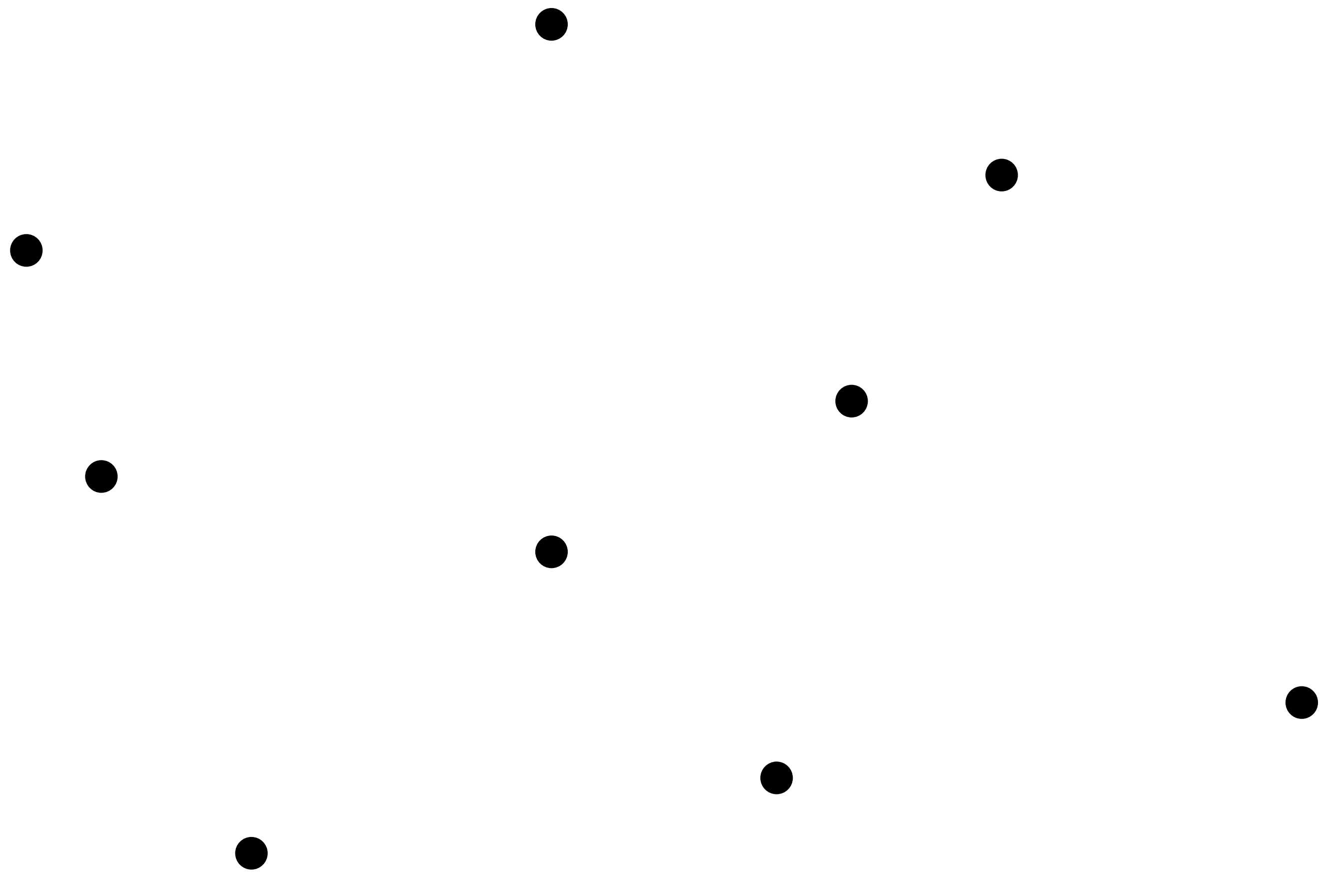
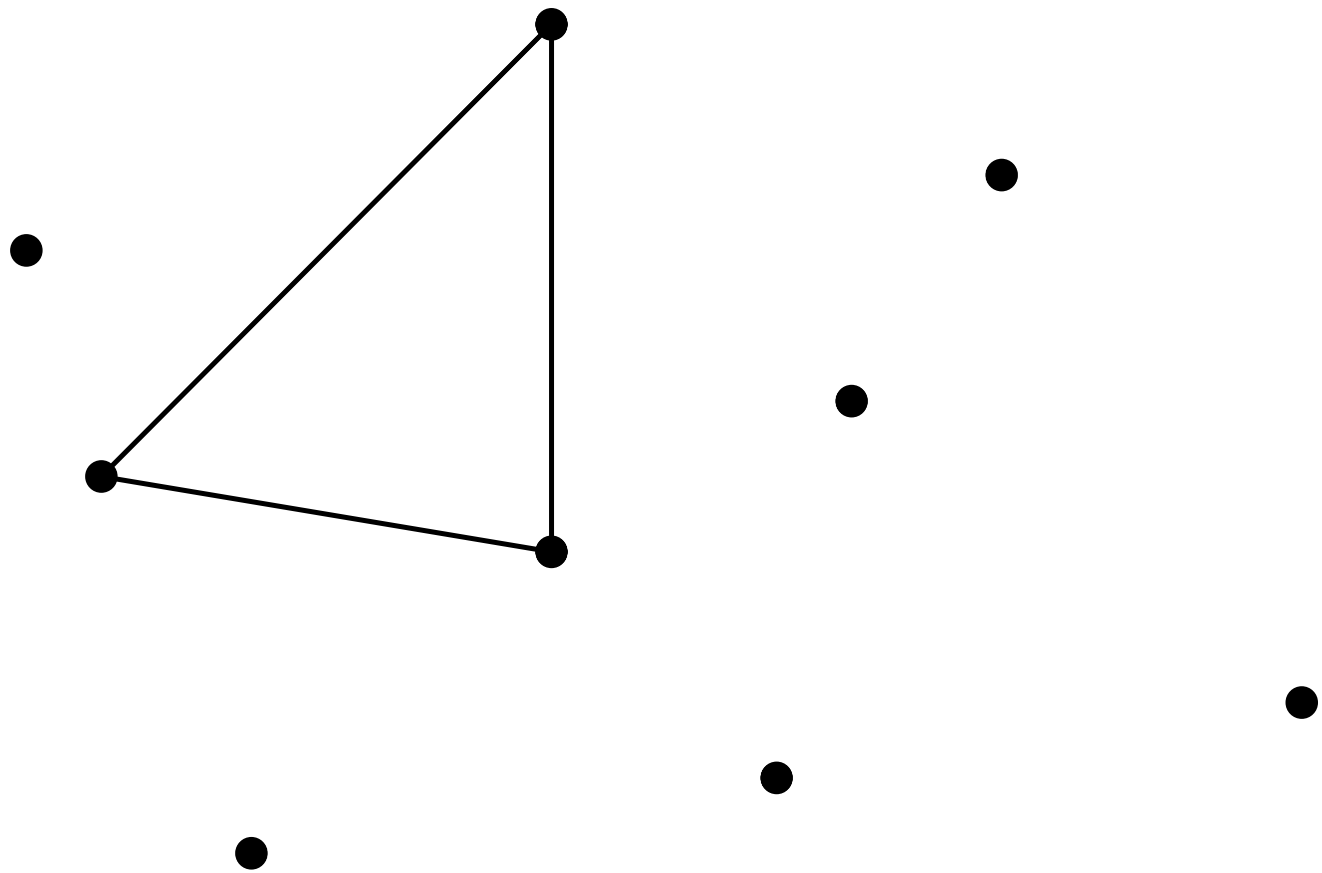
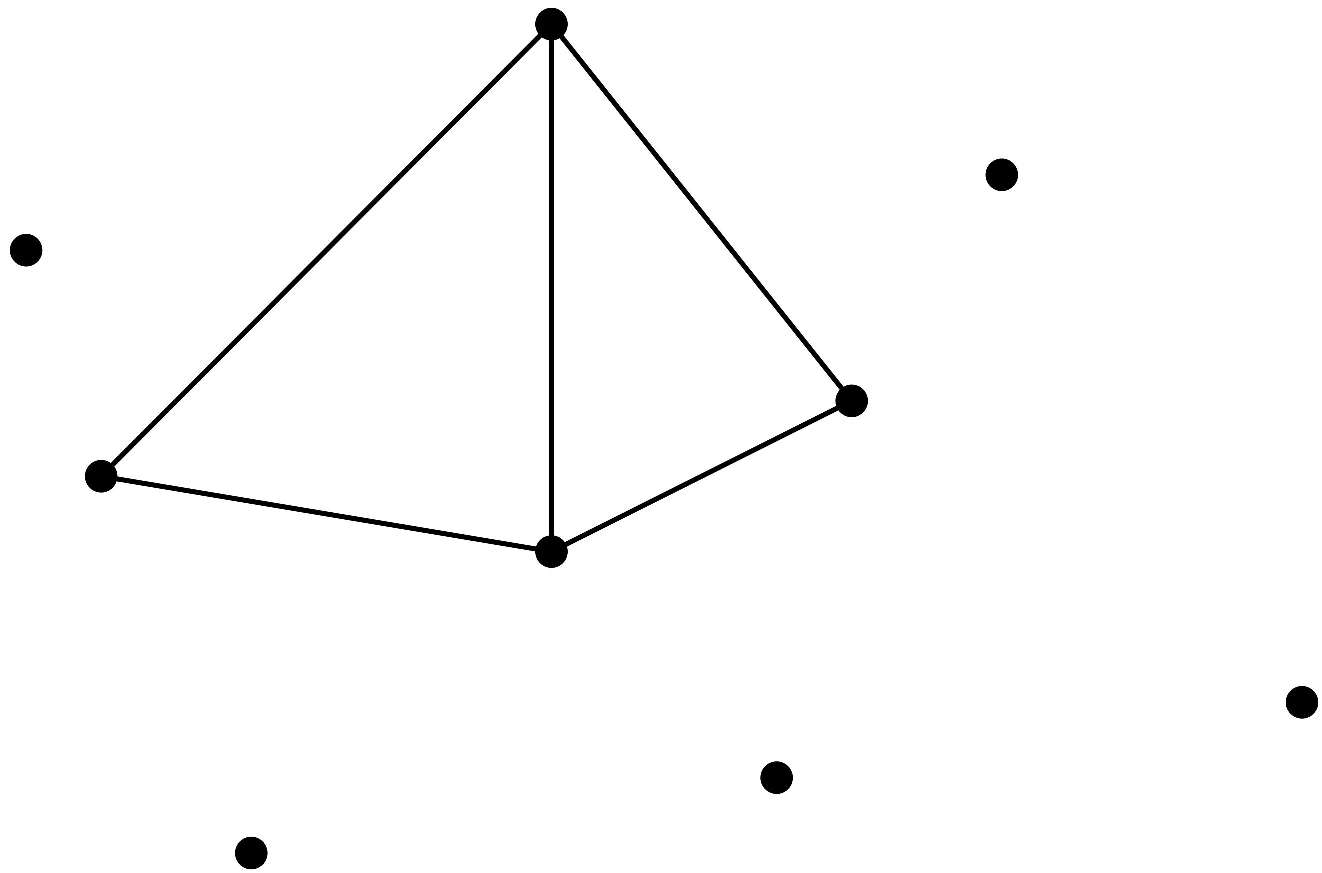
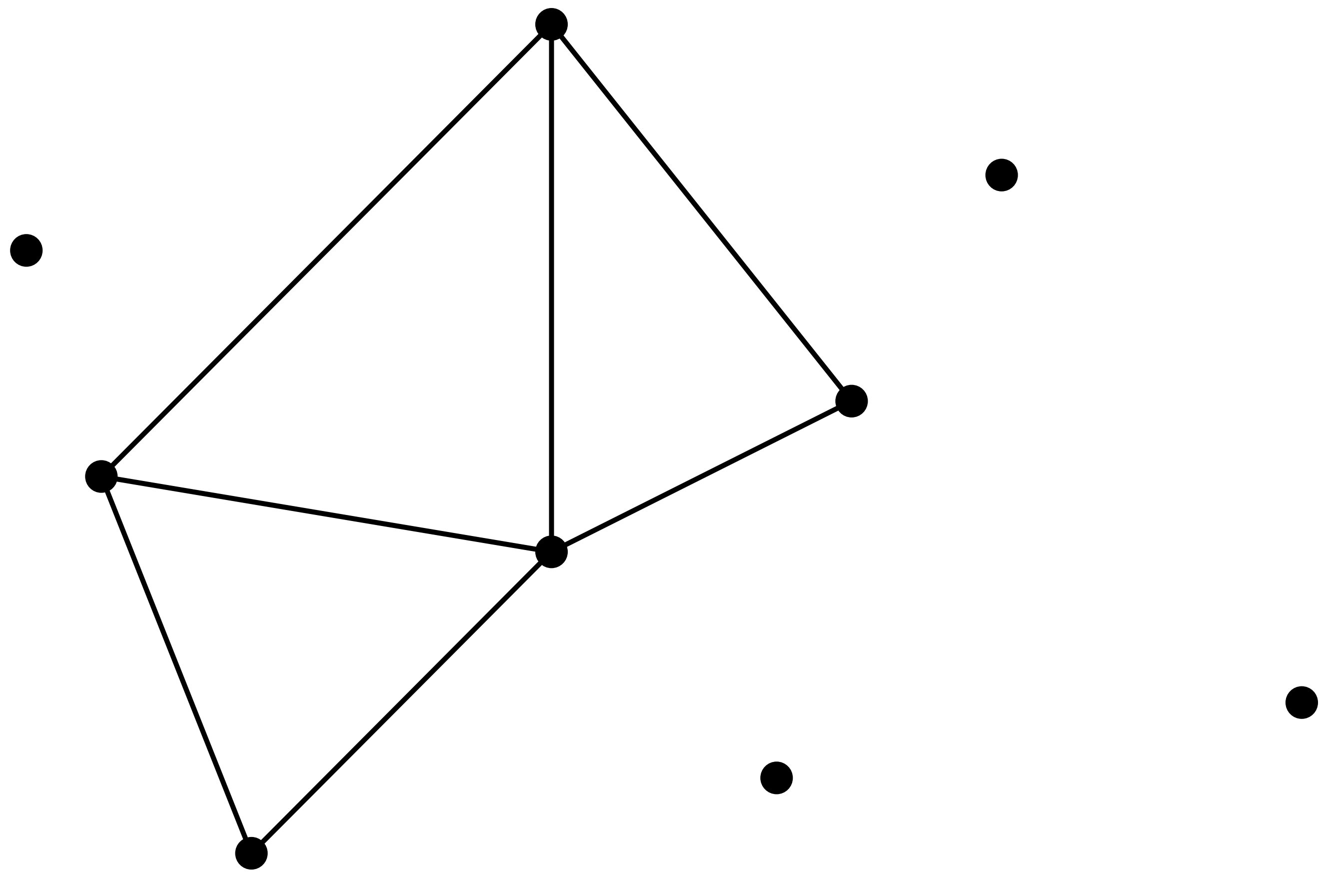
Nonobtuse Triangulation of PSLGs

