) + 1 \geqslant 1 + \frac{1}{2} (\sqrt{2n} - 1) \sqrt{2n} \geqslant 1 + \frac{1}{2} \sqrt{2n} \sqrt{2n} > n$$

so the number of regions would be more than we need. Thus, for n points, we only need $O(\sqrt{n})$ lines to divide them.

But the question is, can we always partition P with an arrangement of lines that has no parallel lines and at most two concurrent lines?

Special case: $n = 7$ points

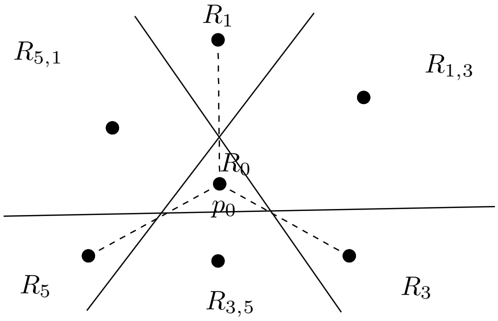
This semester in our computational geometry class, professor Sue Whitesides proposed the Big Dipper problem: given a set of 7 points (appearing as the Big Dipper constellation of stars), separate each point into its own region by an arrangement of lines.



Here, with 3 non-parallel lines with at most two concurrent, we get $1 + \frac{1}{2} 3 (3 + 1) = 7$ regions, which allows us to fit the 7 points into the arrangement so that each region gets one point.

Morphing an arrangement of lines to fit the points

I approached the Big Dipper problem first by placing arbitrary lines on the plane, and then rearranging them so that they put each point in its own region. Here, the idea is that any non-parallel arrangement of 3 lines with at most two lines concurrent will have an inner triangle, and the other regions of the arrangement will border on the outside of this triangle.



Thus, I devised an algorithm that finds the arrangement of lines that partitions each point in its own cell by finding a point p_0 of P to be in the center triangle, placing the lines around this point, and then shifting the lines so that the partitioning is correct.

If we position the three vertices of the central triangle R_0 on the line segment between each surrounding point and the inner point p_0, we are at least guaranteed to get these points and the inner point in their own regions with respect to one another. These regions consist of the inner triangle, and the cones (R_1, R_3, R_5) at the corners of the triangle. But the remaining three points may also be in these regions. To remedy this, we morph the arrangement of lines until these points are in the other regions. We'll see why this works.

The formal description of the algorithm follows.

> Input: a set $P \subseteq \mathbb{R}^2$ of 7 points
>
> Output: a set of lines $A(L)$ such that each point of P is in its own region.
>
> For each $p_0 \in P$
>
> > `(* We suppose` p_0 `is the centre point. *)`
> >
> > $i = 0$
> >
> > Order $\{p_1, p_2, ..., p_6\} = P - \{p_0\}$ in order of increasing angle of polar coordinates centered at p_0
> >
> > `(* Define the vertices of the triangle *)`
> >
> > For each $p_i \in \{p_1, p_2, ..., p_6\}$ and $i \equiv 1 \pmod 2$
> >
> > > Let $q_i = p_0 + \frac{1}{2}(p_i - p_0)$
> >
> > `(* Define the initial set of lines. *)`
> >
> > Let $L_{1,3} = \{q_1 + \lambda(q_3 - q_1) \mid \lambda \in \mathbb{R}\}$
> >
> > Let $L_{3,5} = \{q_3 + \lambda(q_5 - q_3) \mid \lambda \in \mathbb{R}\}$
> >
> > Let $L_{5,1} = \{q_5 + \lambda(q_1 - q_5) \mid \lambda \in \mathbb{R}\}$
> >
> > $L = \{L_{1,3}, L_{3,5}, L_{5,1}\}$
> >
> > Let R_0 be the region bounded by L. R_0 is then the central triangle.
> >
> > If $p_0 \in R_0$
> >
> > > `(* Correct the arrangement of lines *)`
> > >
> > > Let R_1 be the region bounded by the lines that contains p_1
> > >
> > > Let R_3 be the region that contains p_3
> > >
> > > Let R_5 be the region that contains p_5
> > >
> > > Let $R_{1,3}$ be the region adjacent to R_0, R_1, R_3.
> > >
> > > Let $R_{3,5}$ be the region adjacent to R_0, R_3, R_5.
> > >
> > > Let $R_{5,1}$ be the region adjacent to R_0, R_5, R_1.
> > >
> > > For $j \in \{1, 3, 5\}$
> > >
> > > > If the previous $p_k \in R_j$ then increase $q_i - p_0$ until this is not true.
> > > >
> > > > If the next $p_k \in R_j$ then increase $q_i - p_0$ until this is not true.
> > >
> > > Return L

