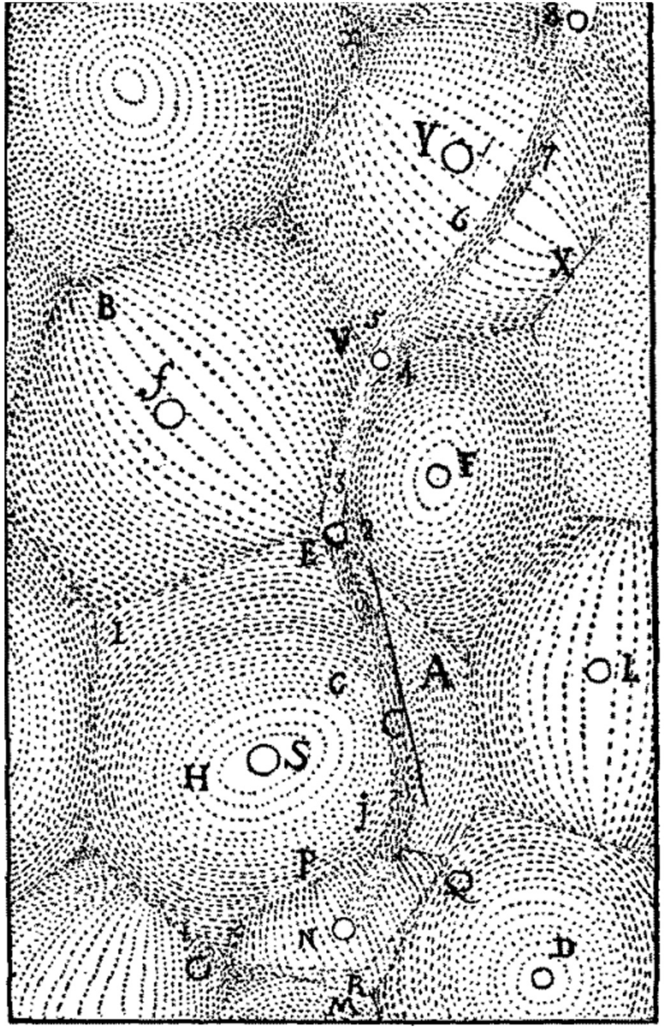
Voronoi Diagrams & Delaunay Triangulations: Related Tilings, Applications and Generalizations

Boris Ter-Avanesov

Some History

- The first published Voronoi diagram dates back to 1644:
  - In the book “Principia Philosophiae” by the famous mathematician and philosopher Rene Descartes.
  - He claimed that the solar system ‘consists of vortices. In each region, matter is revolving around one of the fixed stars.’
- Worked on by Kepler, Dirichlet, and Thiessen but rigorously studied and defined by Voronoi.



Voronoi Diagrams

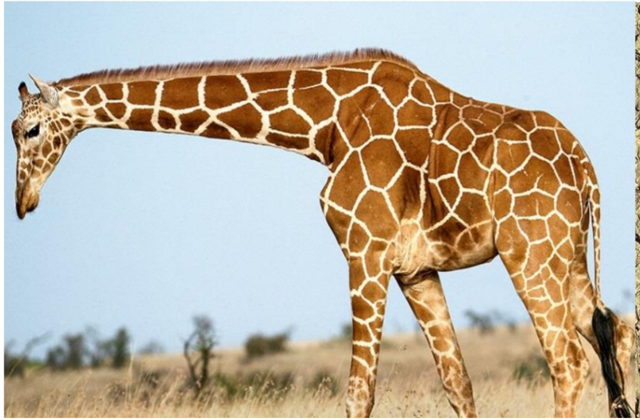
- Georgy Feodosevich Voronoy (1868-1908*).

- A certain decomposition of a given space X into cells, induced by a distance function and by a tuple of subsets $(P_k)k \in K$, called the generators or sites. The Voronoi cell or region V_k associated with the site P_k is the set of all the points in X whose distance to P_k is not greater than their distance to the union of the other sites P_j.

Glossary for VDs Elsewhere

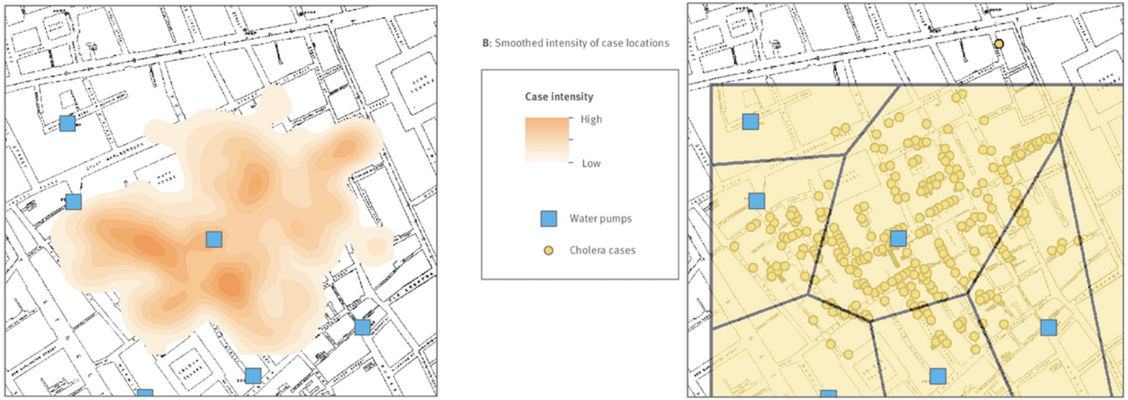
Field of science:	Term used:
Mathematics:	Voronoi diagram, Dirichlet tessellation
Biology and Physiology:	Plant polygons, Capillary domains, Medial axis transform
Chemistry and Physics:	Wigner-Seitz zones
Crystallography	Domains of action, Wirkungsbereich
Meteorology and Geography:	Thiessen polygons

- Wide range of applications* and appear in nature.
- Glossary above is not complete...



Application Example: Maps

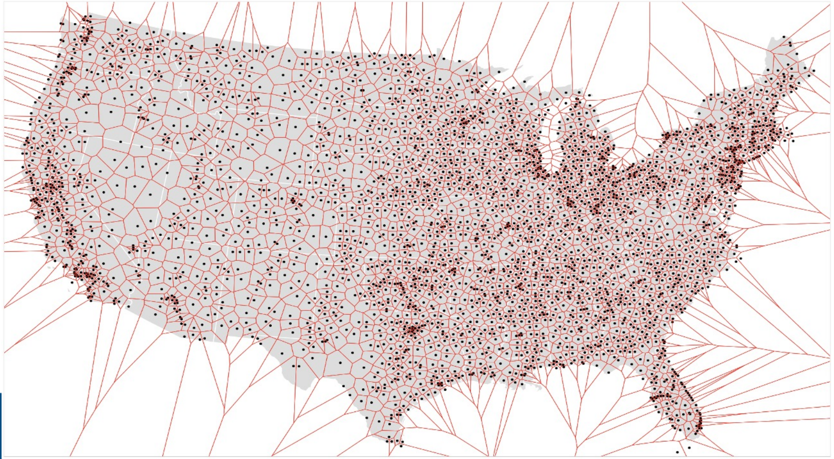
- Identifying and tracking cholera, John Snow, 1854, London:
  - Used data on the number of victims and locations of disease cases.
  - Divided London into sections, each having a separate water source.
  - Plotted his data on a map, creating a Voronoi diagram.
  - Found that most deaths were in houses provided by a particular pump in Broad Street, Soho.
  - Before that the cause of disease was unknown.



More Maps...

- Map showing all airports and the voronoi regions for their locations.
  - Picking locations for future airports.
- Map showing voronoi regions of fire trucks (manhattan distance).
  - Nearest fire truck is in light green region, same as the fire.
  - No other truck can reach any light green point faster than this truck.
- A fun example:
  - World's territories have been redrawn.
  - Closest capital city determines a voronoi region.

U.S. Airports Voronoi



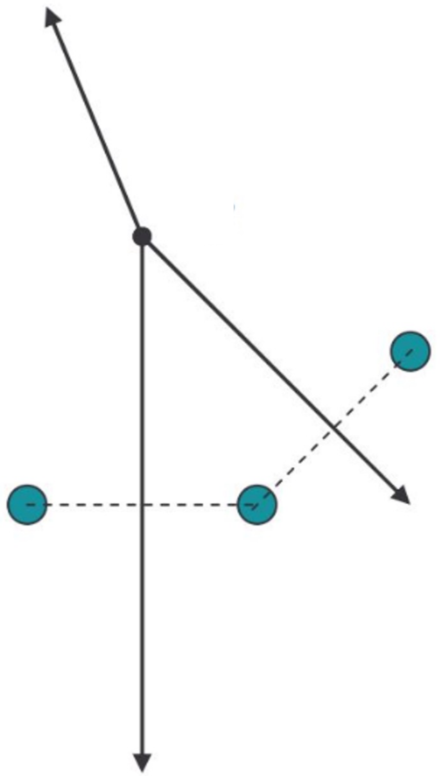
Voronoi Diagrams on R2

- Denote the location of a point p_i as (x_{i1}, x_{i2}). Let $P = \{ p_1, p_2, \ldots, p_n \} \in R2$, where $2 \leq n < \infty$ and $p_i \neq p_j$ for $i \neq j$, be the set of generators.
- The Voronoi Region of p_i is $V(p_i) = \{ p_k ;\ d(p_k, p_i) \leq d(p_k, p_j) ,\ \forall j \ni i \neq j \} = V_i$
  - Convex and connected...
- The set given by $V = \{V(p_i), \forall i\}$ = union of all V_i , is the Voronoi diagram of P.
- The Voronoi diagram is composed of three elements: point sites/their corresponding regions, edges, and vertices.

Edges, and Vertices

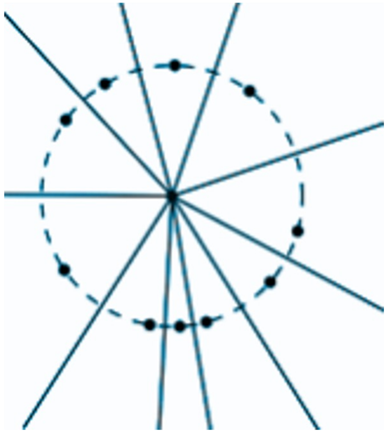
- An edge between Voronoi regions is their intersection.
  - If $e(p_i , p_j) \neq \emptyset$, the regions are considered adjacent.
  - Points on an edge are equidistant to the generators of the voronoi regions separated by this edge.
  - The set of edges surrounding a Voronoi region can be referred to as the boundary of the region.

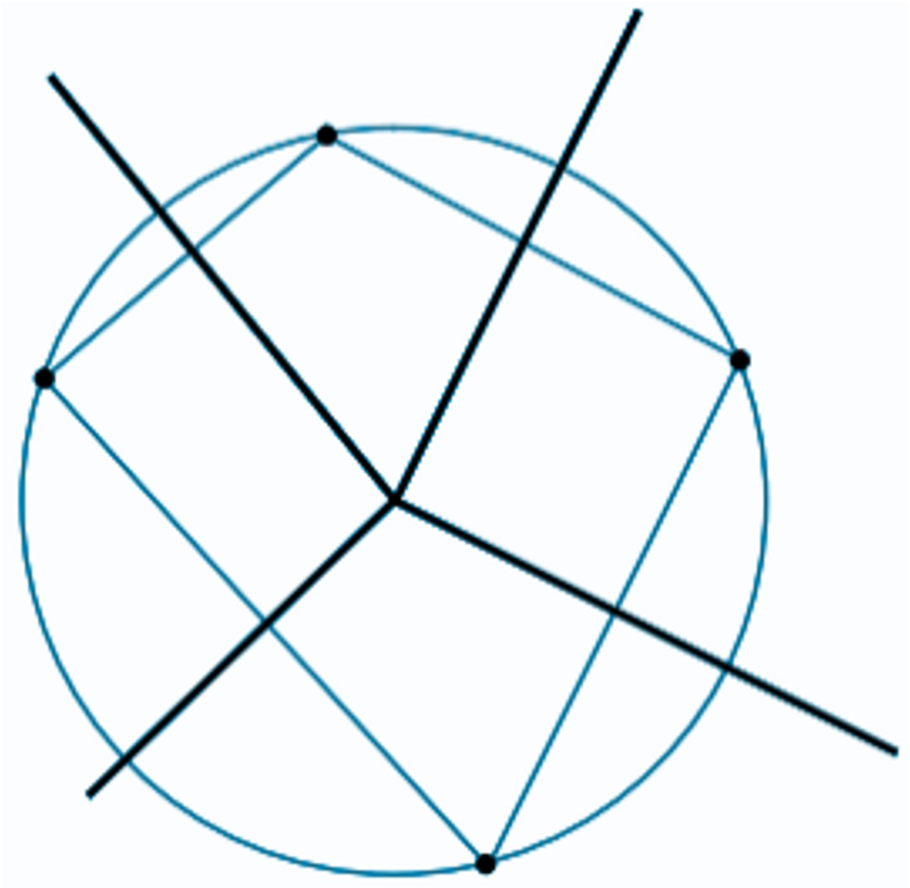
- Vertices (q_i) are the endpoints of edges/any point that is equidistant from the three (or more) nearest generator points.
  - The number of edges that meet at a vertex is called the degree of the vertex.
  - If $\forall q_i \in V$, degree(q_i) = 3, then V is considered to be non-degenerate.



Four Co-circular Points, Degenerate Voronoi Vertex Example

- ‘General position’ assumption = no four sites on circumcircle of any VD vertex so this example would not be possible*.
- Note: this does not guarantee that all vertices are non-degenerate.
  - Can still get more than four co-circular sites.
  - In some papers ‘general position’ = no four or more co-circular sites.





Dominance Regions, Half-Planes

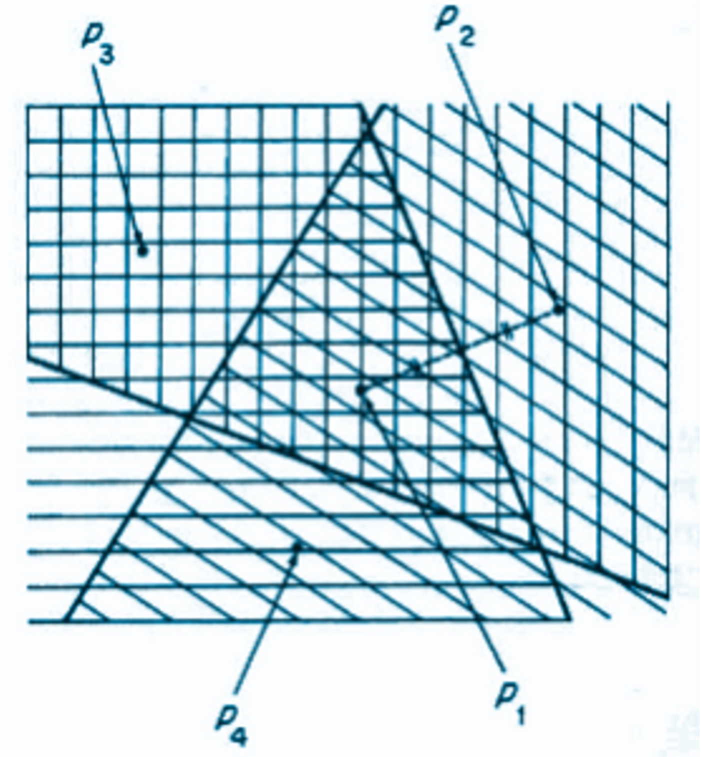
- Given any two generators, p_i and p_j, the dominance region, $H(p_i, p_j)$, of p_i over p_j, is the set consisting of every point of the plane that is closer to p_i than p_j or equidistant from the two.
- In VD on R2, all $H(p_i, p_j)$ are half-planes.
- Voronoi regions can be redefined as intersection of dominance regions.

- Construction of VDs from Half-Planes:

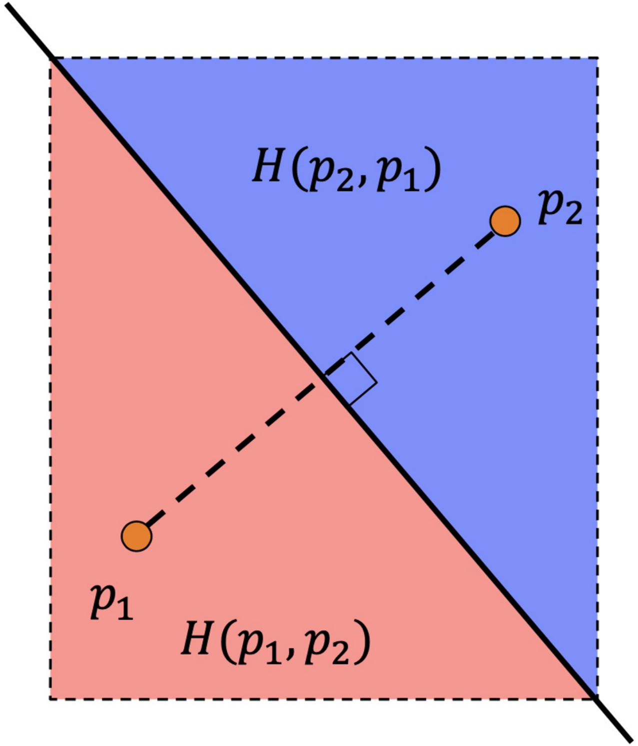
  Let $P = \{p_1, p_2, \ldots, p_n\} \in R2$ be a set of generator points.

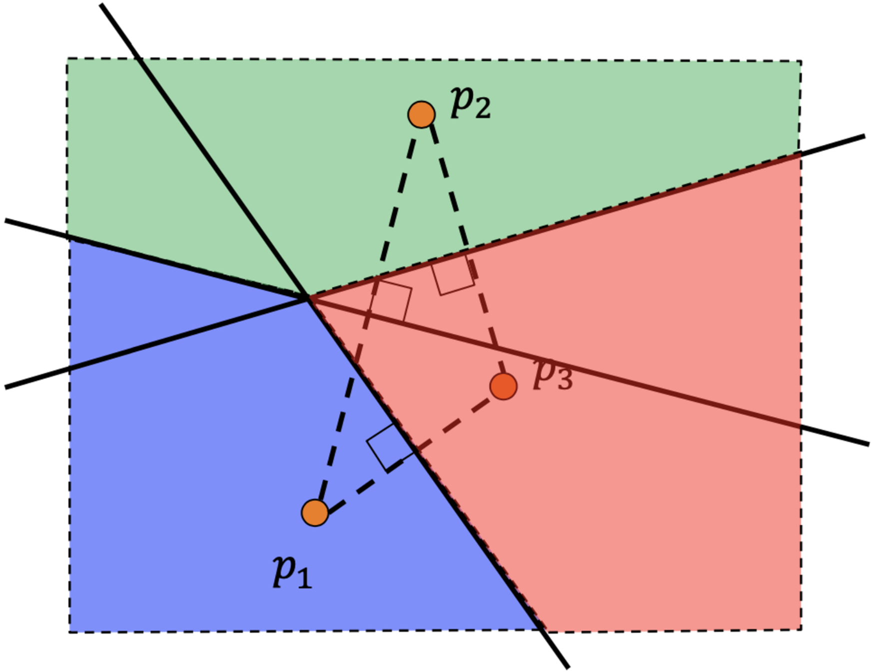
  $V_i = \bigcap_{j \in Z+ \leq n} \{H(p_i, p_j)\}$ = Voronoi region associated with p_i.

- Hence, Voronoi regions are non-overlapping convex polygons.



Simple Examples with Dominance Regions

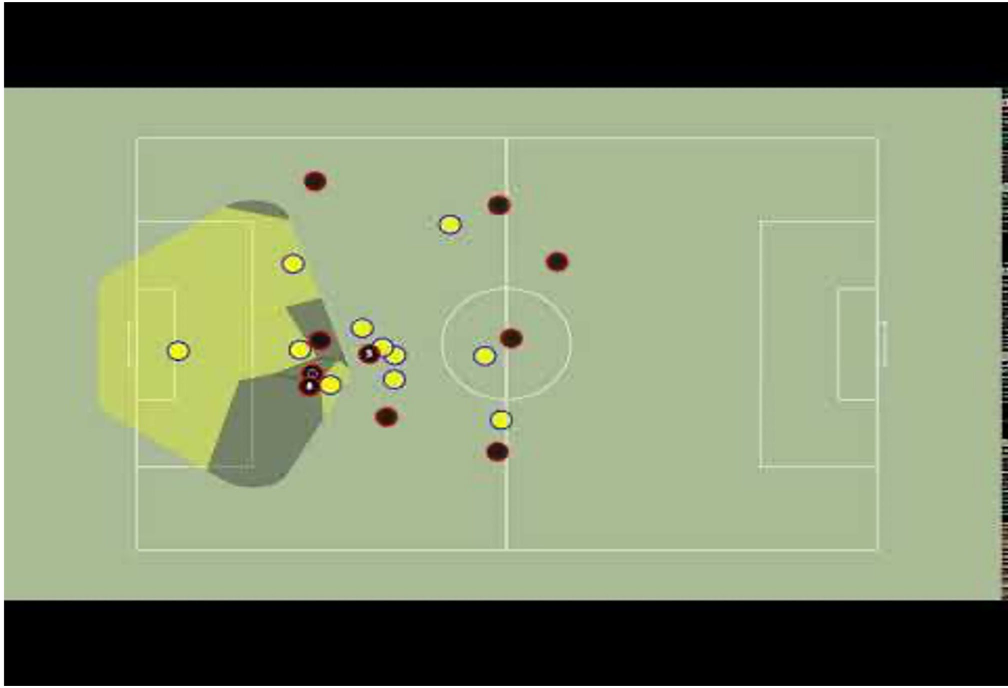




Application Example: Sports (Soccer)

- If the goal of the defender is to get anywhere in the field faster than the opposing player, the best way of achieving that is **to be closer than the attacker to the dangerous areas.**
- A video representation of the 5th goal in Brazil’s 7–1 defeat against Germany at 2014 FIFA World Cup:

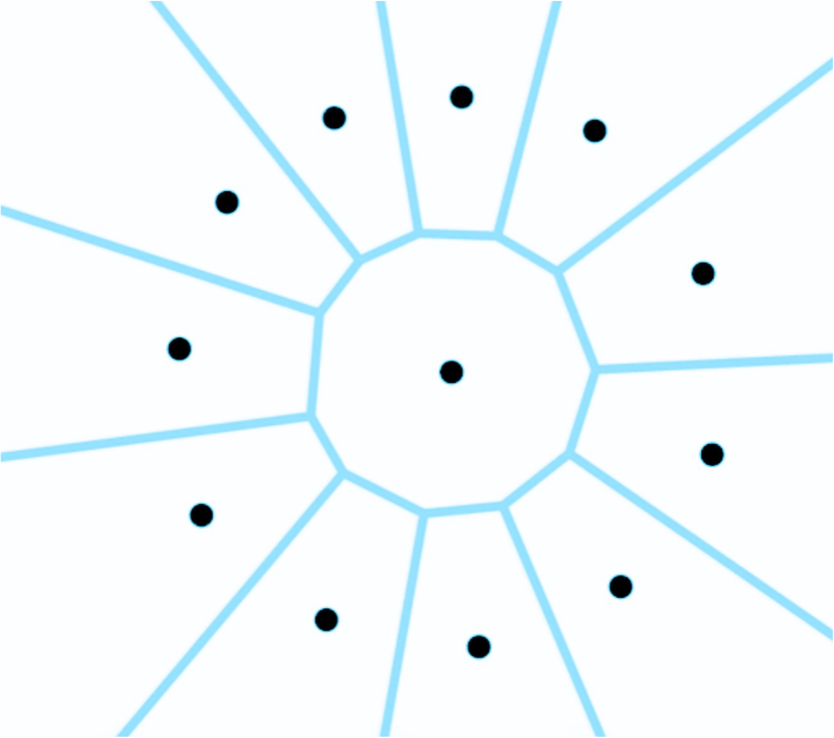
1. David Luiz, the Central Defender closer to the goal, covers an area bordering three attackers. Not optimal.
2. He leaves his position mid-play trying to intercept the ball.
3. This gives away the area in front of the box to the attackers.
4. He then misses his tackle.
5. From here (skipping some details) the remaining defenders and goalie get overwhelmed.
6. Area in front of goal changes from mostly yellow to mostly grey.



- Idea of **voronoi regions to analyse player positioning** can be used in many other sports.