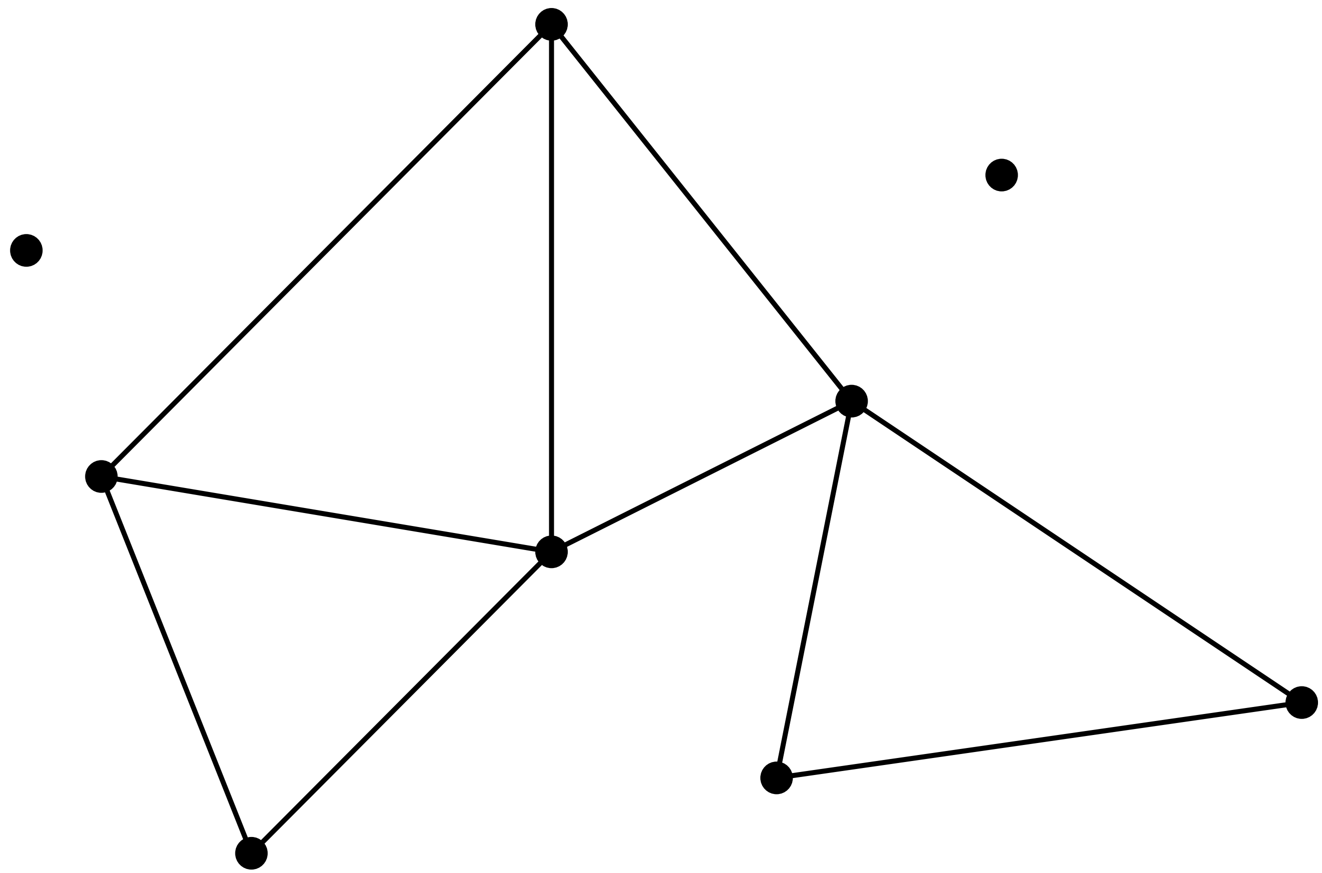
Christopher J. Bishop
SUNY Stony Brook

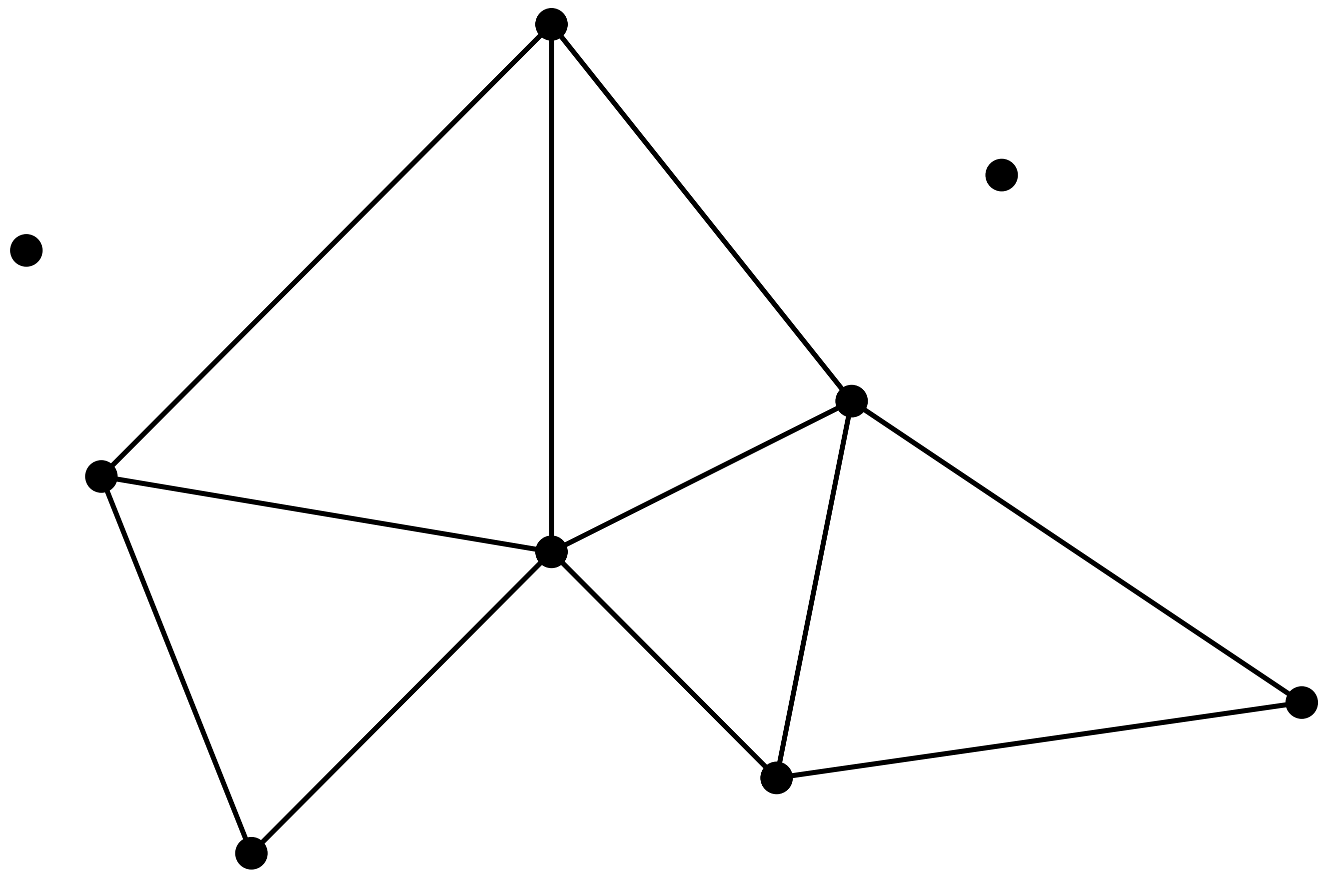


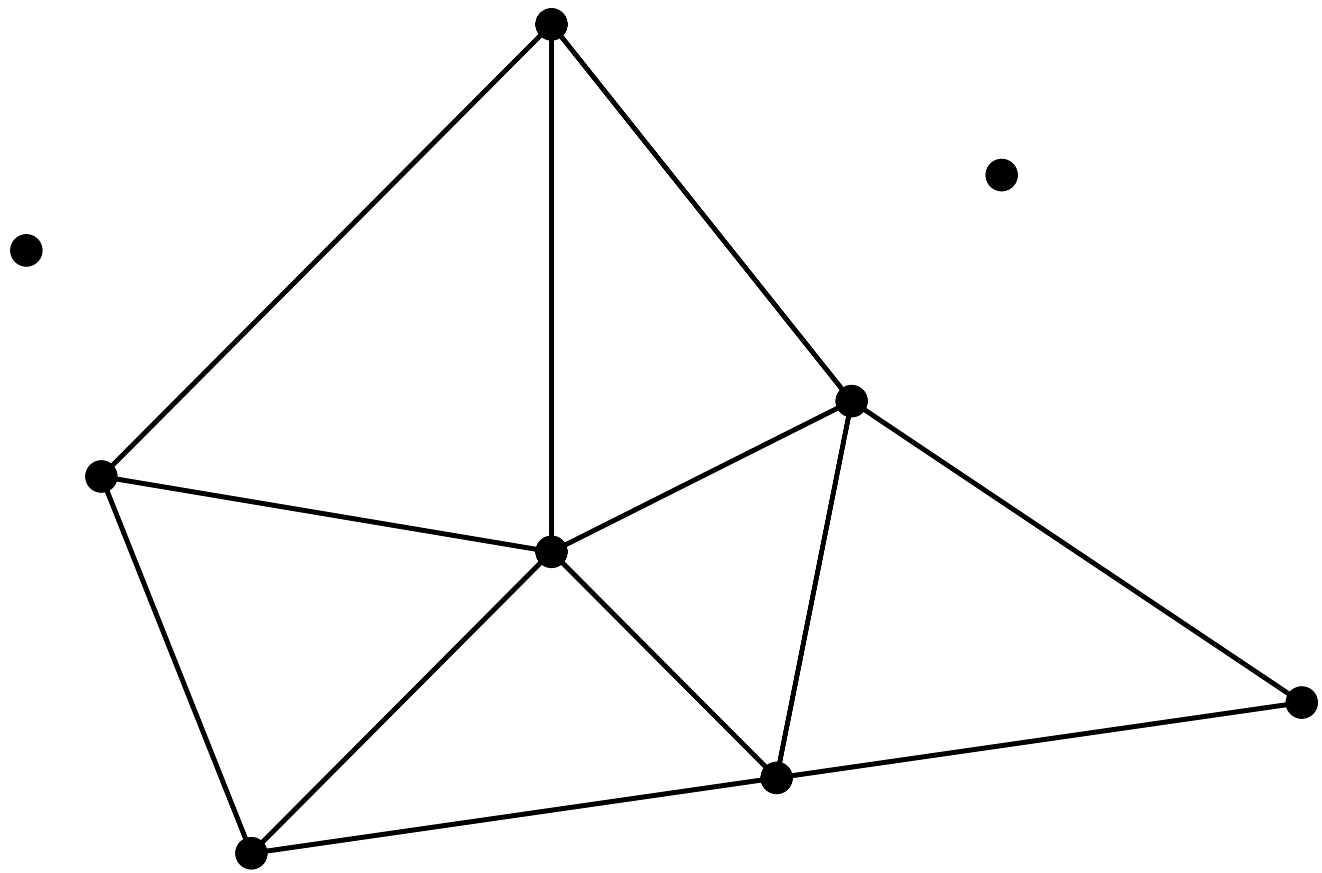


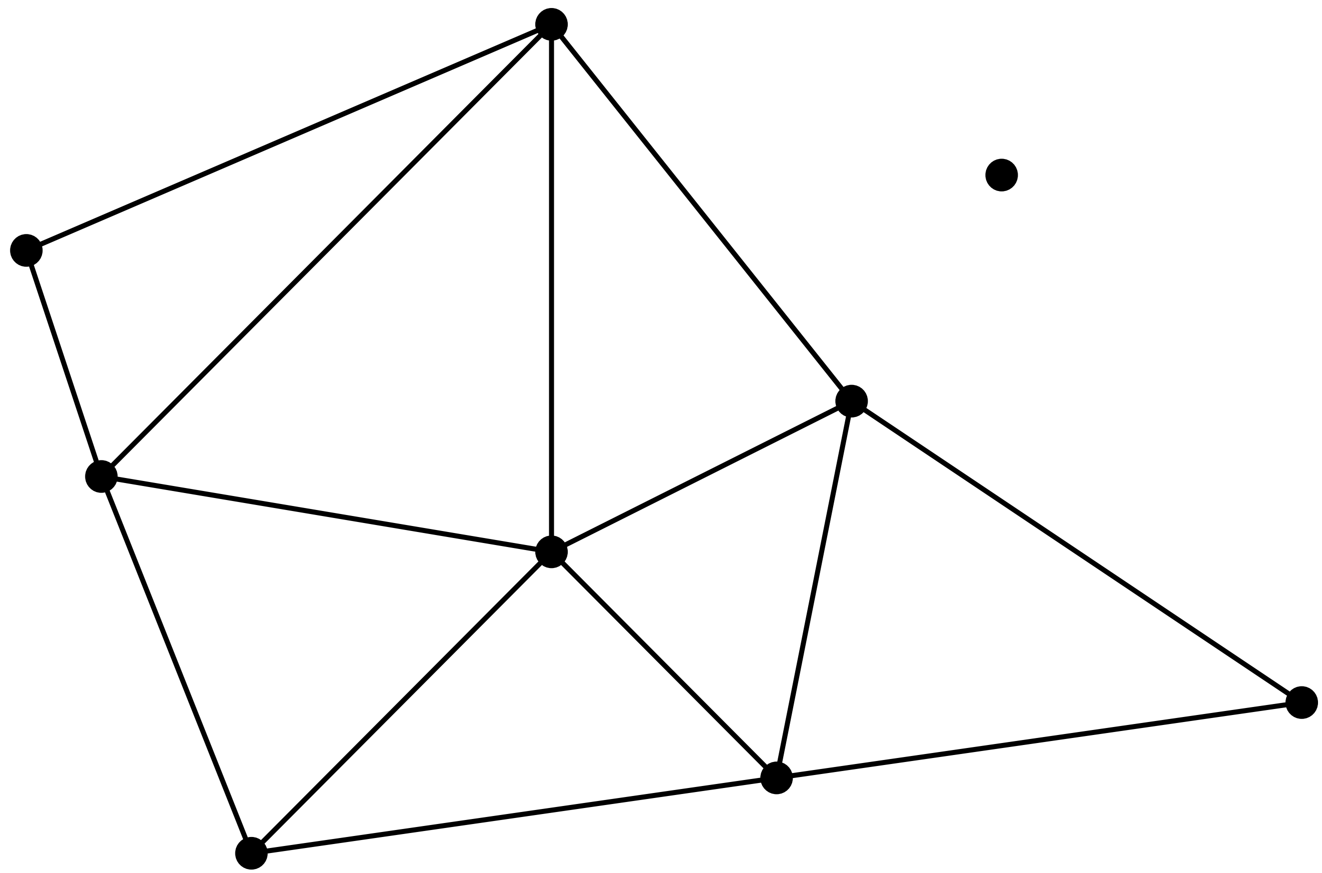


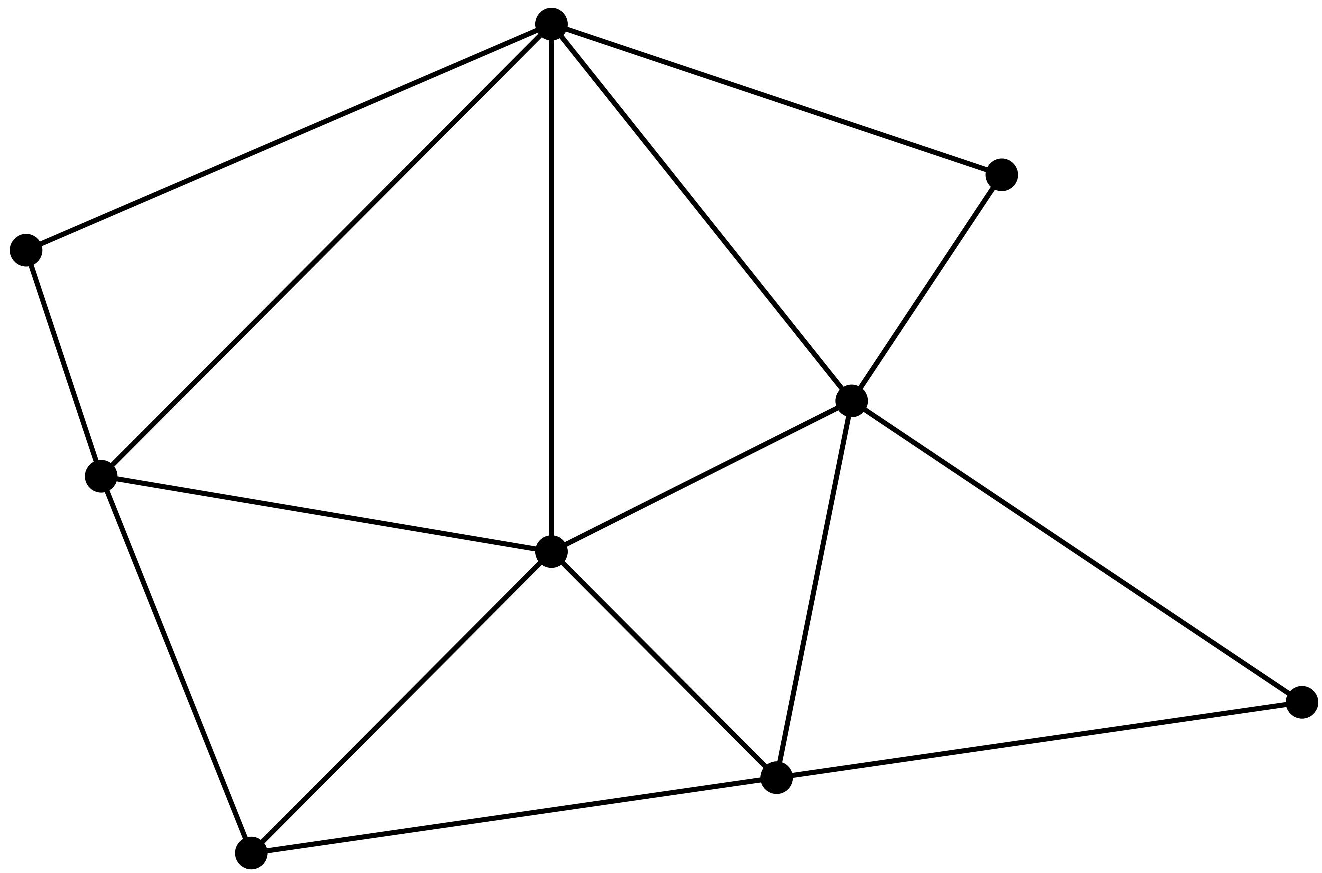


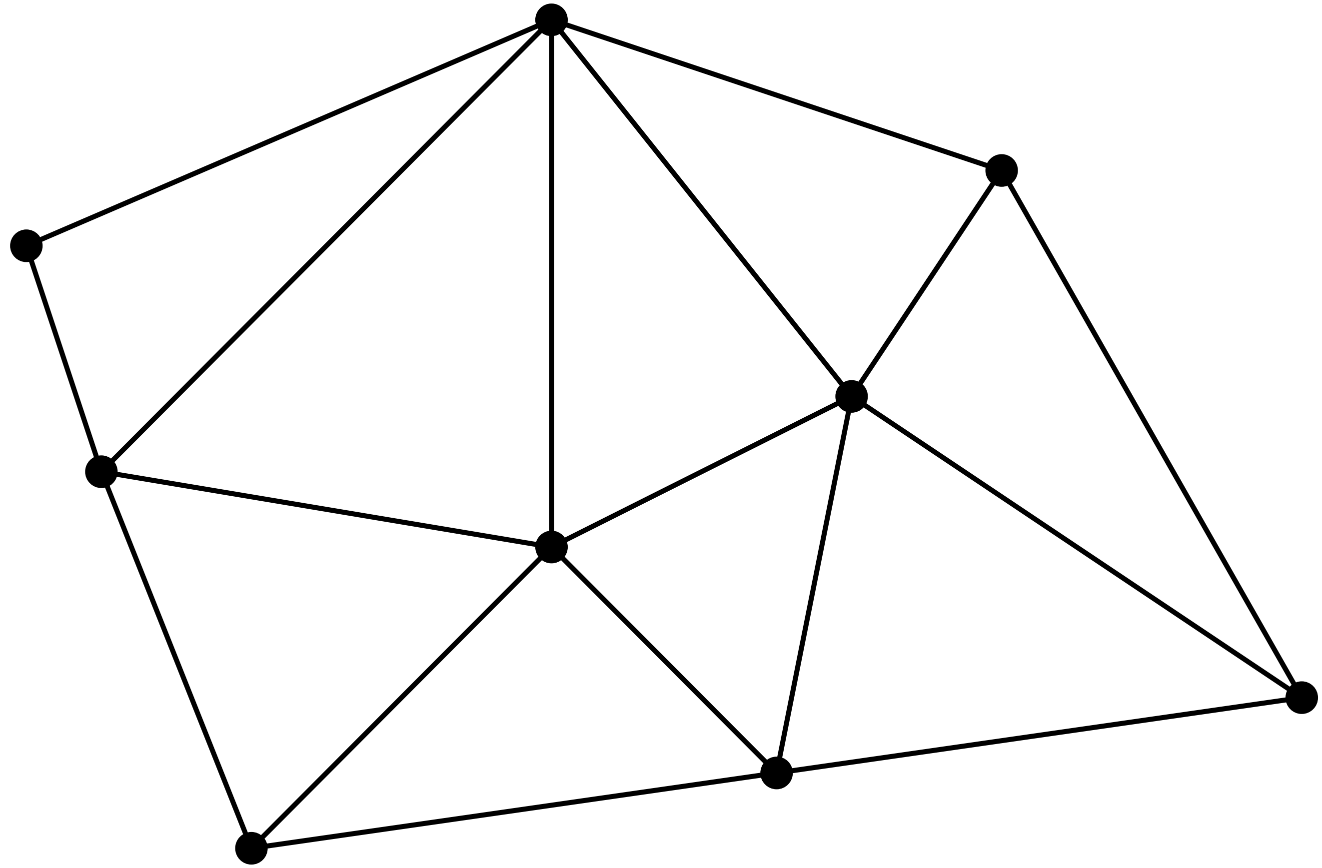




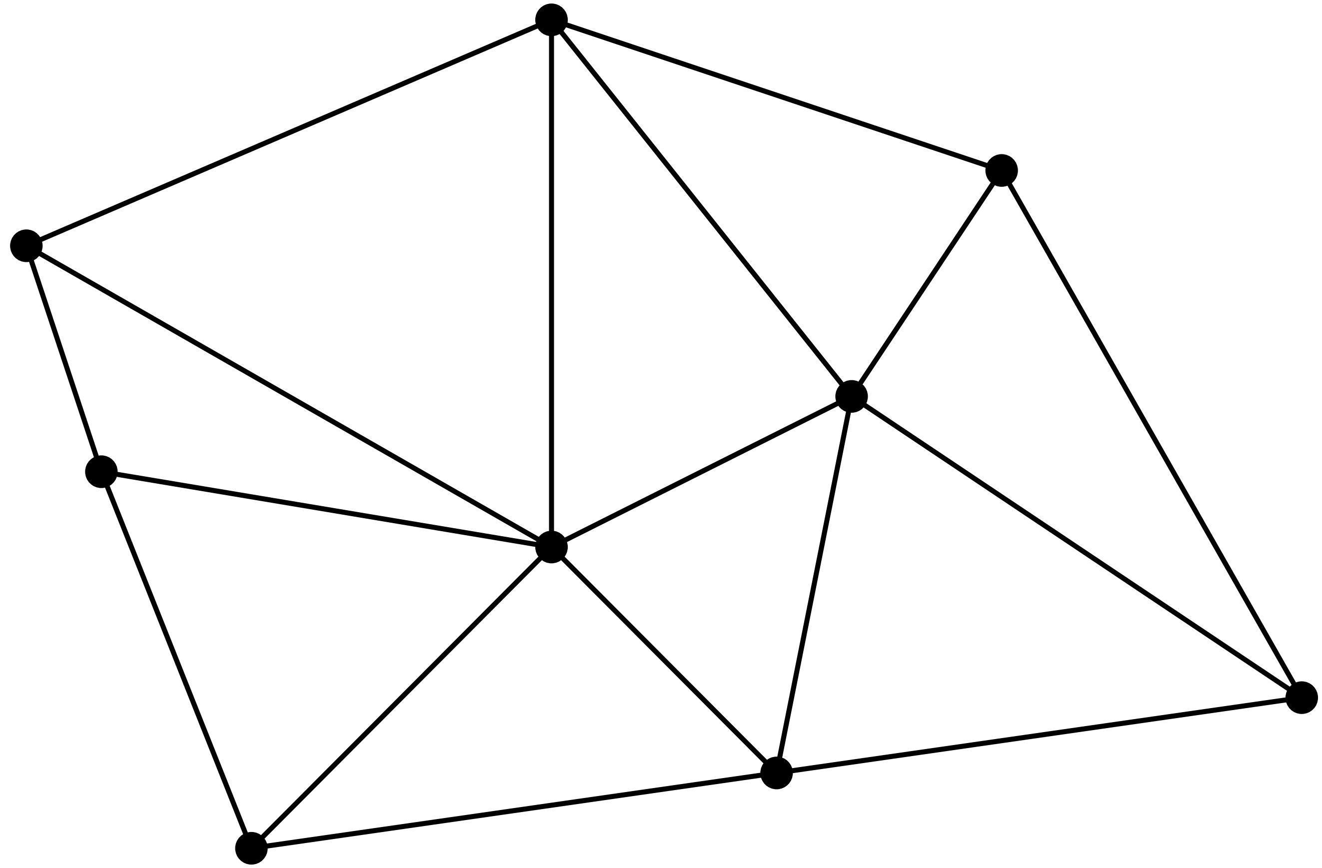




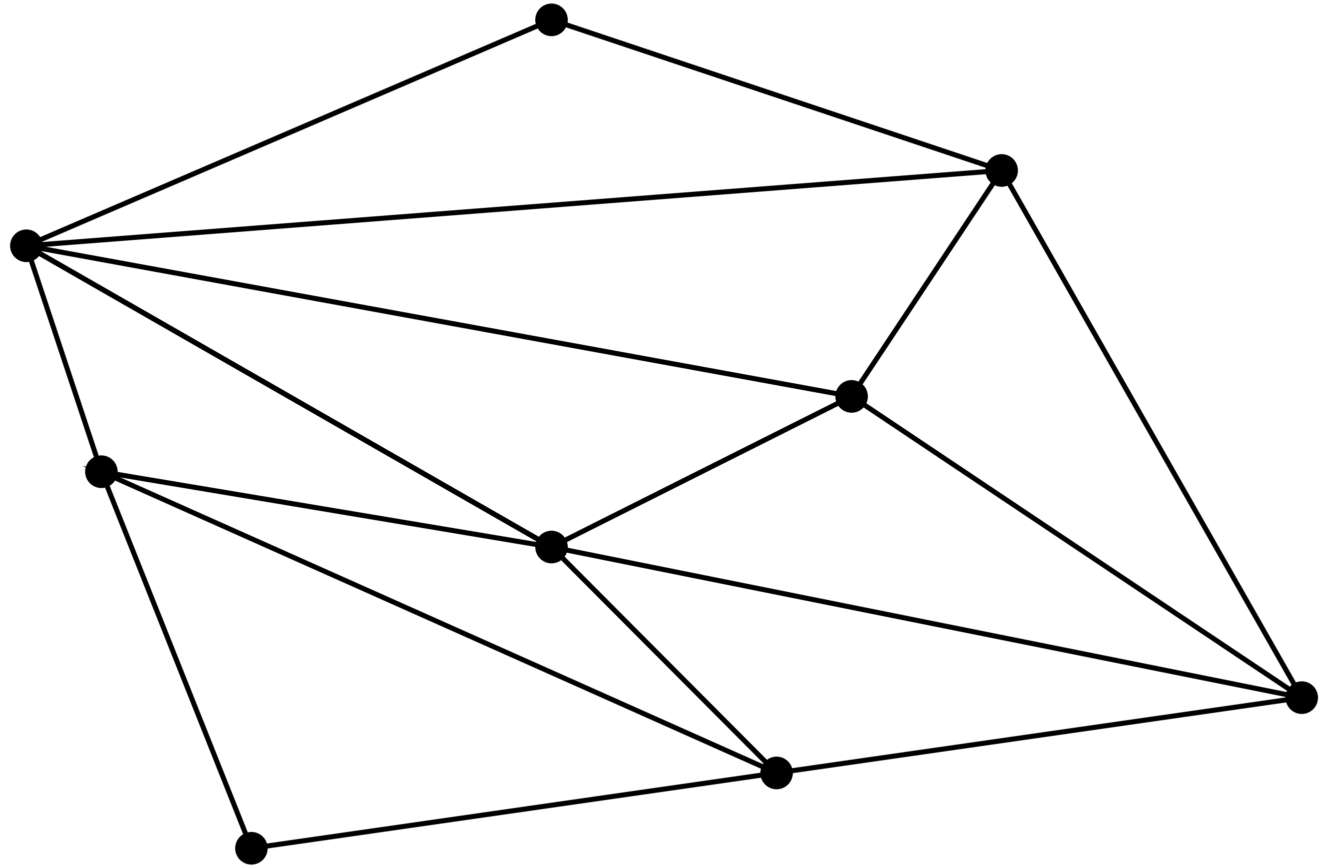




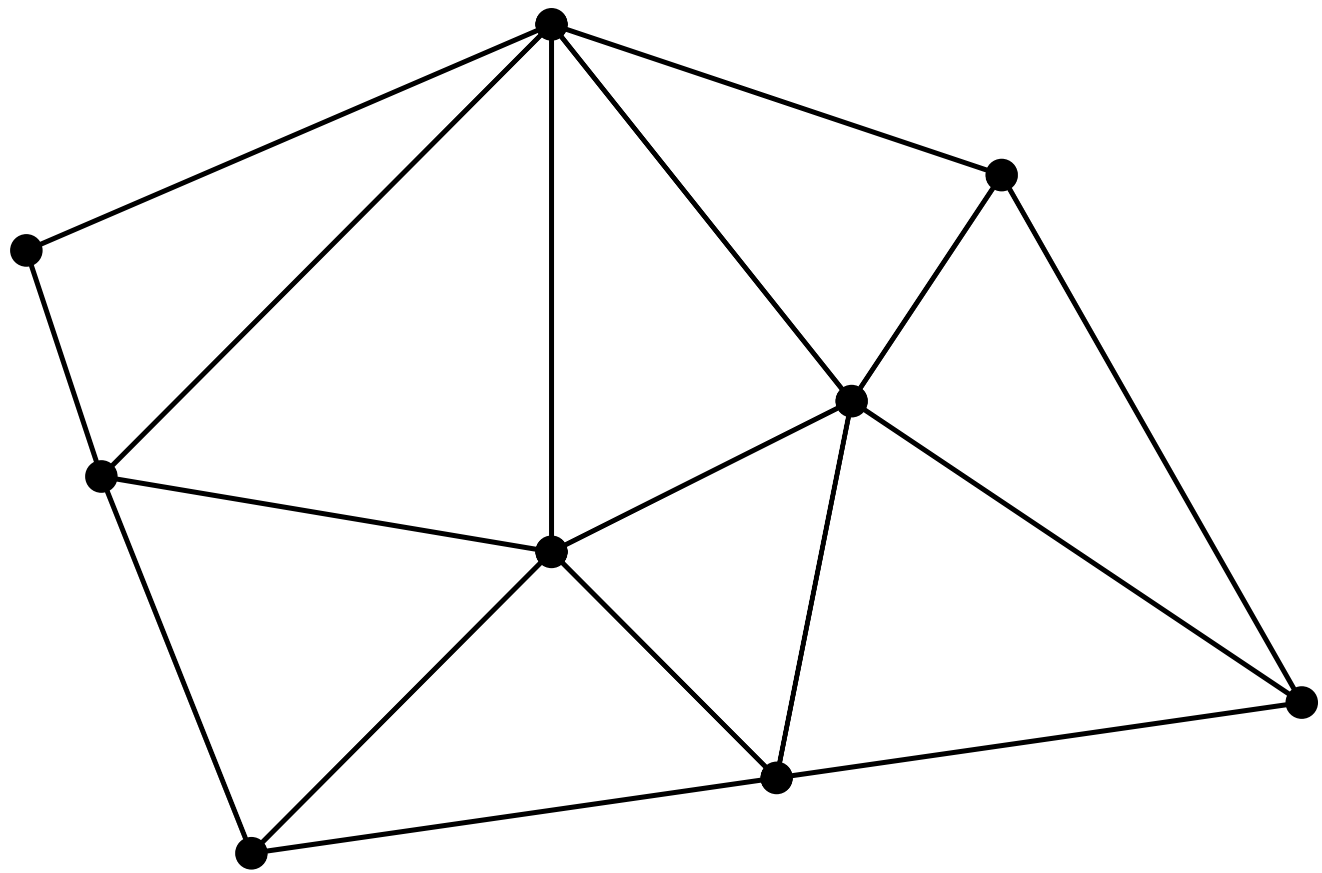
Triangulation of a point set



Another triangulation (flipped a diagonal)

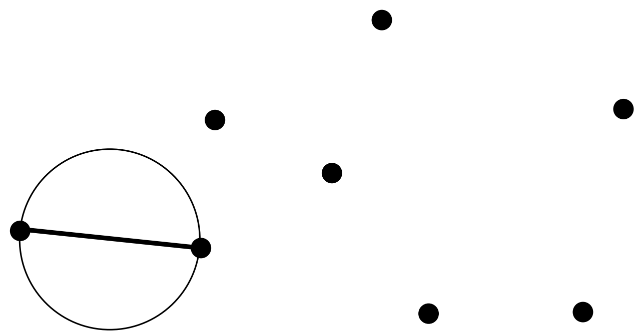


Yet another



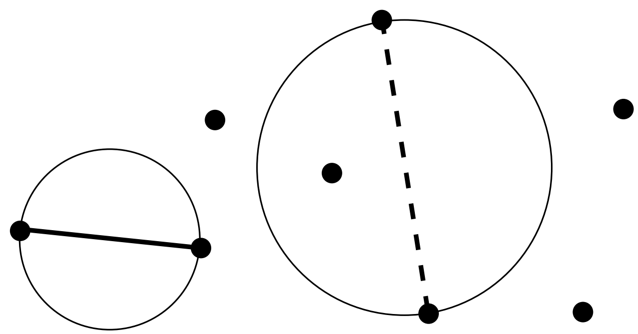
This one is “best” = smallest maximum angle.

The segment $[v, w]$ is a **Gabriel** edge if it is the diameter of an open disk containing no points of V.



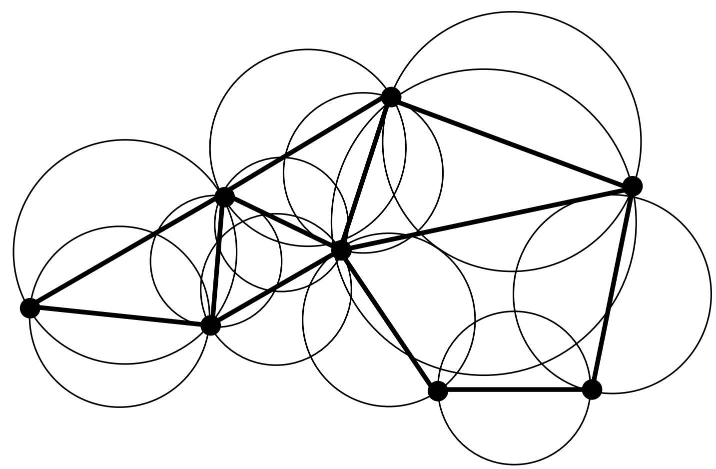
Gabriel edge.