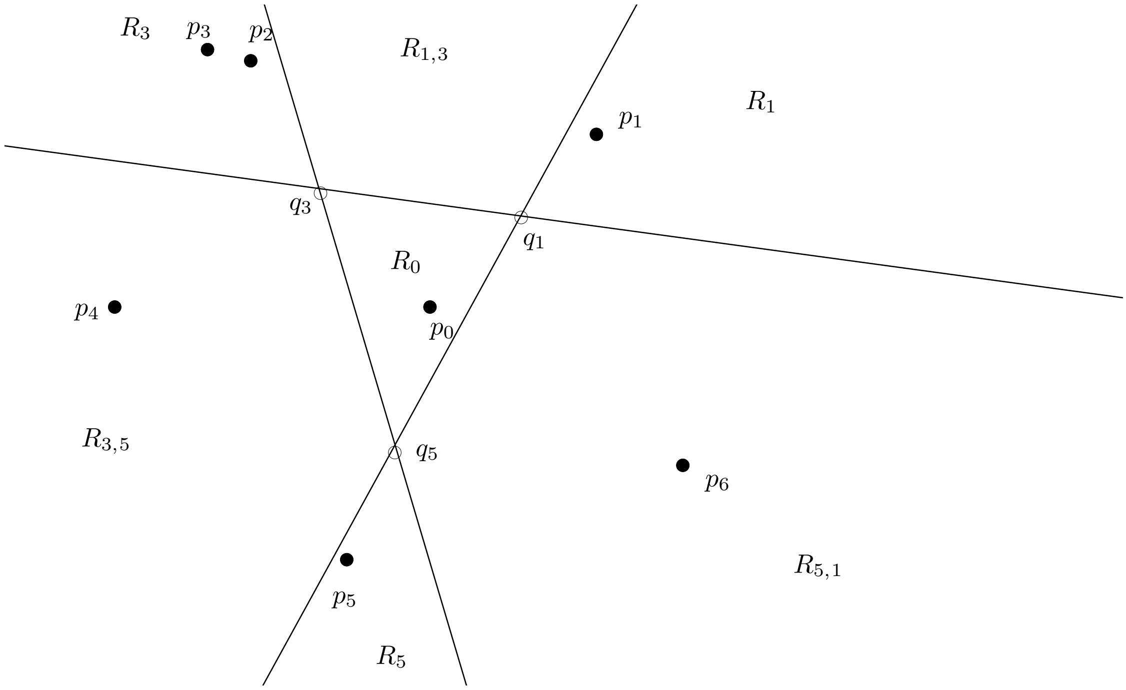
When I say p_{prev}, I mean p_1 if $j = 2$, p_6 if $j = 1$, etc. When I say p_{next}, I mean p_6 if $j = 5$, p_1 if $j = 6$.