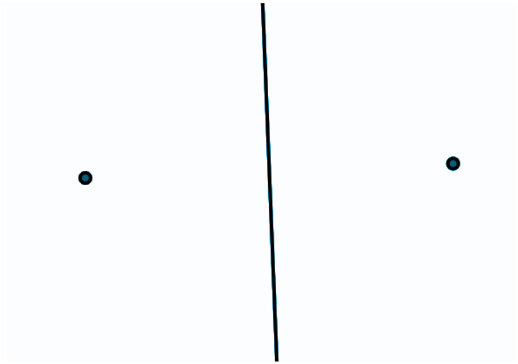
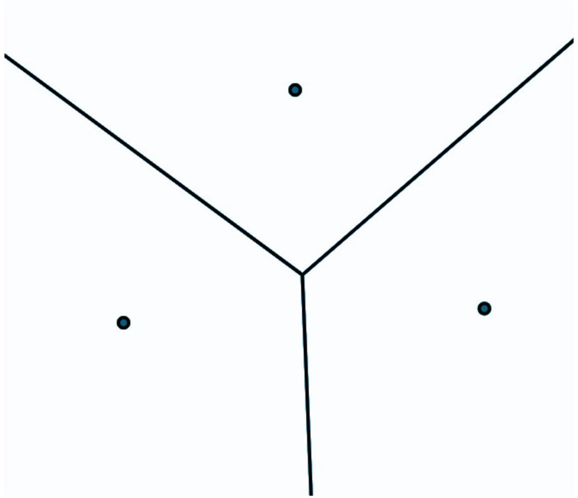
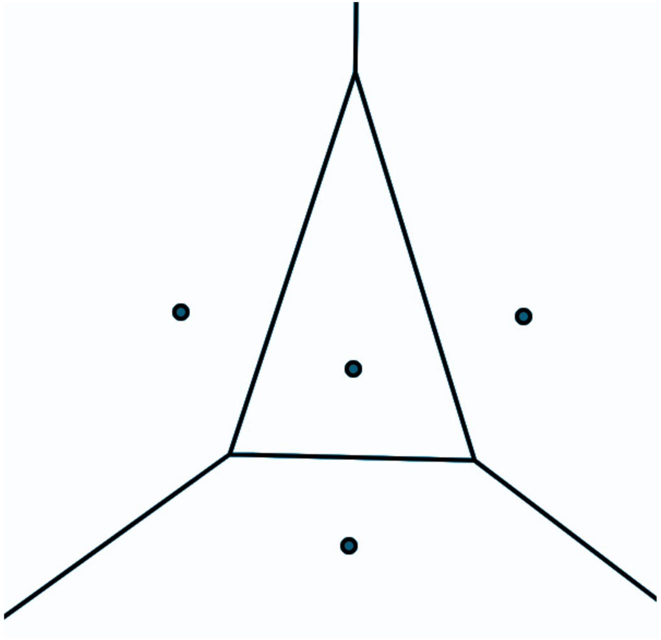
Q: How many edges can a voronoi region have?

- A:
  - When a point is surrounded by all of the other points, its voronoi region will have n-1 sides.



Q: Can all regions have n-1 edges?

- A:
    - This is possible up to n=4.
    - NB: After that this fails due to Euler's Formula.

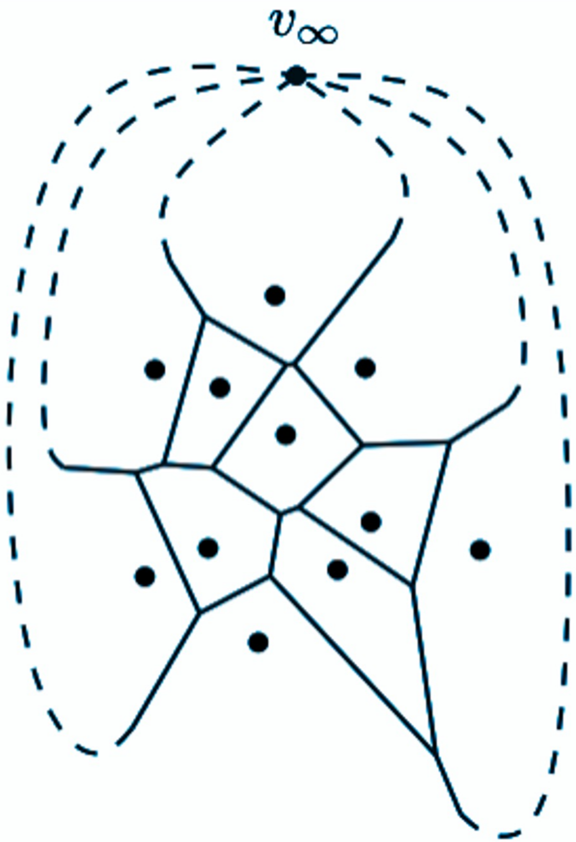




Graph Theory and Voronoi Diagrams

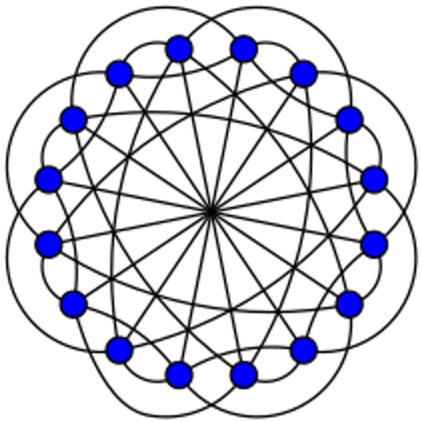
- Connected graph if points not all collinear.
- Let n, n_e, and n_v be the number of generator points, Voronoi edges, and Voronoi vertices of a Voronoi diagram, respectively.
  - $n_v - n_e + n = 1$ (Euler's Formula...)
  - If P has more than 3 elements, $n_e \leq 3n - 6$ and $n_v \leq 2n - 5$
- Much more...

Euler’s Formula Adapted to VDs

- Euler’s formula for plain-connected graphs is :
  - V - E + F = 2.
- Edges must be ‘finite’/bounded by vertices at the ends.
  - This is not the case in voronoi diagrams of points in the plane.
  - Need to add extra vertex ‘at infinity’ to connect to the endpoints of infinite edges.
  - Then the formula holds, with extra vertex accounted for:
    - $(n_v + 1) - n_e + n = 2$



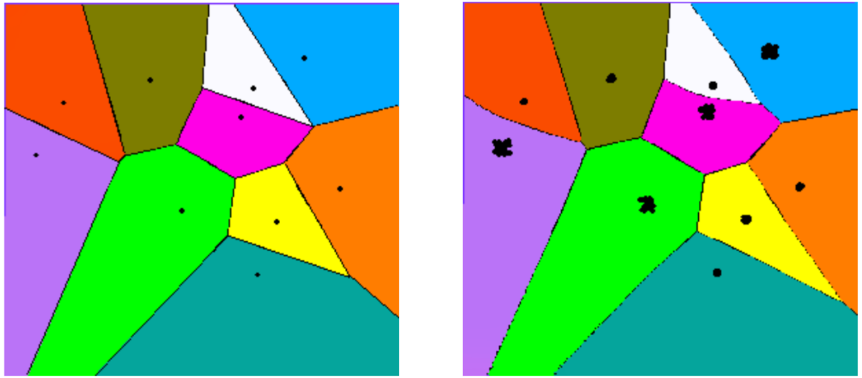
Proof of n_e and n_v Upper Bounds



- Sum of all degrees of vertices is twice the number of edges (handshaking lemma).
- Every vertex has degree at least 3.
  - Thus, $2*n_e \geq 3*(n_v + 1)$ [1].
- $n_v - n_e + n = 1$ (Euler's Formula).
  - $n_v + 1 = (2 - n) + n_e \geq (2 - n) + (3/2)*(n_v + 1)$ which gives $n_v + 1 \leq 2*(n - 2)$.
  - Hence, $n_v \leq 2n - 5$.
- Combining with eq. [1] gives $n_e \leq 3n - 6$.

Geometric Stability of Voronoi Diagrams

- Q: Does a small change of the sites, e.g., of their position or shape, yield a small change in the corresponding Voronoi cells?
  - In reality, we get approximate voronoi diagrams due to lack of information or errors.
    - Want to know if our VDs are close to reality…
      - Shape of sites not perfect points.
      - Exact locations of sites not known.

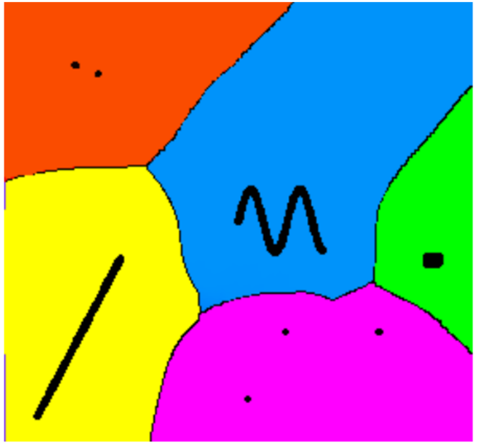


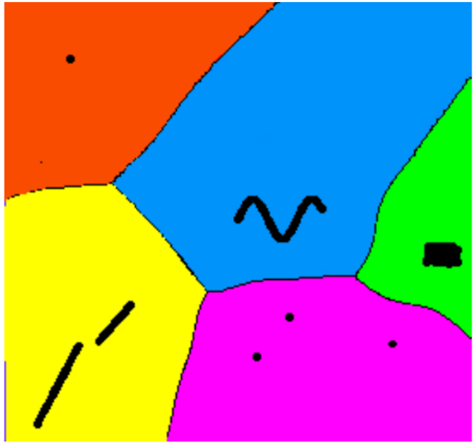
- Formalised Q:

Suppose that $(P_k)_{k\in K}$ is a tuple of non-empty sets in X. Let $(R_k)_{k\in K}$ be the corresponding Voronoi diagram. Is it true that a small change of the sites yields a small change in the corresponding Voronoi cells, where both changes are measured with respect to the Hausdorff distance? More precisely, is it true that for any $\epsilon > 0$ there exists $\Delta > 0$ such that for any tuple $(P'_k)_{k\in K}$, the condition $D(P_k, P'_k) < \Delta$ for each $k \in K$ implies that $D(R_k, R'_k) < \epsilon$ for each $k \in K$, where $(R'_k)_{k\in K}$ is the Voronoi diagram of $(P'_k)_{k\in K}$?

Geometric Stability of Voronoi Diagrams

- Conditions on space (not just Rd):
  - Uniformly convex.
  - Normed.
  - Induced distance metric.
- Conditions on points:
  - Closed and convex set.
  - Bound on the distance between neighbouring sites.
  - Uniform bound on diameter of sites.

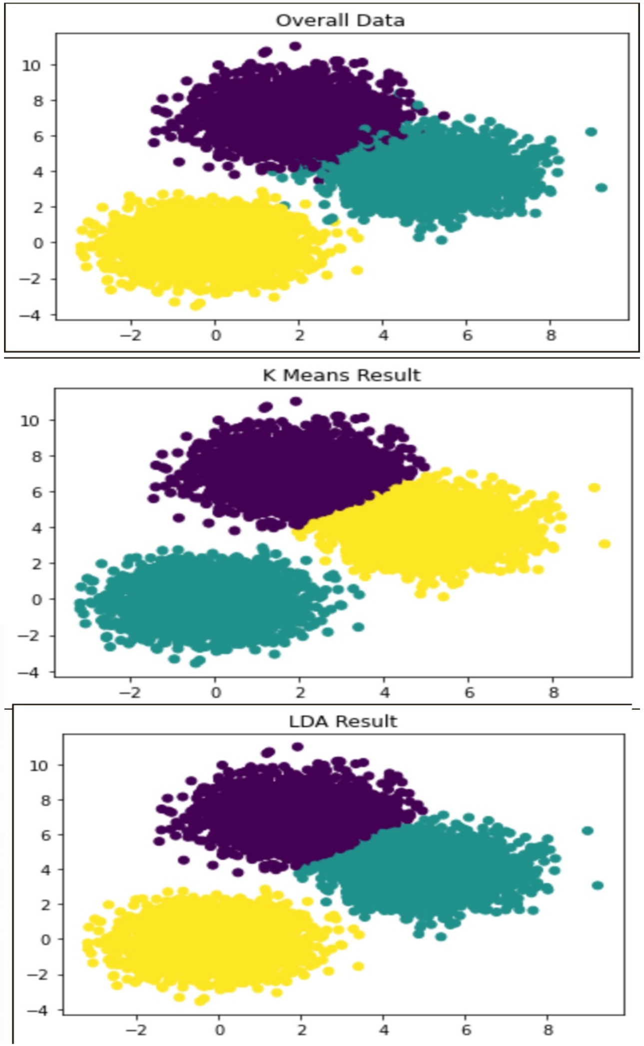




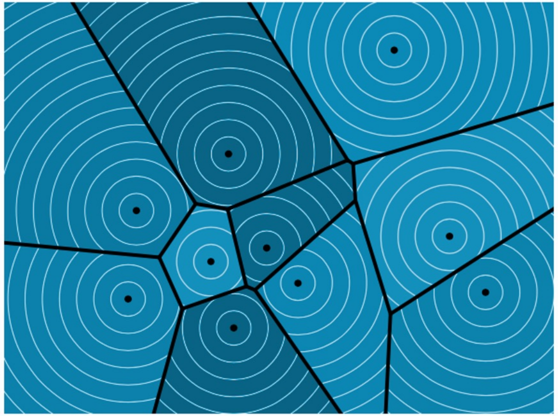
- Informal answer:
  - If there is a common positive lower bound on the distance between the sites, and the distance to each of them is attained,
    - Small enough change of the (possibly infinitely many) sites yields a small change of the corresponding Voronoi cells.
      - Both changes are measured with respect to the Hausdorff distance.
    - In other words, the shapes of the real cells and the corresponding perturbed ones are **almost the same.**

Application Example: Data Analysis

- LDA and K-Means
  - Voronoi Diagrams can be used to visualise K-means clustering and LDA.
  - In LDA with many classes, given some assumptions of uniformity on the priors of the data, the decision boundaries produced form a straight edge VD.



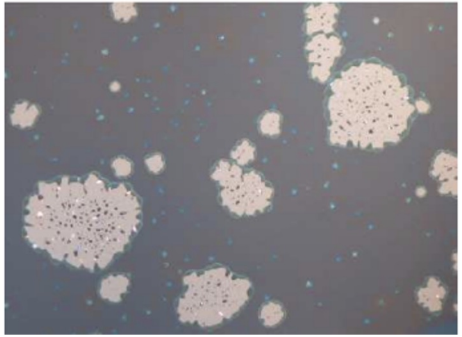
- Issues/complexity for higher d data...
- Other variations/extensions of VDs more useful for applications (CVD, Additively-weighted VDs, Approximate VDs, Higher Order).

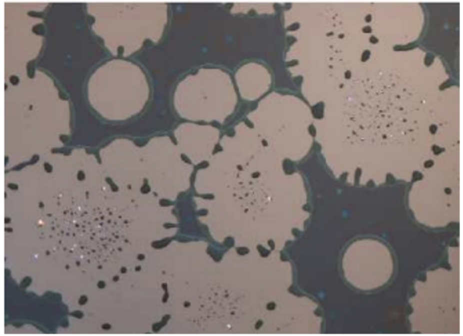


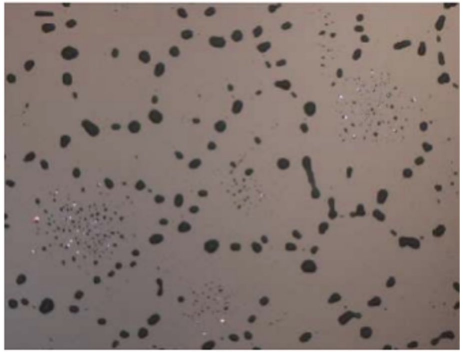
Application Example: Voronoi Growth Models

- N points (nucleation sites), P = $\{p_1, \cdots, p_n\}$, at positions $x_1, \cdots, x_n$, in R2 or a bounded region of R2 .
- To get ordinary Voronoi diagram V(P) of P:
  - Assumption 1) Each point p_i ($i = 1, \cdots, n$) is located simultaneously.
  - Assumption 2) Each point p_i remains fixed at x_i throughout the growth process.
  - Assumption 3) Once p_i is established, growth commences immediately and at the same rate l_i in all directions from p_i.
  - Assumption 4) l_i is the same for all members of P.
  - Assumption 5) Growth ceases whenever and wherever the region growing from p_i comes into contact with that growing from p_j ($j \neq i$).
- Together, assumptions 1-5 define the Voronoi Growth Model.

- An example application: Model crystal growth about a set of nucleation sites.
- Here, assumptions 1-5 are equivalent to assuming:
  - ‘an omni directional, uniform supply of crystallizing material to all faces of the grind crystal in the absence of any absorbable impurities.’
  - Assumption 3 also implies that the rate of growth of the volume of a crystal will be proportional to its surface area.



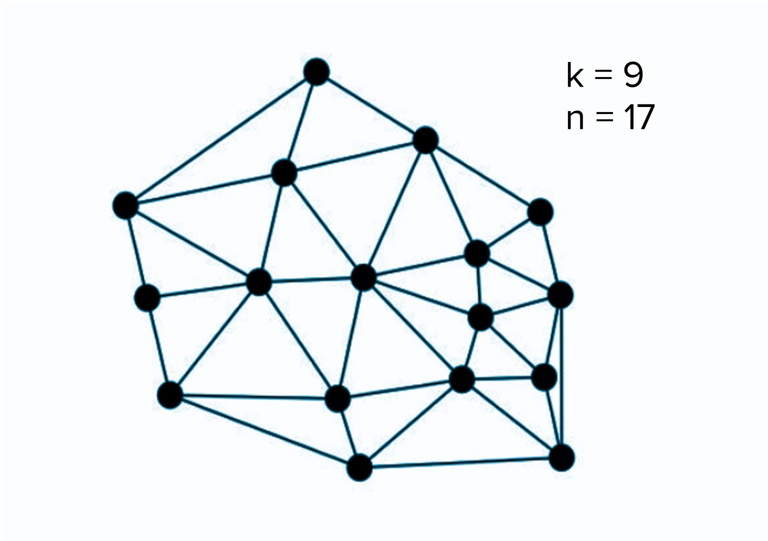




Triangulations

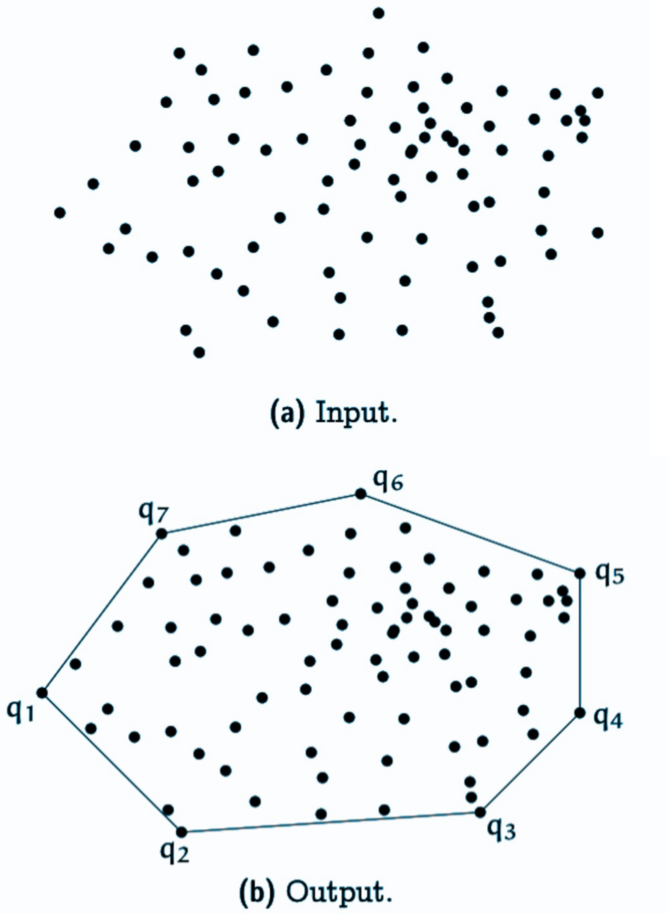
- A triangulation of P is a planar subdivision whose faces are triangles and whose vertices are the points of P.
- P has many triangulations and they all have the same number of triangles and edges.

- More precisely, assuming that the number of points on the convex hull of P is k, any triangulation of P has,
  - 2n−k−2 triangles.
  - 3n−k−3 edges.

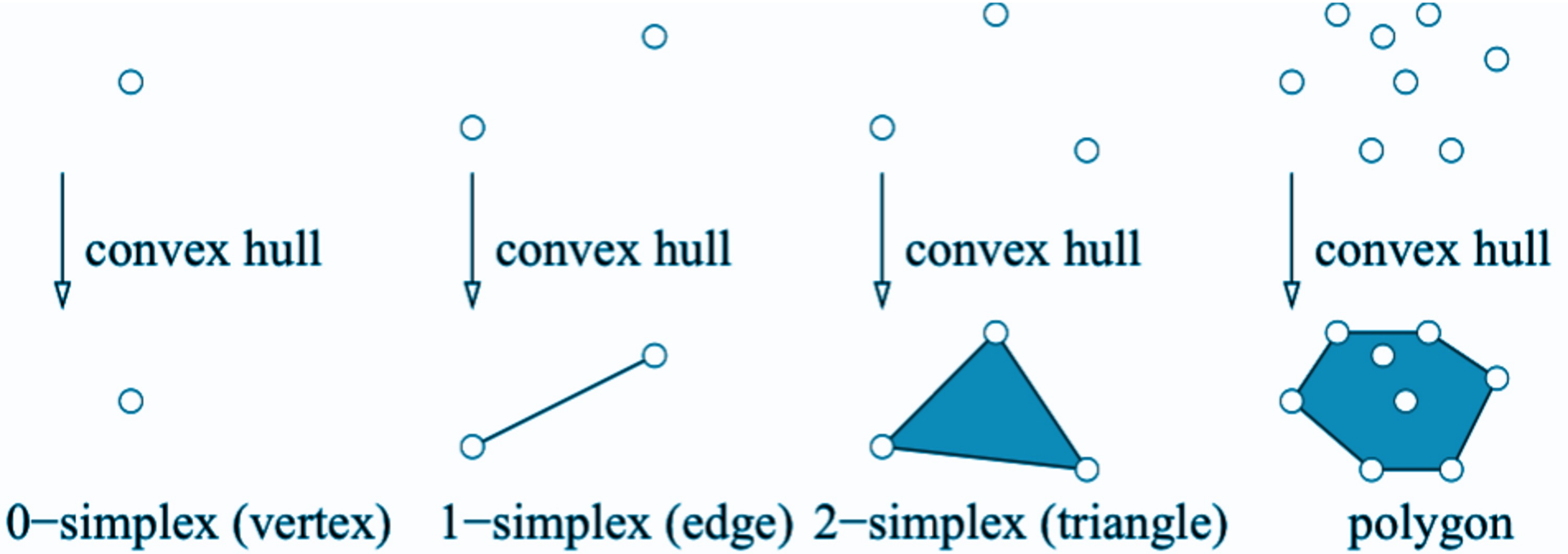


Convex Hull of P

- For a set of points P, the convex hull of P or CH(P) is defined as
    - Smallest (area) convex set that contains all of the points.
    - The intersection of all convex sets containing the points.
    - Set of all convex* combinations of points.
        - A linear combination of points is ‘convex’ if all coefficients are non-negative and sum to 1.
    - The union of all simplices* with vertices as the points of P.
        - ‘Simplices’: Point, line-segment, triangle, tetrahedron…
- Given a finite set of points, and a triangulation of the corresponding convex hull (having the points as its set of vertices):
    - For points in R2, the boundary is a convex polygon.
    - For points in R3, the boundary is a convex polyhedron.