The segment $[v, w]$ is a **Gabriel** edge if it is the diameter of an open disk containing no points of V.

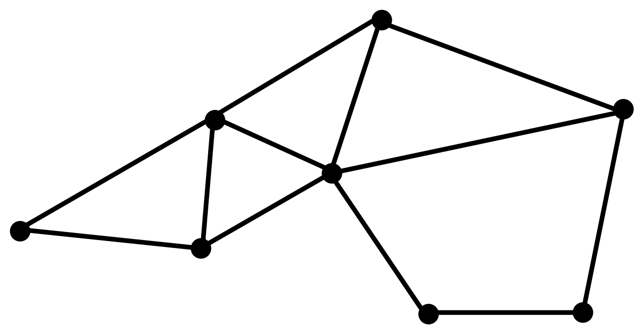


Not a Gabriel edge.

The segment $[v, w]$ is a **Gabriel** edge if it is the diameter of an open disk containing no points of V.



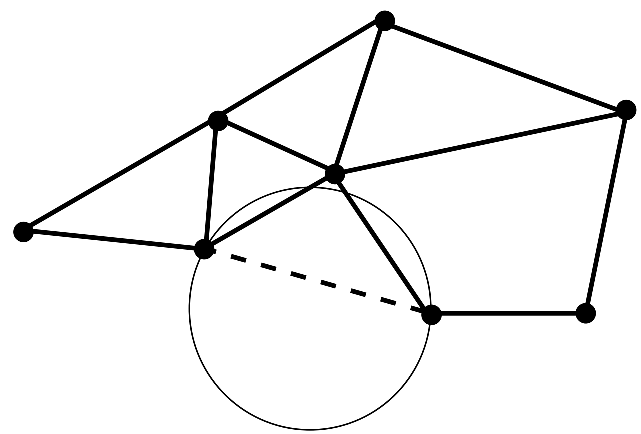
The segment $[v, w]$ is a **Gabriel** edge if it is the diameter of an open disk containing no points of V.



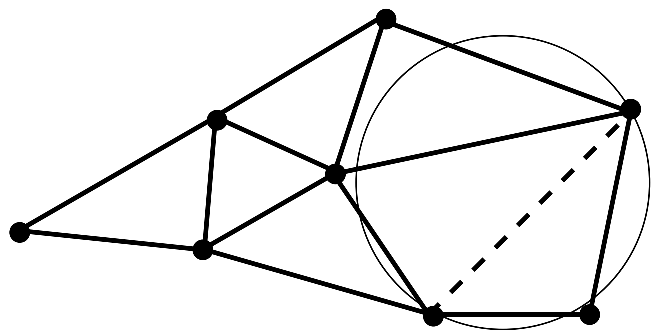
Gabriel graph contains the minimal spanning tree.

Gabriel and Sokol, *A new statistical approach to geographic variation analysis*, Systematic Zoology, 1969.

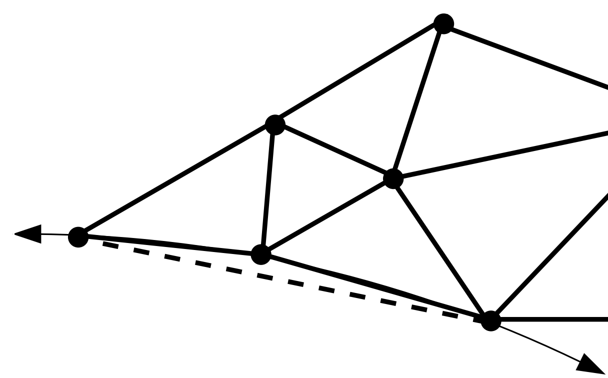
Gabriel edge is a special case of a **Delaunay** edge: $[v, w]$ is a **chord** of an open disk not hitting V.



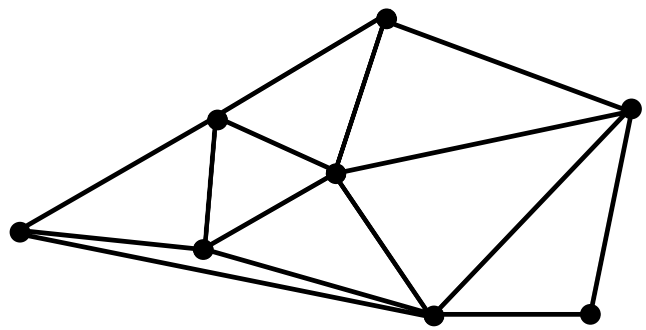
Gabriel edge is a special case of a **Delaunay** edge: $[v, w]$ is a **chord** of an open disk not hitting V.



Gabriel edge is a special case of a **Delaunay** edge: $[v, w]$ is a **chord** of an open disk not hitting V.

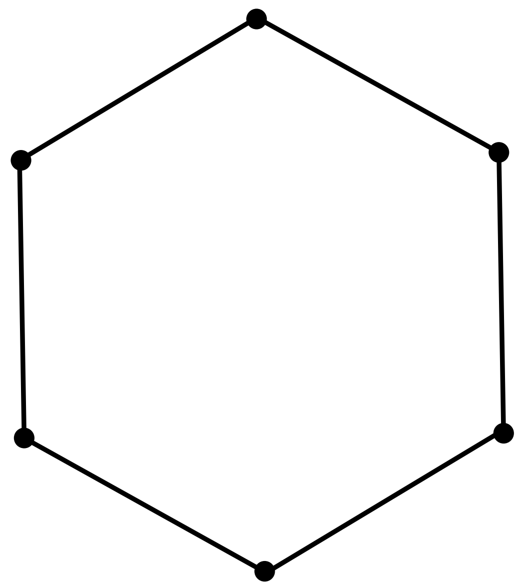


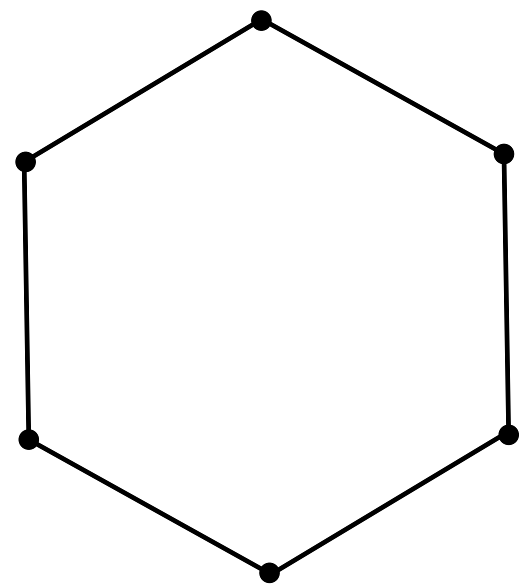
Gabriel edge is a special case of a **Delaunay** edge: $[v, w]$ is a **chord** of an open disk not hitting V.



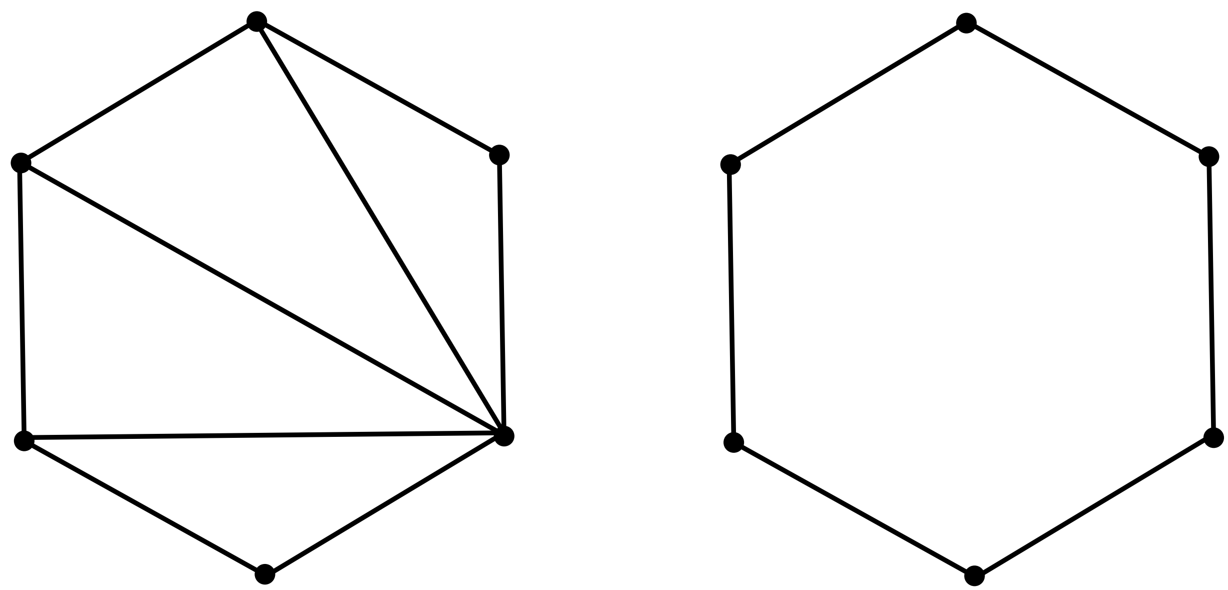
Delaunay edges give a triangulation that minimizes the maximum angle, i.e, has the "best" geometry if only original points are used.

We can add extra points (Steiner points) to get better shaped triangles.

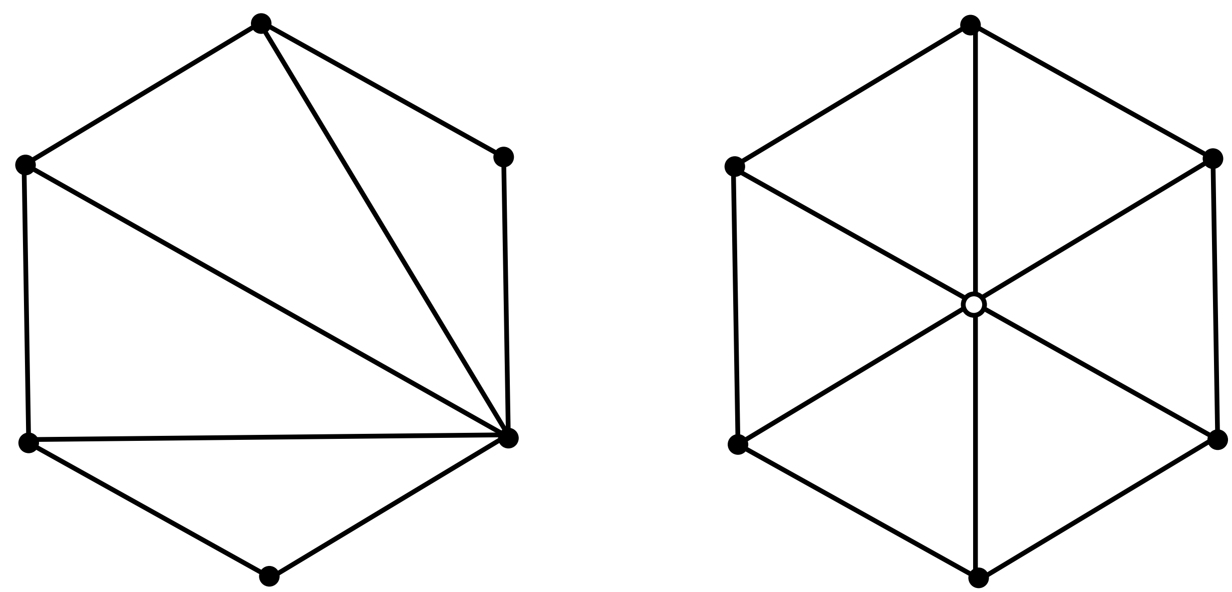




We can add extra points (Steiner points) to get better shaped triangles.

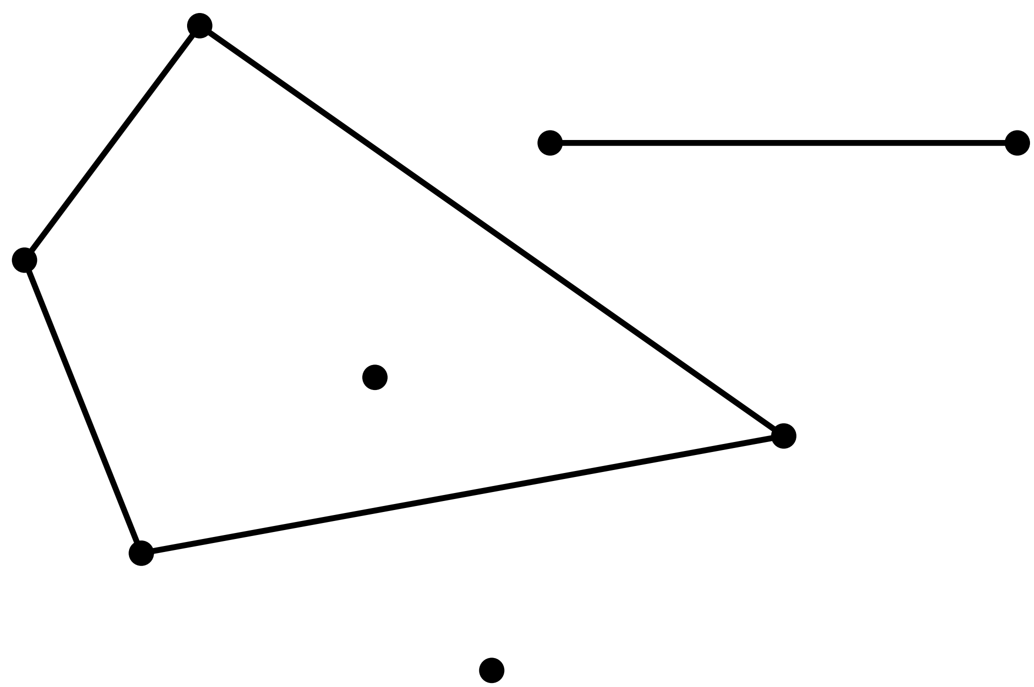


We can add extra points (Steiner points) to get better shaped triangles.



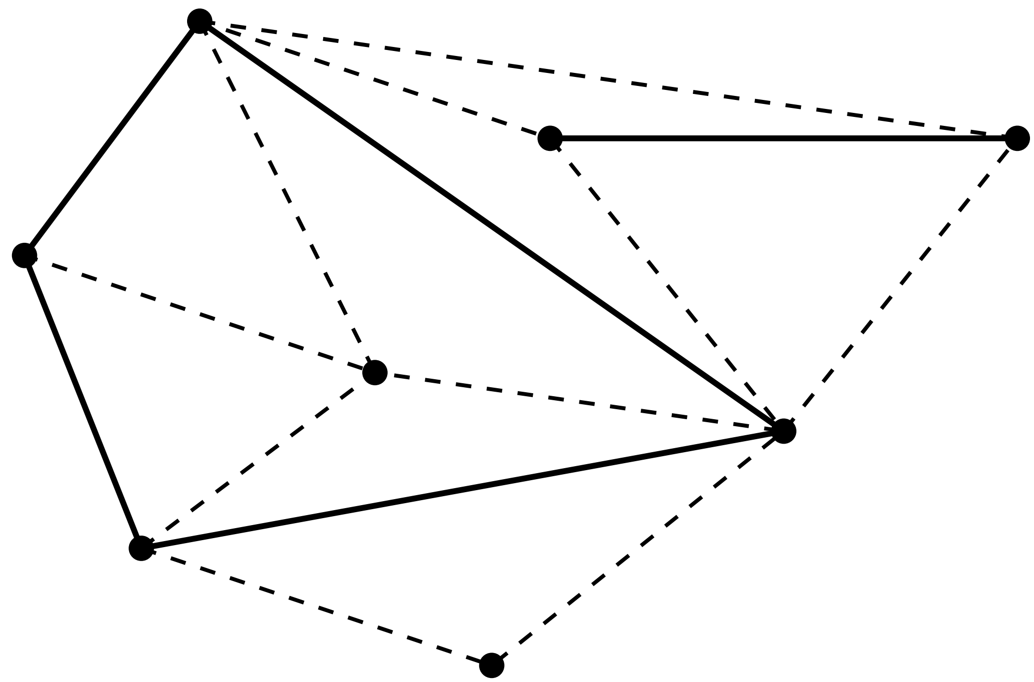
Sometimes we force certain edges in the triangulation.

A **Planar Straight Line Graph** (PSLG) is a finite point set plus a set of disjoint edges between them.



Sometimes we force certain edges in the triangulation.

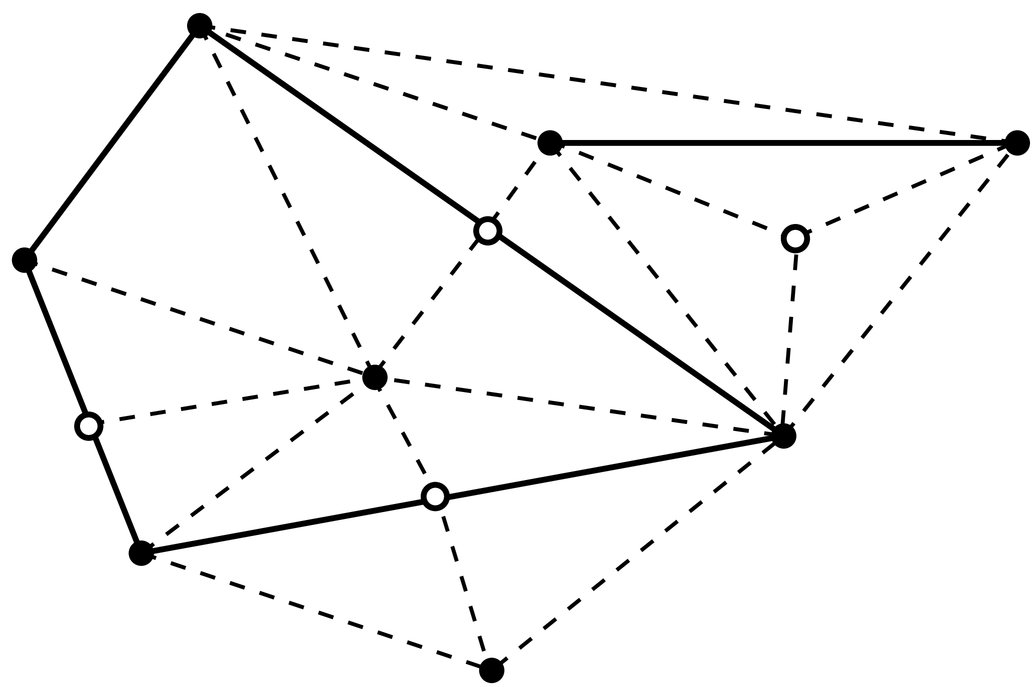
A **Planar Straight Line Graph** (PSLG) is a finite point set plus a set of disjoint edges between them.



A triangulation of a PSLG is a triangulation of a point set which covers the edges of the PSLG.

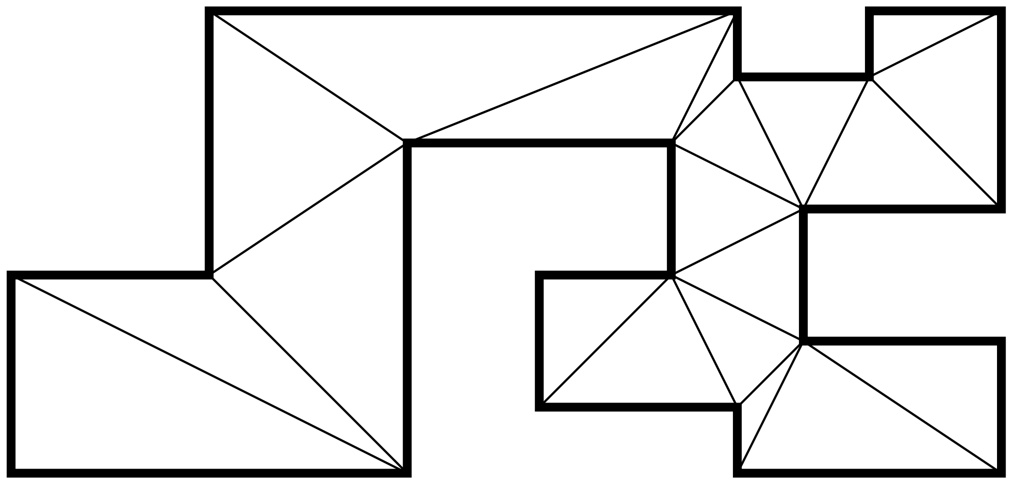
Sometimes we force certain edges in the triangulation.

A **Planar Straight Line Graph** (PSLG) is a finite point set plus a set of disjoint edges between them.



A triangulation of a PSLG is a triangulation of a point set which covers the edges of the PSLG.

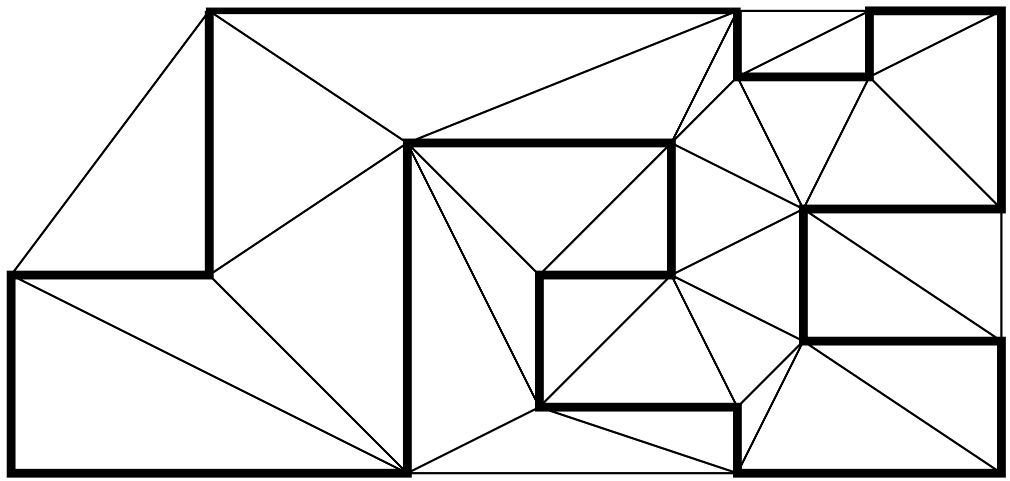
Special case of PSLG is a polygon.



A triangulation of a polygon only covers the interior.

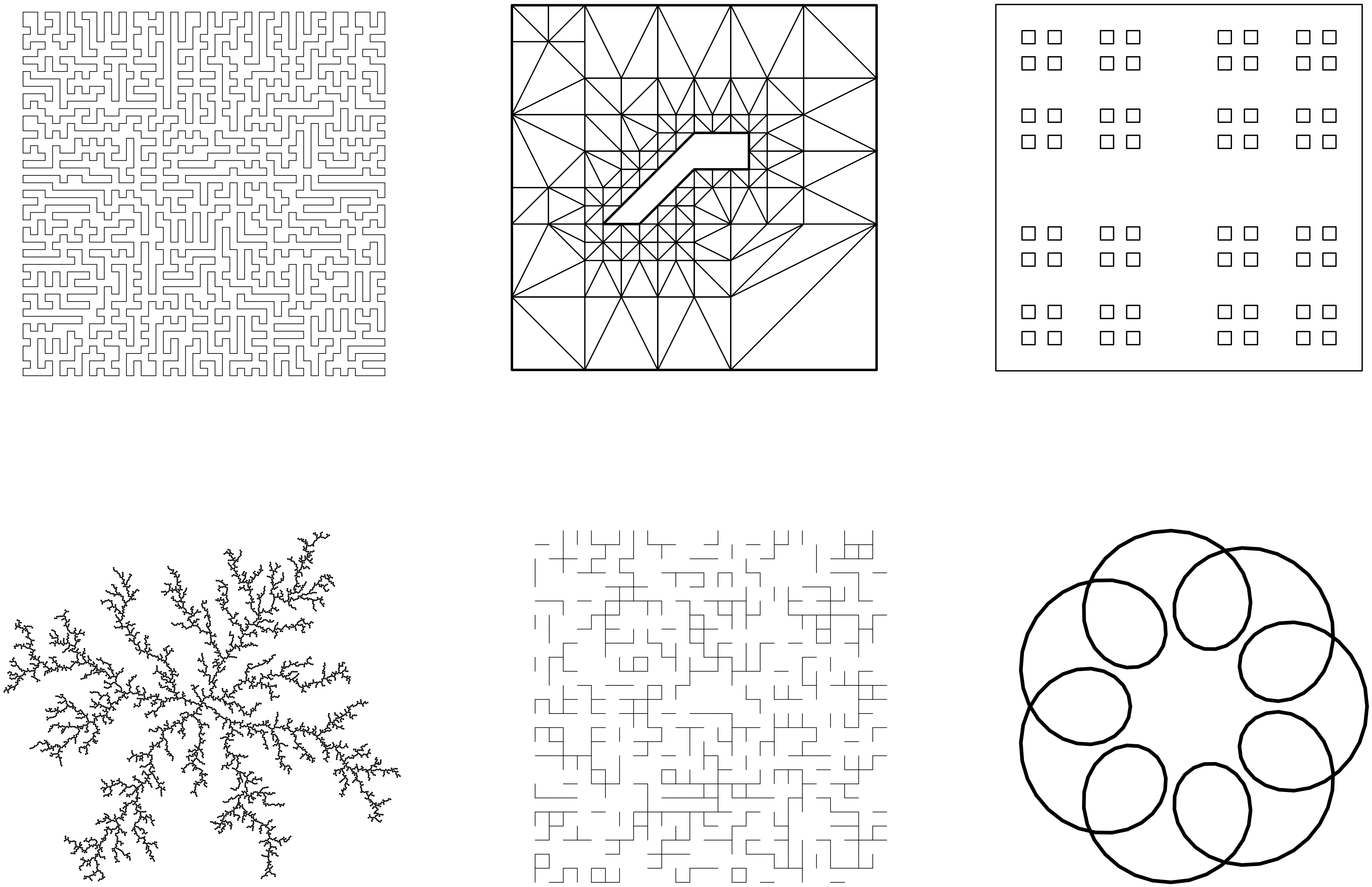
A triangulation of a polygon has a tree structure.