Proof of correctness

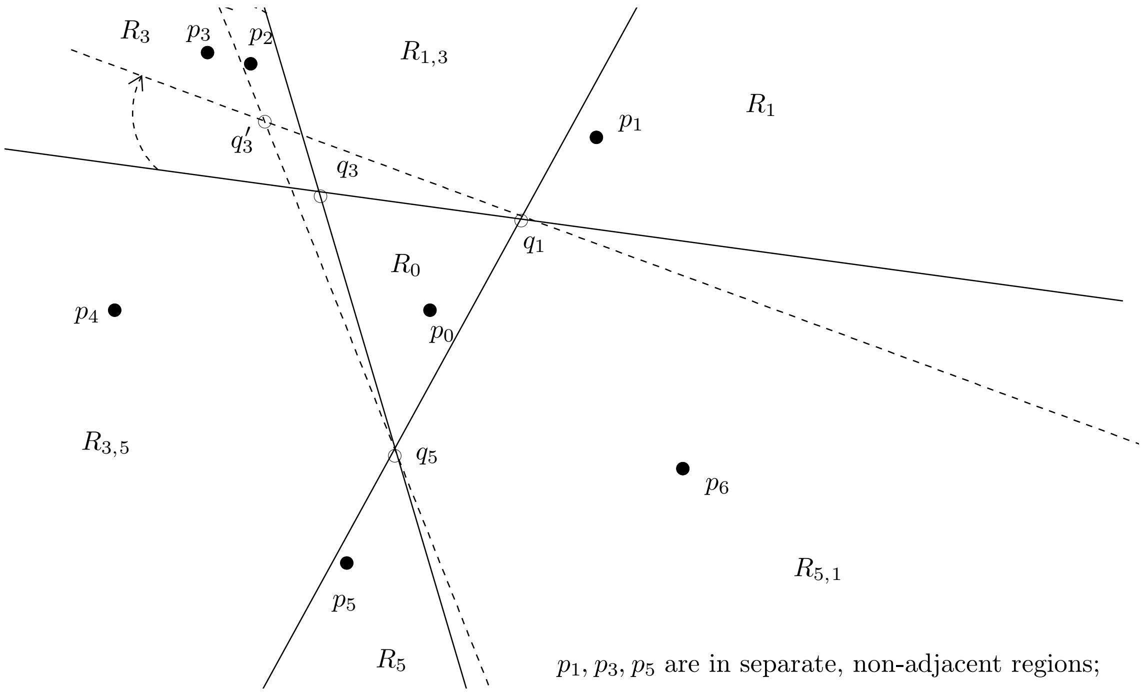
We show that each p_i is in a different region.

First, note that R_1, R_3, and R_5 must be opposite of different corners of the triangle defined by $\{q_1, q_3, q_5\}$. This is because for odd i, p_i is colinear with q_i and p_0, and is farther along $q_i - p_0$ than q_i. This places p_i in the region opposite of R_0 bounded by two lines. This happens exactly when $p_0 \in R_0$. Such regions R_1, R_3, R_5 could not intersect because their regions are defined by different corners of the triangle.

Furthermore, R_1, R_3, R_5 could not be adjacent. This follows from the fact that if $p_0 \in R_0$, then R_i are opposite the corners of the triangle.