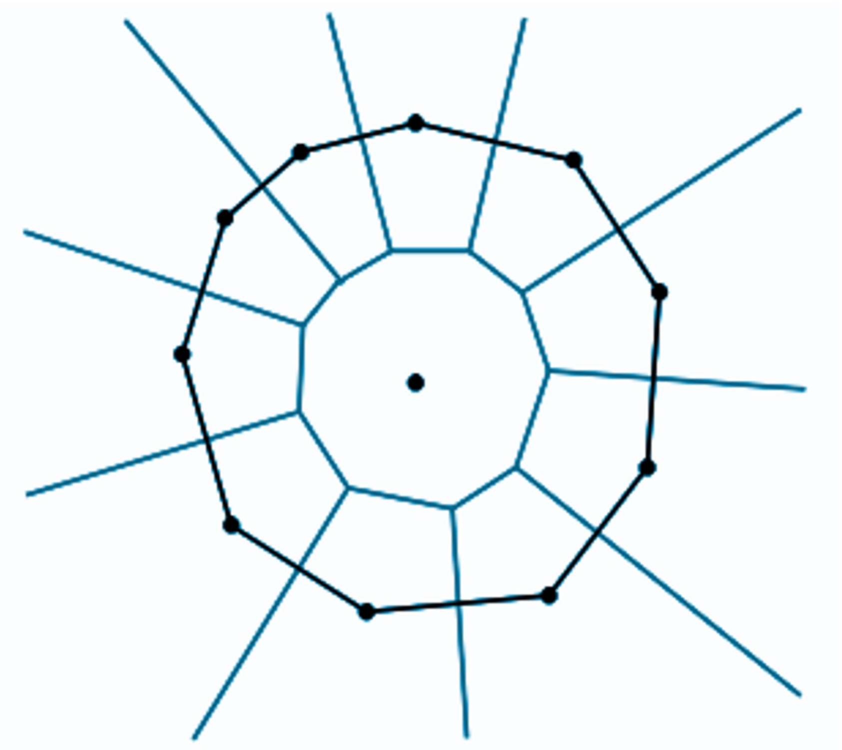


Examples of Simplices and Convex Hulls



Voronoi Regions of Sites on CH(P)

- Regions can be bounded or unbounded.
- Every point contained in an unbounded region of the diagram lies on the convex hull of the set P.
- Example with one bounded region and n-1 unbounded regions.

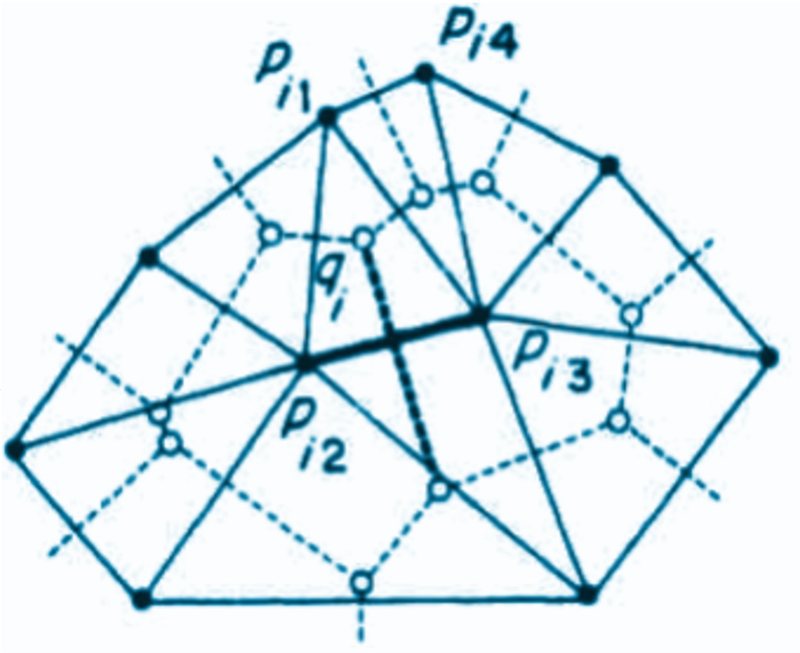


Delaunay Triangulations

- **Boris Nikolayevich Delaunay** or **Delone** March 15, 1890 – July 17, 1980 .
- Theorem [Delaunay, 1934]: The straight-line dual of Vor(P) is planar, and is a triangulation.
- The Delaunay tessellation D(P) is obtained by connecting with a line segment any two points p_1 , p_2 of P for which a circle C exists that passes through p_1 and p_2 and does not contain any other site of P in its interior or boundary.

Delaunay Triangulations From VD in R2

- Assume non-collinearity. Given a Voronoi diagram V(P), join all pairs of generator points whose Voronoi regions share an edge.
  - In the Delaunay tessellation of P, or D(P), there exists the edge p_ip_j if and only if $e(p_i,p_j) \in V(P) \neq \emptyset$.
  - If this tessellation consists of only triangles, we call it a Delaunay triangulation.
    - Iff all vertices of V(P) are non-degenerate/points in 'general position'.
- (assuming non-degeneracy) Voronoi edges and Delaunay edges are orthogonal*.

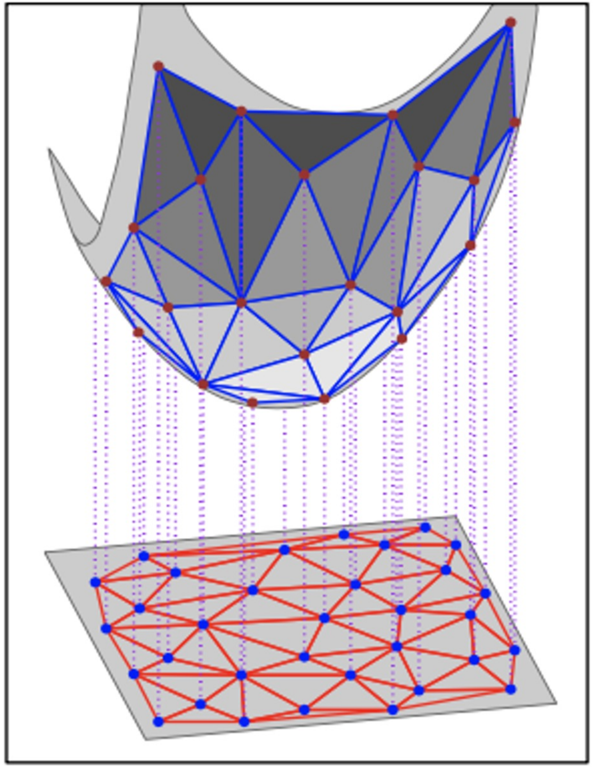


- Voronoi diagram (dashed lines) and a Delaunay triangulation (solid lines) of the same generator set P (solid dots).

Delaunay to Convex Hull

- Union of external edges in D(P) is boundary of convex hull of P.
  - Q-hull algorithm for D(P).

- Let Ψ be the paraboloid $z = x^2 + y^2$. For any point $p = (p_x, p_y)$ in R2, define the vertical projection/lifted image of p onto Ψ to be point $p^{\uparrow} = (p_x, p_{y,}, p^2_x + p^2_y)$ in R3. Given a set of points P in the plane, let $P^{\uparrow}$ be the projection of every point in P onto Ψ.
- Let the *lower convex hull* of $P^{\uparrow}$ be the part of the convex hull visible to an observer at $z = -\infty$.
- (Theorem) Let P be any set of points in the x-y plane. Then the projection of the *lower convex hull* of $P^{\uparrow}$ back onto the plane is the Delaunay triangulation of P.
  - Proof: Must show equivalence of Delaunay and LCH conditions.



Delaunay and Lower Convex Hull Conditions

- Delaunay:

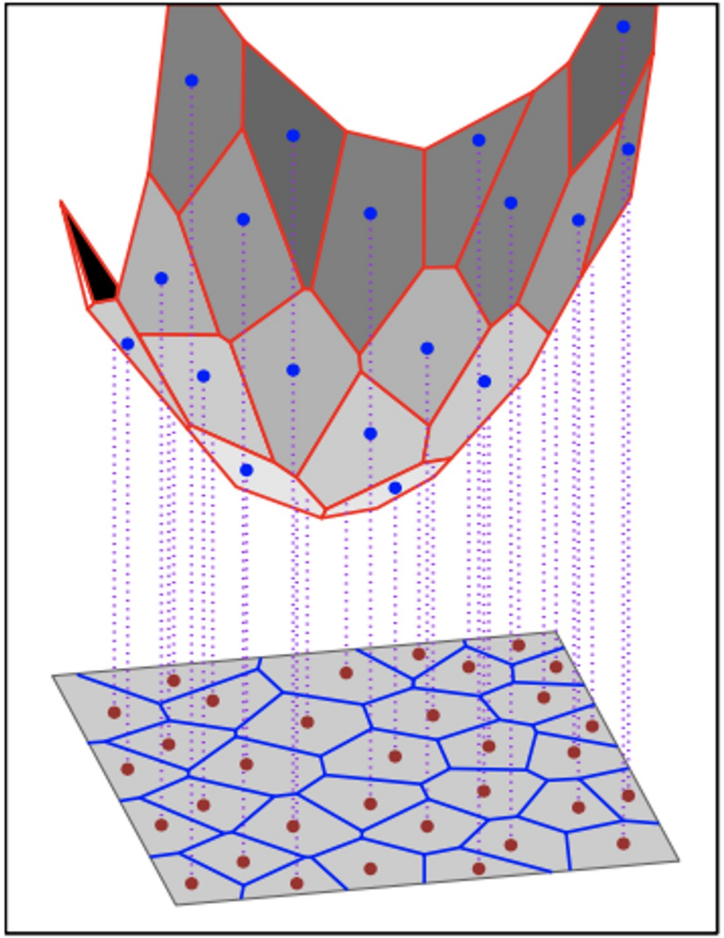
Three sites in P form a Delaunay triangle if and only if no other site of P lies within the circumcircle defined by the sites.

- Lower Convex Hull:

Three points $p^{\uparrow}$, $q^{\uparrow}$, and $r^{\uparrow}$ in $P^{\uparrow}$ form a face of the LCH of $P^{\uparrow}$ if only if no other point of $P^{\uparrow}$ lies below the plane passing through $p^{\uparrow}$, $q^{\uparrow}$, and $r^{\uparrow}$.

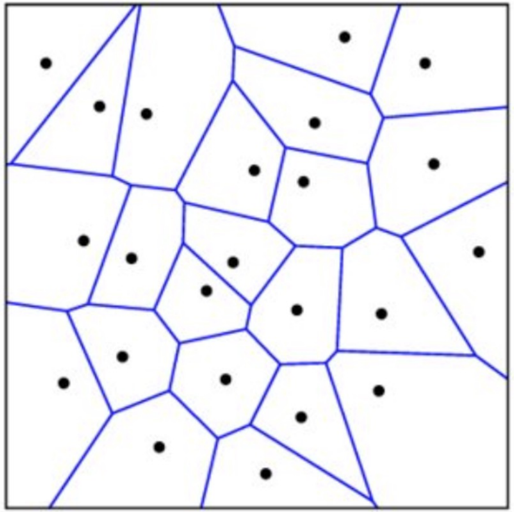
Voronoi to Convex Hull

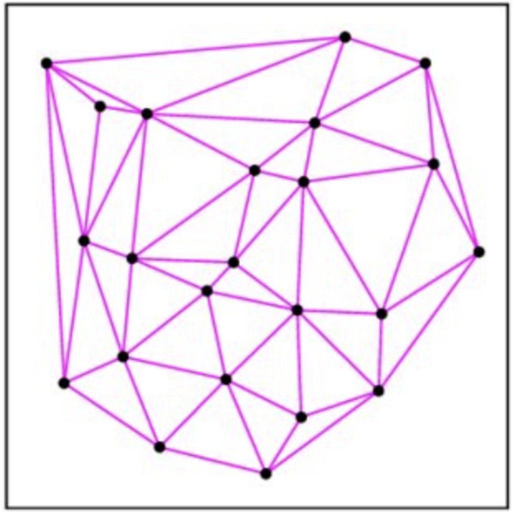
- (Theorem) Given a set of points P in the x-y plane, let U(P) be the upper envelope of the tangent hyperplanes passing through each point $p_i^{\uparrow}$ for p_i in P.
  - Then the VD of P is equal to the vertical projection on the x-y plane of the boundary complex of U(P).



Correspondence Between Voronoi and Delaunay

- Given a non-degenerate Voronoi diagram V(P) and a Delaunay triangulation D(P),
  - Let Q and Qd be the sets of Voronoi vertices and Delaunay vertices, respectively.
  - Let E and Ed be the sets of Voronoi edges and Delaunay edges, respectively.
  - Let Cd be the set of circum- centers of Delaunay triangles.
- Then the following holds:
  - (i) Qd =P
  - (ii) Cd = Q
  - (iii) |Ed| = |E|

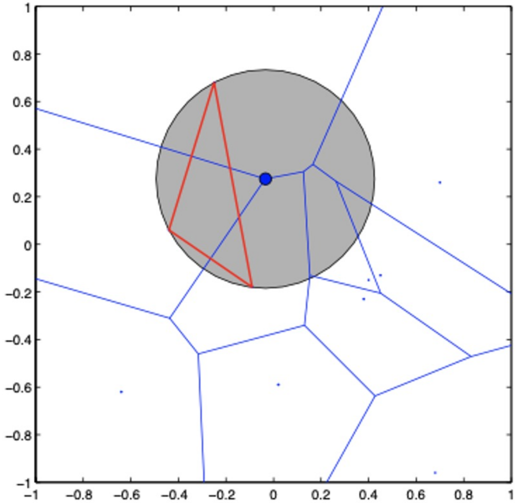




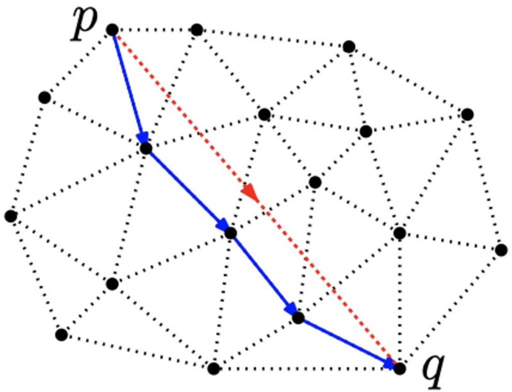
- (i) Says that each generator point p_i is a vertex of a Delaunay triangle.
- (ii) Says that the circumcenter of each Delaunay triangle corresponds to a Voronoi vertex.
- (iii) Says that the number of Delaunay edges is equal to the number of Voronoi edges.

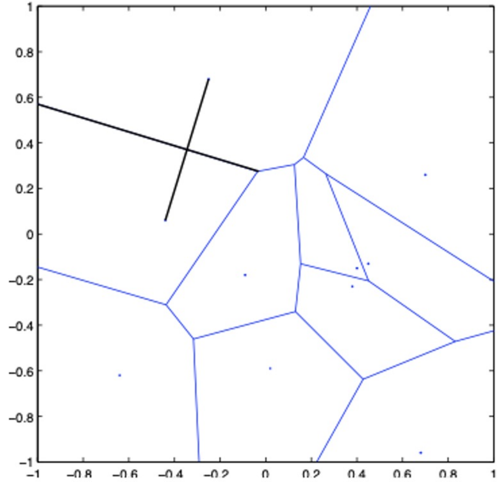
DTs and VDs on R2: Properties

- Voronoi Diagrams:
  - Each triangle of the Delaunay triangulation is associated with a single vertex of its dual Voronoi tessellation (ii on previous slide). That Voronoi vertex is located at the center of the circumscribed circle of the triangle. (Circumcircle property)
  - Each cell edge of the Voronoi tessellations is uniquely associated with one cell edge of the dual Delaunay triangulation (iii on previous slide);
    - If the pair of edges intersect (or if the lines segments are extended to a point where they intersect), then the intersection point will bisect the line segment connecting generators. (Bisection property)



- Delaunay Triangulations:
  - Let D(pi,pj) be the shortest path along the Delaunay edges of D(P) from pi to pj. Then, D(pi,pj) ≤ c ∗ d(pi,pj), where c ≈ 2.42 and d(pi,pj) is Euclidean distance.
  - The Delaunay triangulation maximizes the minimum angle;
    - i.e., compared to any other triangulation of the points, the smallest angle in the Delaunay triangulation is at least as large as the smallest angle in any other.
    - This property does not hold in higher dimensions.



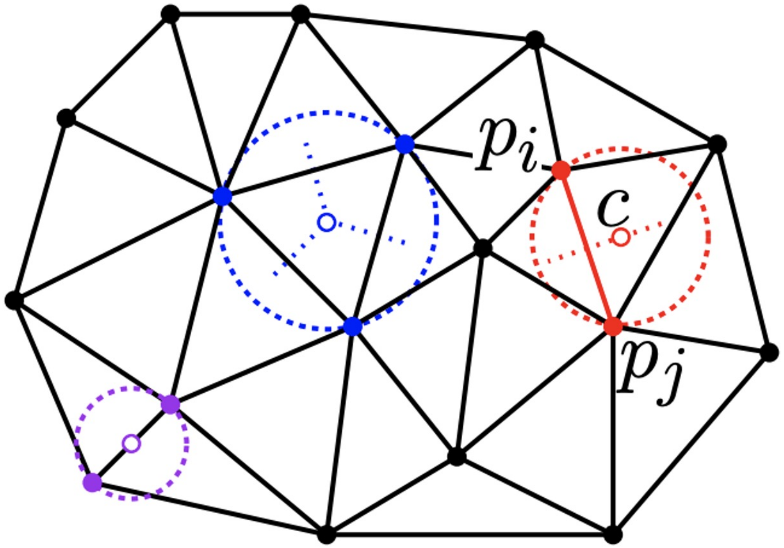


Empty Circle and Closest Pair Properties (DT)

- Empty Circle:

  For every triangle in the DT of P, its circumcircle does not contain any other points of P inside and is centered at the corresponding voronoi vertex.

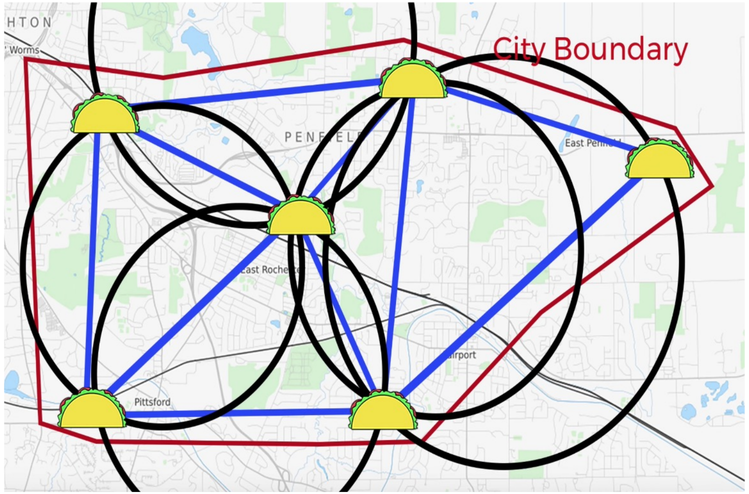
- If every triangle in a triangulation of P has the empty circle property, then the triangulation is the Delaunay triangulation (Delaunay lemma).
- Closest Pair of points are always neighbours in the Delaunay Triangulation.

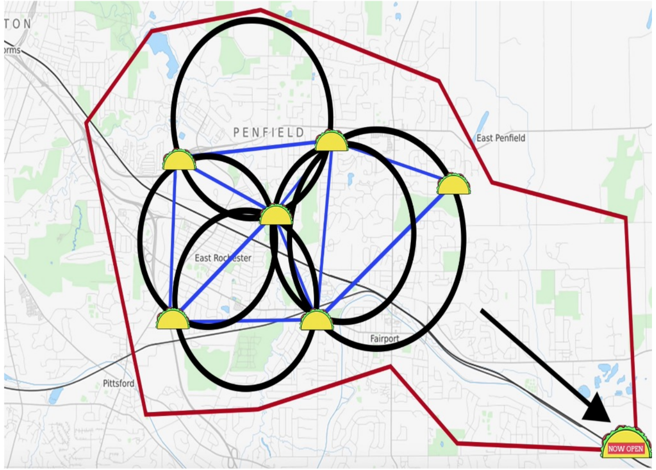


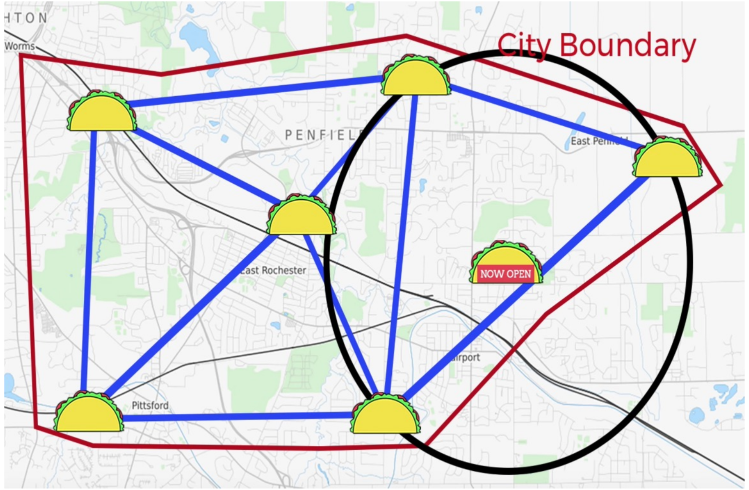
- Empty Circle shown in red, Circumcircle shown in blue, and Closest Pair shown in purple.

Empty Circle in Practice

- Largest empty circle use case:
  - You run a taco truck company and want to decide where to place a new truck.
  - Ideally, you construct it as far as possible from existing locations, to decrease competition between your own trucks.
  - But it also has to be within the border of the area you operate in.
    - Solution: At the centre of the largest empty circle in the DT with existing trucks as vertices.
    - Caveat: If boundary extends beyond all empty circles, then pick furthest point on boundary.



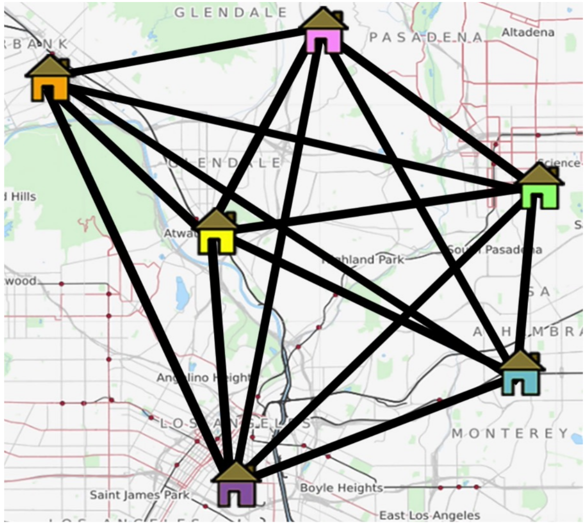


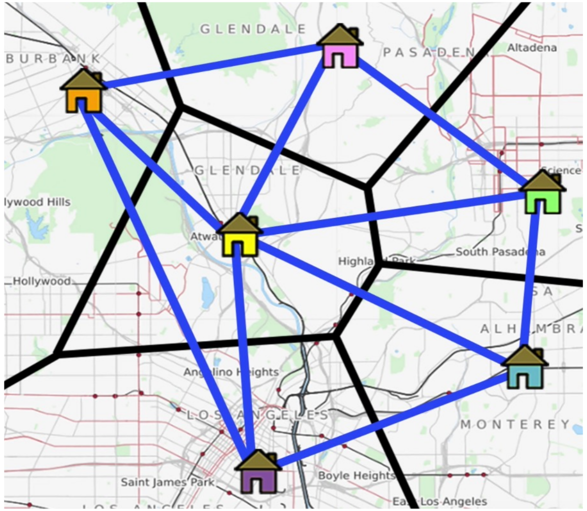


Closest Pair in Practice

- A group of friends debates which pair lives closest:
  - Comparing all distances between pairs of houses is long...
  - Instead, find the shortest edge of DT with houses as vertices.
- This works since DT edges connect every house to its closest neighbour.
- For large n this makes a significant difference.

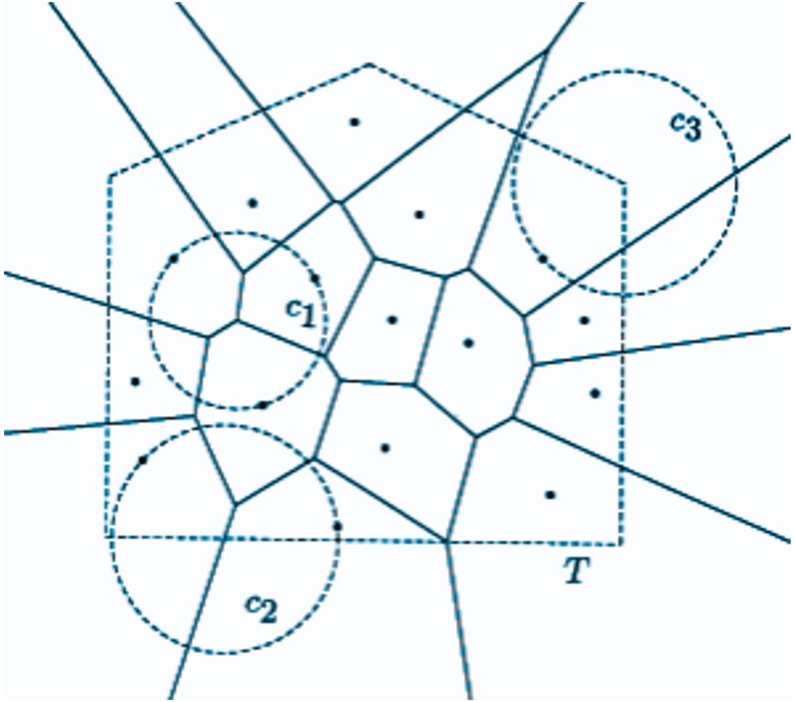
n	n(n-1)/2	3n - 6
10	45	24
100	4,950	294
1,000	499,500	2,994
10,000	49,995,000	29,994





Voronoi Regions Form a Decomposition of the Plane

- Let p_i be an arbitrary point in the plane. We center a circle, C, at p_i and let its radius grow from 0.
- At some stage the expanding circle will, for the first time, hit one or more generators/sites. There are three different cases:
    1. If C hits three or more sites, then p_i is a Voronoi vertex.
    2. If C hits exactly two sites, p_s and p_z, then p_i lies on a Voronoi edge separating the voronoi regions of p_s and p_z.
    3. If C hits exactly one site, p_s, then p_i belongs to the V_s .