Special case of PSLG is a polygon.



A triangulation of a polygon only covers the interior.

A triangulation of a polygon has a tree structure.



More examples of PSLGs

Two conflicting goals: add Steiner points so we

- Triangulate with best geometry (angles bounded)
- Triangulate with least complexity (fewest elements).

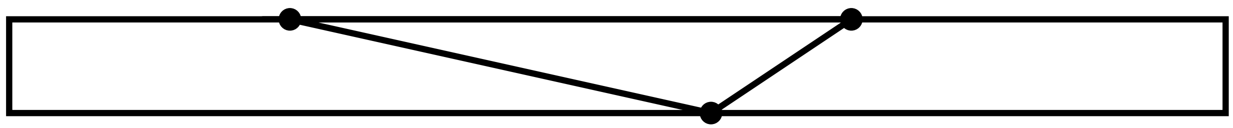
Compromise: find best uniform angle bounds that allow complexity bounds depending only on n.

Nonobtuse triangulation ($\leq 90°$) is best we can do.

Why?

For $1 \times R$ rectangle

$$\text{number of triangles} \gtrsim R \times (\text{smallest angle})$$



So uniform complexity $\Rightarrow$ no lower angle bound.

If all angles are $\leq 90^\circ - \epsilon$ then all angles are $\geq 2\epsilon$.

So nonobtuse triangulation is best we can hope for.

(There are estimates with lower angle bounds, but they depend on geometry.)

Some history:

• Nonobtuse triangulation is always possible (no complexity bounds): Burago, Zalgaller 1960 and Baker, Grosse, Rafferty, 1988

• $O(n)$ for points sets: Bern, Eppstein, Gilbert 1990

• $O(n^2)$ for polygons: Bern, Eppstein 1991

• $O(n)$ for polygons: Bern, Mitchell, Ruppert 1994

• If there is a nonobtuse triangulation, there is an acute triangulation: Maehara 2002, Yuan 2005

• Many heuristics for nonobtuse triangulation.

No known bounds for PSLGs.

Nonobtuse triangulation of PSLGs seems hard, so what if we replace " $\leq 90°$ " with " $\leq \theta < 180°$ "?

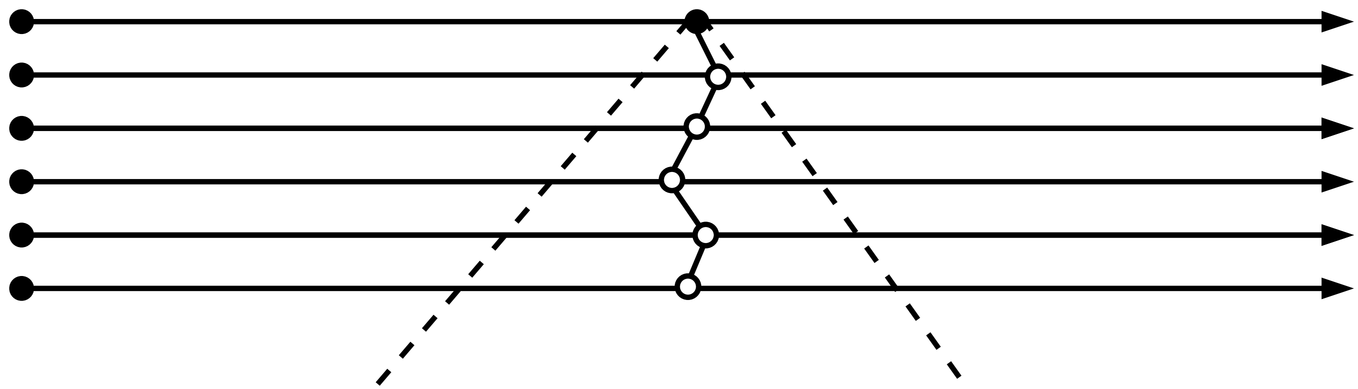
Nonobtuse triangulation of PSLGs seems hard, so what if we replace " $\leq 90°$ " with " $\leq \theta < 180°$ "?

- S. Mitchell (1993): every PSLG has $O(n^2)$ triangulation with all angles $\leq 157.5° = \frac{7}{8}\pi$

Nonobtuse triangulation of PSLGs seems hard, so what if we replace " $\leq 90°$ " with " $\leq \theta < 180°$ "?

- S. Mitchell (1993): every PSLG has $O(n^2)$ triangulation with all angles $\leq 157.5° = \frac{7}{8}\pi$
- Tan (1996): same for angles $\leq 132° = \frac{11}{15}\pi$.

Any bound $< 180^\circ$ sometimes requires n^2 vertices.



Applications of non-obtuse triangulations:

- Discrete maximum principle (Ciarlet, Raviart, 1973)
- Convergence of finite element methods (Vavasis, 1996)
- Fast marching method (Sethian, 1999)
- Meshing space-time (Ungör, Sheffer, 2002)
- Machine learning

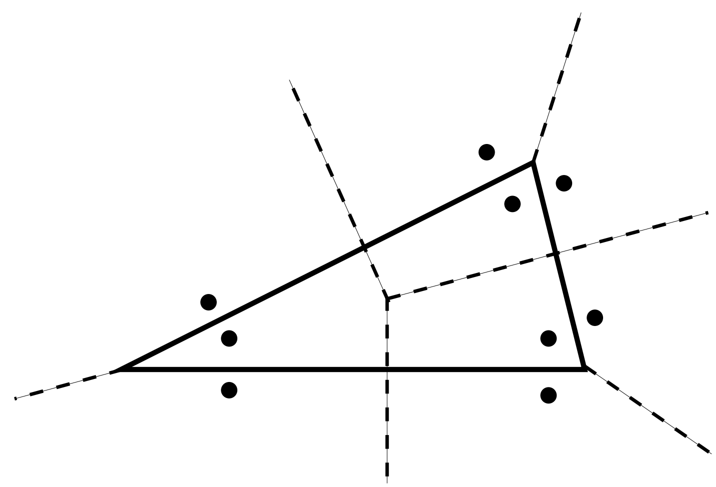
Salzberg, Delcher, Heath, Kasif, 1995, *Best-case results for nearest-neighbor learning*.

Given polygon Γ find point sets I, O so that

$$\mathrm{int}(\Gamma) = \{z : \mathrm{dist}(z, I) < \mathrm{dist}(z, O)\},$$

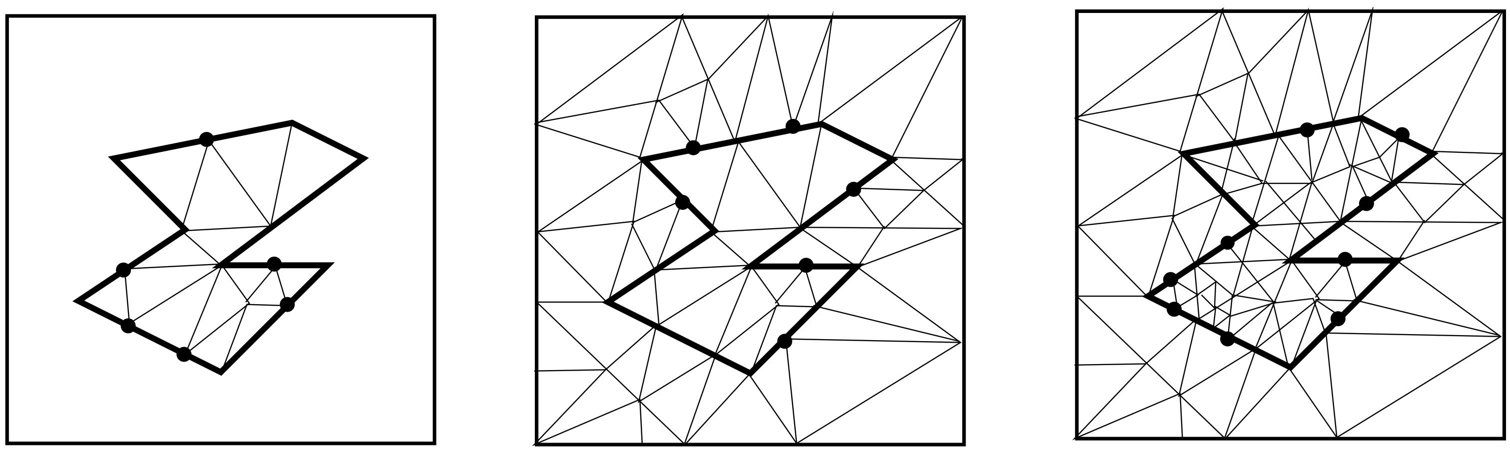
i.e., Voronoi diagram of $I \cup O$ covers Γ.

Easy for nonobtuse triangles.



S-D-H-K reduce nearest neighbor learning to simultaneous nonobtuse triangulation of both sides of Γ.

We can triangulate one side, then the other, but this makes new vertices on polygon. Then retriangulate first side. This creates more vertices. ...



Do a polynomial number of points suffice?

Theorem (B, 2010): Every PSLG has a non-obtuse triangulation with $O(n^{2.5})$ elements.

Theorem (B, 2010): Every PSLG has a triangulation with all angles $\leq 90° + \epsilon$ and $O(n^2/\epsilon^2)$ elements.

Theorem (B, 2010): Every PSLG has a non-obtuse triangulation with $O(n^{2.5})$ elements.

Theorem (B, 2010): Every PSLG has a triangulation with all angles $\leq 90^\circ + \epsilon$ and $O(n^2/\epsilon^2)$ elements.

Theorem (B, 2010): Every PSLG has a quadrilateral mesh with $O(n^2)$ elements, all angles less than 120° and all new angles greater than 60°.

The non-obtuse triangulation problem reduces to:

Theorem: For any PSLG Γ of size n there is set of size $O(n^{2.5})$ whose Gabriel graph covers Γ.

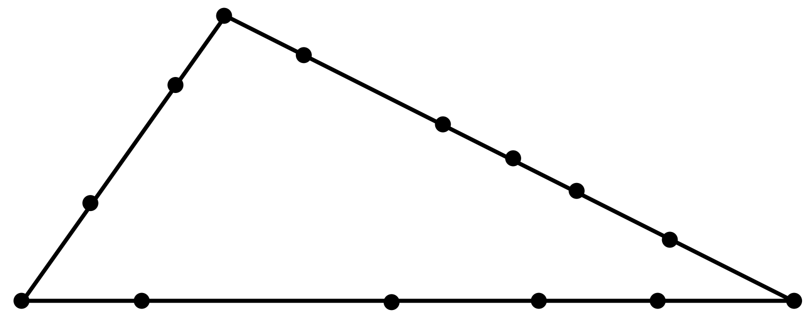
The non-obtuse triangulation problem reduces to:

Theorem: For any PSLG Γ of size n there is set of size $O(n^{2.5})$ whose Gabriel graph covers Γ.

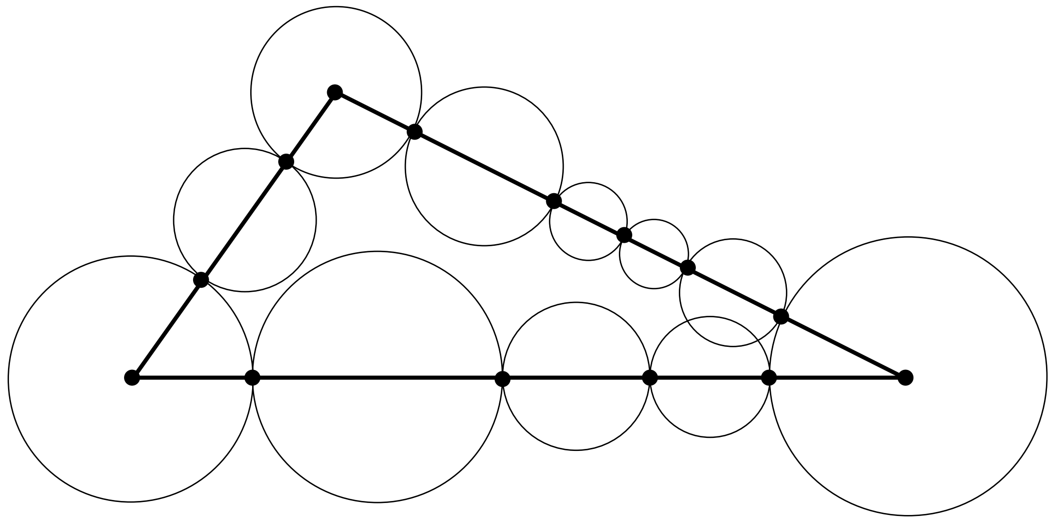
Suffices to consider Γ a triangulation.

Follows ideas of Bern, Mitchell and Ruppert (1994).

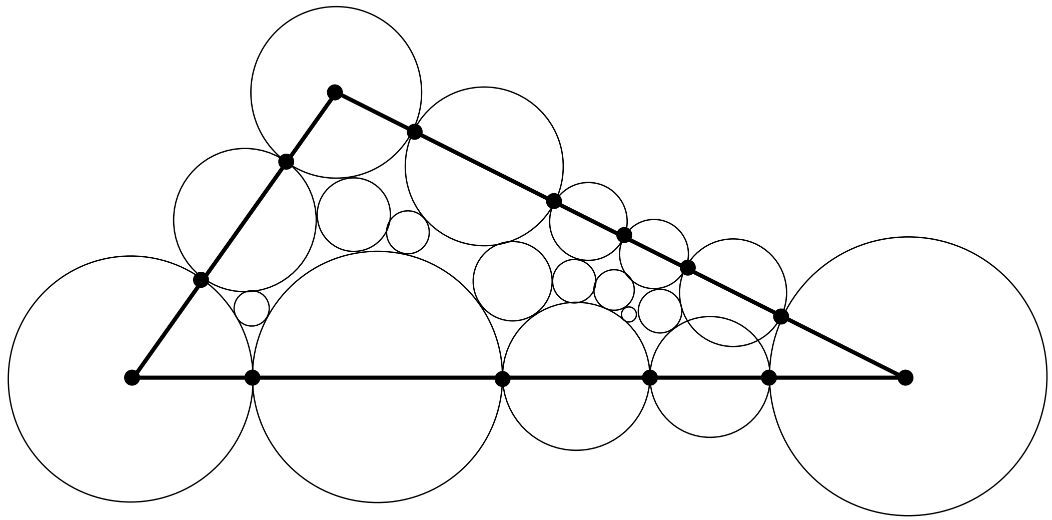
Proof of reduction:



Proof of reduction:

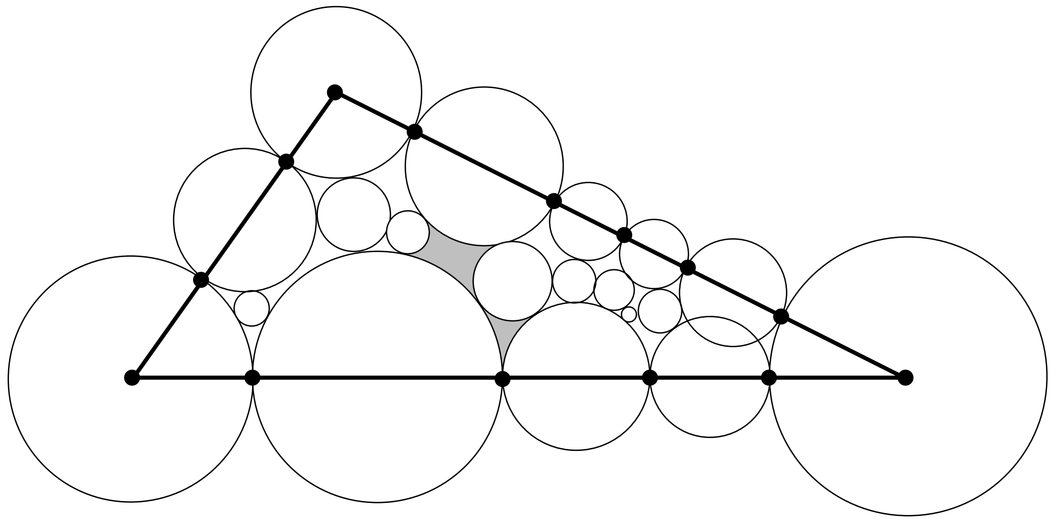


Proof of reduction:



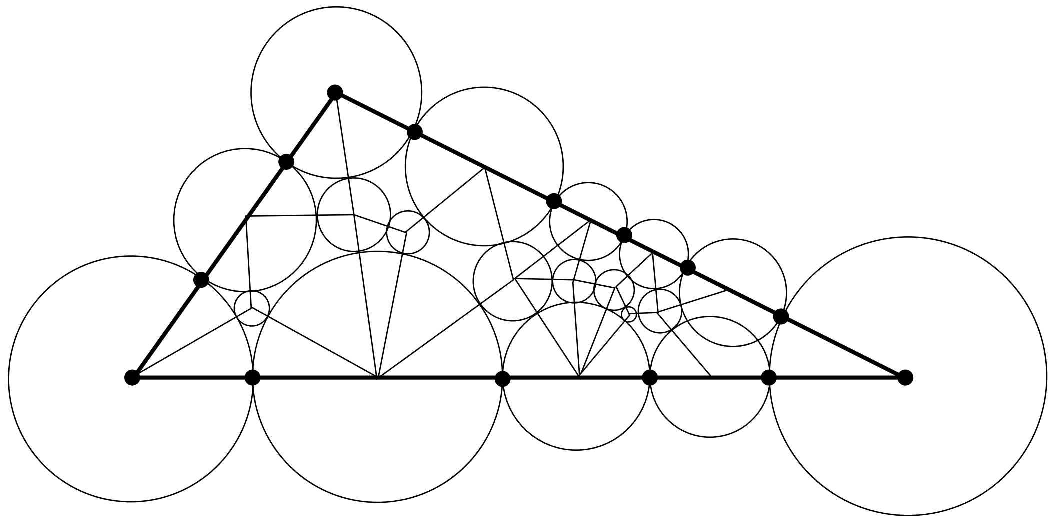
• Pack interior with disjoint disks so only 3-sided and 4-sided regions remain.

Proof of reduction:



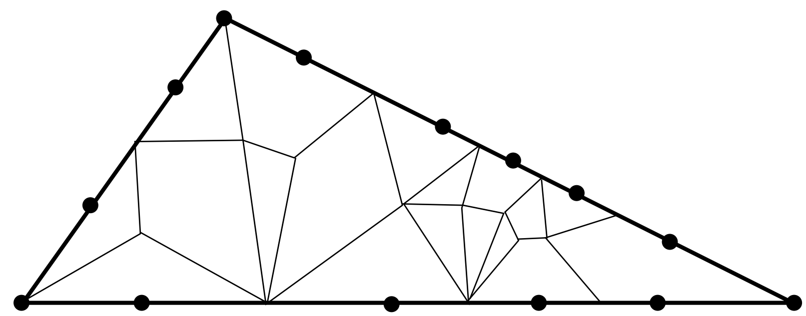
- Pack interior with disjoint disks so only 3-sided and 4-sided regions remain.

Proof of reduction:



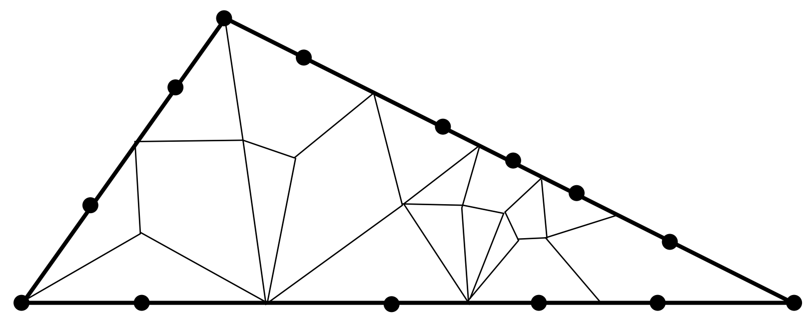
- Pack interior with disjoint disks so only 3-sided and 4-sided regions remain.
- Connect centers.

Proof of reduction:



- Pack interior with disjoint disks so only 3-sided and 4-sided regions remain.
- Connect centers.
- Divides triangle into triangles and quadrilaterals.

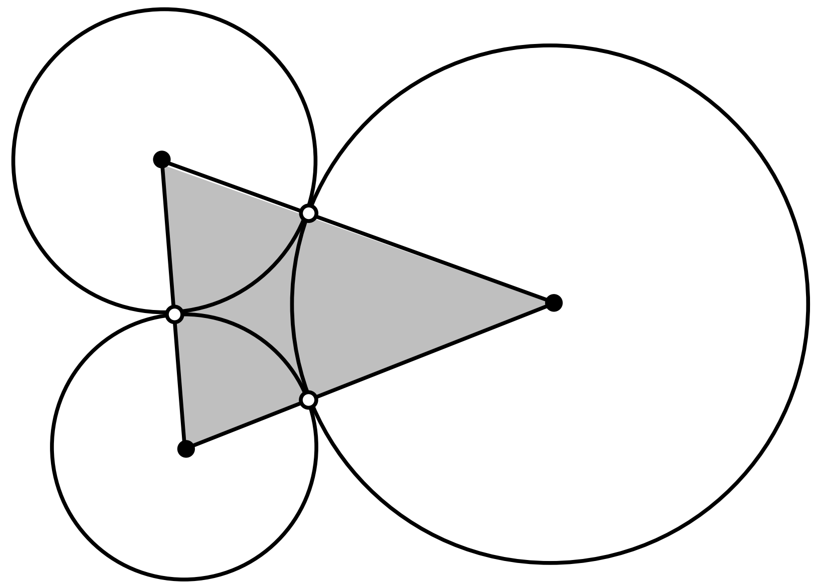
Proof of reduction:



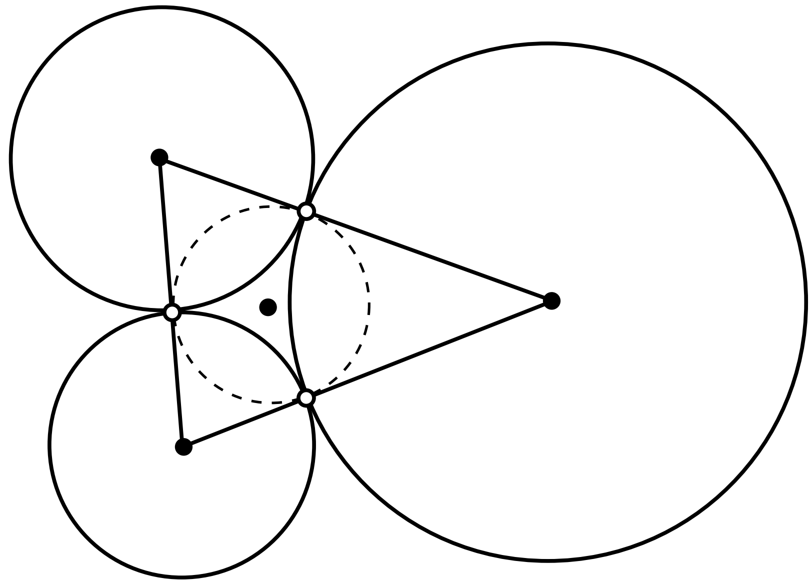
- Pack interior with disjoint disks so only 3-sided and 4-sided regions remain.
- Connect centers.
- Divides triangle into triangles and quadrilaterals.

We want to nonobtusely triangulate each region without adding new vertices along boundary. Several cases.