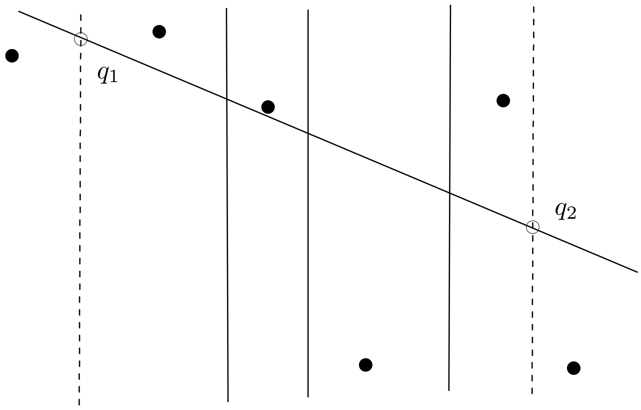
The question arises of whether we can be sure p_2, p_4, p_6 are in separate regions. Suppose $p_0 \in R_0$. I claim that $p_2 \in R_{1,3}$, $p_4 \in R_{3,5}$, $p_6 \in R_{5,1}$. Note for even-number i, p_i is between p_{prev} and p_{next} in angular order. Let $p_{\text{prev}} \in R_{\text{prev}}$ and $p_{\text{next}} \in R_{\text{next}}$ and R_{mid} be the region between them, which exists because R_{prev} and R_{next} are not adjacent. Then $p_i \in R_{\text{prev}} \cup R_{\text{mid}} \cup R_{\text{next}}$. But depending on whether we processed p_{next} or p_{prev} first, we would have moved q_{next} or q_{prev} closer to p_{next} or p_{prev} until $p_i \notin R_{\text{next}}$ or $p_i \notin R_{\text{prev}}$. Since this is done for each odd i, this process will make it so that each odd p_i will be in its own region. Since the even p_i's are separated by the odd regions, which are not adjacent, the even p_i's must be in their own regions:



Although we solved the problem for $n = 7$, it's hard to see how this approach can be taken with an arbitrary number of lines. We got lucky because this arrangement of lines has only one triangle. Others have many triangles. Then what do you do?

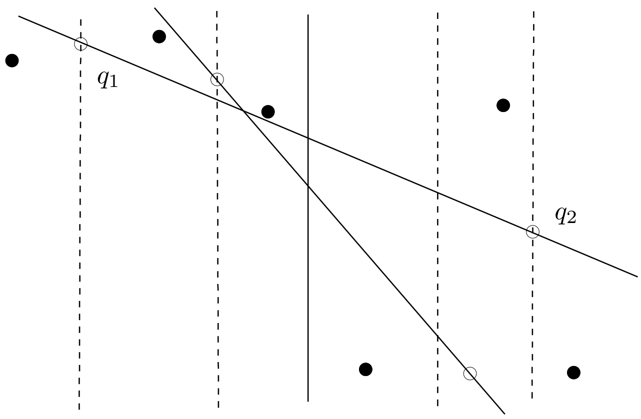
Decremental approach

Another approach to dividing n points with an arrangement of lines also starts with an arrangement of lines that may not correctly partition them at first. Here, however, we start with the naive solution: $n-1$ parallel lines between each two points. However, we may remove two lines and replace them with one line.



When we remove a line, we make a new region that has two points in it instead of one. So if we remove two lines, and take the midpoints q_1, q_2 of the pairs of points that are in the same region, then the line passing through q_1 and q_2 divides each pair.

At least we know how to proceed heuristically. We can remove more lines, but they may not necessarily be remedied by a line through the midpoints of the pairs points that are in the same region.



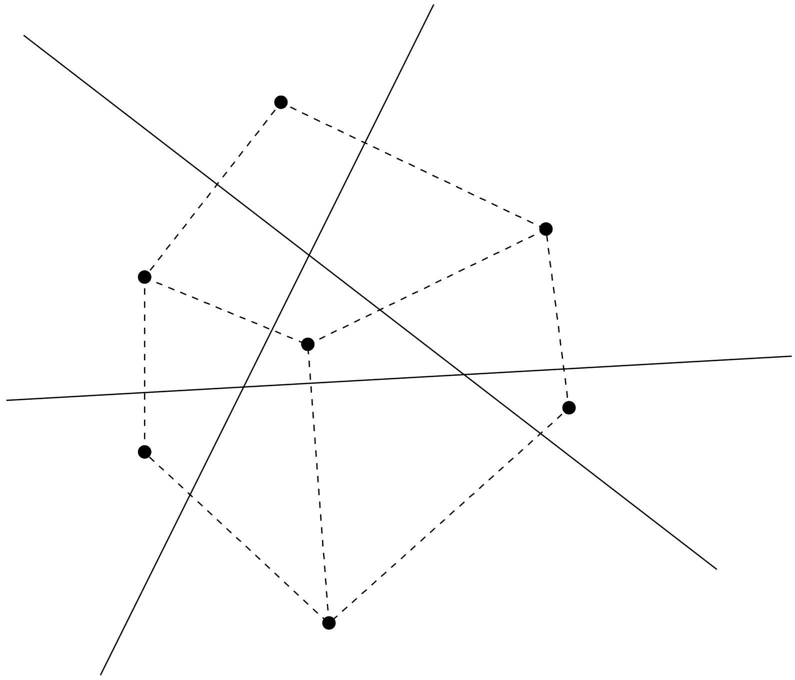
In the above example, we've removed four of the initial parallel lines, and added two remedying lines.