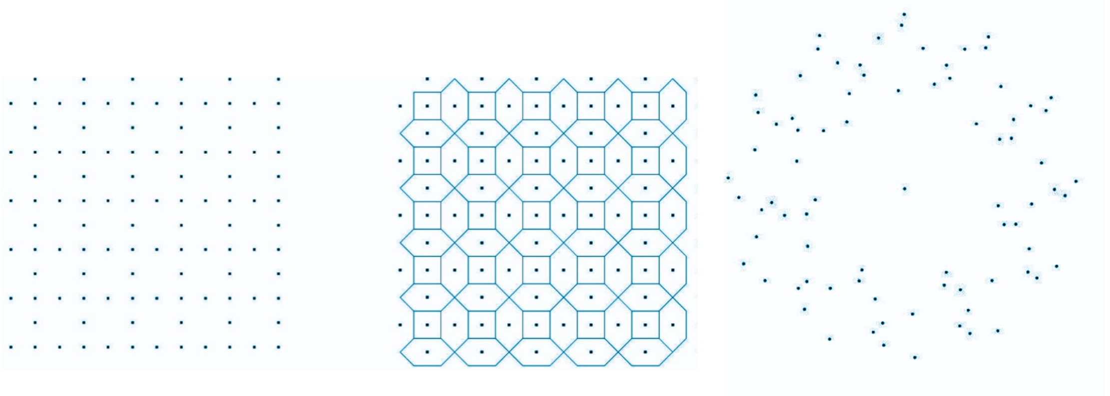
Voronoi Tessellations

- Voronoi Regions:
  - Cover the space*.
    - Finite number of points...
  - Do not overlap.

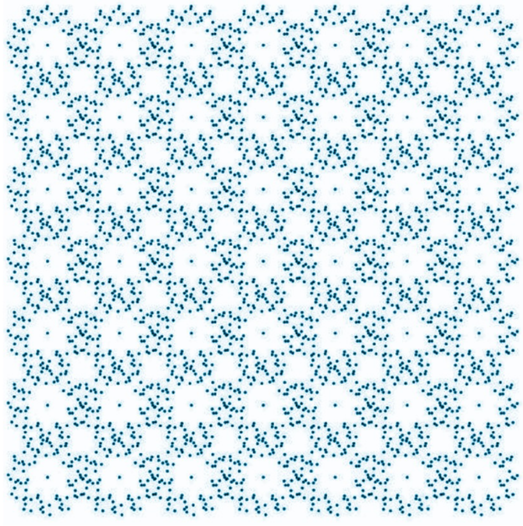
- A ‘regular’ arrangement of the seed points guarantees that all of the voronoi regions are the same shape.
  - Single prototile criterion is met.

- Semi-regular set of points gives a twin-tile tessellation of hexagons and smaller squares.

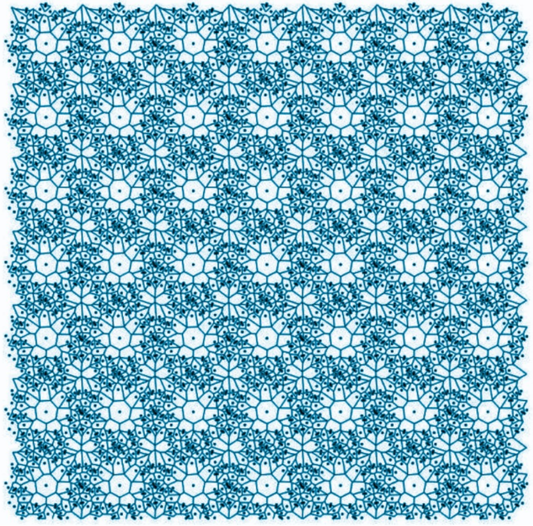
- Voronoi pattern arising from 7 × 7 XY translation of the pattern shown in Figure A is shown in Figure B.
  - Consists of eight types of polygons.



A

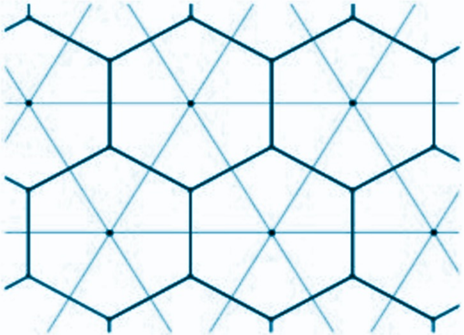


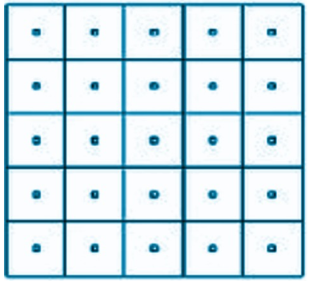
B

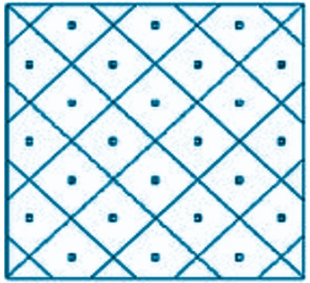


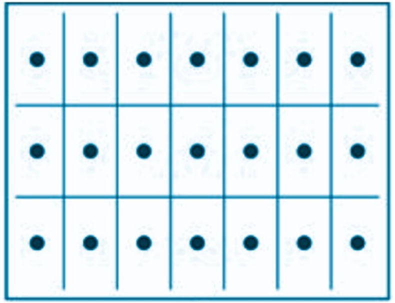
From Regular Lattices of Points

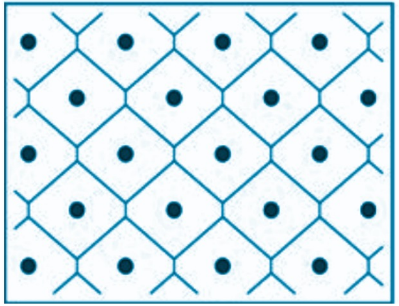
- The Voronoi tessellation of __ lattice points:
  - Triangular, forms a grid of regular hexagonal cells.
  - Square, forms a grid of squares.
  - Rectangular, forms a grid of rectangles.
  - Rhombic, forms a grid of rhombus-like hexagonal cells.
  - Oblique, forms a grid of elongated hexagonal cells.
  - Honeycomb (non-lattice), forms a grid of equilateral triangles.

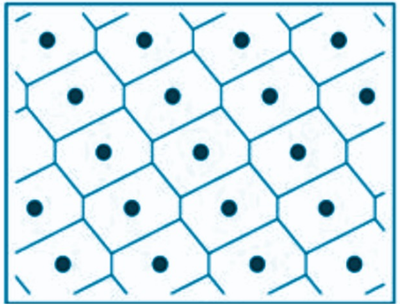


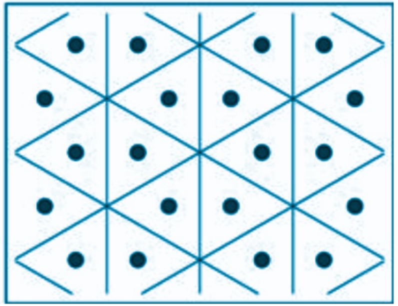












Regular vs Semi-regular Tilings (Reminder)

- Regular = use one convex regular polygon prototile.
- Semi-regular = use a set of two or more convex regular polygons as prototiles.
  - Also, every vertex has to have the same arrangement/order of polygons around it.
- 3 regular tilings of the plane are:
  - with squares, triangles, and hexagons.
- Dual = exchange the vertices and the faces*.
- Dual of semi-regular tilings of the plane:
  - Connect the centres of each tile and pass through the midpoint of each edge.
  - Only one vertex condition in the semi-regular tilings ⇒
    - The duals are each comprised of a single shape.
- Duals not always unique…

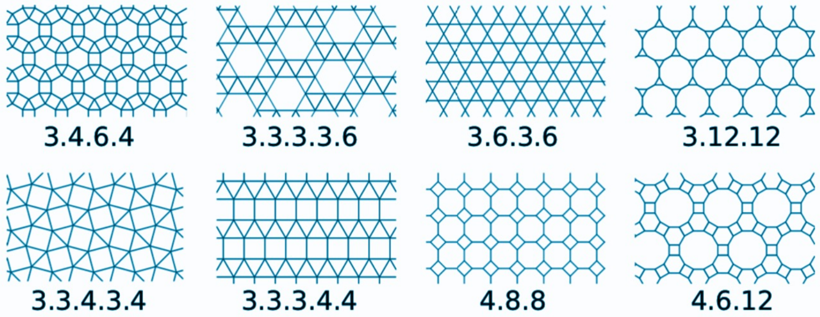


Figure 1: The eight semiregular tessellations

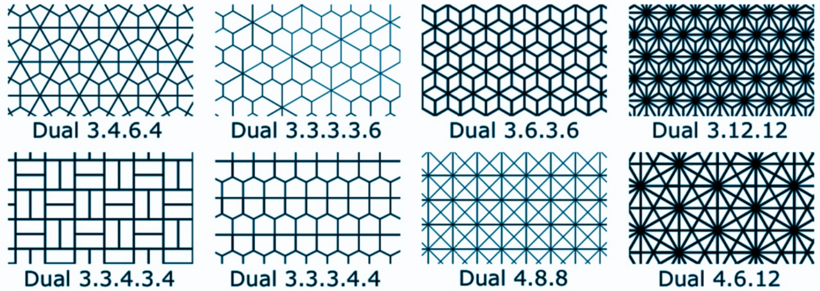


Figure 2: The duals of the eight semiregular tessellations.

11 Archimedean Tilings of the Plane Question

- Q: Which of the 11 Archimedean Tilings and their duals can produce **unique** DTs from the VDs of their vertices?

- Hint: conditions on arrangement of points for VD and for corresponding DT to be unique.
- A:
  - They are all excluded due to the conditions of **no three or more collinear** points and **no four or more points on a circle** for uniqueness of DT.

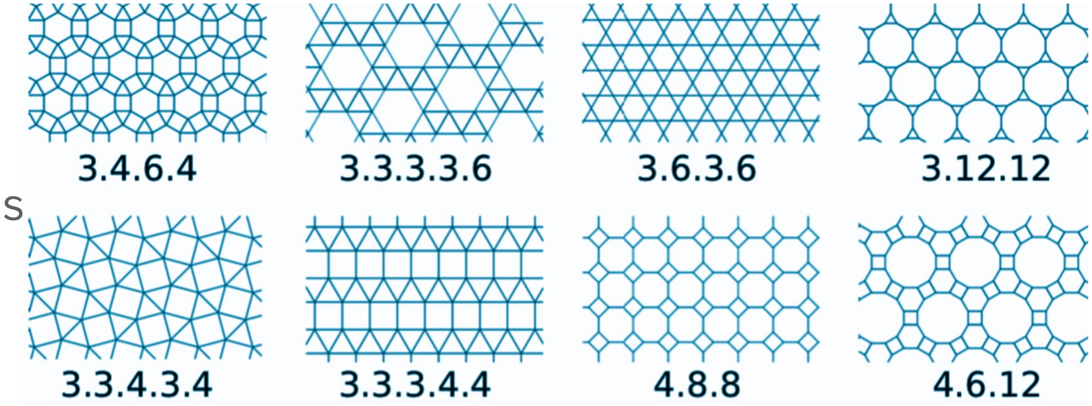


Figure 1: The eight semiregular tessellations

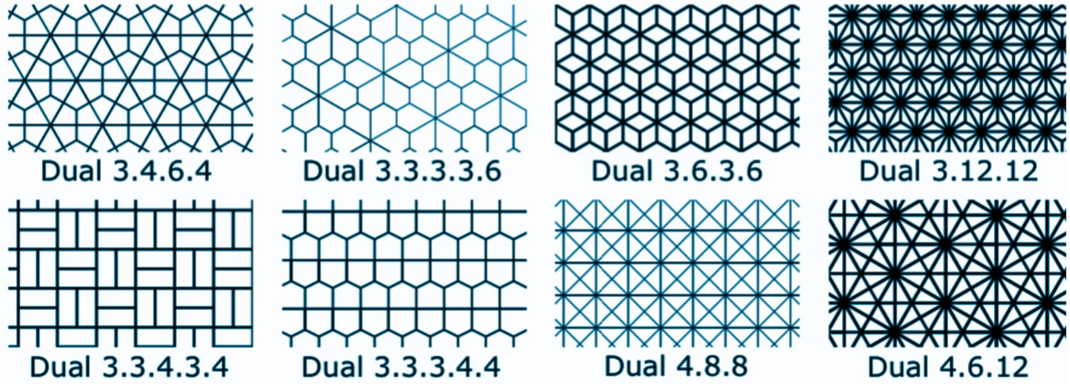
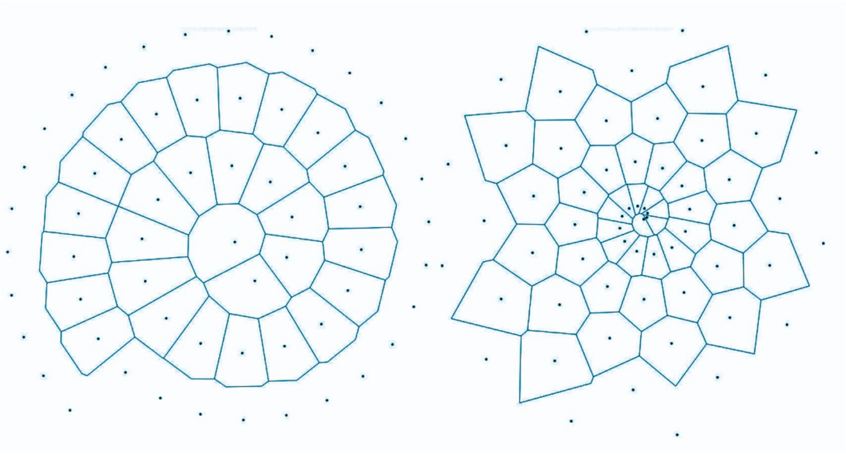
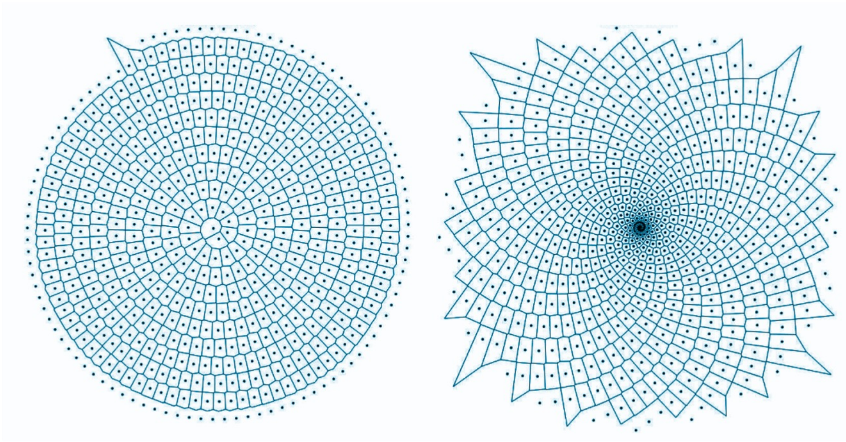


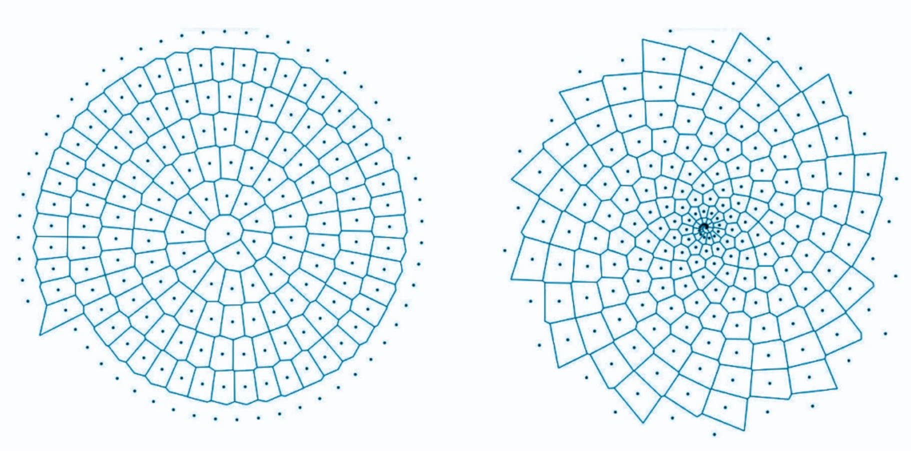
Figure 2: The duals of the eight semiregular tessellations.

VDs from Archimedean Spirals

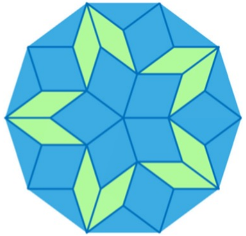
- Have aesthetic appeal and appear in nature a lot, phyllotaxis.
    - Useful for visualising and studying many natural processes.
    - Produce beautiful mosaic art.





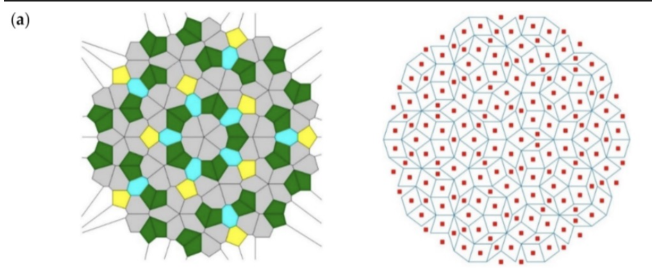


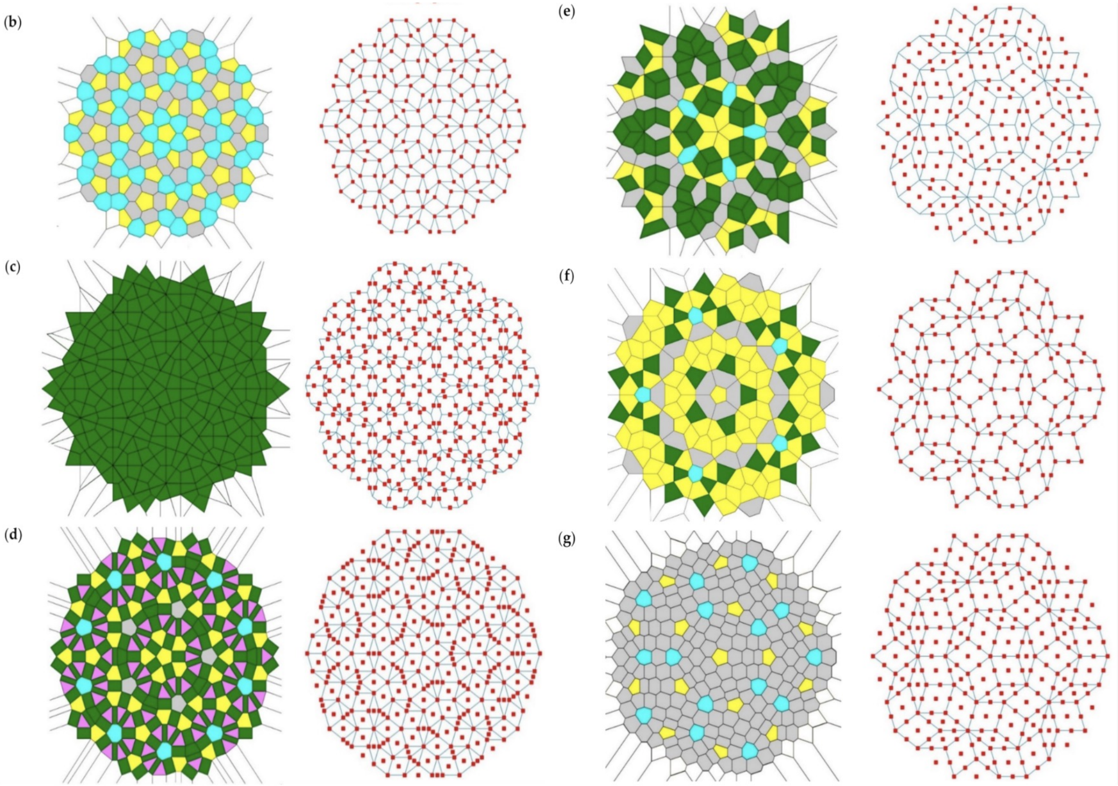
VDs from Penrose Tilings



- The Voronoi tessellations (right) and Penrose tiling (left).
  - Color mapping: magenta polygons are triangles, green are tetragons, yellow are pentagons, grey are hexagons, blue are heptagons.
- Three Examples of VDs generated by Penrose tilings:
  - *a*-type; centers of rhombs taken as seeds.
  - *b*-type; vertices of rhombs taken as seeds.
  - *c*-type; centers of the edges of rhombs taken as seeds.
- (d)-(g) shows combinations of seed points from three main types.
  - Note that *a-, b-, c-, ab-, ac-, bc-* and *abc*-type Voronoi diagrams possess the same groups of symmetry. Five-fold rotational symmetry and mirror plane symmetry, from Penrose tiling.

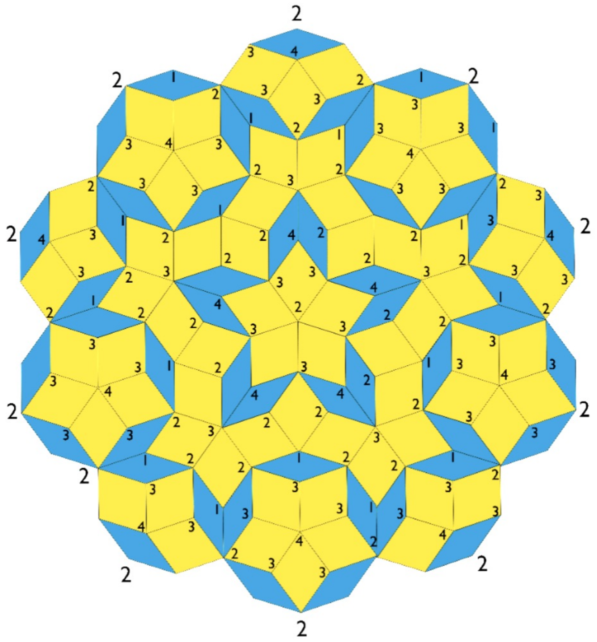
Diagram Type	Polygons Number, n_{pol}	Polygon Types Number
a	140	4
b	141	3
c	290	1
ab	375	5
ac	205	4
bc	161	4
abc	221	3

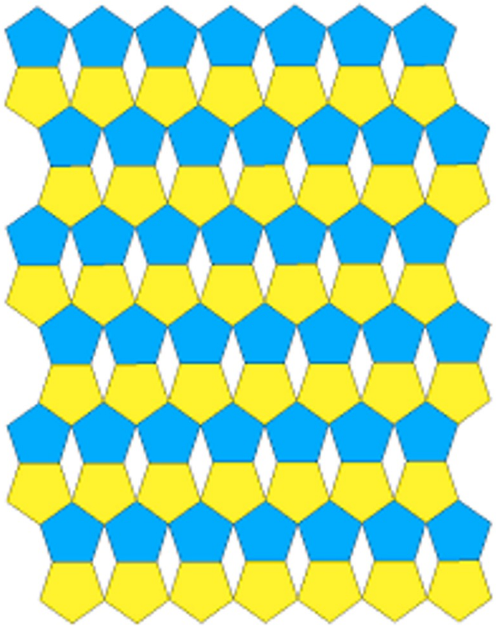


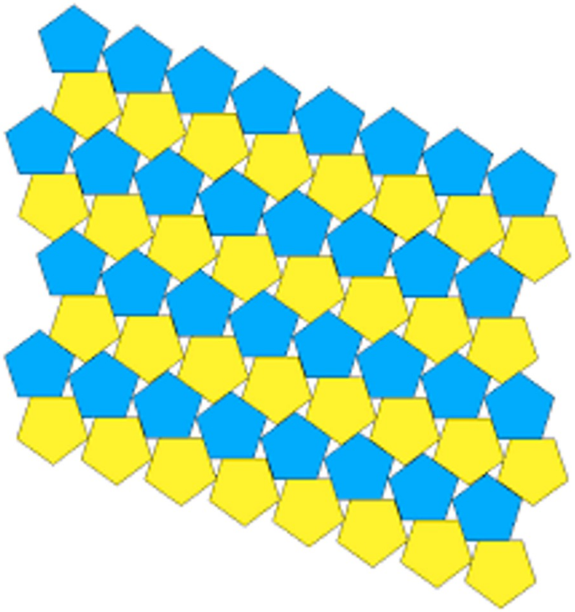


Aperiodic vs Periodic Tilings (Reminder)

- A tiling is *periodic* if there exists a translation that leaves the tiling invariant and it is non-periodic in any other case.
- A set of prototiles that always tiles non-periodically is called *aperiodic*.







