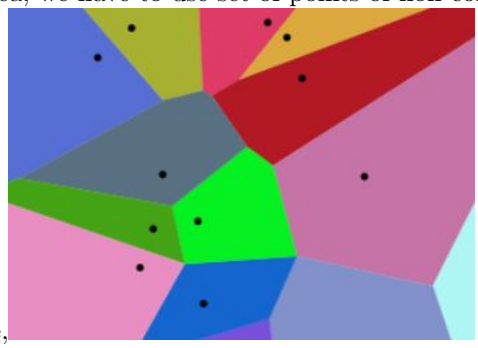
CSE355

Zhiming Fan

Nov 2018

1.

a). For this question, we could use the techniques Voronoi diagram.Suppose this plane has infinite area.One Voronoi edge has to be created between two points on the convex hull, and this edge is also the bisector between two points. However, the bisector line will not be able to appear as a wrapping edge of Voronoi cell.Because the wrapping edge must be drawn between two vertices, but we are not able to draw the edge to wrap the cell. Namely, in order to draw a cell that has finite area, we have to use set of points of non-convex hull.

For example, 

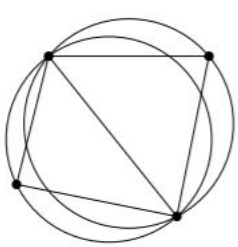
The area of green is finite but the other are not.

b).

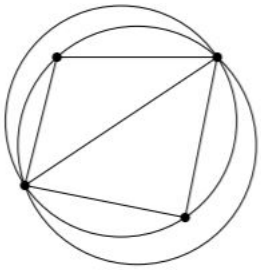
By the definiiton of Delaunay triangulation, the edge has to be the bisector of the nearest vertices. Firstly, we draw a diagram on the plane and create Voronoi diagram and connect all the bisectors.We can see the nearest neighbor vertices will be on the circumcircle of red points. 

But those points are not neighboring will not construct this Delaunay triangulation. Suppose we ar e able to draw the Delaunay triangulation between

far-away vertices, however, this Delaunay triangulation will be considered as failure triangulation.



(a) Delaunay triangulation.



(b) Non-Delaunay triangulation.

For this picture, the left side only connects nearest vertices but right one chooses arbitrary vertices. Thus this is a non-Dalaunay triangulation.

c).

To my understanding, the n-regular polygons are n sides convex polygons with same length of each side.

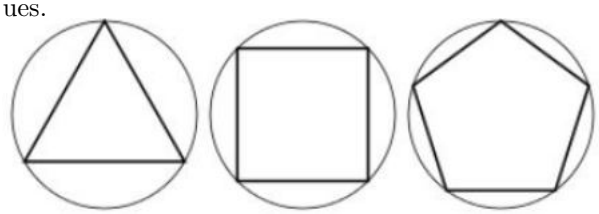
However, the bisector line will still intersect at a common point as the sides increase.

The intersect pointer will not change or increase.

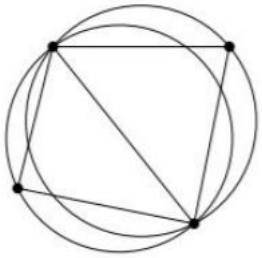
It just cuts the plane into equal pieces of area if the center of regular polygon at center of this plane.

So the number of Voronoi cells = the number of sides of n-regular polygon.

For the Delaunay triangulation, we will draw circle of nearest points. But the circle will be overlapping if we repeatly draw circle as the triangulation continues.



d).

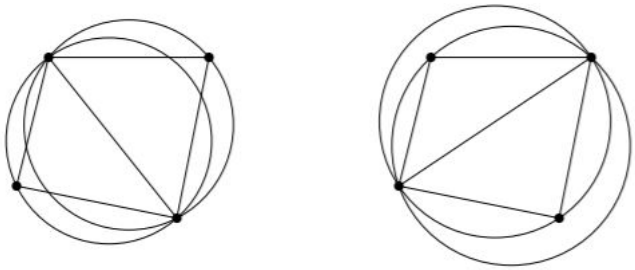


2.

a).

Firstly, we have Pi.It provides us a growing circle. Once the growing circle reaches nearest point, we call it Pj. We also use growing circle on Pj. If no

points are covered in the intersection region between Pi and Pj. We connect two points. We also need to find 3rd nearest point by growing circle method. If the the three points satisfy Pi-Pj ¡= MaxPj-Pm,Pj-Pm. Connect three points otherwise we delete the last nearest edges. It continues until all the points are utilized in the program.
The time complexity will be $O(n^3)$.
b).
We can draw the circumcircle of the three vertices Pi, Pj and Pm. We also draw the RNG.
For each point, we use a growing circle to find the nearest neighbor. When two neighboring circles reach each other, we can get the forbidden region. If no other points are covered inside the forbidden region, we could connect two points. We could imagine that the nearest points can form Delaunay triangulation like this picture. In the other words, the procedure of constructing RNG can be done by Delaunay triangulation as well. By contradiction, once we find any point locates inside the forbidden region, the two points which provides us growing circle will not be the nearest points. So we could prove the RNG $\subseteq$ D(p).



(a) Delaunay triangulation. **(b)** Non-Delaunay triangulation.

c).
We could use growing circle to divide this plane by Delaunay triangulation. After that, we could design a faster algorithm to delete edges between vertice in the plane. The procedure can be $O(n^2)$. For example, once the nearest pair of vertices cover the points in the region, we may delete edges and the cursor move the other nearest points. The time complexity will be reduced greatly as well. The total time complexity $O(n^3)$ will be decreased to $O(n^2)$. Because we will consider how to delete the existing points instead of creating edges between them.