Alternative approaches

Now, we shall look at this problem from a different perspective. In particular, we transform it into problems in dual spaces. Will these new problems be any easier?

Taking the dual graph

It is possible that taking the dual graph created by the intersection of lines dividing the points, if any, will give ideas about the characterization of points that can or cannot be divided by lines as such.



Notice each face of this dual graph has the same degree as each intersection of the set of lines, which is 4.

A naive approach to find a set of lines that will divide the points this way could be to try and put such a graph on the vertices. Then, we only need to find sets of consecutive edges through which we could put a line.

The only question is, what nodes of this graph correspond to which vertices so that the graph can be drawn as an arrangement of lines? Can this always be done?

Transformation of problem into line space

Suppose we were to transform the space in which we pose this problem into the one in which we transform the x coordinate of a point to a slope m and the y coordinate to y-intersect $(-y)$. More precisely, transform $p=(m,b)$ to line $\{y=m\,x-b\}$.

Since this transformation preserves y order between lines and points, the transformed points will also divide the plane into "cells" into which lines will fit. Here, if line l is above p, then the point's line p^* is above the line's point l^*.

Suppose we have 7 points. Then we need 3 lines to divide, so the transformed space will have 3 points that will divide the plane into cells. We can describe these cells in terms of whether they are above or below which points. So we can describe each cell as a vector $(\pm, \pm, \pm)$ called a **signature**, indicating whether it is above or below p_1, above or below p_2, and above or below p_3.

But then the question becomes: how do we find these cells for a given set of n lines?

Conclusion

In general, an arrangement of lines in the plane divides it into a number of regions quadratic in the number of lines. Efficient algorithms are out there for finding these regions and putting them to use.

But when you are given a set of points to begin with, and you want to partition those points with an arrangement of lines, things are trickier. Essentially what you are doing is finding semispaces that optimally divide the set of points. Intersections of these semispaces correspond to the points of each region.

Given that the number of regions of an arrangement of m lines is $O(m^2)$, the number of lines needed for n regions dividing n points is $O(\sqrt{n})$. But there has to be inner points that are confined within polygonal regions, and so it may be hard to choose a set of lines that will divide the points evenly. It doesn't always work to fit a simple arrangement of lines to a set of points. A decremental approach starting with the naive arrangement of $n-1$ lines seems to work for a wider range of point sets.

My recommendations for further research are thus as follows:

1. Study how transformation of this problem into dual scenarios can make the solution easier

2. Try to find a general characterization for when exactly a set of points can be partitioned by an arrangement of a certain number of lines

References

1. Purdy, G.B. Triangles in Arrangements of Lines. Discrete Mathematics 25 (1979) 157-163.

2. Agarwal, Pankaj. Partitioning Arrangements of Lines. Discrete & Computational Geometry 5 (1990) 533-573.

3. Alon, N., Gyori, E. The Number of Small Semispaces of a Finite Set of Points in the Plane. Journal of Computational Theory Series A 41 (1986) 154-157.

4. Canham, R.J. A Theorem on Arrangements of Lines in the Plane. Israel J. Math 7 (1969) 393-397.