ELSEVIER

Computational Geometry 23 (2002) 171–182

Computational Geometry
Theory and Applications

www.elsevier.com/locate/comgeo

An aperiodic tiles machine ☆

José Cáceres [a,*], Alberto Márquez [b]

[a] *Departamento de Estadística y Matemática Aplicada, Universidad de Almería, 04120 Almería, Spain*
[b] *Departamento de Matemática Aplicada I, Universidad de Sevilla, 41012 Sevilla, Spain*

Received 8 May 2001; received in revised form 31 July 2001; accepted 1 October 2001

Communicated by K. Mehlhorn

Aperiodic Tilings from VDs of Penrose Point Sets (1)

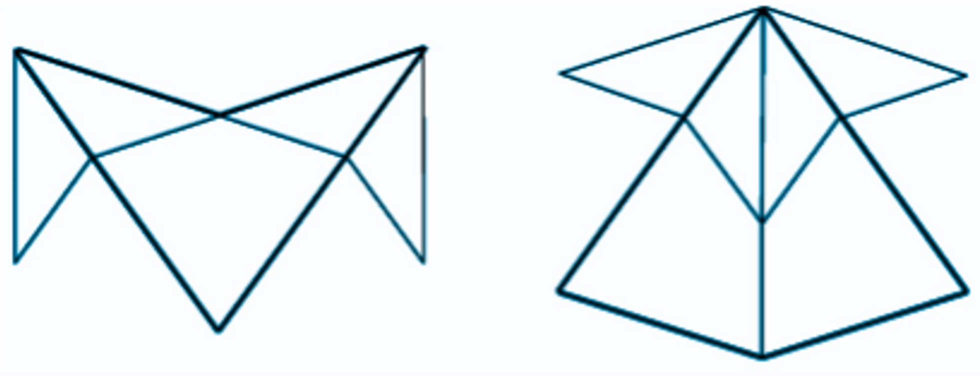
- Dealing with infinite aperiodic set of sites.
- A Voronoi diagram of an appropriate aperiodic site set is a aperiodic tiling of the plane.
  - Since this yields a way to obtain new sets of prototiles by placing points in known prototiles, it is effectively a sort of ‘machine’ for generating aperiodic sets of prototiles.
- In fact, from any aperiodic set of prototiles, it is possible to construct infinitely many of those sets using Voronoi diagrams.
- A Penrose point set:
  - Consider a finite set of points on Penrose’s kite and dart.
  - Tile the plane by those tiles to get an infinite point set in the plane.
    - The Penrose tiling which generates it, is the *underlying tiling*.

Aperiodic Tilings from VDs of Penrose Point Sets (2)

- New aperiodic tilings constructed by choosing some points on Penrose's prototiles.
  - The set of prototiles does not depend on the underlying tiling used for generating the Penrose point set but on the position of the points lying on the original kites and darts.
- Possible to give some matching rules which guarantee the aperiodicity of the prototiles (left out, read paper).
- **Theorem:**
  - *Given a Voronoi diagram of a Penrose point set, there exists a set of matching rules for its Voronoi–Penrose prototiles such that any tiling following those rules induces a Penrose tiling. Reciprocally, if the two sets of prototiles induced by two Penrose tilings are the same, then the matching rules for those sets of prototiles are the same.*
- *Corollary:*
  - *Given a set of Voronoi–Penrose prototiles, there exists a set of matching rules which enforces them to be aperiodic.*
- **Theorem (*):**
  - The Voronoi diagram of a Penrose point set is a non-periodic plane tiling.
    - Finitely many distinct Voronoi regions.

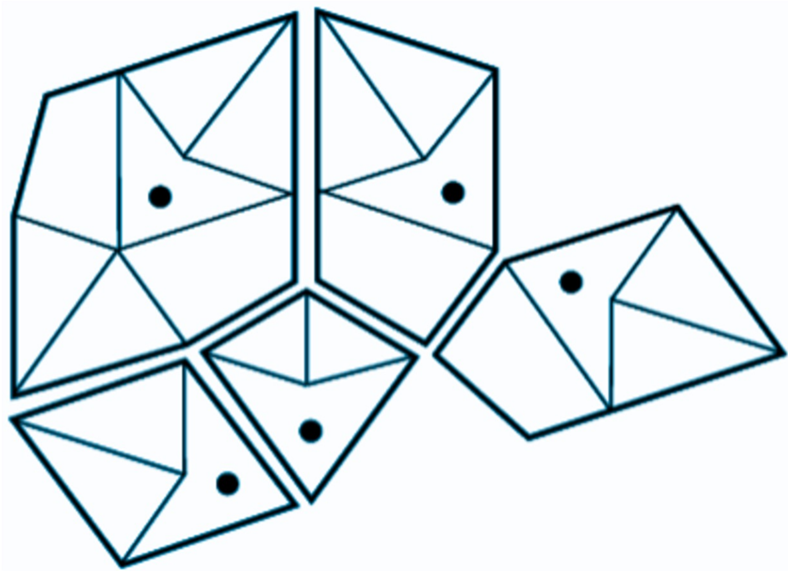
Aperiodic Tilings, Unique Composition Rule

- All aperiodic sets of prototiles discovered so far, depend on a surprising phenomenon discovered by Penrose. This phenomenon is called composition or inflation by Conway.
- It is a simple method to produce complicated tilings beginning with a patch or even a single tile.
- In the case of Penrose tilings, imagine that every dart is cut in half and then all short edges of the original pieces are glued together.
    - The result is a new tiling by larger darts and kites as shown in the figure.
- When this operation acting on a tiling or a patch of tiles is reversed, is called *decomposition* or *deflation*.
- The important point about the composition operation is its uniqueness which is closely related to the aperiodicity of prototiles.
- **Theorem:** *If a tiling has a unique composition that leaves the tiling invariant then it is non-periodic.*
    - *Hence, if there exists exactly one composition acting on any tiling of a set of prototiles then this set is aperiodic.*
- This is used in the proof of **Theorem (*)** ...
    - Simply put, the uniqueness of composition rule is inherited by VD from Penrose tiling.



Properties of VDs of Penrose Point Sets

- It is possible to obtain an aperiodic prototile set as big as we want with this method.
- Lemma: A PPS is non-periodic.
- **Theorem:**
  - *For all* n ∈ N , n ≥ 5, *there exists a set of at least,* n *Voronoi–Penrose prototiles.*
  - *<u>Proof:</u> The figure shows a set of prototiles that has five elements. Adding suitable new points yields new regions. Thus, only need to add enough points to get the result.*
- Proposition:
  - *Every patch of tiles in a Voronoi diagram of a PPS is congruent with infinitely many patches in every tiling by the same prototiles.*



Aperiodic Tilings from DTs of Penrose Point Sets

- Any VD of PPS has vertices with degree > 3 (degenerate) producing a Delaunay tessellation (not unique, not triangulation).
  - Use any of the triangulations produced by further subdivision.
- **Theorem**: The Delaunay triangulation of a Penrose point set is an aperiodic tiling.
- Can draw original darts and kites on the Delaunay triangles in order to obtain matching rules.
  - The prototiles obtained in this way are called *Delaunay–Penrose prototiles*.
- Lemma:
  - *Given a Penrose triangulation of a Penrose point set, there exists a set of matching rules for its triangles such that any tiling following those rules induces a Penrose dart and kite tiling.*
- *Corollary:*
  - *Given a Delaunay triangulation of a Penrose point set, there exists a set of matching rules for its triangles which enforces them to be aperiodic.*
- It is possible to obtain an aperiodic prototile set as big as we want with this method.
- Corollary:
  - *For all* $n \in N$, $n \geq 4$ *there exists a set of at least,* n *Delaunay–Penrose prototiles.*

Some of the Stones Left Unturned

- Paper presents a Voronoi–Penrose set of prototiles of size 5. Is 5 the minimum size of such kind of collections?

- Find an aperiodic set S of prototiles such that placing points on each element of S and following the process described in the paper, the same set is obtained.

Many Options for Construction

- VDs:
  - The Linde–Buzo–Gray algorithm - Yoseph Linde, Andrés Buzo and Robert M. Gray. (1980).
  - Lloyd's algorithm - Lloyd, Stuart P. (1982).
  - Fortune’s Sweep algorithm - Steven Fortune. (1986).
  - Jump flooding algorithm - Rong, Guodong; Tan, Tiow-Seng (2006-03-14).
  - ...
- DTs:
  - *Bowyer-Watson Algorithm*. - Bowyer, Adrian (1981). Watson, David F. (1981).
  - Divide and Conquer algorithm(s) - Guibas, Leonidas; Stolfi, Jorge (1985). Dwyer, Rex A. (1987).
  - Randomized/Systematic incremental algorithm - Guibas, Leonidas J.; Knuth, Donald E.; Sharir, Micha (1992).
  - Edge-Flip algorithm(s) - Hurtado, F.; M. Noy; J. Urrutia. (1999).
  - Radial Sweep-hull with flips - D. A. Sinclair. (2010).
  - Rip-Tent Randomized Incremental - Blelloch, Guy; Gu, Yan; Shun, Julian; and Sun, Yihan. (2018).
  - ...

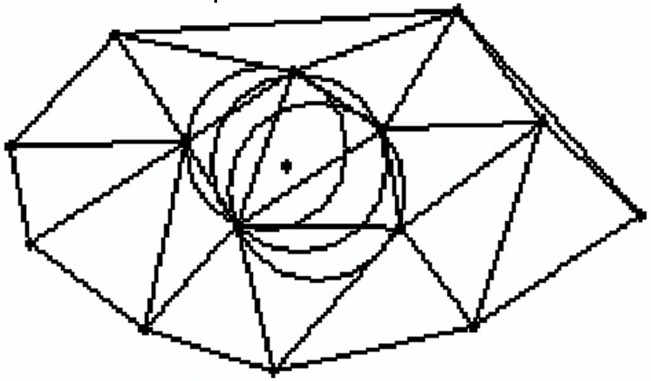
- More efficient to construct VD from DT, especially for large number of points.

Bowyer-Watson Algorithm (1981)

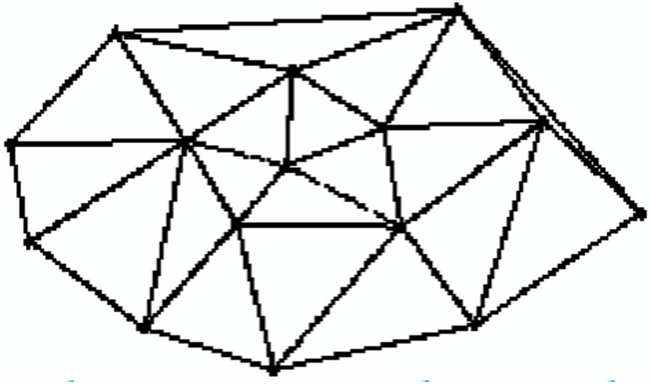
- Adds points sequentially into an existing Delaunay triangulation.
  - Usually starting from a very simple triangulation that encloses all the points to be triangulated.
- The algorithm proceeds as follows for each point that is going to be added:
  - 1. Add a point to the triangulation.
  - 2. Find all existing triangles whose circumscribing circle contains the new point. This can be done to find the triangle which contains the new point first. Then the neighbours of this triangle are searched and then their neighbours, etc., until no more neighbours have the new point in their circumscribing circle.
  - 3. Delete these triangles; this creates a convex cavity.
  - 4. Join the new point to all the vertices on the boundary of the cavity.

- O(n^2) time but theoretical optimum O(n*log(n)).

Step 2: Circumscribing circles that contain the new point.

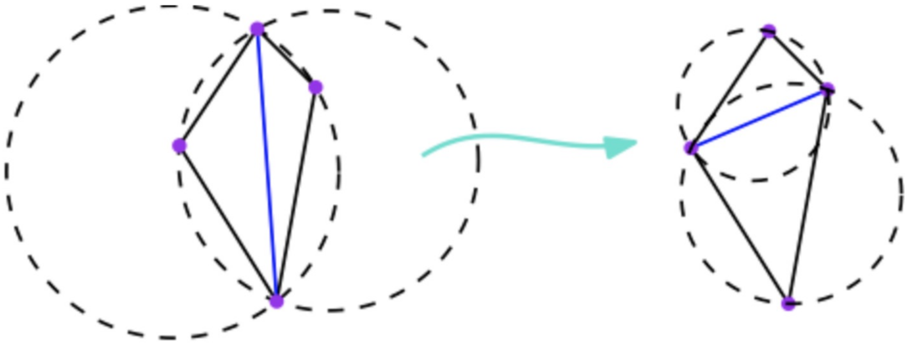


Step 4 yields triangulation below...



Edge-Flipping Preliminaries

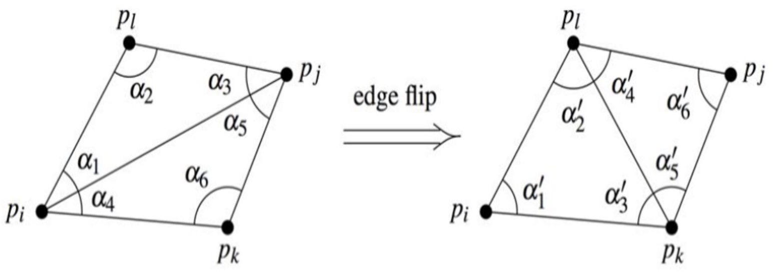
- Suppose that an edge ab is shared by the triangles abc and abd. If abc has the empty-circle property then d must be outside the circle through abc.
- If d is outside the circle through abc, then c is outside the circle that goes through abd, and the other way around.
    - In this case edge ab is called legal.
- An edge ab is then illegal if d is inside the circle through abc.
    - If d is inside the circle through abc then c is inside the circle that goes through abd, and the other way around.
- Edge Flip:
    - Consider an edge ab shared by the triangles abc and abd. Flipping edge ab to edge cd will create triangles cda and cdb.
        - Claim: If edge ab is illegal then edge cd is legal, and the other way around.



Edge-Flipping Algorithm (1999)

- The algorithm:
  - Construct arbitrary triangulation T including all desired points in P.
  - Push all edges in T onto a stack and mark them.
  - While stack is not empty do
    - pop edge ab from stack and unmark it,
    - if ab is not legal flip edge and update T,
    - for each new edge ac, ad, bc, bd: if not marked, push onto stack and mark it.

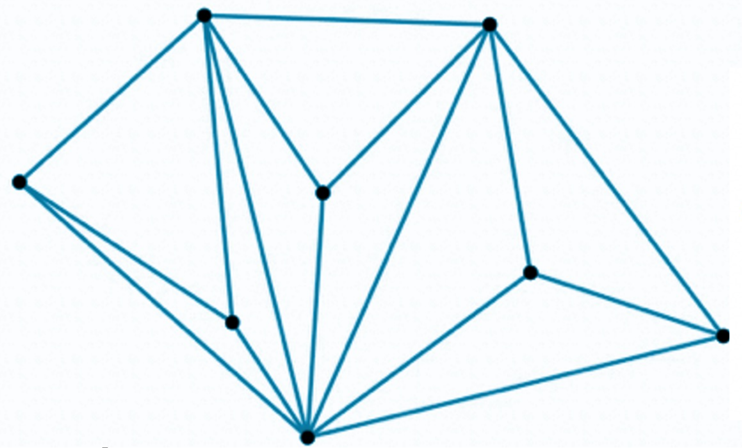
- O(n^2) time.
- Why it works: Angle vector
  - The angle vector A(T) of T is the vector of all angles of the triangles of T , sorted in increasing order.
  - Recall that all triangulations on P have the same number of triangles, therefore they will all have an angle vector of the same size.
  - Claim: When we flip an illegal edge to a legal edge, the angle vector increases.
  - Claim: The angle vector of the Delaunay triangulation is the maximum angle vector among all triangulations of P (Consequence of DT maximising minimum angle).

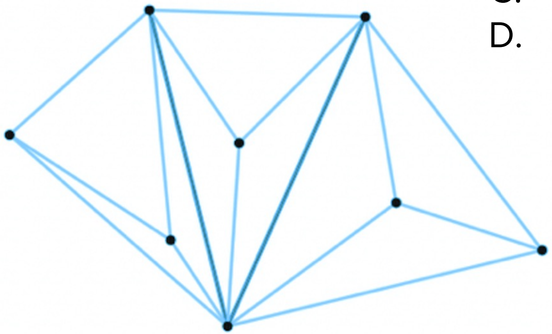


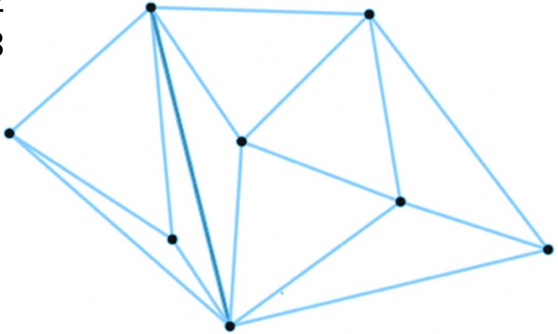
- Angle vector $(\alpha_1,\ldots,\alpha_6)$
- Illegal edge : not angle optimal
  (min angle is not maximal)
- Flipping an illegal edge increases angle vector

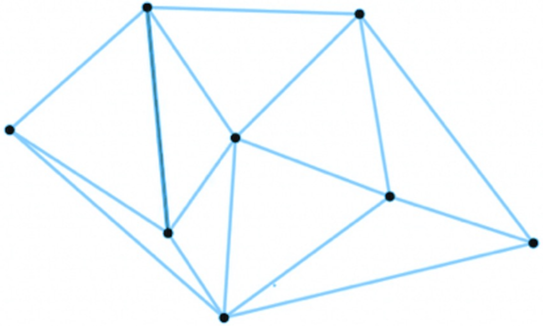
Q: How many edges need to be flipped to make all edges legal?

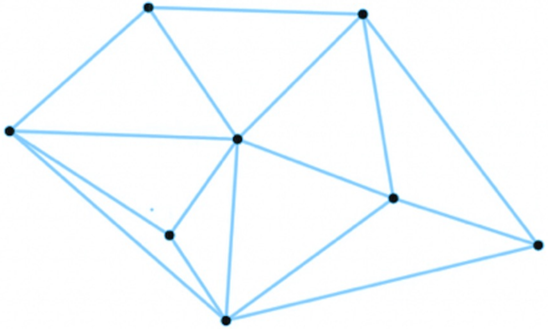
A. 0
B. 1
C. 2
D. 3





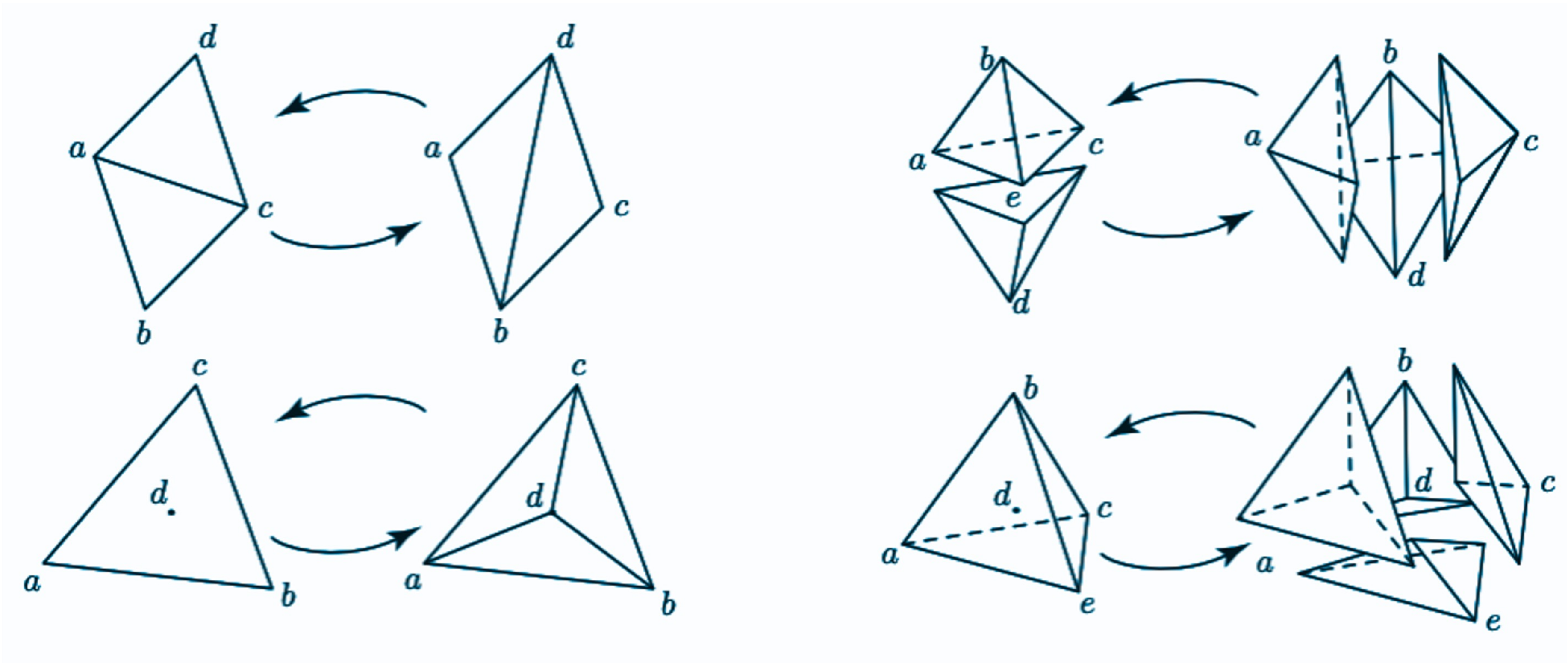






- A:
  - Initially, there are two illegal edges.
  - But flipping the second produces a new illegal edge.
  - Total number of flips needed to remove initial illegal edges and the one created is 3.

Edge Flipping in Higher Dimensions



Radial Sweep-hull with Flips (2010)

S-hull operates as follows: For a set of unique points $\mathbf{x}_i$ in R2:

1. sellect a seed point $\mathbf{x}_0$ from $\mathbf{x}_i$.
2. sort according to$|\mathbf{x}_i - \mathbf{x}_0|^2$.
3. find the point $\mathbf{x}_j$ closest to $\mathbf{x}_0$.
4. find the point $\mathbf{x}_k$ that creates the smallest circum-circle with $\mathbf{x}_0$ and $\mathbf{x}_j$ and record the center of the circum-circle $\mathbf{C}$.
5. order points $[\mathbf{x}_0, \mathbf{x}_j, \mathbf{x}_k]$ to give a right handed system: this is the initial seed convex hull.
6. resort the remaining points according to $|\mathbf{x}_i - \mathbf{C}|^2$ to give points $\mathbf{s}_i$.
7. sequentially add the points $\mathbf{s}_i$ to the porpagating 2D convex hull that is seeded with the triangle formed from $[\mathbf{x}_0, \mathbf{x}_j, \mathbf{x}_k]$. as a new point is added the facets of the 2D-hull that are *visible* to it form new triangles.
8. a non-overlapping triangulation of the set of points is created. (This is an extremely fast method for creating an non-overlapping triangualtion of a 2D point set).
9. adjacent pairs of triangles of this triangulation must be 'flipped' to create a Delaunay triangulation from the initial non-overlapping triangulation.