First case: decompose 3-region into right triangles

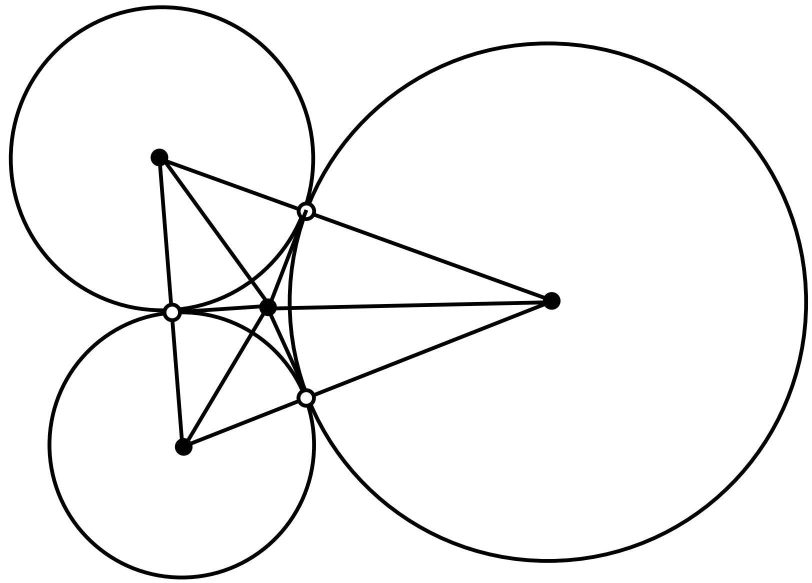


First case: decompose 3-region into right triangles



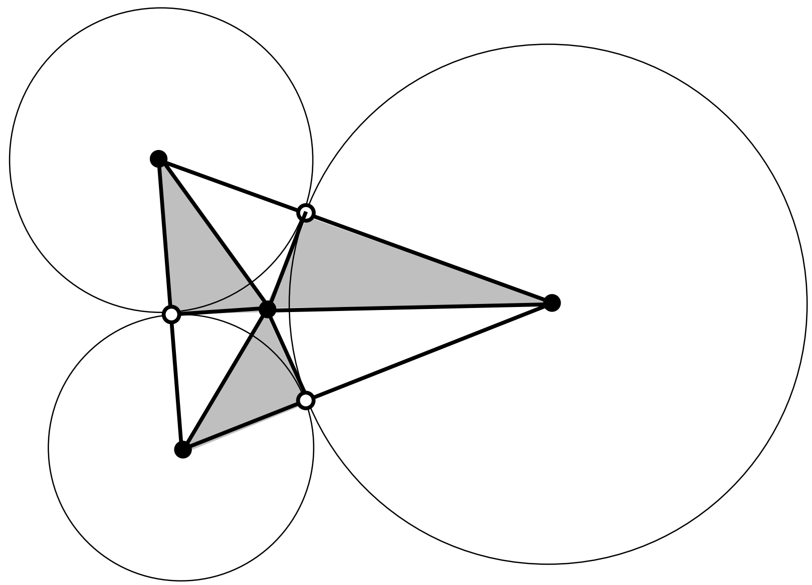
- Add center of circle through the three tangent points.

First case: decompose 3-region into right triangles



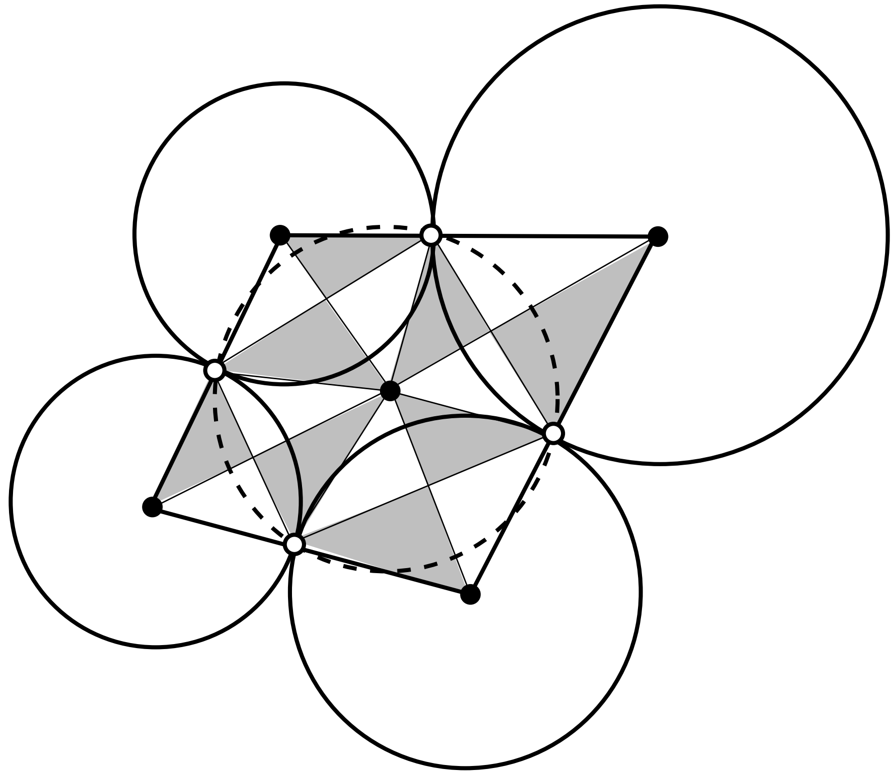
- Add center of circle through the three tangent points.
- Connect center to tangent points and centers of circles

First case: decompose 3-region into right triangles



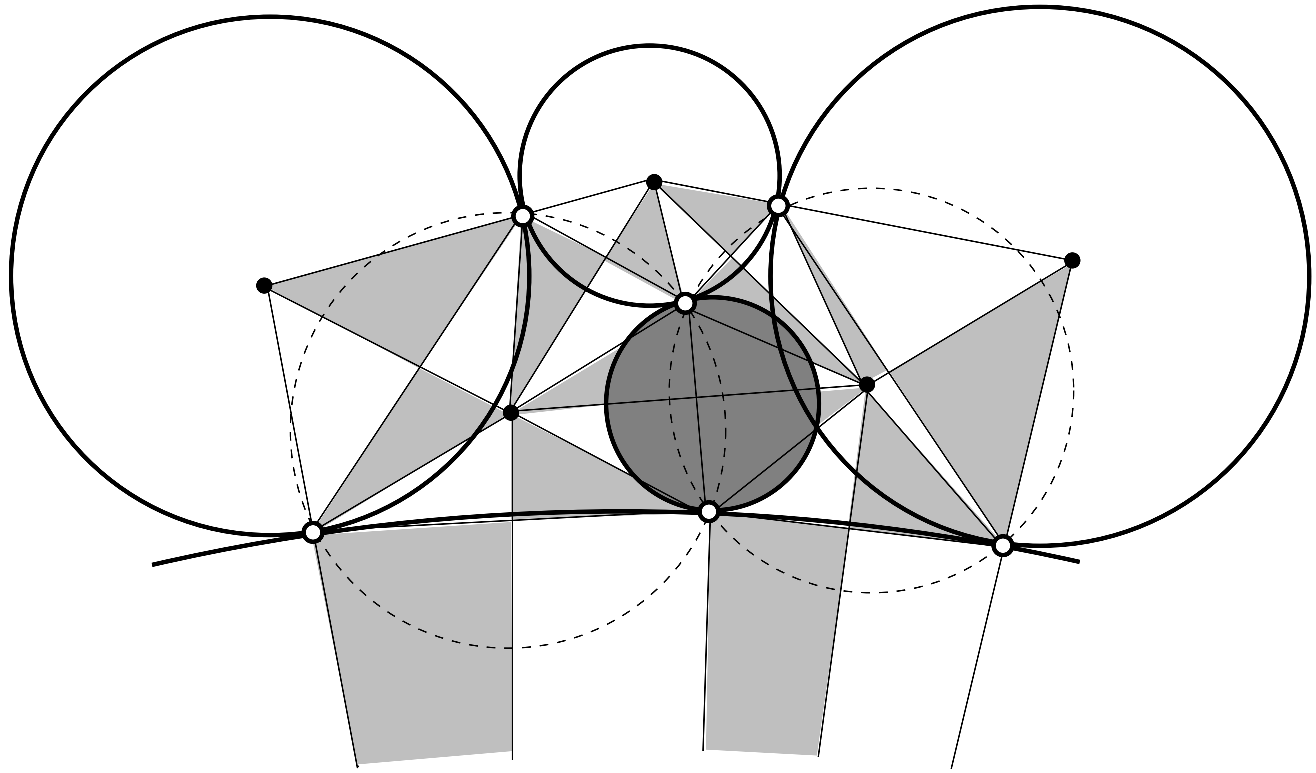
- Add center of circle through the three tangent points.
- Connect center to tangent points and centers of circles

The 4-regions are similar (but several cases arise).



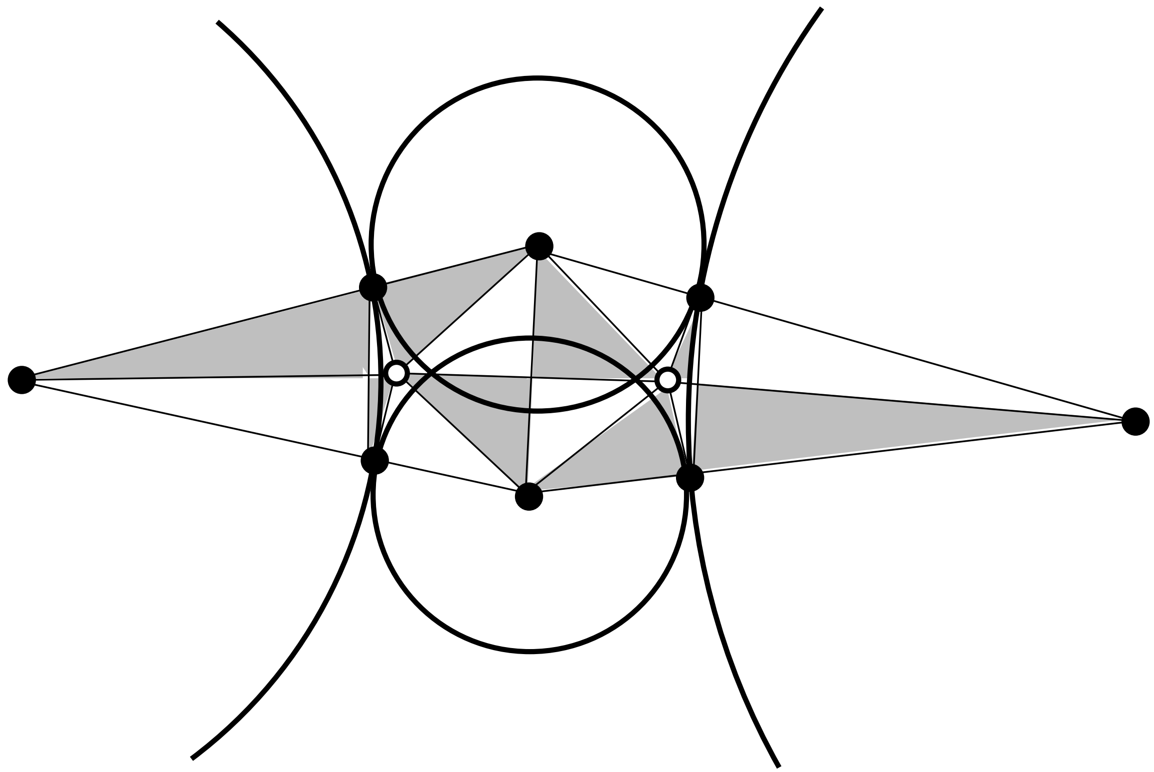
So Gabriel covering implies a nonobtuse triangulation.

The 4-regions are similar (but several cases arise).



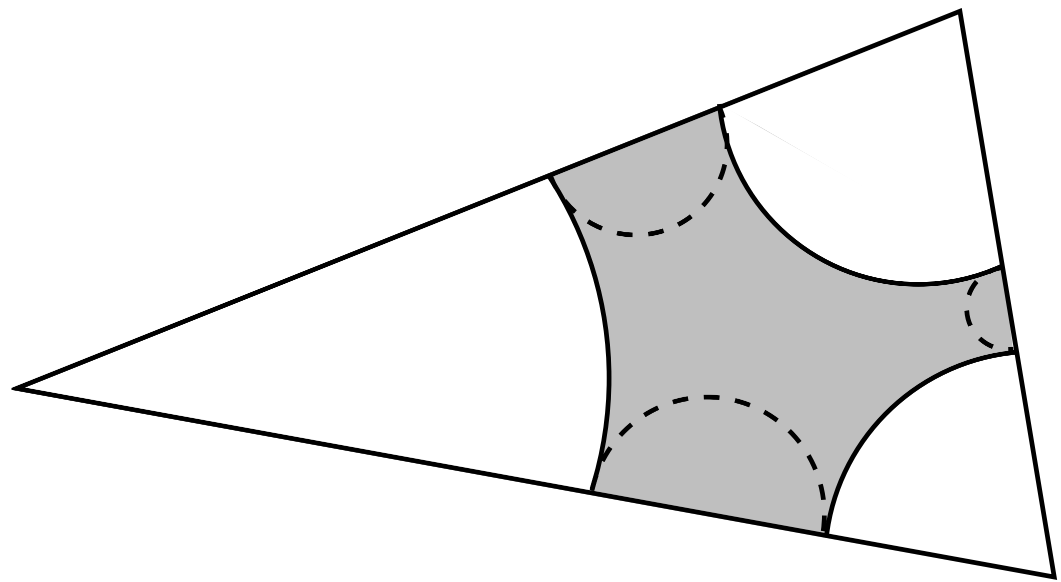
So Gabriel covering implies a nonobtuse triangulation.

The 4-regions are similar (but several cases arise).



So Gabriel covering implies a nonobtuse triangulation.

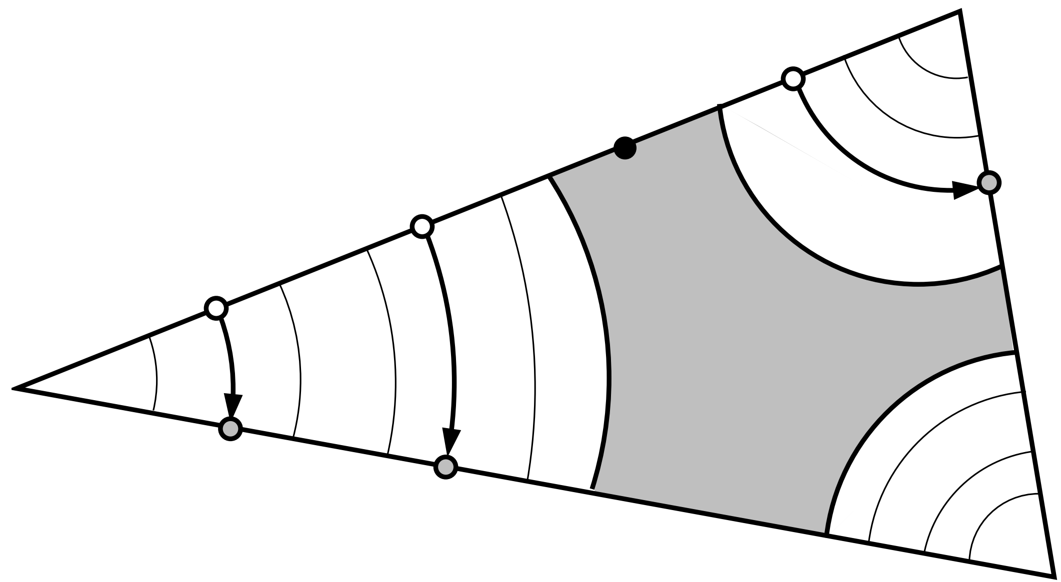
Find Gabriel cover: first partition triangle



Thin parts = corners, Thick part = central region

Thick sides are diameters of disjoint disks.

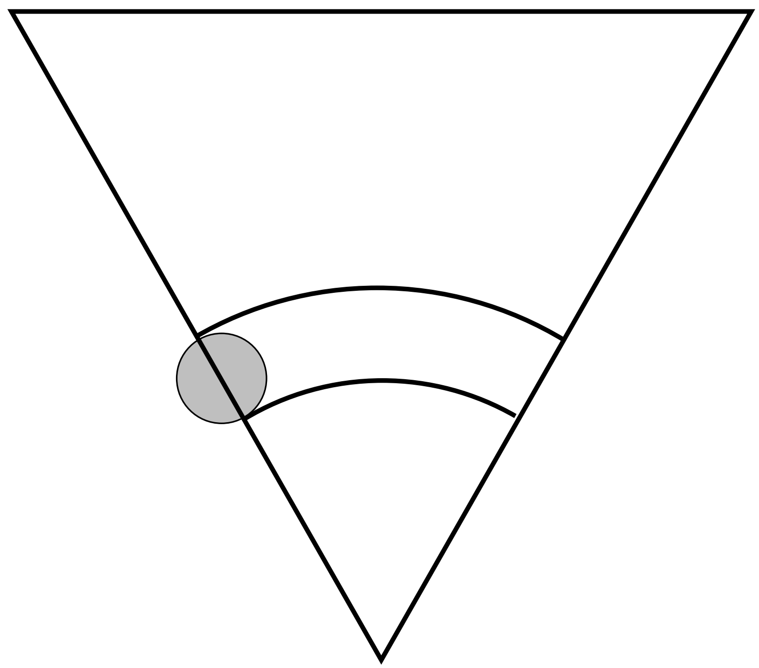
Find Gabriel cover: first partition triangle

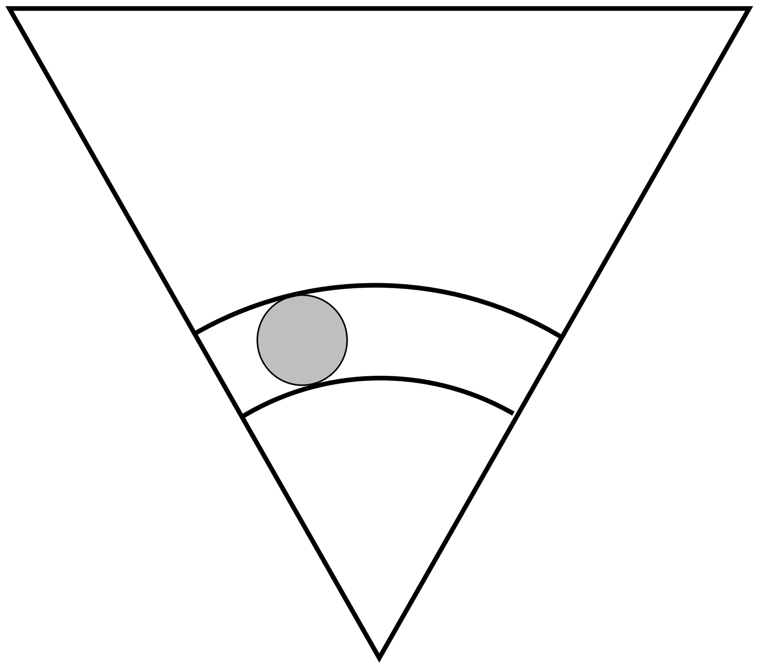


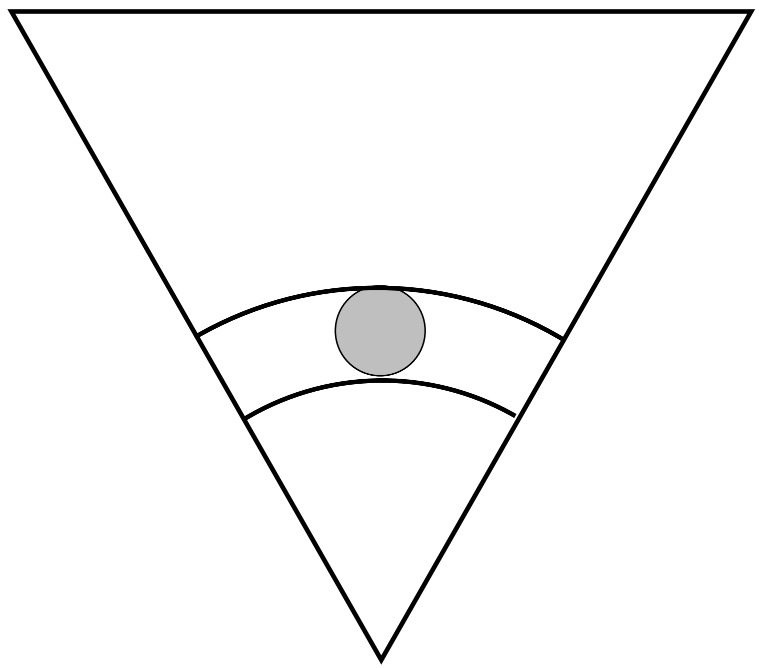
Thin parts foliated by arcs concentric with vertices.

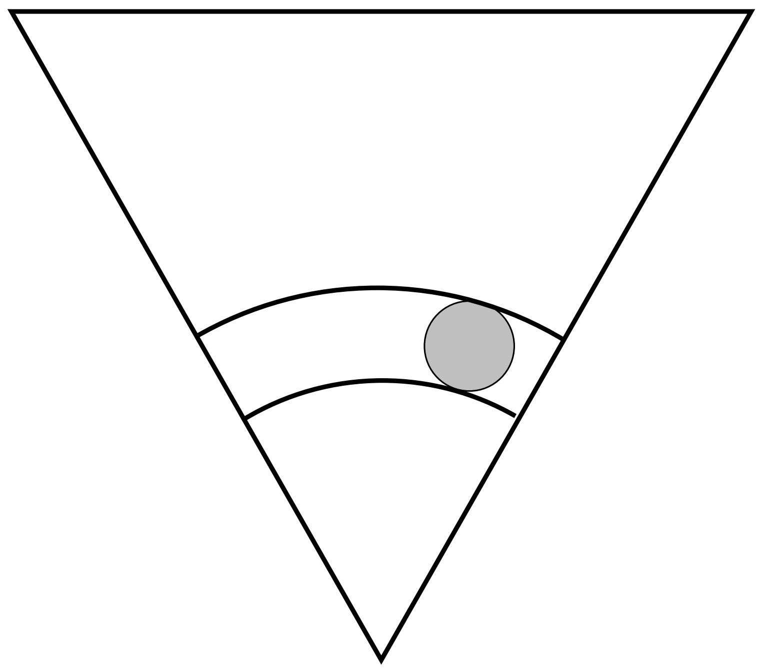
These arcs form tubes of fixed width.

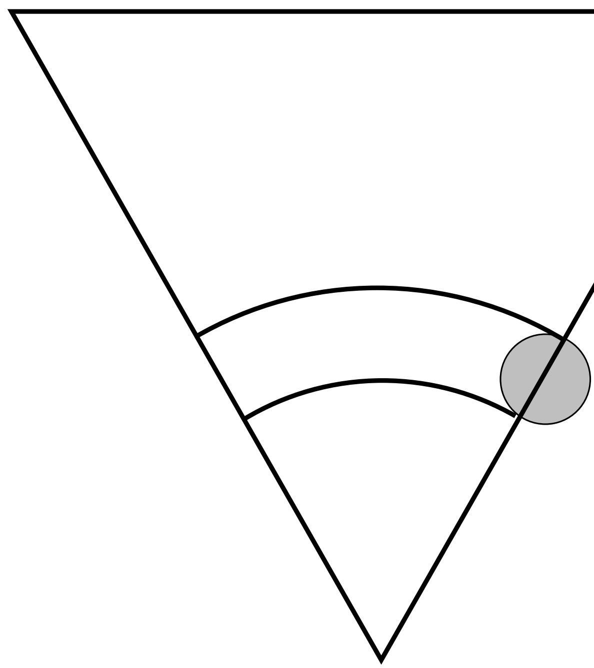
The tube is swept out by a disk of fixed size.