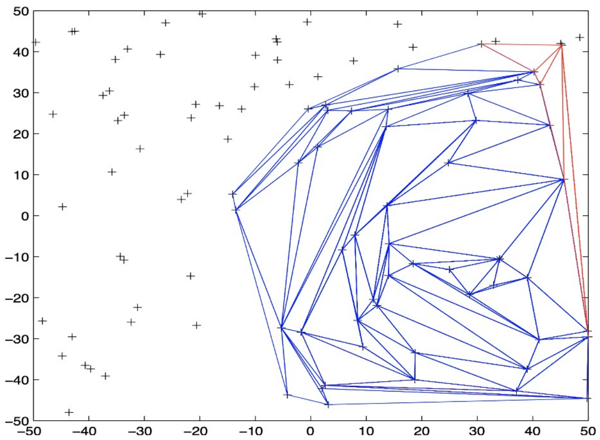


Figure 2: *Randomly generatesd set of points in R2. Sequential addition of triangles as the convex hull is swept through the ordered set of points (new triangles in red).*

O(n*log(n))

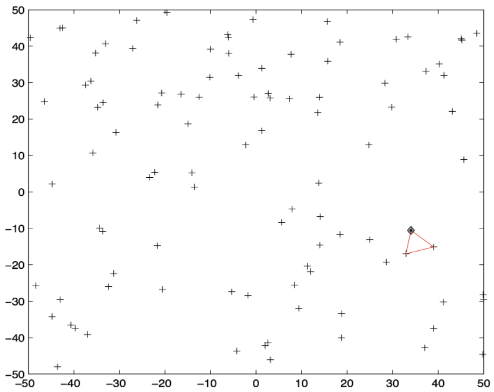


Figure 1: *Randomly generatesd set of points in R2. The seed point for the triangulation is marked along with the triangle associated with the smallest circum-circle through it and its nearest neighbour.*

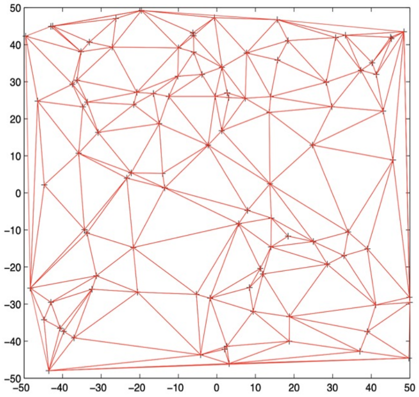


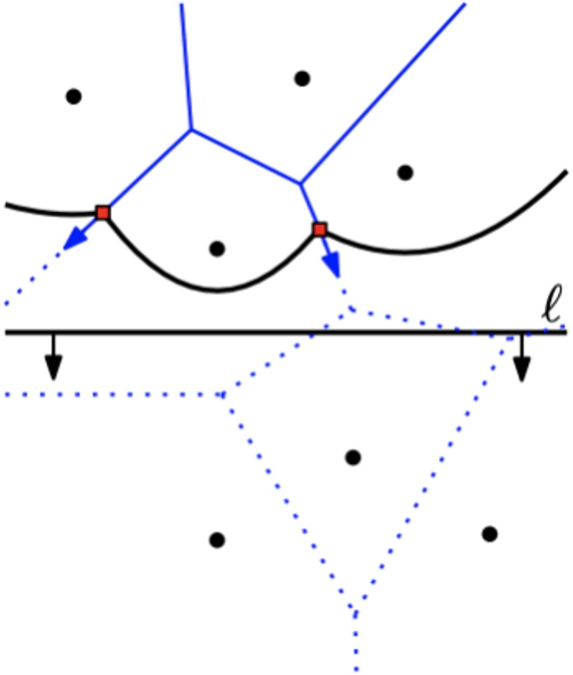
Figure 3: *Delaunay triangulation of the set of points in R2 generated by s-hull.*

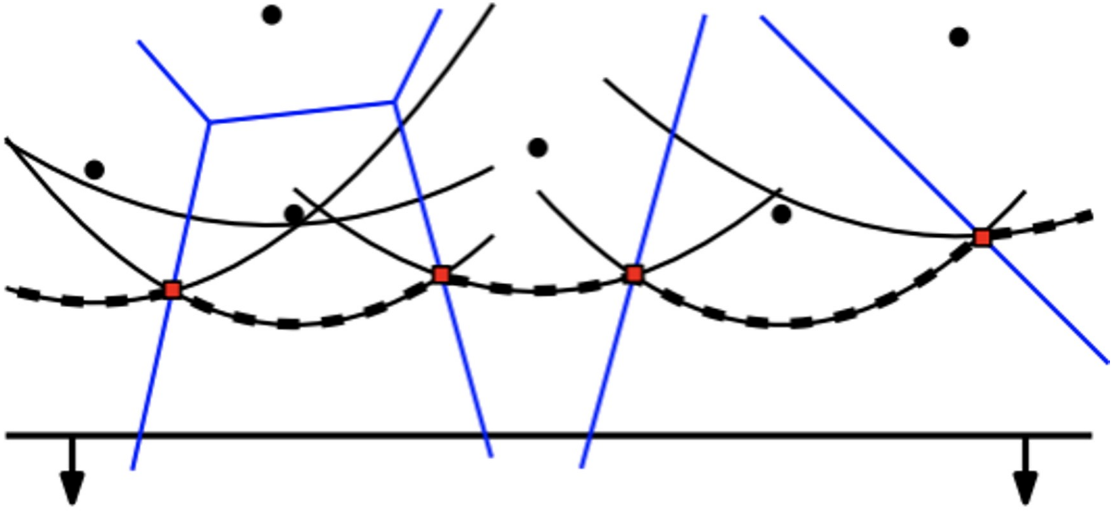
Fortune’s Sweep Algorithm (1986)

- O(n*log(n)) time.
- Not for computing DT, yields VD directly.
- The *sweep line* (red) is a straight horizontal line and moves through the plane, from left to right.
- The *beach line* (black) is a series of parabolas. A parabola has a site's center as its focus and points inside the parabola are sure to belong to the site, while points outside the parabola are still uncertain.
- Points to the left of the beach line have been incorporated to the VD.
- Points between the sweep and beach lines are still uncertain.

Q: The beach line has breakpoints, what do they represent?

- A:
  - The breakpoints move tracing out the VDs edges.





- Emergence of a new break point(s) (from formation of a new arc or a fusion of two existing breakpoints) identifies a new edge.
- Voronoi vertices are identified when two break points meet (fuse).
  - Decimation of an old arc identifies new vertex.

Special Section on SMI 2019

Delaunay Lofts: A biologically inspired approach for modeling space filling modular structures

Sai Ganesh Subramanian[a], Mathew Eng[b], Vinayak R. Krishnamurthy[a,*], Ergun Akleman[b]

[a] J. Mike Walker '66 Department of Mechanical Engineering, Texas A& M University, College Station, TX 77843, United States
[b] Department of Visualization & Computer Science and Engineering, Texas A& M University, College Station, TX 77843, United States

A R T I C L E I N F O

Article history:
Received 11 March 2019
Revised 21 May 2019
Accepted 22 May 2019
Available online 29 May 2019

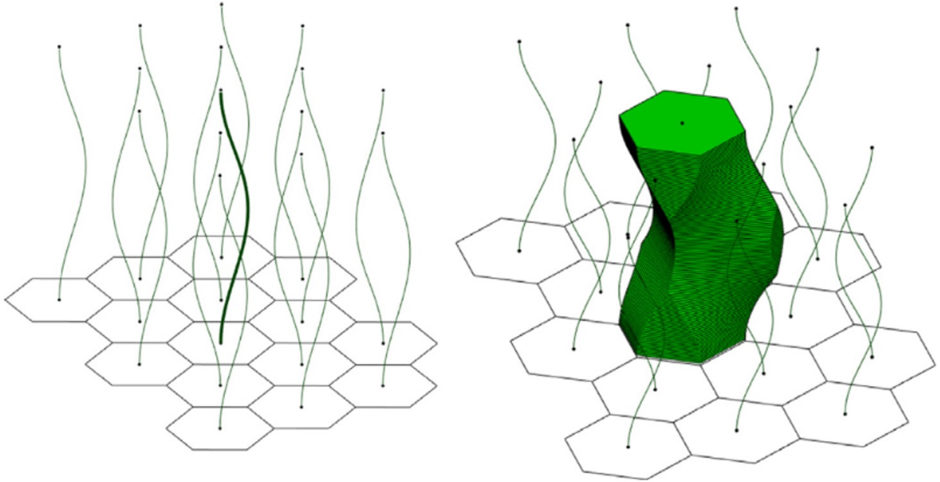
Keywords:
Tessellation of space
Space filling structures
3D tile design

A B S T R A C T

In this paper, we present a simple and intuitive approach for designing space filling tiles in 3D space. Our approach is inspired by “scutoids” — shapes that were recently reported to occur in epithelial cells due to topological changes between the extremal (apical and basal) surfaces of epithelia. Drawing from this discovery, we develop the theoretical and computational foundations leading to a generalized procedure for generating *Delaunay Lofts* — a new class of scutoid-like shapes. Given two extremal surfaces, both with Delaunay diagrams, Delaunay Lofts are shapes that result from Voronoi tessellation of all intermediate surfaces along the curves joining the vertices of Delaunay diagrams that defines the extremal tessellations. This, combined with the use of wallpaper symmetries allows for intuitive design of complex space filling regular and semi-regular tilings in 3D space.

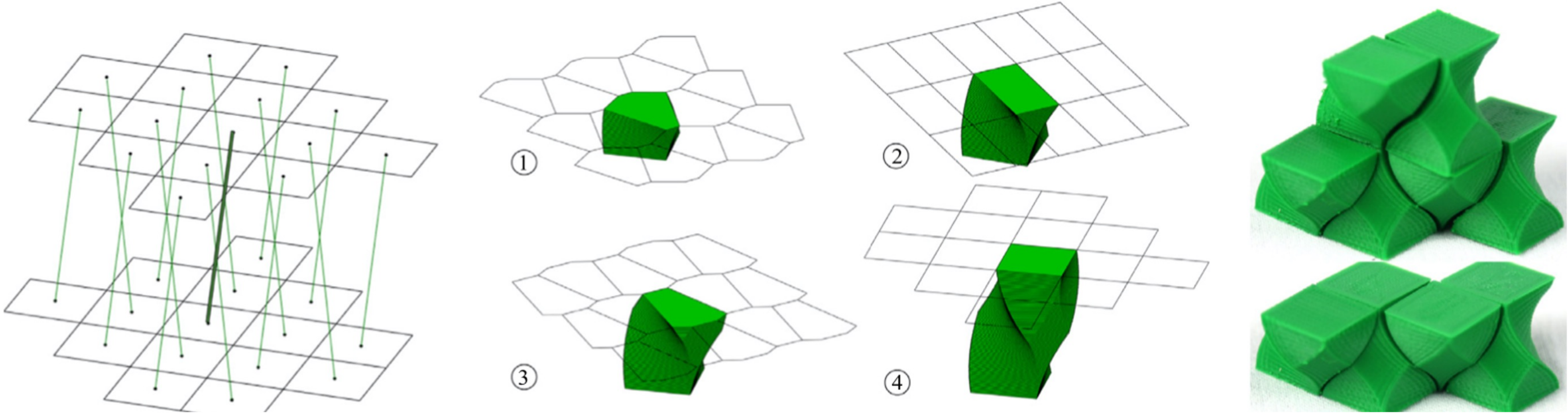
Motivation and Intro to Delaunay Lofts

- Inspired by a discovery by Gómez-Gálvez, Vicente-Munuera, and many other collaborators:
  - Observed that a simple polyhedral form, which they call *'scutoids'*, commonly exists in epithelia cells in the formation of skin cells.
  - Demonstrated that having this polyhedral form in addition to prisms provides a natural solution to three-dimensional packing of epithelial cells.
  - Showed that in skin cells, the top (apical) and bottom (basal) surfaces of the cellular structure are Voronoi patterns.
- The paper introduces an approach to construct non-polyhedral space filling shapes that are 'scutoid-like' using VDs and DTs.
  - Stack multiple planar VDs/DTs vertically with space in between, and interpolate the curves connecting the sites of neighbouring layers.
- NB: No general position assumption:
  - Cyclic polygons are useful in this context.
- Use of wallpaper symmetries to ensure regular/semi-regular layers.



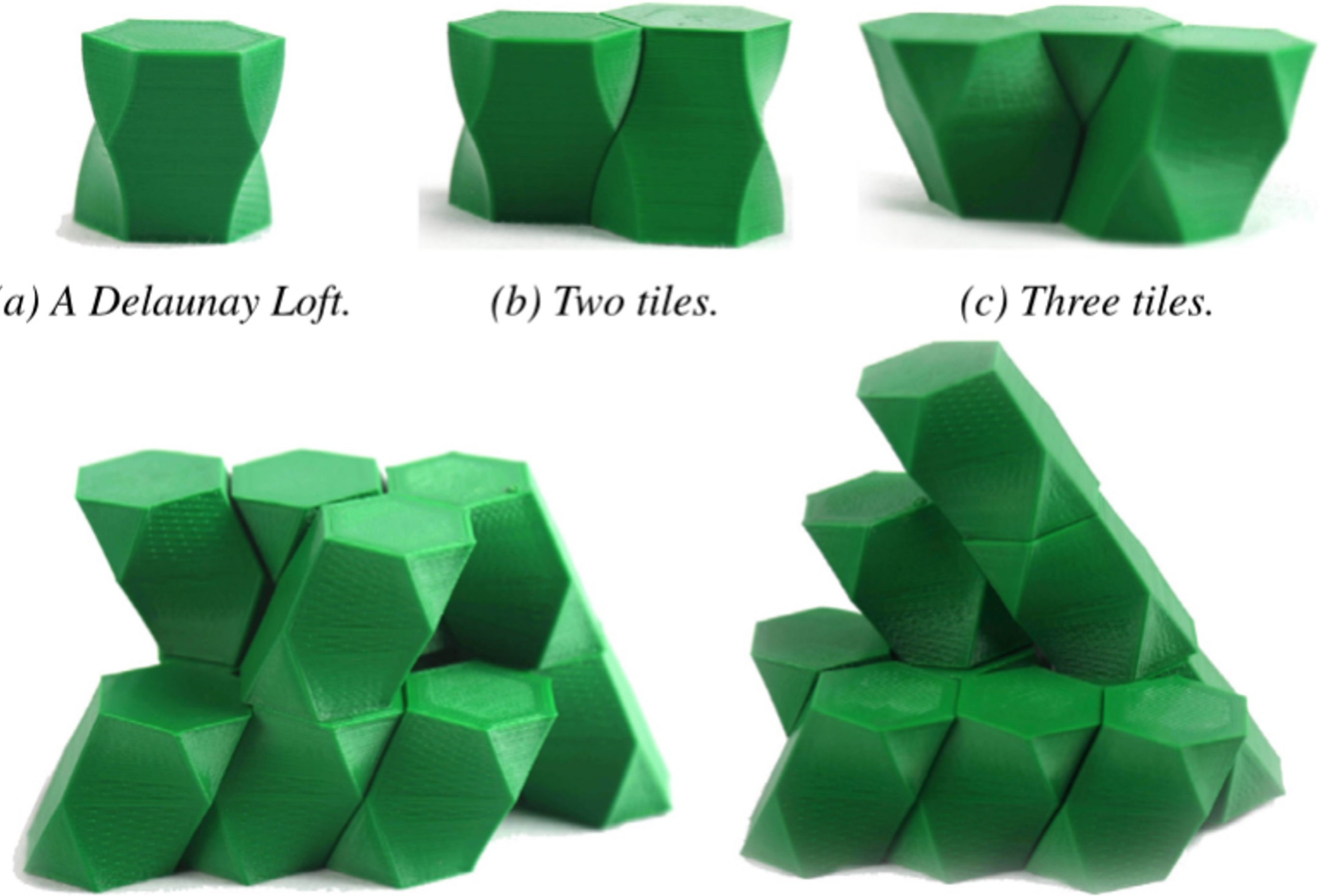
Methodology (1)

- A distance function that reduces Voronoi decomposition for a given set of curves in space into a set of Voronoi decompositions with respect to points on slices of those curves.
- This approach makes the construction algorithm independent of the complexity of the control curve while providing well-defined boundaries.
- The Voronoi decomposition obtained with these control curves gives us a natural interpolation of 2D Voronoi diagrams producing Delaunay Lofts.
- Figure:
  - (a) Control curves that interpolates Voronoi sites are shown in green. Voronoi sites are black dots.
  - (b) Voronoi decomposition in set of layers.
  - Construct Delaunay Lofts by interpolating the Voronoi polygons. The resulting Delaunay Lofts in this case are shown (c).
  - NB: (b) Motion of Voronoi sites produces hexagonal grids, which causes a change of topology from quadrilateral to hexagon and back to quadrilateral (2-4).



See it Fill Space

- This DL is created as an interpolation of three layers of tilings, namely: (1) a regular hexagonal; (2) a square and; (3) another regular hexagonal tilings, which is translation of the first hexagonal tiling.
- The interpolating control lines are straight lines.

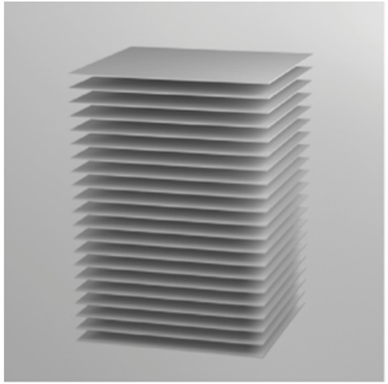


(a) A Delaunay Loft. (b) Two tiles. (c) Three tiles. (d) Ten tiles. (e) Ten tiles.

‘General Construction Algorithm’

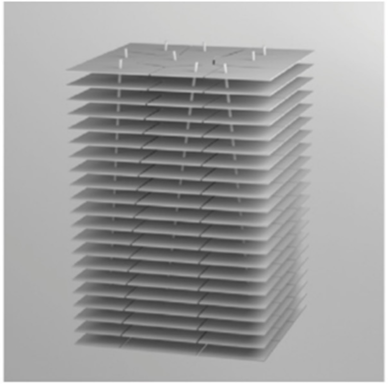
- 1. Sample *N* number of constant *z* planes/layers from a rectangular prism. (a)
- 2. Design *M* number of curves inside of the rectangular domain. (b)
- 3. Find the intersection of curves with intermediate layers. For each layer, compute its Voronoi partitioning by using intersection points with that particular layer as Voronoi sites. (c)
- 4. Offset each Voronoi polygon the same amount using Minkowski difference. Note that this process can also change topology of the polygons. (d, e)
- *Remark*: Step 4. ensures the production of separable Delaunay Lofts while printing a tiling as a whole. The offset is half the width of the 3D printing nozzle.
- 5. Treating each vertex as a single manifold, insert edges between consecutive vertices thereby turning each original face into a 2-sided face, which is 2-manifold. (f)
- 6. Connect 2-sided faces with insert-edge operations to form a single genus-0 object. (g)
- This series of edge insertions results in a Delaunay Loft. (h)

See it Happen

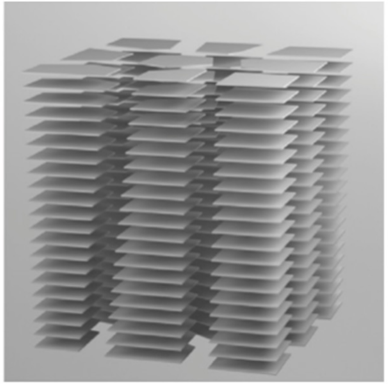


(a) Initial Layers.

(b) The control curves.



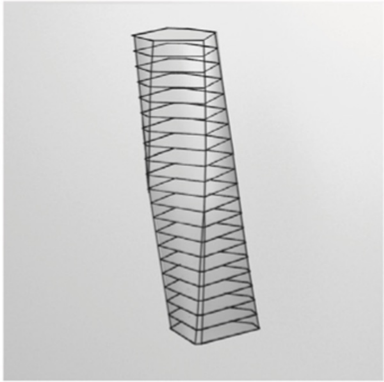
(c) Voronoi decomposition of layers.



(d) Offset polygons.

(e) Offset polygons for one Voronoi Loft.

(f) Construction of 2D sided (i.e. 2-manifold) faces.

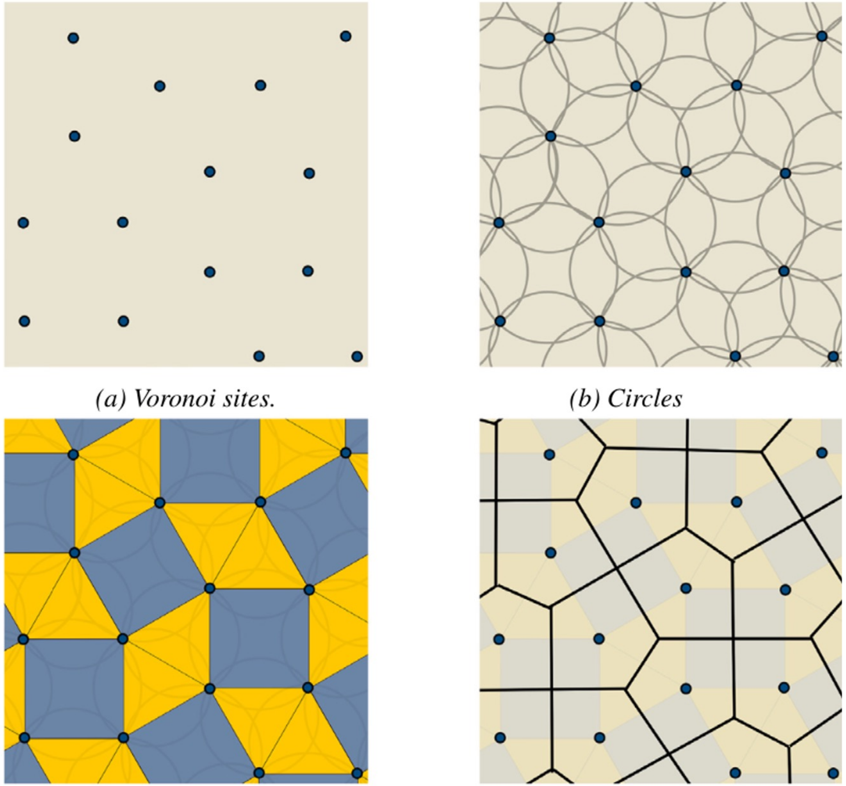


(g) Connecting 2-sided faces by inserting edges.

(h) Final Voronoi Loft.

Q: Why are cyclic polygons useful?

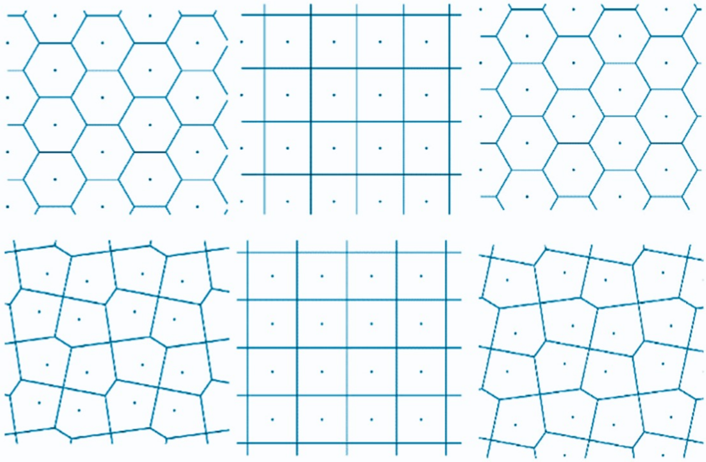
- Figure shows an example to demonstrate how to design single polygon tilings as dual meshes of regular or semi-regular tilings.
- Note that each polygon in semi-regular Delaunay diagram is regular, and therefore, cyclic.
  - (c) Delaunay diagram that is a semi-regular mesh with the same vertex figure, which is 3.4.3.4.3.
  - The corresponding Voronoi Diagram in (d) is a tiling that consists of the same polygons, which are pentagons.
    - This property holds for all semi-regular tilings.
- To precisely control polygonal topology changes, use the fact that when n points are inscribed on a circle, they form a convex cyclical polygon and their *n*-perpendicular bisectors to the sides are always concurrent and the common point is always the center of the circle ((b)).
  - This property helps to design desired control curves by directly controlling the number of sides and vertex valances of a Voronoi tessellation in every layer.
  - Essentially can create the desired Delaunay diagrams in some layers (not neighbouring) and then interpolate their vertex positions to other layers (in between).
  - Therefore, controlling how many triangles share the same circle in key layers, makes it possible to change mesh topology in any desired layer.

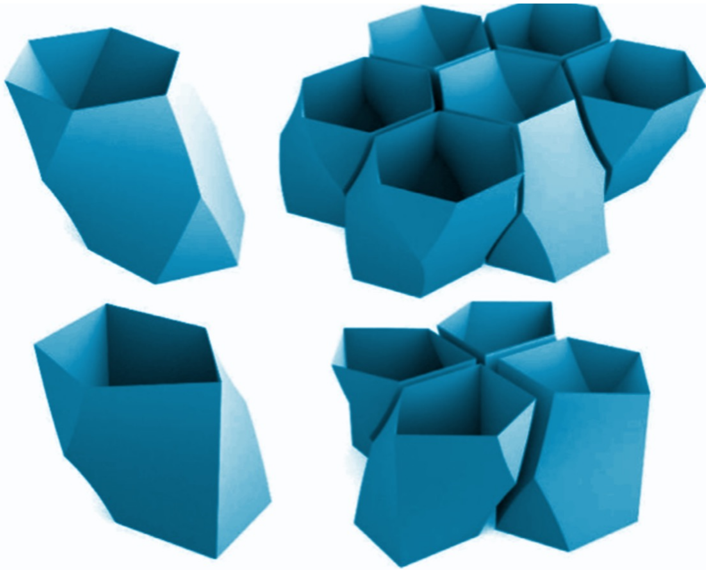

(a) Voronoi sites. (b) Circles (c) Delaunay diagram. (d) The corresponding Voronoi Diagram.

Methodology (2)

- Since any regular polygon is also a cyclic polygon, regular polygon tilings are good candidates to design Delaunay diagrams.
    - Fig: The three patterns (left to right) show the Voronoi diagrams from the bottom, middle and the top layer of interpolation.
        - First row shows the 464 Delaunay Lofts.
        - Second row shows the Delaunay Lofts obtained by interpolating 3.4.3.4.3 patterns.

- Construction algorithm should allow for exploring a vast array of shapes that are accessible with this method.
    - Only constraint is to have the total number of points in the neighbouring layers to be equal.
    - To ensure repeatability of the Delaunay Lofts, must also have some geometric regularity in the tilings of the layers.



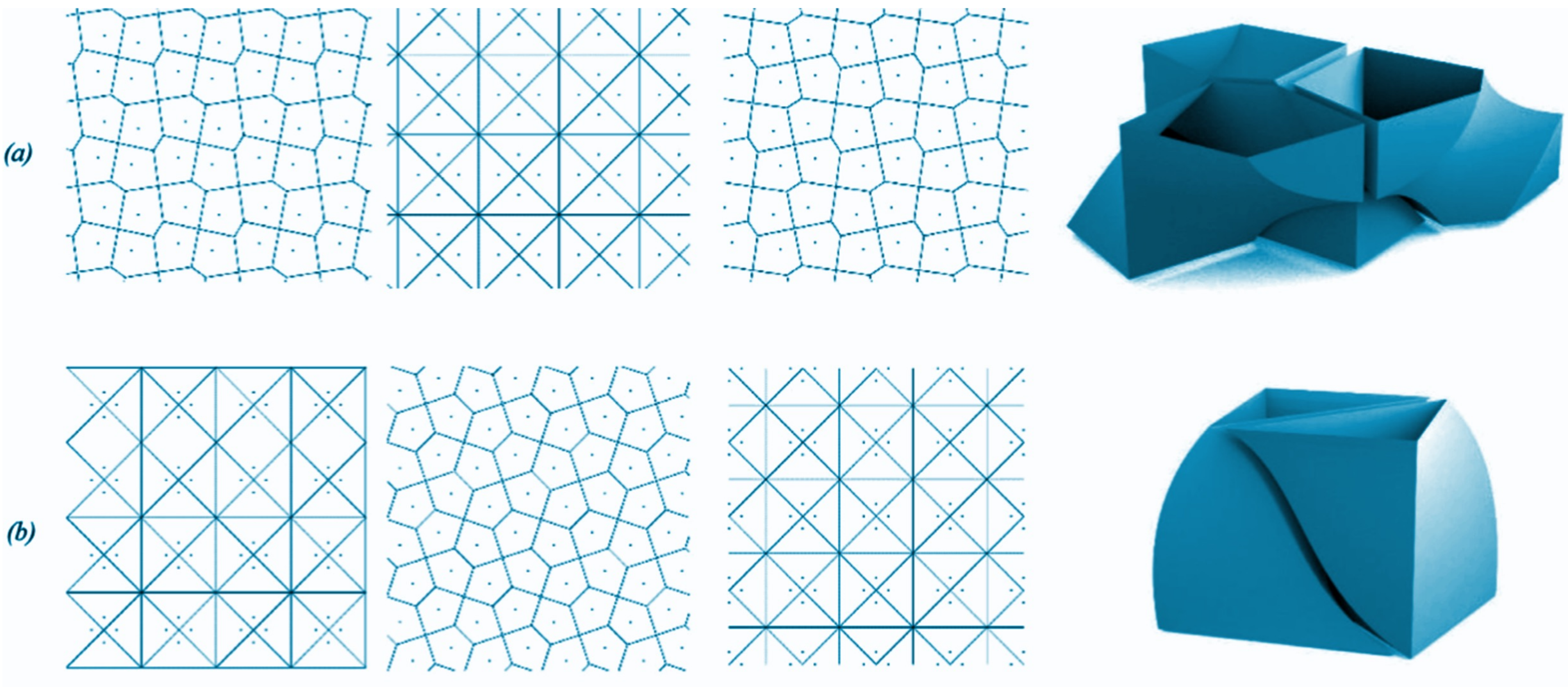


Methodology (3)

- Broadly speaking, the authors of this paper had two main aims for Delaunay lofts:
  - First, for it to be possible to compose space filling patterns with the shapes.
  - Second, for the pattern to be composed of ideally a single (or at least a finite set of) repeatable shape(s).
- The first condition is naturally satisfied by the strategy to use Voronoi partitioning (since any such partitioning is guaranteed to fill space of any given dimension).
  - The space filling condition is satisfied regardless of how the Voronoi sites are distributed on each of the extremal surfaces (as long as we can establish a on-to-one correspondences between the sites on each surface).
- The authors addressed the second condition of repeatability through their method of construction and design based on wallpaper symmetries.

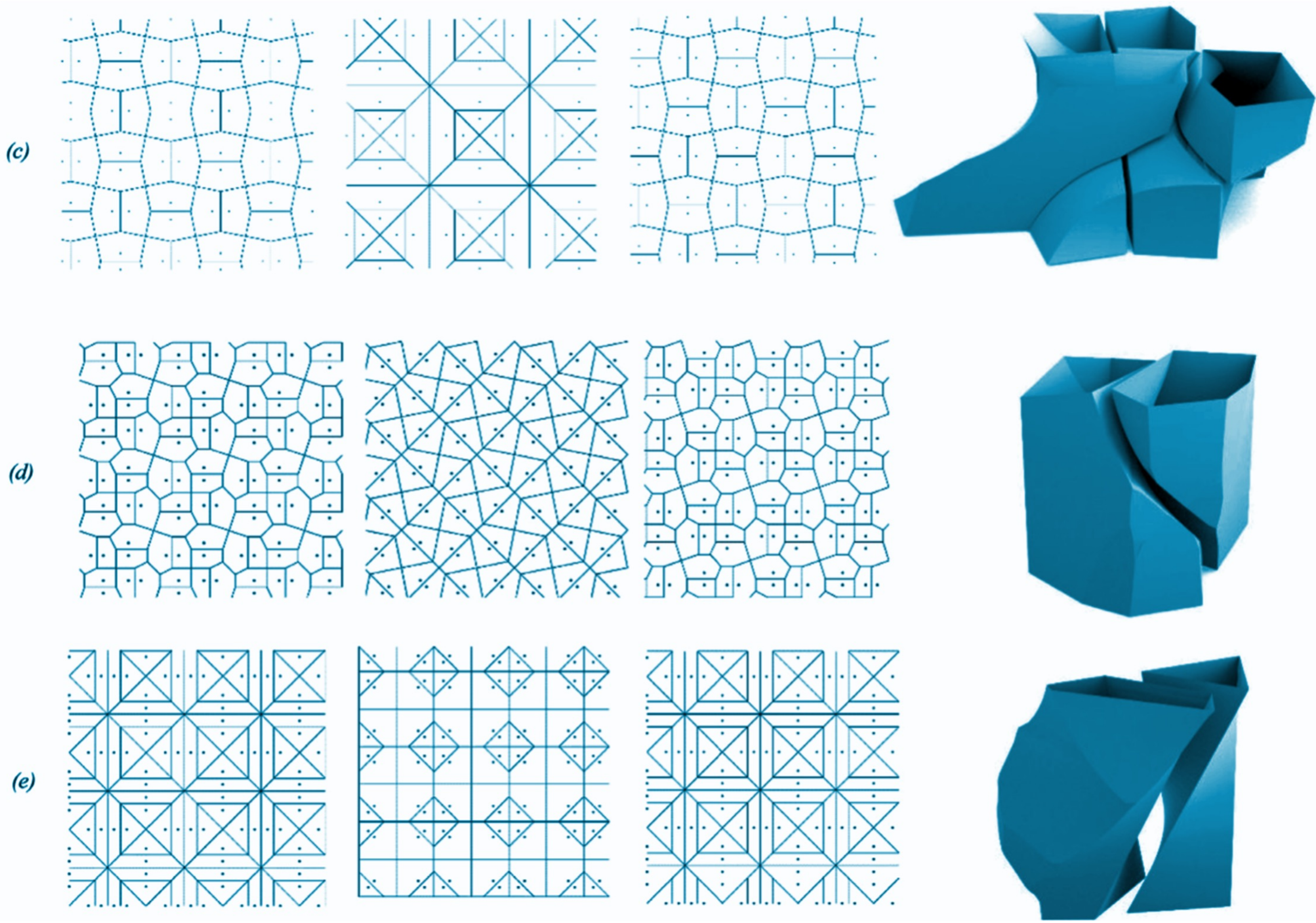
Examples of Lofts and Intermediate Tillings

- Figure shows more Delaunay Lofts from applying wallpaper symmetry patterns to the control curves in 3D.
- Each row shows the transition of Voronoi from the bottom layer, middle layer and top layer followed by the corresponding tiling.
- (a) shows a transition from Pentagon in the Top and bottom layer with a Triangle in the middle.
- (b) shows Triangle on top and bottom with a Pentagonal Middle layer.



More Examples

- The Delaunay Lofts (c), (d) and (e) go through a series of transitions within Triangle, Quadrilateral and Pentagonal layers.
    - This is the reason for the intricate shapes of these Lofts.



So Many Possible Shapes...

- The design space of shapes that can be composed using this approach is **extremely** rich. This is due to three facts:
  - First, the **construction algorithm does not assume any specific shape of the control curves** — as long as they intersect each slicing plane at a unique point thus maintaining the number of sites per slice. This alone provides many possibilities in terms of obtaining seemingly complex geometries.
  - Second, the 17 wallpaper symmetries result in several possibilities in terms of the tiling configurations that may be possible with our approach.
  - Finally, the distance functions utilized in examples are only *L*2-norms. Generalizing to *Lp*-norms will lead to even more unusual shapes that we have currently demonstrated.

Why Else are Delaunay Lofts Cool ?

- Whether DLs are powerful shapes in terms of withstanding stress, torsion and fatigue, has not been tested in practice yet but, in theory, they should.
  - If true in practice, DLs could be applied to come up with completely new designs and structures that could have greater strength.
- Another advantage of the approach to construct DLs is that it can easily be used in combinatorial optimization.
- Therefore, this approach could take the industry to the next level of material optimization and unveil endless possibilities of geometric designs with Delaunay Lofts.
- Scutoids have many applications in biology and biochemistry.
  - Scutoids can be constructed with the layering approach used for DLs.
  - Some DLs are near-perfect replicas of scutoids.

Nonparametric Functional Approximation with Delaunay Triangulation

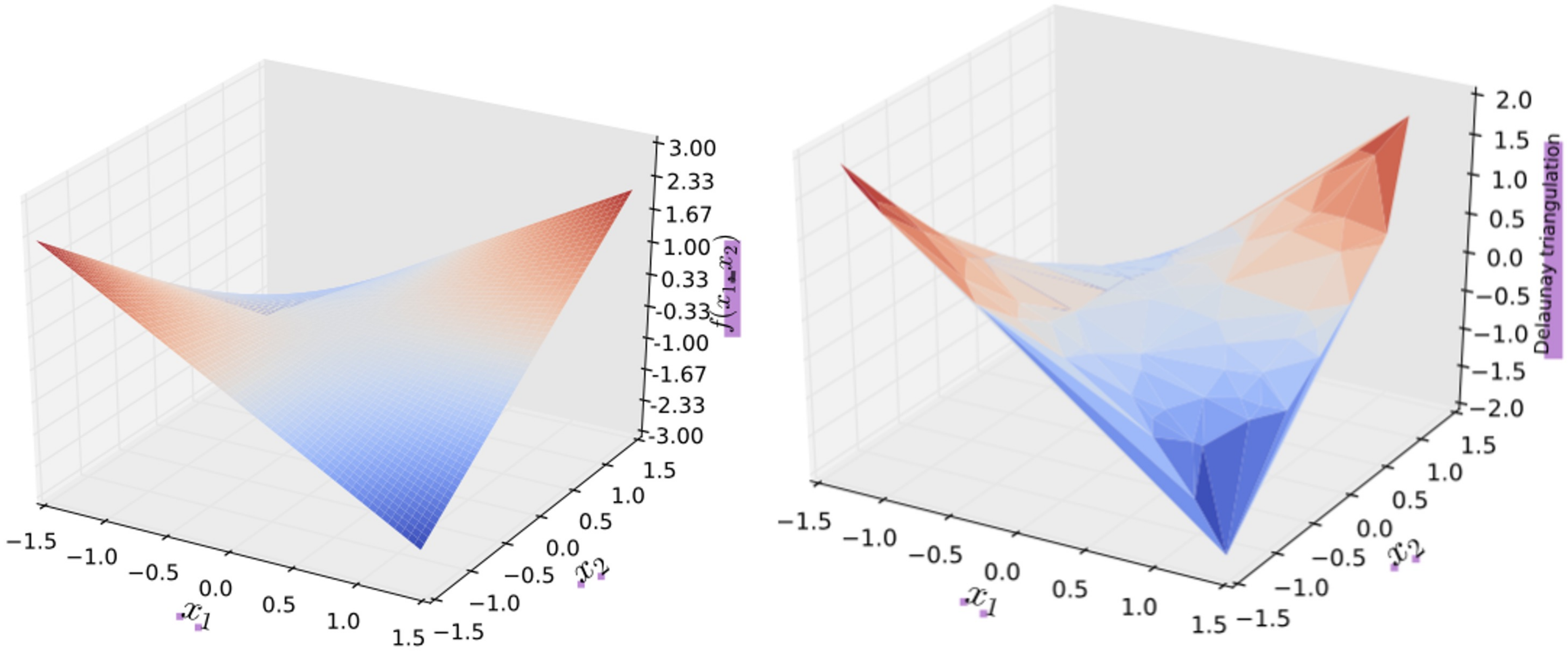
Yehong Liu[1] and Guosheng Yin[2]

Department of Statistics and Actuarial Science
The University of Hong Kong
Pokfulam Road, Hong Kong
Email: [1]liuyh@hku.hk and [2]gyin@hku.hk

June 4, 2019

Delaunay Triangulation Learners (ML)

- Want to approximate a low-dimensional smooth functional with small errors or without errors.
  - Paper focuses on low-dimensional settings ($p < n$) for simplicity.
  - Could be possible to generalize DTL to high-dimensional settings.
- DTL idea:
  - (1) DTL fits a piecewise linear model, which can be interpreted easily. .
  - (2) DTL separates the feature space in a geometrically optimal way.
  - (3) Compared with general triangulation methods (e.g., random triangulation), where the geometric structure of the triangles can be difficult to analyze, the Delaunay triangulation has many nice geometrical and statistical properties.
  - (4) Based on the construction of the DTL, we can define the regularization function geometrically to penalize the roughness of the fitted function.
  - (5) DTL can accommodate multidimensional subspace interactions in a flexible way, which is useful when the output of the model is dependent on the impact of a group of covariates.
  - (6) DTL is formulated as a differentiable optimization problem, which can be solved via a series of well-known gradient-based optimizers.
- The DTL can be used as an alternative approach to substituting a nonlinear activation function in the neural network.
- The DTL can also be generalized into manifold learning approaches, with multi-dimensional input and output.



- (a) The true function $f(x_1,x_2) = x_1x_2$, (b) the Delaunay triangulation of the surface based on 100 random samples.

Formalization of DTLs

- Given the data {X,Y} from a probabilistic model (joint distribution) $P_{X,Y}$, where Y = $\{y_i\}_{i=1}^n$ and X = $\{x_i\}_{i=1}^n$, the DTL is formalized as follows:
  - (1) Delaunay Partition
    - Partition the feature space with a triangle mesh by conducting Delaunay triangulation on X = $\{x_i\}_{i=1}^n$ and obtain a set of simplices, denoted as D(X).
  - (2) Parabolic Lifting
    - Let $\Psi = (\psi_1, \ldots, \psi_n)^T$ denote a vector of location parameters. Construct the DTL as a linear interpolation function based on {X,Ψ}, which is denoted as $F_D(x;\Psi)$.
- It is a nonparametric functional learner as the dimension of the parameter Ψ grows at the same pace as the sample size n.
- We define a geometric loss function to analyze the geometric optimality of the Delaunay triangulation with random points. (optimization problem left out, read paper)

Definition 4.1. *Let $\mathcal{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$ $(n > p+1)$ be n points in R^p in general position with a convex hull $\mathcal{H}(\mathcal{X})$, and $\mathcal{T}$ be any triangulation of $\mathcal{X}$. For any point $\mathbf{X} \in R^p$, if $\mathbf{X} \in \mathcal{H}(\mathcal{X})$, the geometric loss function of the points is defined as*

$$H(\mathbf{X}, \mathcal{X}, \mathcal{T}) = \sum_{i=1}^{p+1} \lambda_i \|\mathbf{X}_{(i)} - \mathbf{X}\|^2, \ s.t. \sum_{i=1}^{p+1} \lambda_i = 1, \sum_{i=1}^{p+1} \lambda_i \mathbf{X}_{(i)} = \mathbf{X},$$

where $\mathbf{X}_{(i)}, i = 1, \ldots, p+1$ are the vertices of the triangle in $\mathcal{T}$ that covers $\mathbf{X}$. Otherwise, if $\mathbf{X} \notin \mathcal{H}(\mathcal{X})$, we define $H(\mathbf{X}, \mathcal{X}, \mathcal{T}) = 0$.

Some Properties of DTLs

- Geometrical Asymptotic Properties:
  - DTL makes prediction for a point by fitting a linear interpolation function locally on the Delaunay triangle that covers the point.
- Paper proves results about the convergence rate of the size of the covering triangle in terms of its average edge length.
  - Thm1 shows
  - C1 shows that the vertices of the covering simplex of point X all converge to X in probability.
  - Thm 2 shows ???

Theorem 1. *Let $\mathcal{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$ be i.i.d samples from a continuous density f, which is bounded away from zero and infinity on $[0,1]^p$. Then,*

$$\mathbb{E}[H(\mathbf{X}, \mathcal{X}, \mathcal{D})] = \inf_{\mathcal{T}} \mathbb{E}[H(\mathbf{X}, \mathcal{X}, \mathcal{T})],$$

where $\mathcal{D}$ is the Delaunay triangulation of $\mathcal{X}$.

Theorem 2. *Let $\mathbf{X}, \mathbf{X}_1, \ldots, \mathbf{X}_n$ be i.i.d samples from a continuous density f, which is bounded away from zero and infinity on $[0,1]^p$. If the point $\mathbf{X}$ is inside the convex hull of $\mathbf{X}_1, \ldots, \mathbf{X}_n$, define $T(\mathbf{X})$ as the average length of the edges of the Delaunay triangle that covers $\mathbf{X}$; otherwise, $T(\mathbf{X}) = \|\mathbf{X} - \mathbf{X}_{(1)}\|$, where $\mathbf{X}_{(1)}$ is the nearest neighbor point of $\mathbf{X}$ among $\mathbf{X}_1, \ldots, \mathbf{X}_n$. Then,*

$$n^{1/p} \mathbb{E} T(\mathbf{X}) \to \beta \int_{[0,1]^p} f(\mathbf{x})^{(p-1)/p} d\mathbf{x},$$

where β is a positive constant depending only on the dimension p and density f.

Corollary 1. *Let $\mathbf{X}, \mathbf{X}_1, \ldots, \mathbf{X}_n$ be i.i.d samples from a continuous density f, which is bounded away from zero and infinity on $[0,1]^p$. If the point $\mathbf{X}$ is inside the convex hull of $\mathbf{X}_1, \ldots, \mathbf{X}_n$, define $\mathbf{X}_{(1)}$ as any one of the vertices of the covering Delaunay triangle; otherwise, define $\mathbf{X}_{(1)}$ as the nearest neighbor point of $\mathbf{X}$. Then, $\mathbf{X}_{(1)} \to \mathbf{X}$ in probability.*

Some More Properties

- General Case Properties:
  - For regression problems, the DTL is shown to be consistent.
  - For classification problems, the DTL has an error rate smaller than or equal to 2RB (1 – RB),
    - where RB is the Bayes error rate (the minimum error rate that can be achieved by any function approximator).

Regression

Theorem 3. *Assume that* $\mathbb{E}y^2 < \infty$, *and* $\psi(\mathbf{X}) = \mathbb{E}(y|\mathbf{X})$ *is continuous. Then, the DTL regression function estimate* $\hat{F}_D$ *satisfies*

$$\mathbb{E}|\hat{F}_D(\mathbf{X}) - \psi(\mathbf{X})|^2 \to L^*,$$

where $L^* = \inf_g \mathbb{E}|y - g(\mathbf{X})|^2$ *is the minimal value of the* L_2 *risk over all continuous functions* $g : \mathbb{R}^p \to \mathbb{R}$.

Classification

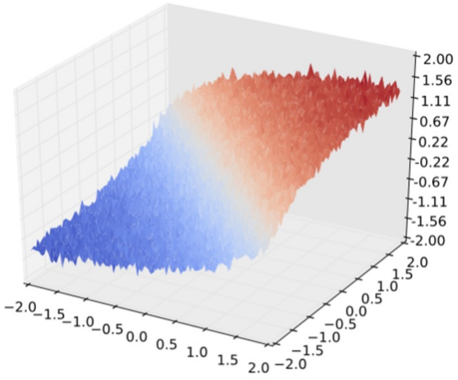
Theorem 4. *For a two-class classification model, if the conditional probability* $\psi(\mathbf{x}) = \mathbb{P}(y = 1|\mathbf{x})$ *is a continuous function of* $\mathbf{x}$. *The mis-classification risk* R *of a DTL classifier is bounded as*

$$R_B \leq R \leq 2R_B(1 - R_B),$$

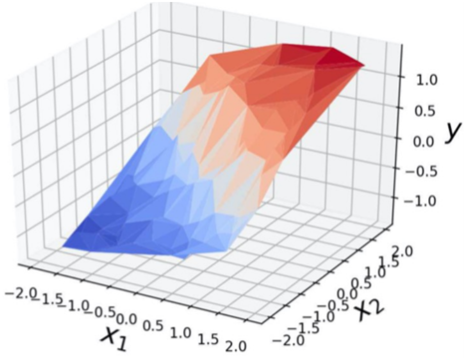
where R_B *is the Bayes error of the model.*

DTLs Compared to Other Statistical Learners (1)

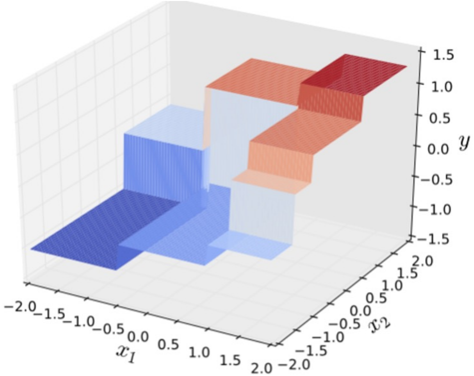
- Comparison between the DTL, decision tree, and multivariate adaptive regression splines (MARS),
  - fitted on the same data generated from a saddle surface model y = arctan(x1 + x2) + ε,
  - where the noise ε follows the normal distribution N(0,0.01):
    - DTL appears to be smoother and more flexible in shape.