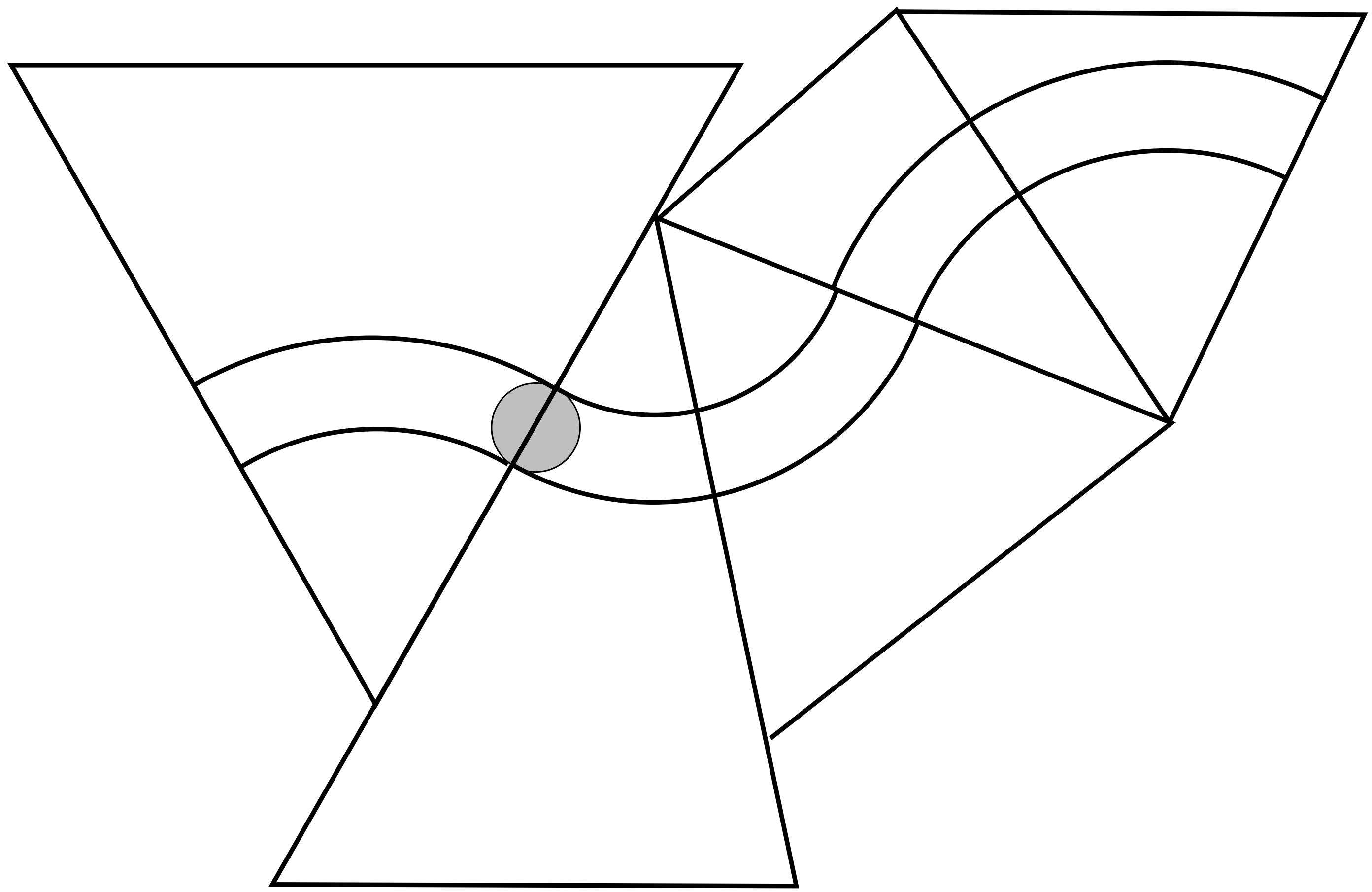


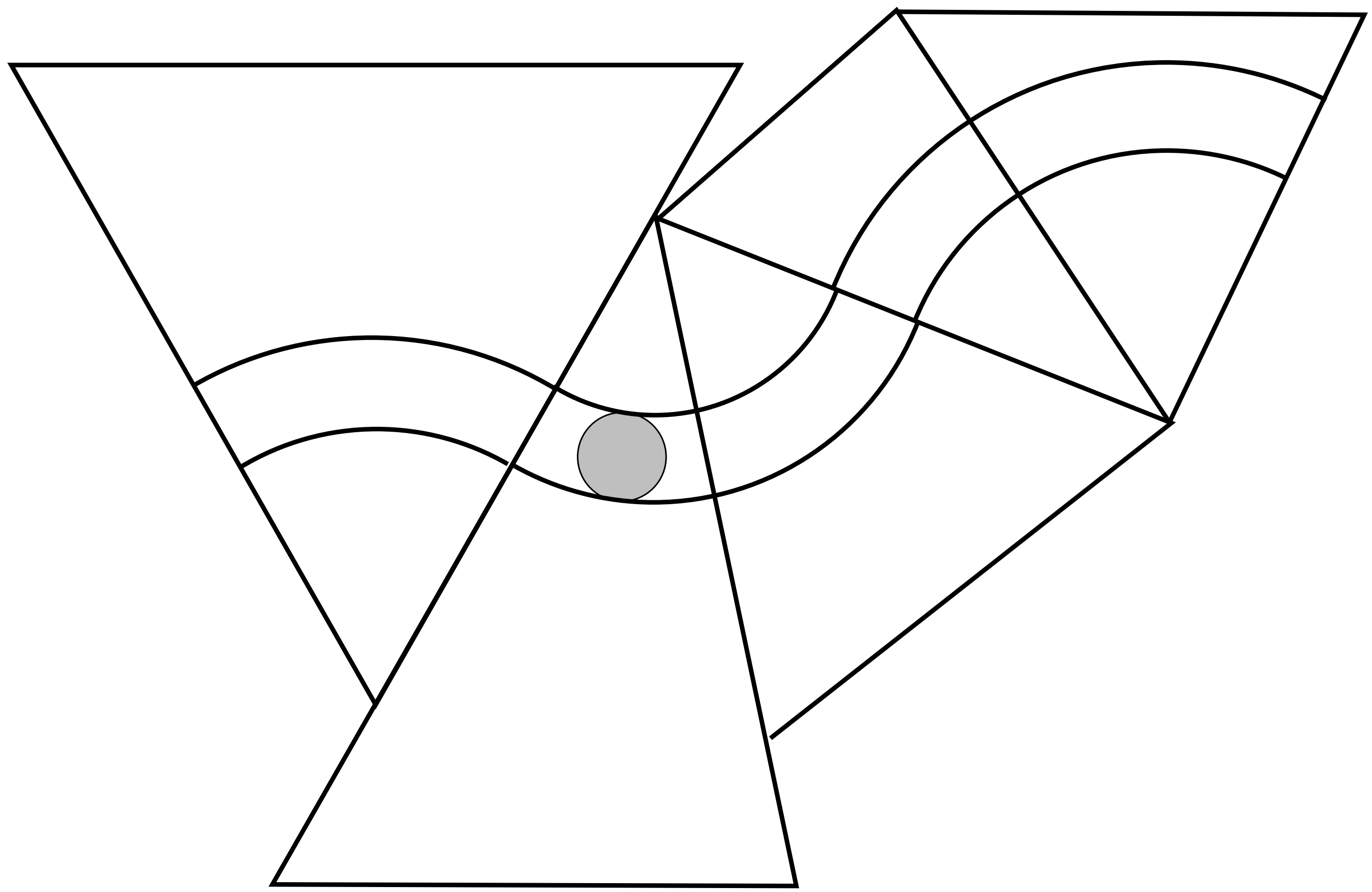


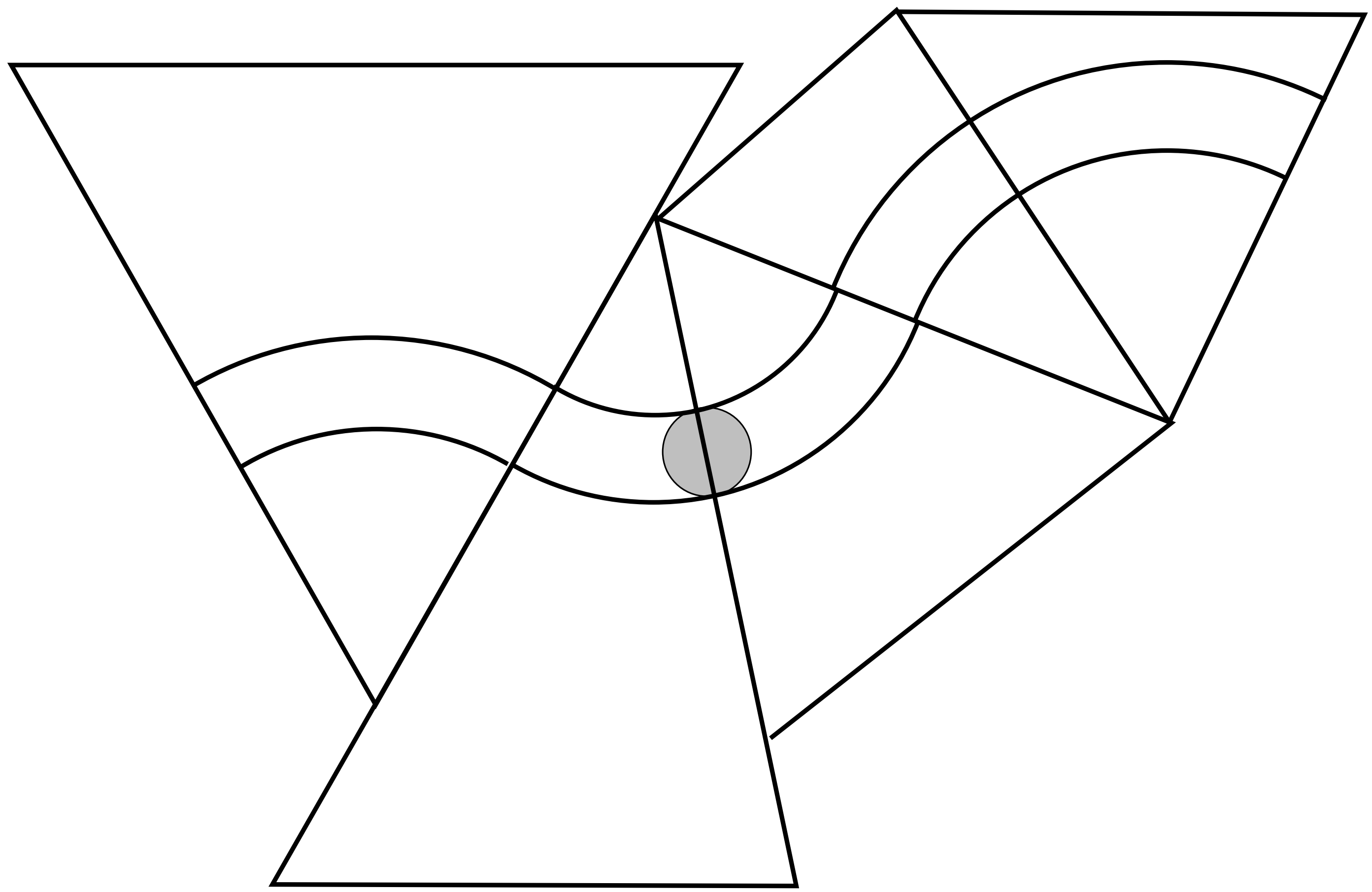


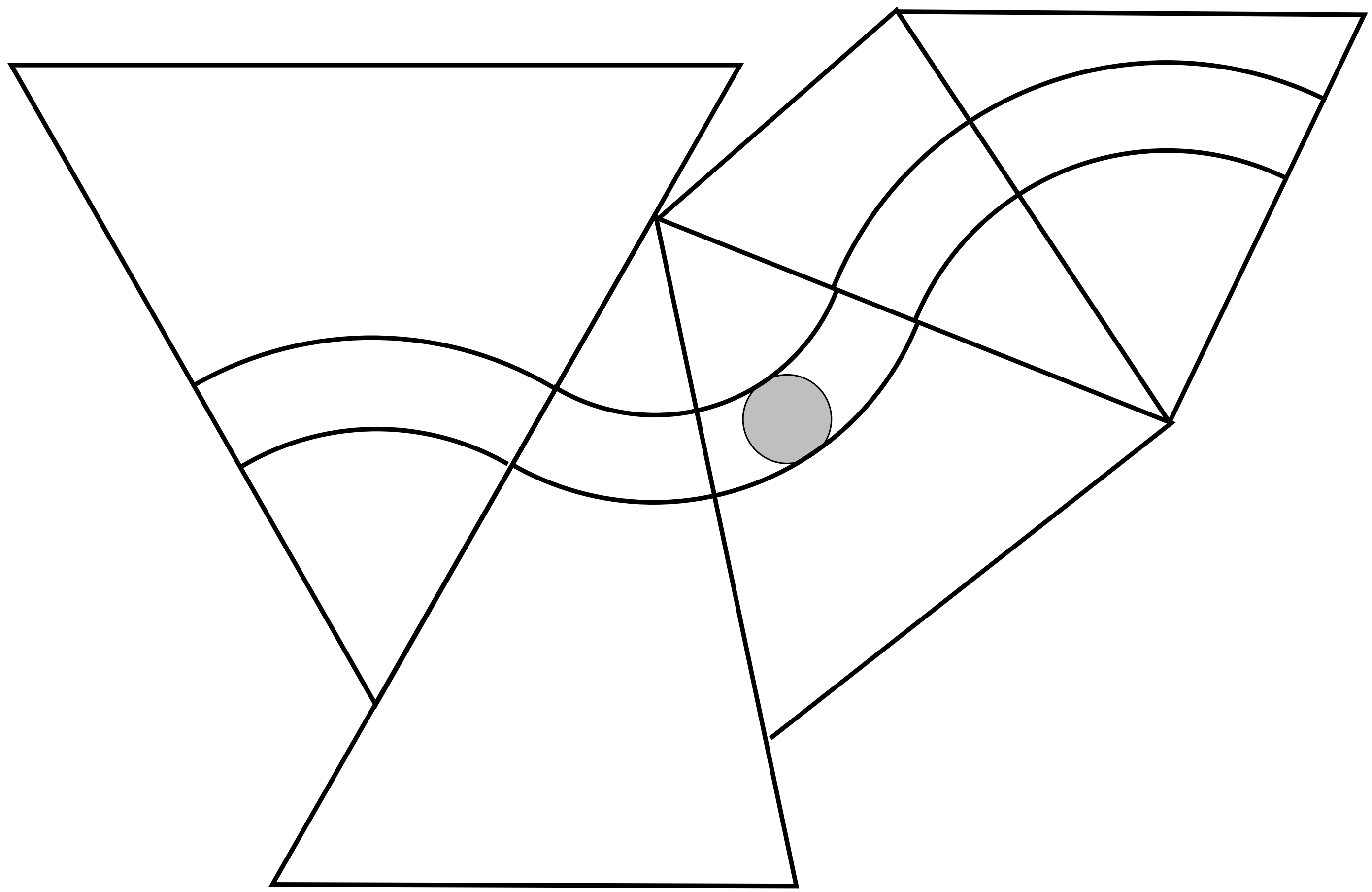


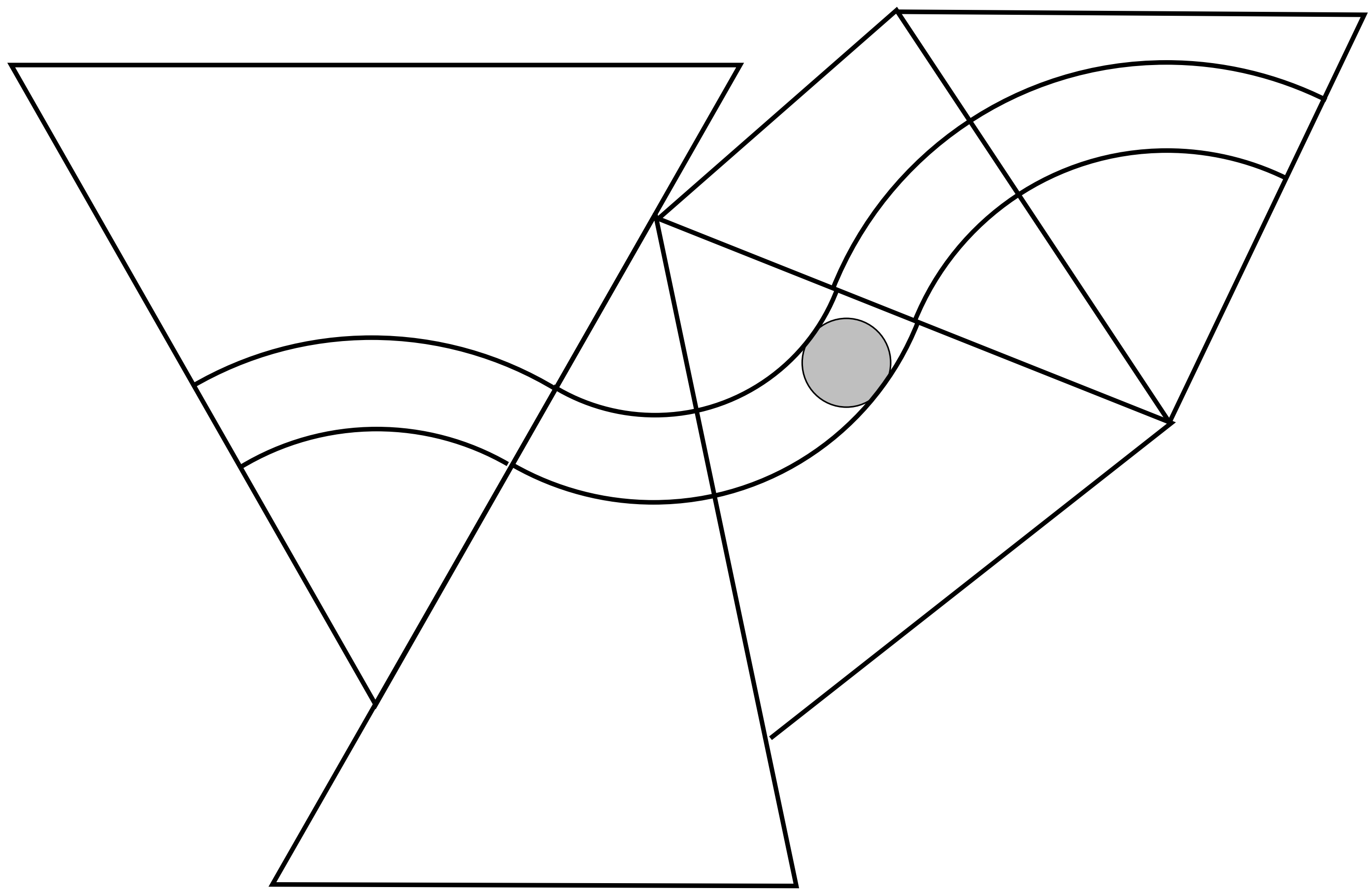


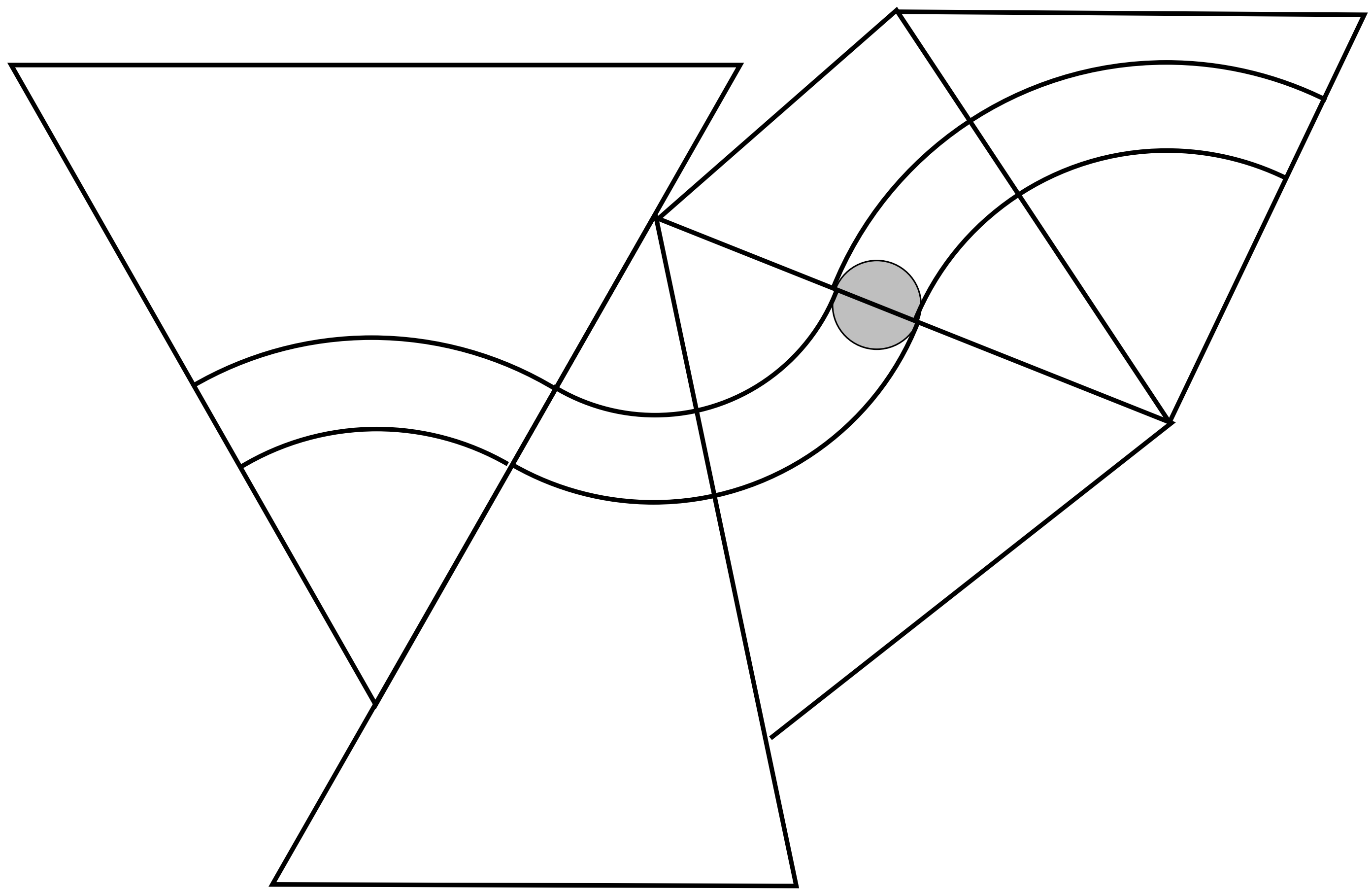


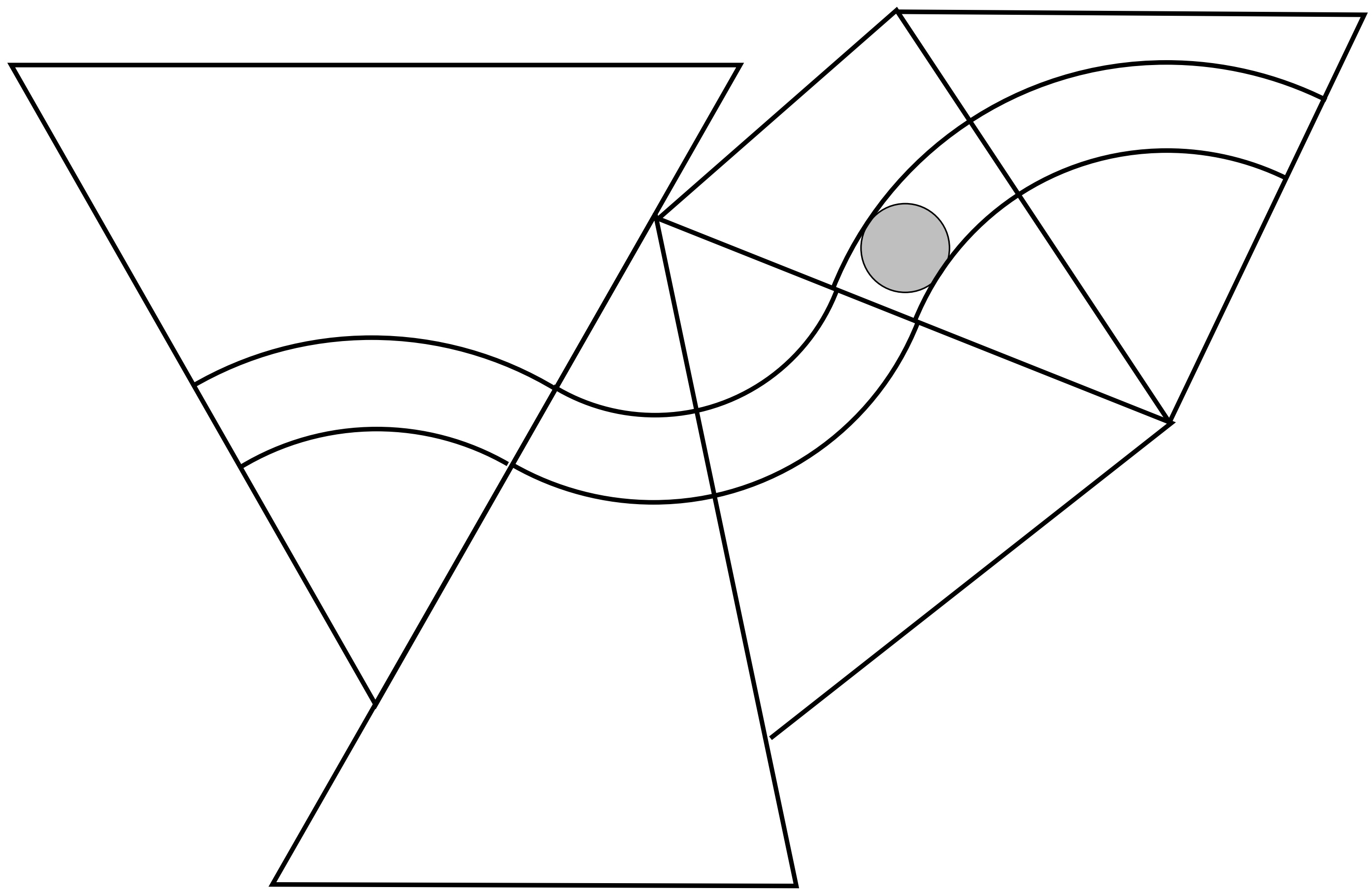


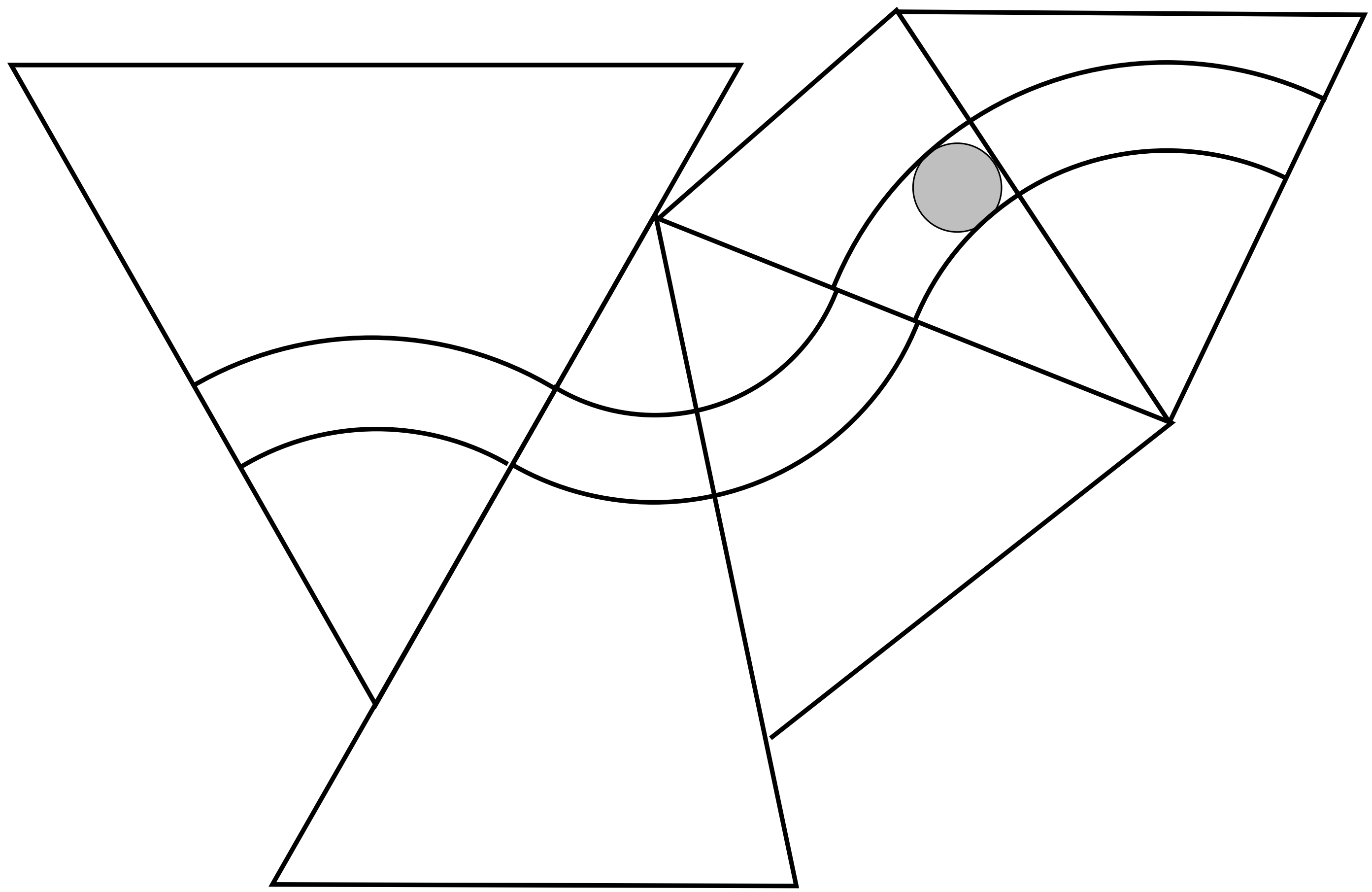


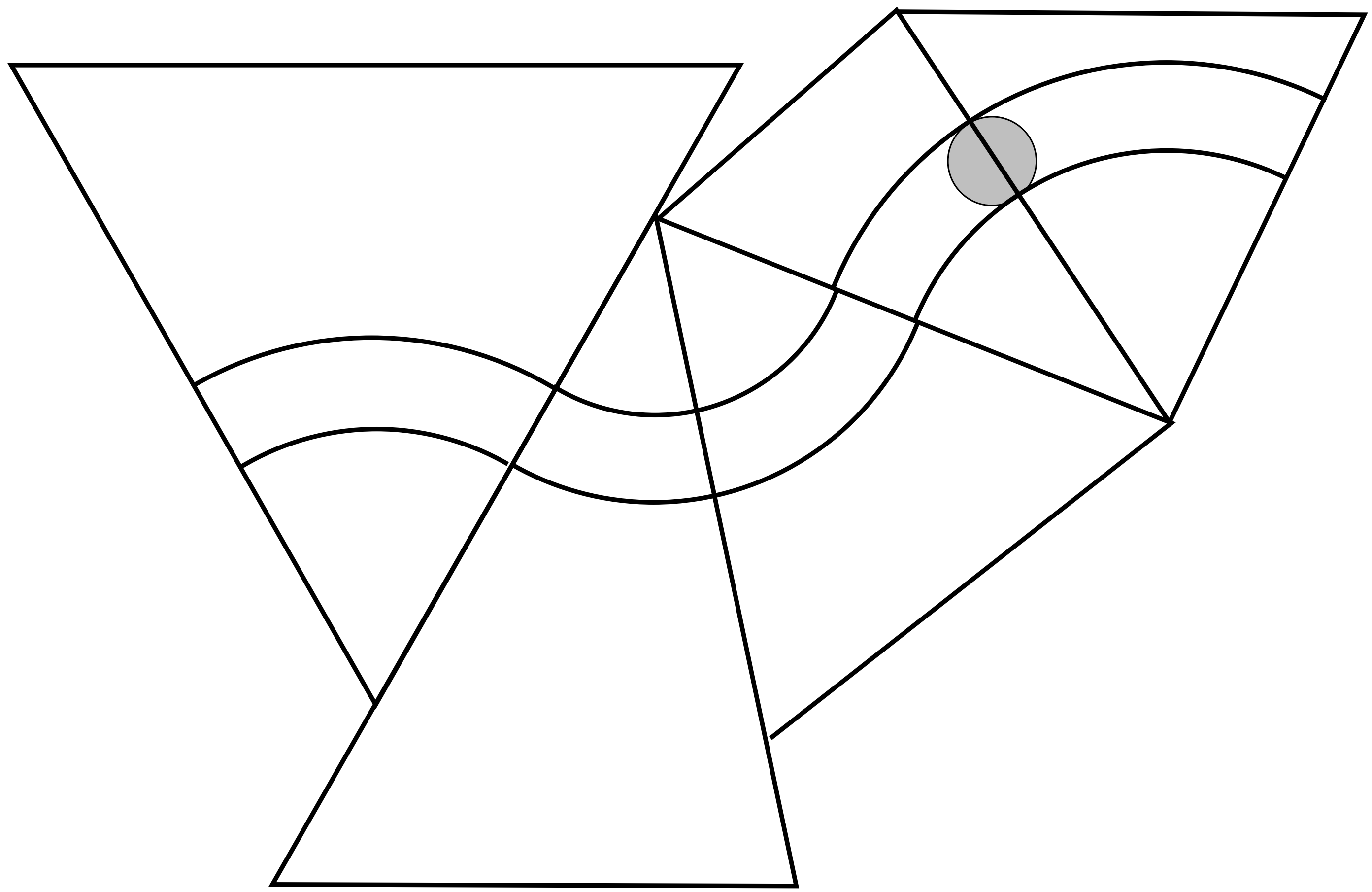


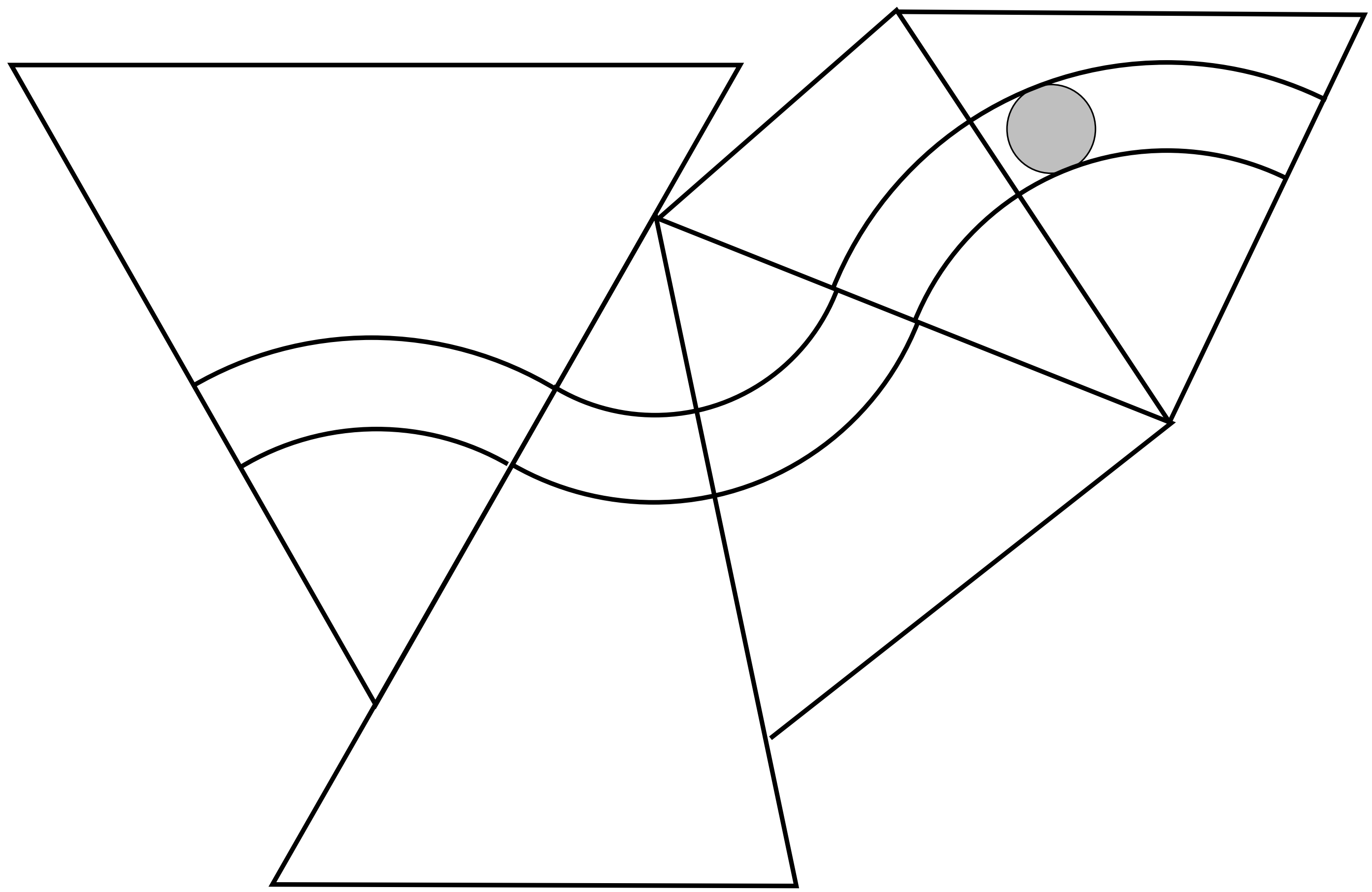


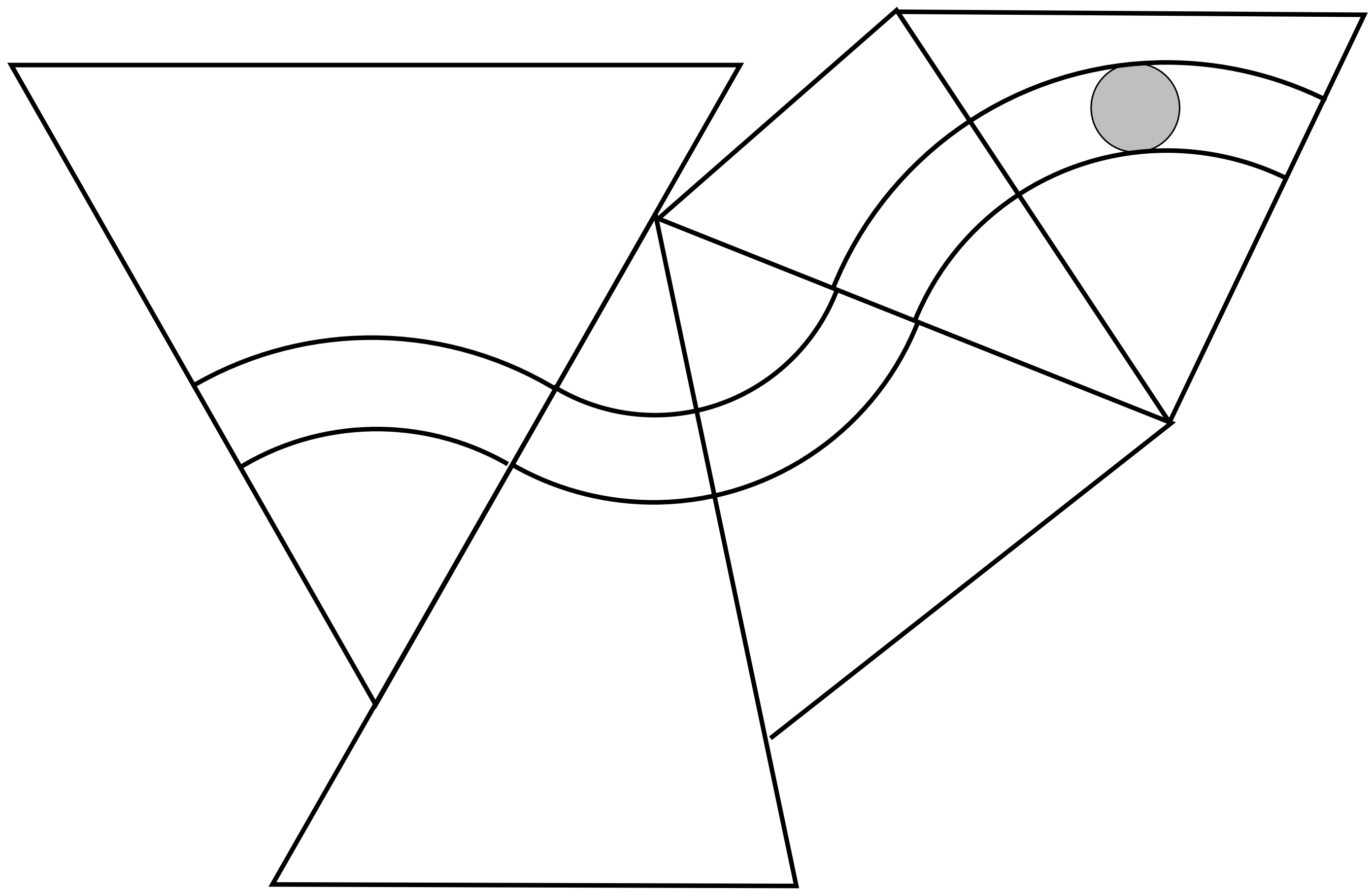


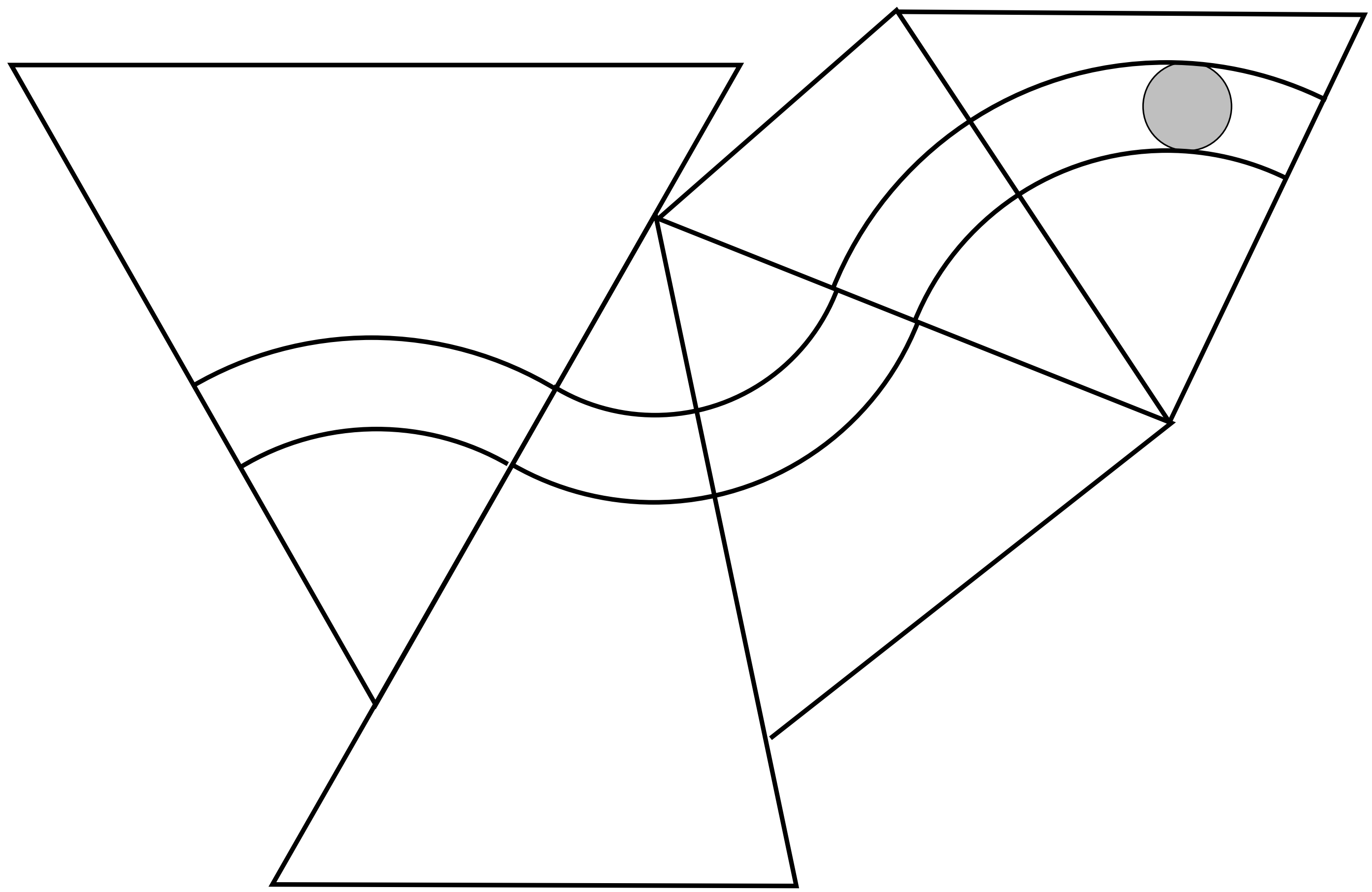


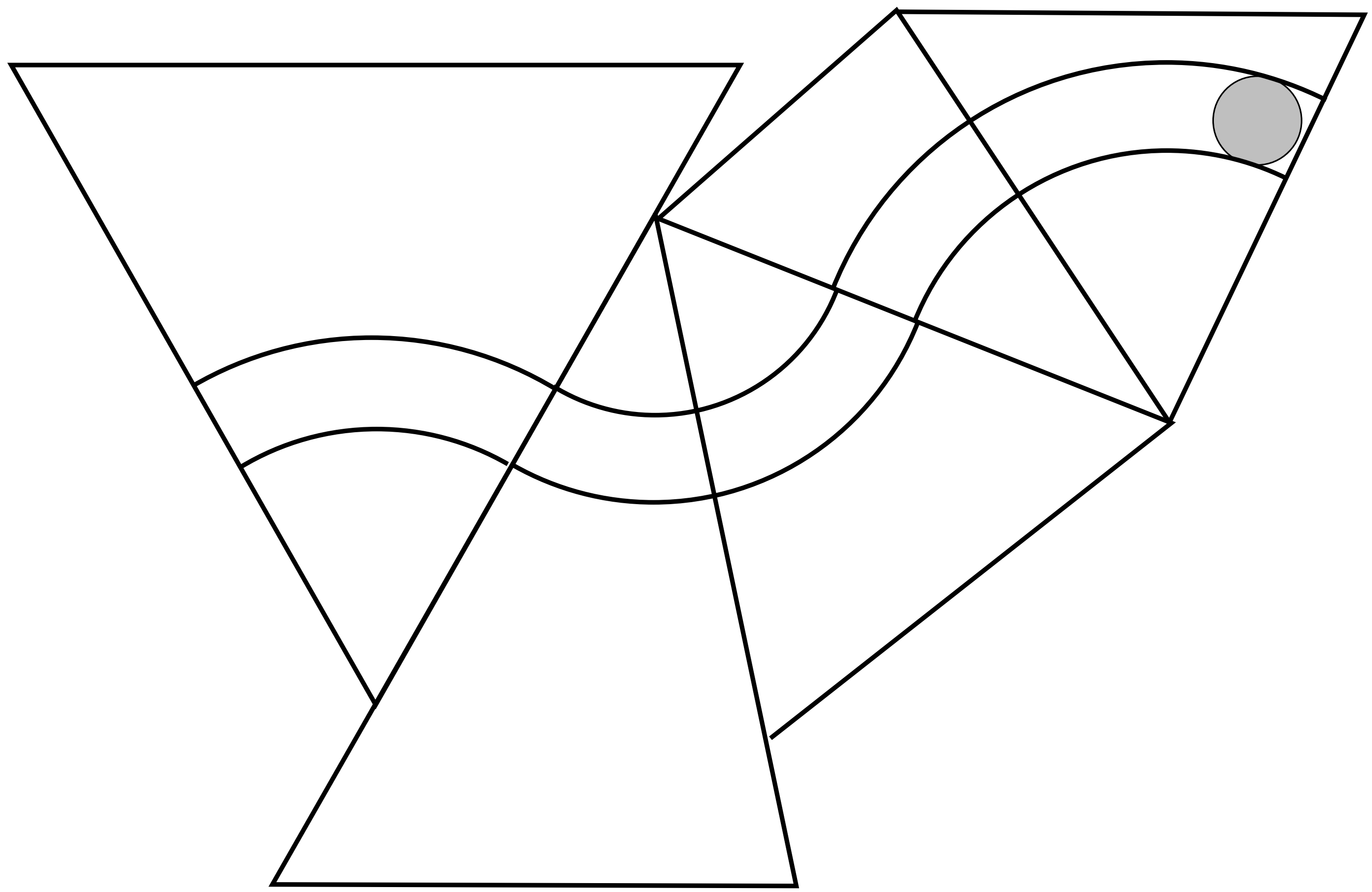


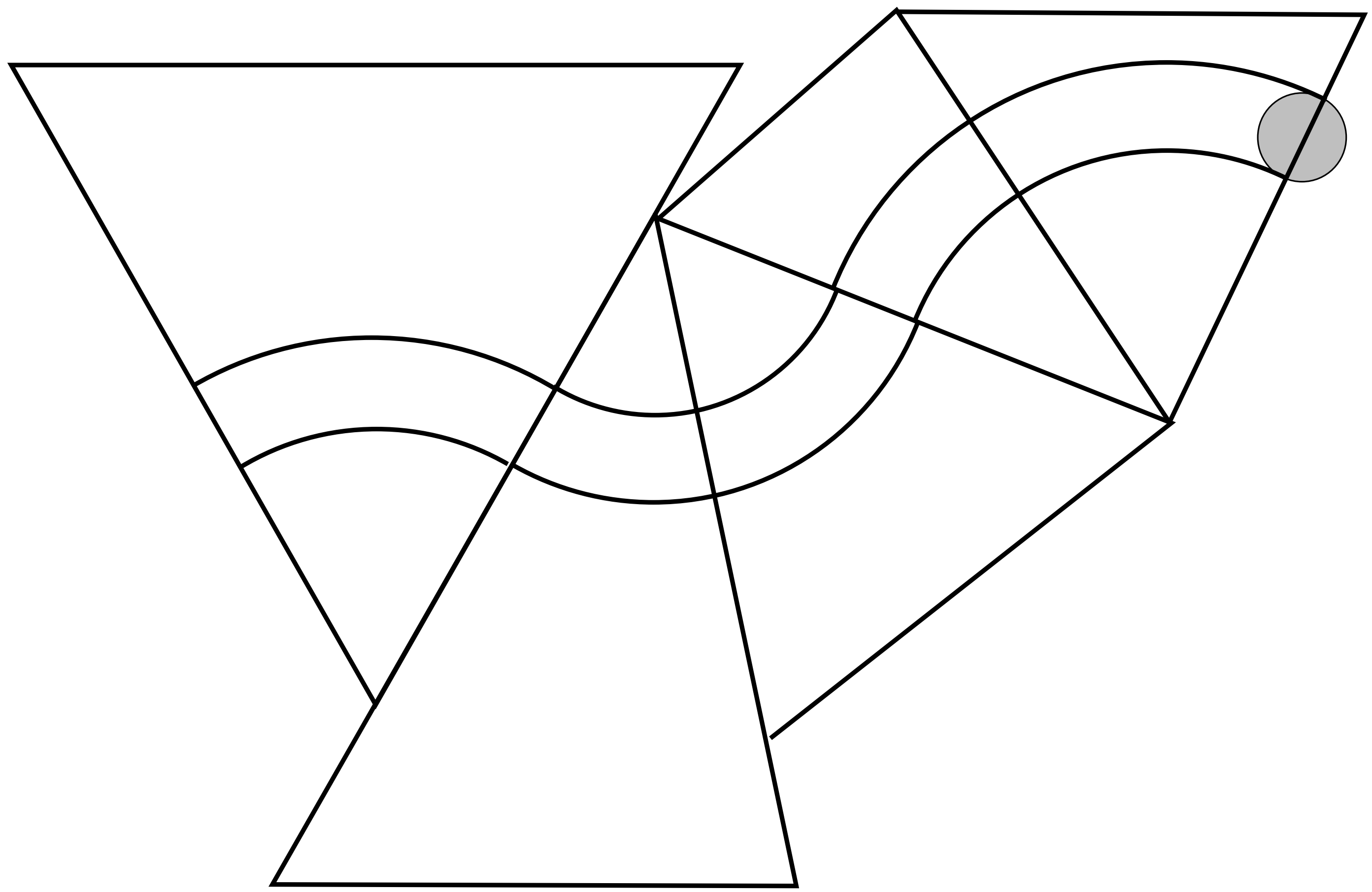


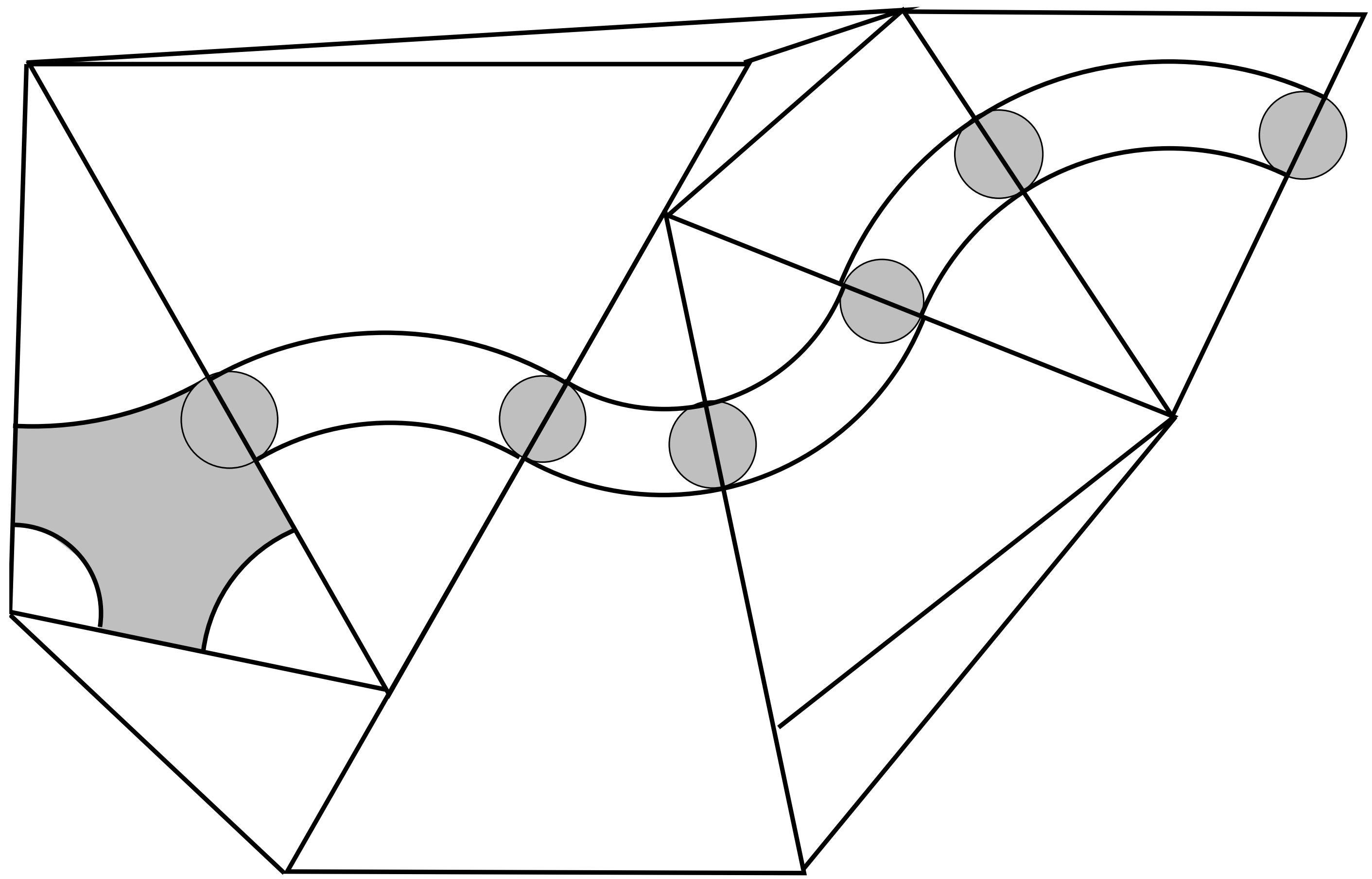






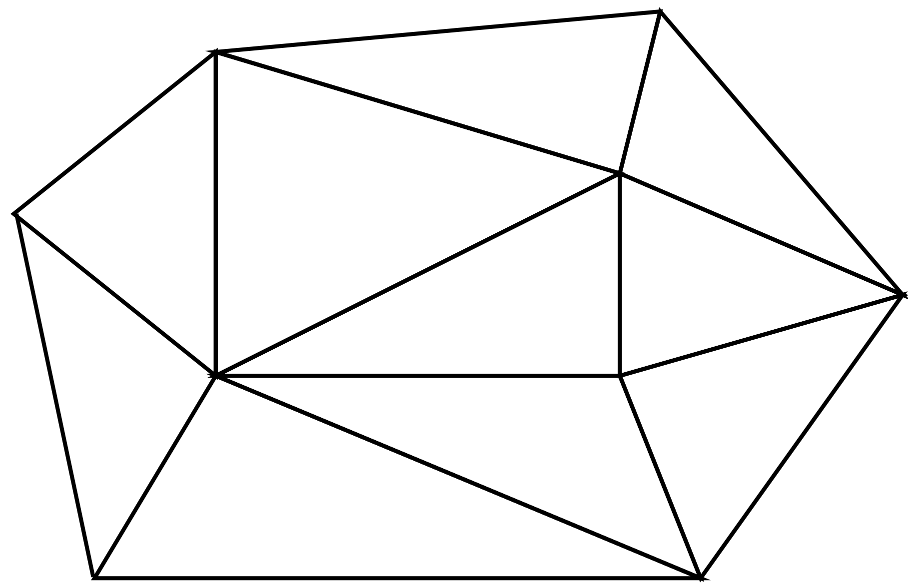




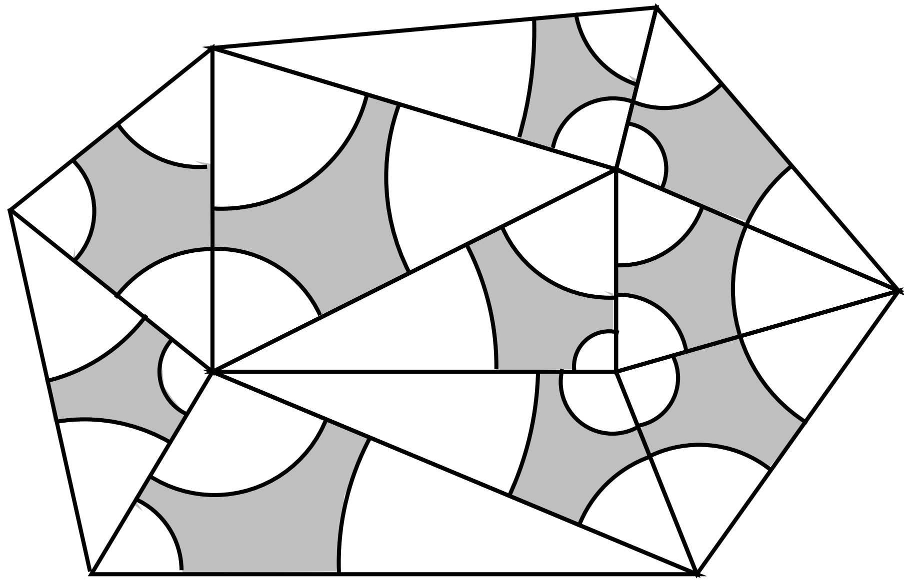


Intersection of tube and triangle edge is a Gabriel edge.

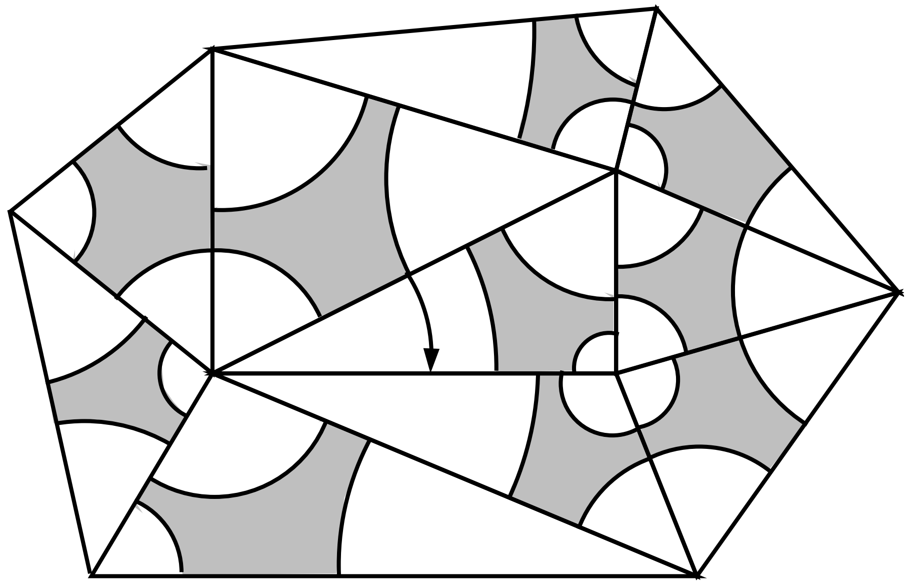