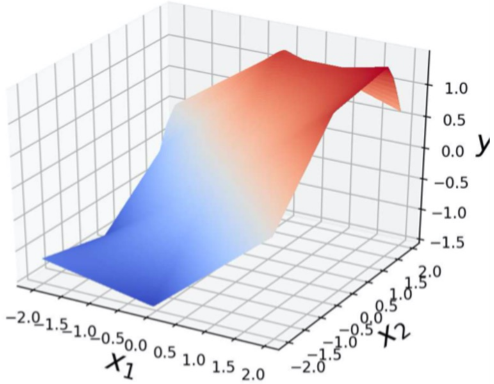
(a) Original data



(b) Delaunay triangulation learner



(c) Tree-based learner

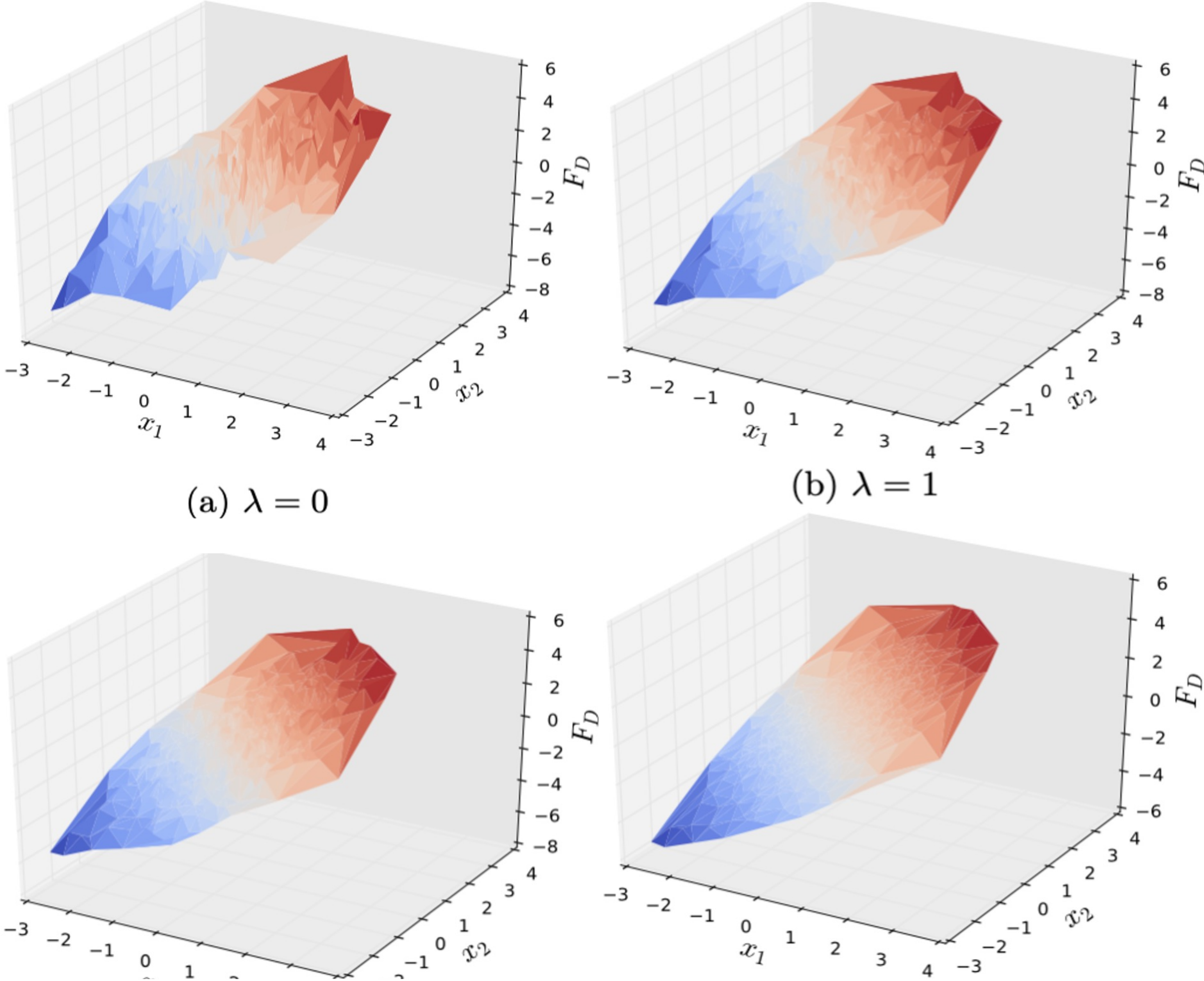


(d) Multivariate adaptive regression splines

- Key:
  - (a) Samples generated from an arc-tangent model,
  - (b) the Delaunay triangulation learner,
  - (c) the tree-based learner,
  - (d) the Multivariate adaptive regression spline.

Advantage of the DTL

- The DTL balances the roughness and smoothness depending on the local availability of data and that can lead to better predictive accuracy.
- Figure shows comparison of the smoothness of the regularized DTL using different values of λ,
  - Data from the linear model y = X1 + X2 + ε.
    - Fitting a two-dimensional linear model under a squared loss function*.
    - Lambda is tuning parameter in front of regularization function (measuring the roughness of DT).
    - Higher lambda means more weight assigned to roughness when penalising. Hence, looks smoother.



(a) $\lambda = 0$

(b) $\lambda = 1$

(c) $\lambda = 2$

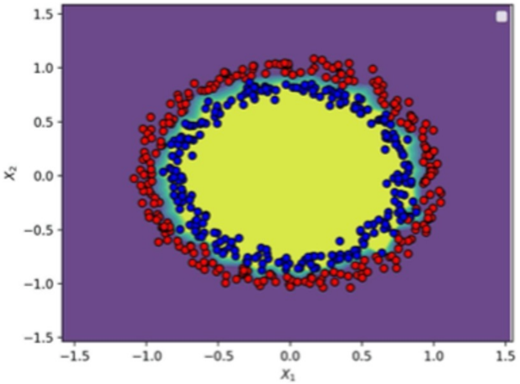
(d) $\lambda = 10$

DTL Compared to Other Statistical Learners (2)

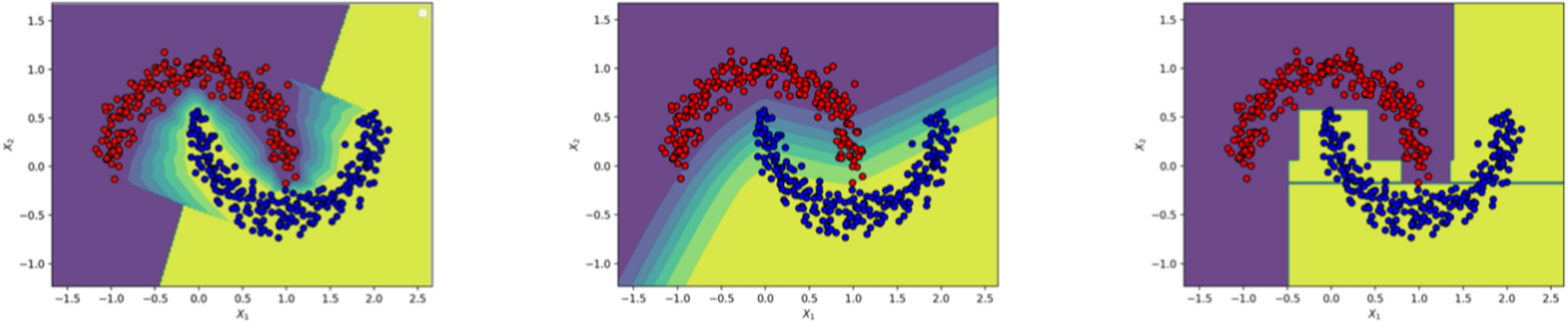
- Three classifiers considered in the paper: the DTL, neural network, and decision tree.
- The estimated probability function of each classifier is plotted with color maps.
- Figures demonstrate the smoothness and robustness of the DTL-based methods, when handling feature interaction problems using artificial data:
    - ‘Moons’ model has two clusters of points with moon shapes, and ‘circles’ model consists of two clusters of points distributed along two circles with different radiuses.
        - Blue y = 1 and red y = 0.

Advantage 1: Local Adaptability

- (a) of both figures:
    - Estimated probability functions of the DTL are piecewise linear in the convex hull of the observations.
        - Advantage when approximating smooth boundary, since it can locally choose to use a piecewise linear model to estimate the probability function.
- (b) less adaptive classification probability (either 0 or 1) using neural network.
- (c) a clear stair-wise shape of the classification region using decision tree.



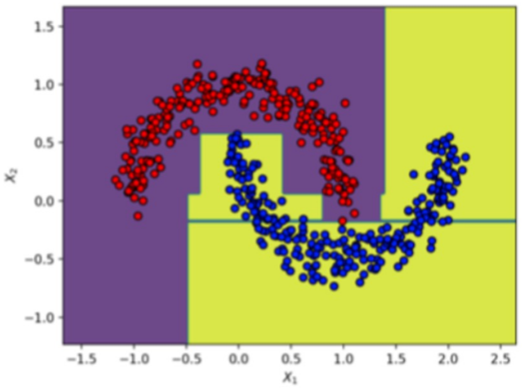
(a) DTL



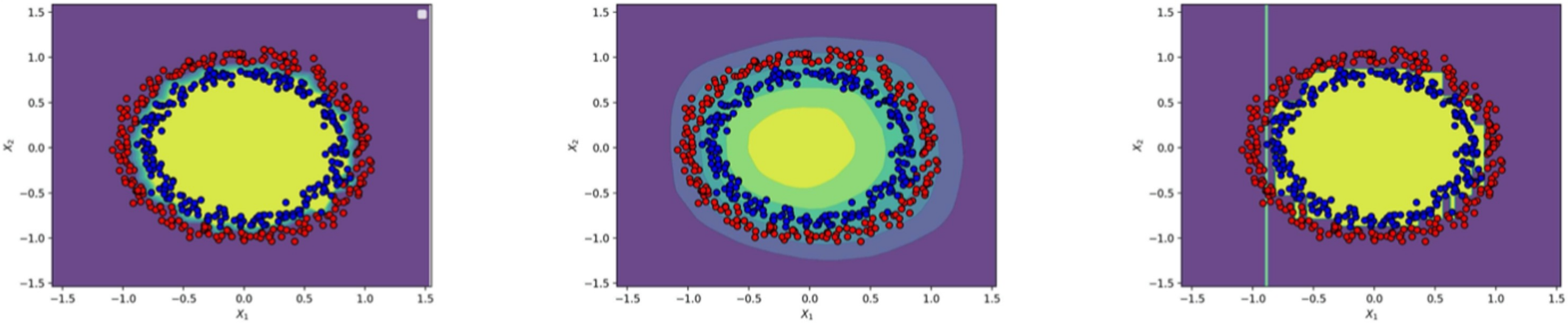
(a) DTL (b) Neural Network (c) Decision tree

Advantage 2: Better Reliability in Dealing with Feature Interactions

- (c) of both figures:
  - Spiky regions in the estimated probability functions using the decision tree.
    - Shows the weakness of the decision tree in capturing the feature interactions.
- (a) vs (b):
  - DTL produces tighter width margins around clusters, increasing reliability of prediction.



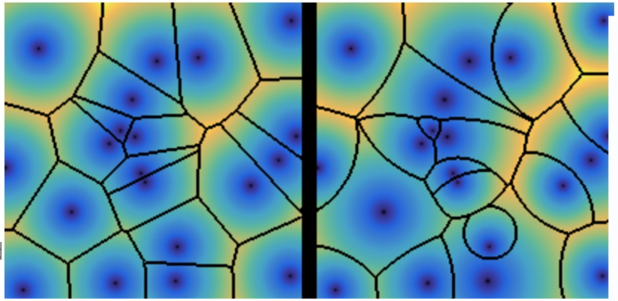
(c) Decision tree

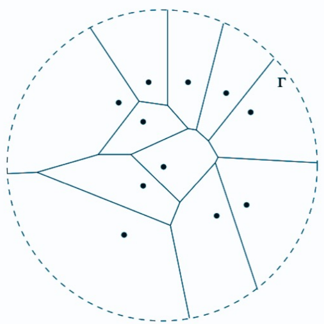


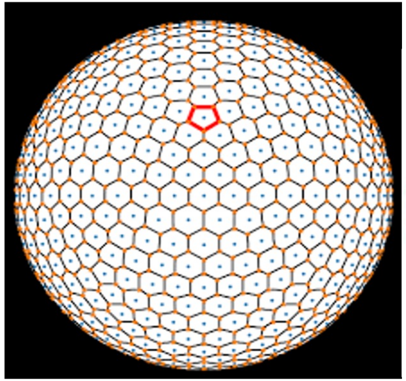
(a) DTL (b) Neural Network (c) Decision tree

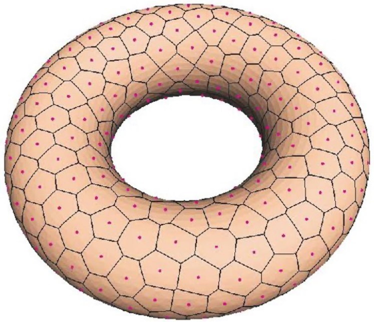
Further Topics

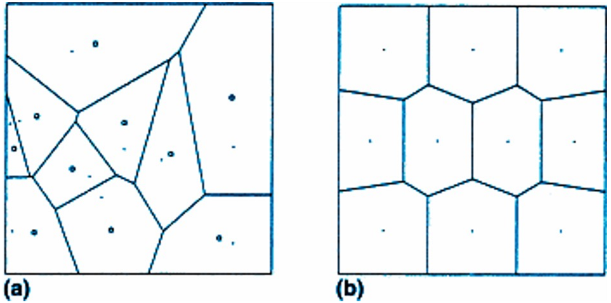
- Voronoi entropy
  - Average, random arrangement, above-average with specific arrangements.
- Voronoi cells of symmetric Delone sets form space-filling polyhedra called plesiohedra
  - Delone sets.
  - Plesiohedra.
  - Cool fact: 38* is upper bound for number of sides of plesiohedron.
- Generalizations and variations:
  - VDs with other structures for sites(lines, circles, polygons, ...).
  - VDs on non-euclidean geometry (hyperbolic, sphere, torus, Laguerre Geometry ...).
  - Higher-order VDs, Furthest Point VDs, VDs in R3+, power diagrams and weighted delaunay triangulations.
  - Weighted VDs, R-WVDs, VWVDs.
  - Bregman VDs, Additive Bregman VDs, VDs of Algebraic Distance Fields.
  - Random VDs, asymptotic properties of RVDs with specified point densities, Voronoi Random Fields, Aspect-Ratio VDs.
  - ‘Thick’ VDs, Perturbed-site VDs, Stable Matching VDs, Tropical VDs, tropical bisector fans, poisson disk sampling.
- Centroidal Voronoi diagrams
  - Mahalanobis CVT, Lp Centroidal Voronoi Tessellation and its Applications.
  - Applications: data, surface reconstruction/mesh generation, obstacle avoidance and path finding.
- Mean-curvature flow of VDs
- Measures of symmetry, continuous measure of symmetry vs voronoi entropy







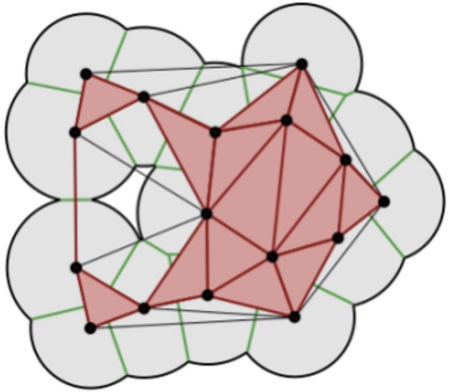
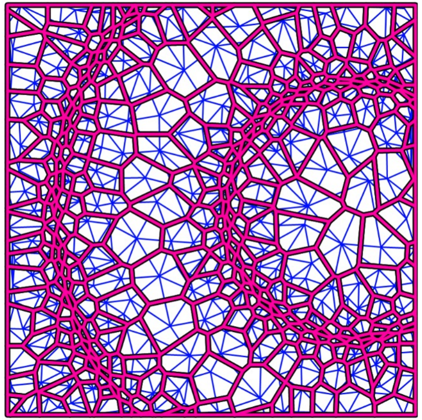
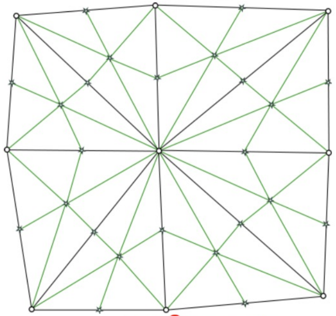
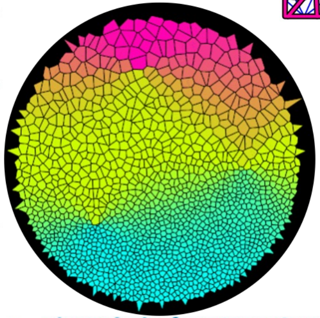
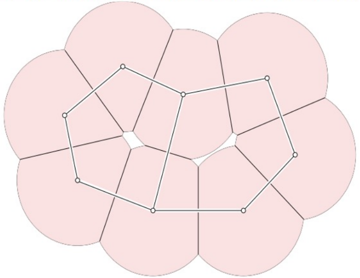
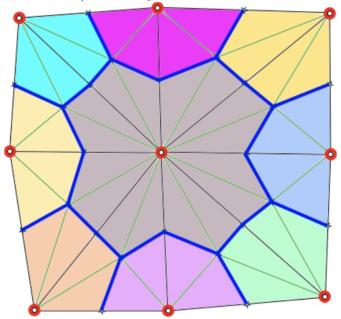
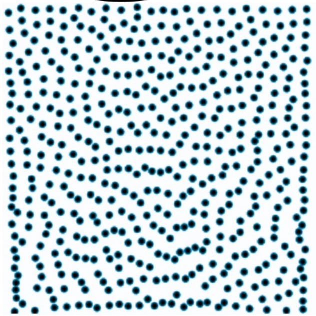
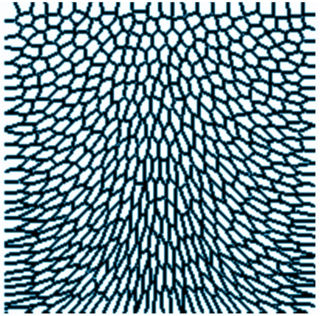




(a) (∘) centroids, (·) generators
(b) (·) generators and centroids

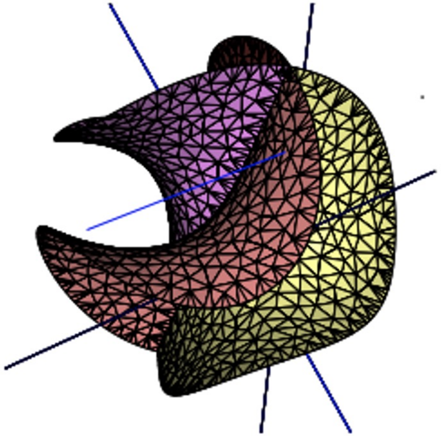
More Further Topics

- More generalisations and variations:
    - Anisotropy of cells, Optimal VDs, Curved VDs.
    - Kinetic VDs, Adaptive VDs, Kinetic DTs, Dynamic DTs.
    - Constrained DTs, conforming DTs, Straight skeleton, medial axis.
    - Fuzzy DTs, Shape DTs, Hodge optimised delaunay triangulations, MAP Decision Rule VDs.
    - Apollonius tessellations, optimal DTs and steiner vertices, Pitteway triangulations.
    - Hybrid VDs, Angular VDs, Variable-radius VDs.
    - Boat-Sail VDs, Time-based VDs with Highways, Riemannian VDs and DTs.
    - Laguerre tessellations.
    - Urquhart graphs, relative neighbourhood graph, Gabriel graphs, Nearest neighbor graphs.
    - DTs with other triangle centres (more than 13200 potential types).
    - VD (circum-centric) dual meshes, barycentric dual meshes, barycentric subdivision, circle packing dual meshes.
    - Piecewise Linear Interpolating Surface of DT, Minimal roughness property.
    - Alpha/Beta Complexes (Duals of VD decomposition of union of expanding disks/spheres).
- Discrete exterior calculus on VDs and DTs, interpolation with VDs and DTs, nearest neighbour interpolation vs natural neighbour interpolation with VDs, barycentric interpolation.
- Complexity of VDs, Random walks on VTs, VDs of Poisson-point processes, VDs as electric networks*
- Seemingly endless applications...

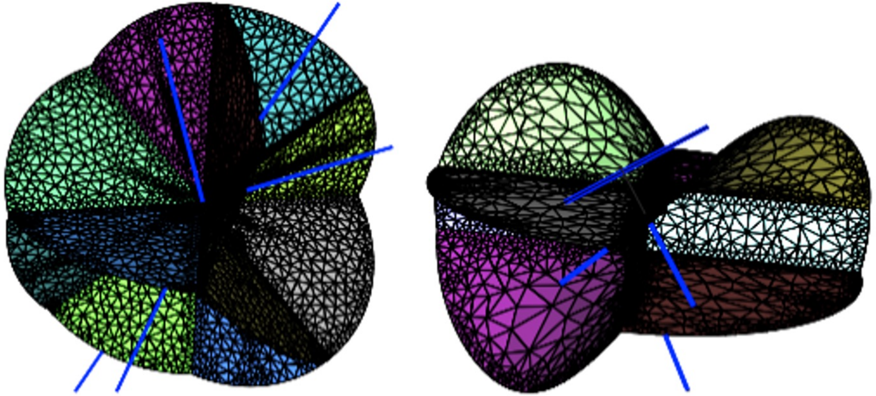
Open Problems

- International Symposium on Voronoi Diagrams in Science and Engineering happened annually 2004-2013
  - Increase work on open problems.
  - Encourage further development and study of applications.
- Compute complexity of Voronoi diagram of a set of lines (or line segments) in three dimensions.
- Algorithm for updating abstract VDs (after deletion of a site) in linear time.
- What is the maximum number of combinatorial changes possible in a Euclidean Voronoi diagram of a set of n points each moving along a line at unit speed in two dimensions?
- Where to place a new site to maximize its Voronoi cell ?



(a)

- (a) VD of 4 lines, obtained by rotating one line around the z-axis. All bisectors meet in that axis.
- (b) VD of 4 lines intersecting in one point.
- (c) VD of 4 lines, two lines intersect and the others are parallel to each of them, respectively.



(b) (c)

Some Questions to Consider

- What do the VDs of the vertices of 11 Archimedean tilings and their duals look like?
  - What about the DTs which are triangulations resulting from further subdivision?
  - What about their DTs which aren't triangulations?
- What is the minimum number of colours that you need when colouring VD so that regions sharing an edge have different colours?
  - Is there a way to predict the minimum number of colours required by seeing some small chunks of the tessellations?
  - What about in higher dimensions?
- Suppose you have a VD and you build a new VD using its vertices as the sites,
  - Does this ever result in the original VD (if it does, when)?
  - What are the conditions for getting regular/semi-regular tiling at this stage?
  - What does the result look like after a couple of iterations?
  - Are there any initial point arrangements that result in the same VD after 3 or more iterations as the first VD?
- What are the geometric and statistical properties of aperiodic tilings produced from VDs and DTs of not PPS but other types of underlying aperiodic tilings?
- Given a VD with some symmetries, do any higher order* VDs of this one have the same symmetries?
- Can the ideas of Delaunay lofts and DTLs be synthesized?

Sources Used

- Wikipedia(s)...
- Grid generation and optimization based on centroidal Voronoi tessellations
- A REVIEW OF PROPERTIES AND VARIATIONS OF VORONOI DIAGRAMS
- Computing Dirichlet Tessellations
- Delaunay Triangulations General Properties
- Convex Hull
- Delaunay Triangulations
- Distances, Edges, Domains
- VDs and DTs
- A Voronoi Poset
- How many facets on average can a tile have?
- Flipping Edges in Triangulations
- Parallelism in RIC Algorithms
- VDs and DTs by S.Fortune
- Frankenstein
- Other Minkowski Cases VDs
- Map Fire - Application
- Application Example: Voronoi Weighting of Samples in MCI
- S-hull Algorithm
- Statistical Topology of Perturbed 2D Lattices
- Real-time Cave Destruction Using 3D Voronoi
- Bowyer-Watson Algorithm
- Geometric Stability
- 2D DT and more
- VDs from DTs and Backwards
- VDs: A Survey of a Fundamental Data Structure
- Voronoi Diagrams
- Voronoi Entropy, Spirals
- Voronoi Tessellations and Applications
- Voronoi Diagrams and Beyond
- Voronoi, Delaunay, and Hulls
- Watson
- Hyperbolic Voronoi Diagram: A Fast Method
- CVD Applications and Algorithms
- US Airports
- Modified VDs and Anisotropy
- VDs and Soccer