Disk lies inside tube or thick part or outside convex hull.



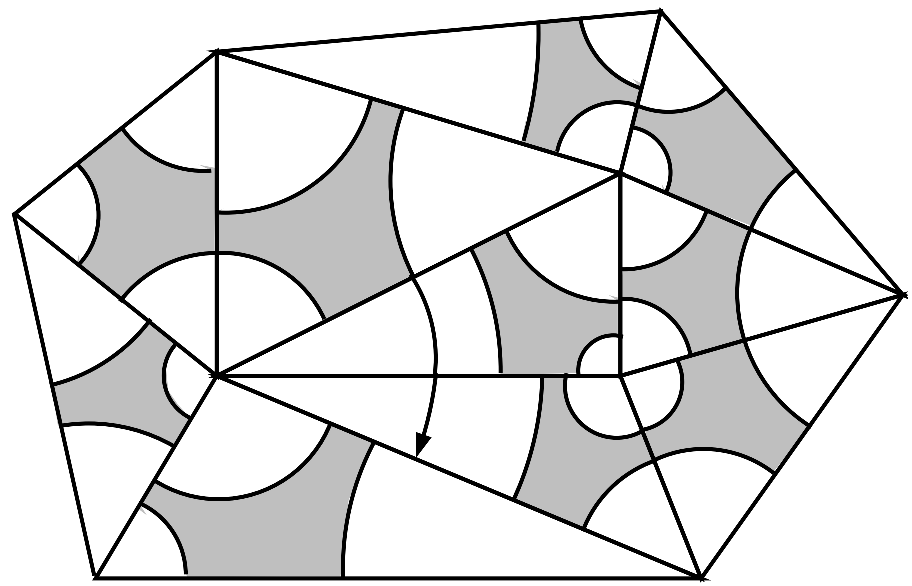
- Start with any triangulation.



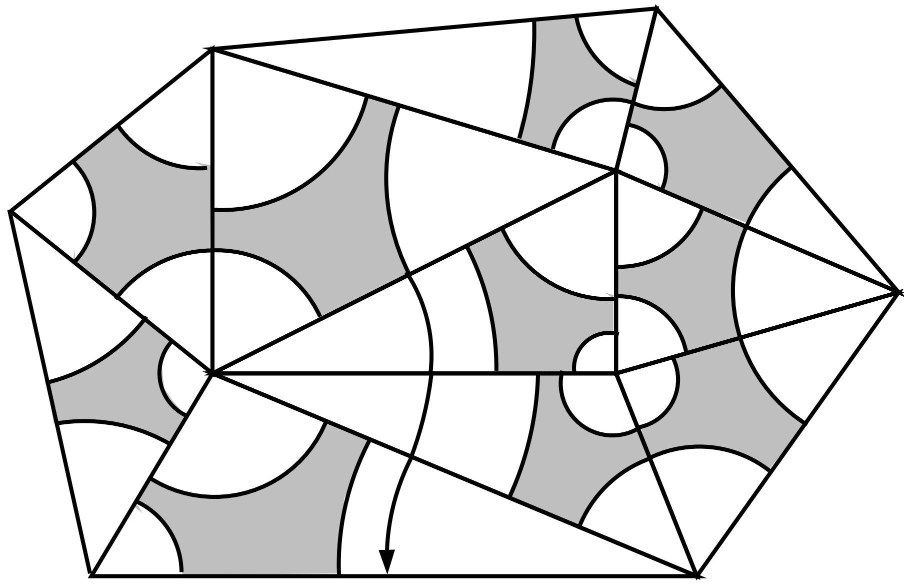
- Start with any triangulation.
- Make thick/thin parts.



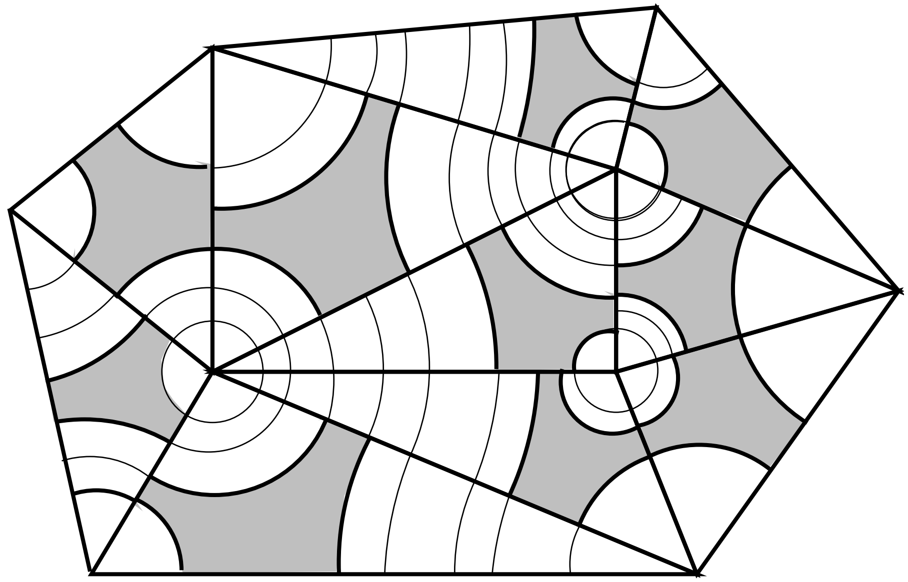
- Start with any triangulation.
- Make thick/thin parts.
- Propagate vertices until they leave thin parts.



- Start with any triangulation.
- Make thick/thin parts.
- Propagate vertices until they leave thin parts.



- Start with any triangulation.
- Make thick/thin parts.
- Propagate vertices until they leave thin parts.

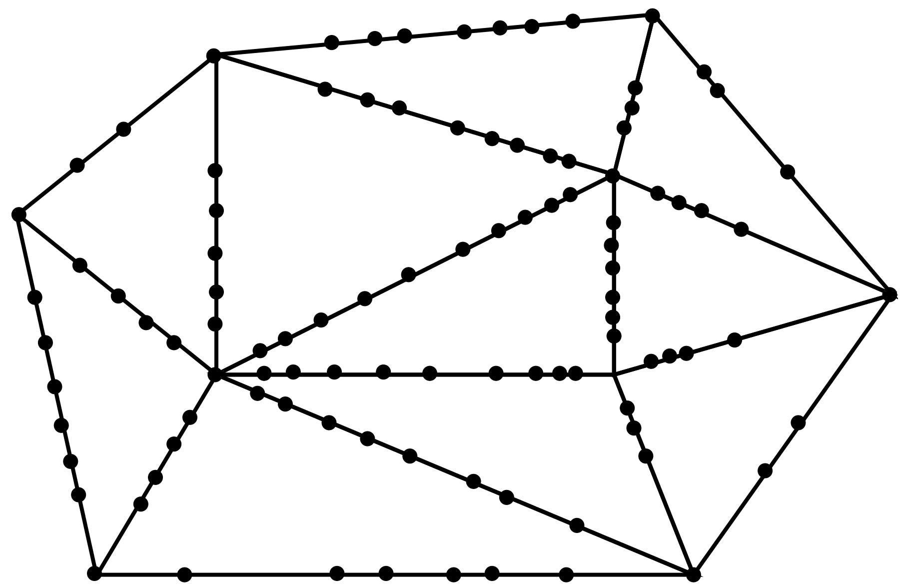


- Start with any triangulation.
- Make thick/thin parts.
- Propagate vertices until they leave thin parts.

Gabriel graph of intersections covers triangulation.

Earlier argument gives nonobtuse triangulation.

But no bound on number of triangles.

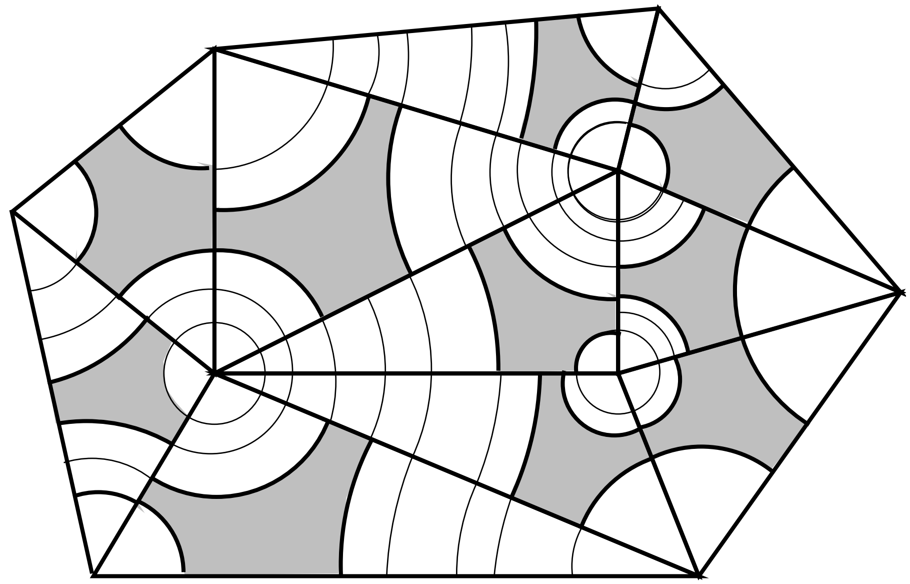


- Start with any triangulation.
- Make thick/thin parts.
- Propagate vertices until they leave thin parts.

Gabriel graph of intersections covers triangulation.

Earlier argument gives nonobtuse triangulation.

But no bound on number of triangles.

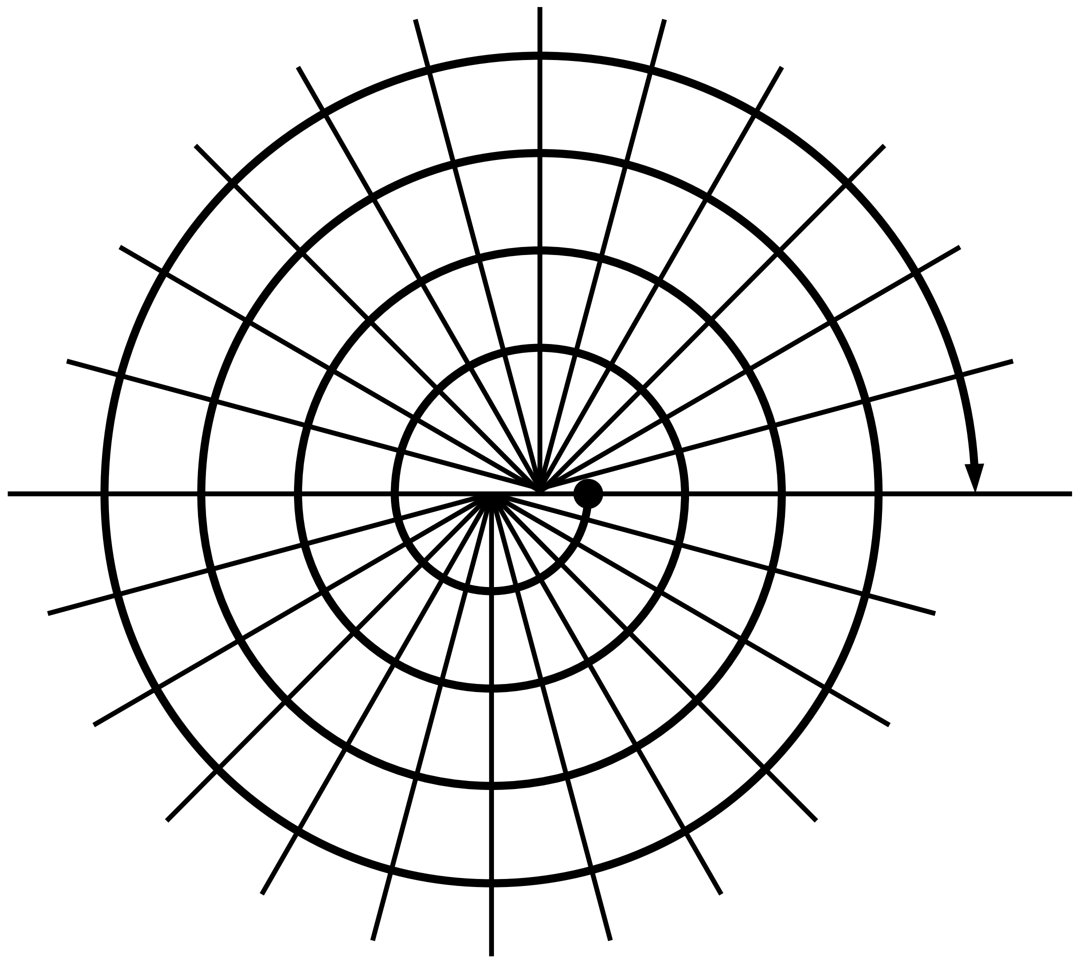


If paths never revisit a triangle, $O(n^2)$ points created.

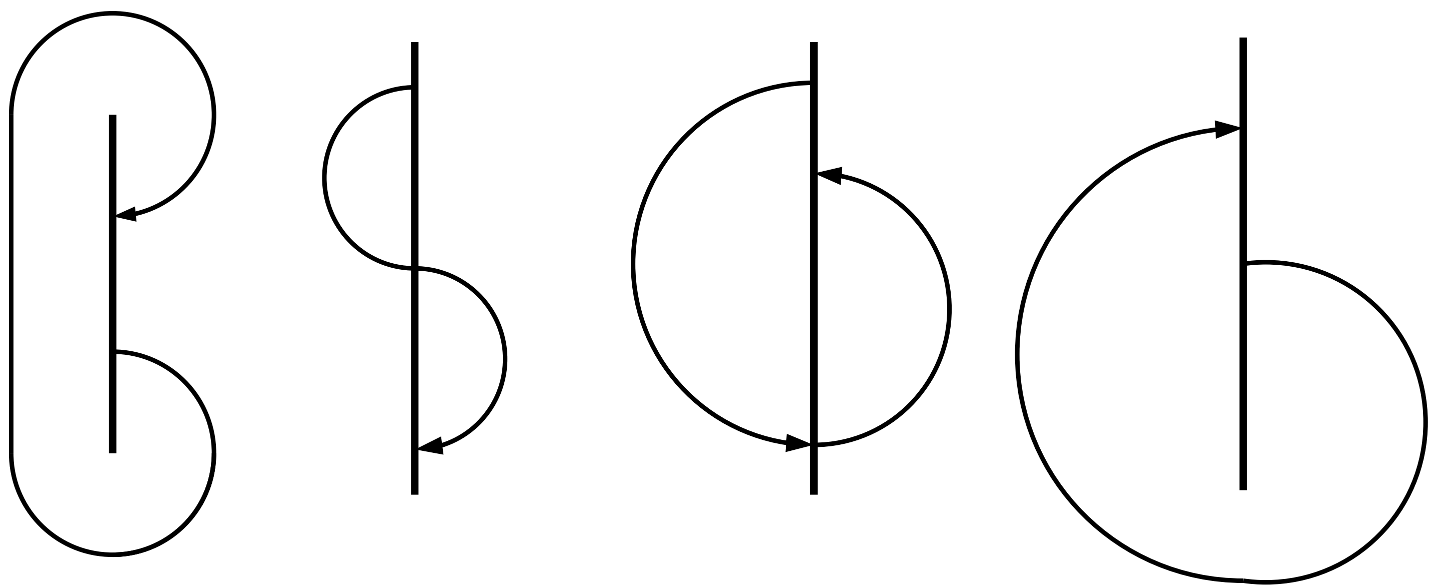
Corollary: Any triangulation of an n-gon has a refinement into $O(n^2)$ right triangles.

Improves $O(n^4)$ bound by Bern and Eppstein (1992).

In general, path can hit same thin part many times.

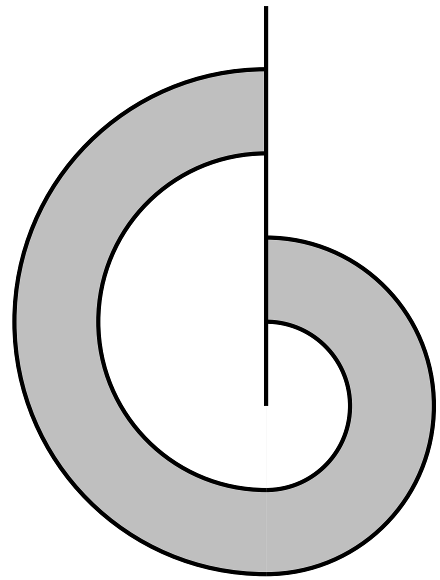


If a path returns to same thin edge at least 3 times it has a sub-path that looks like one of these:

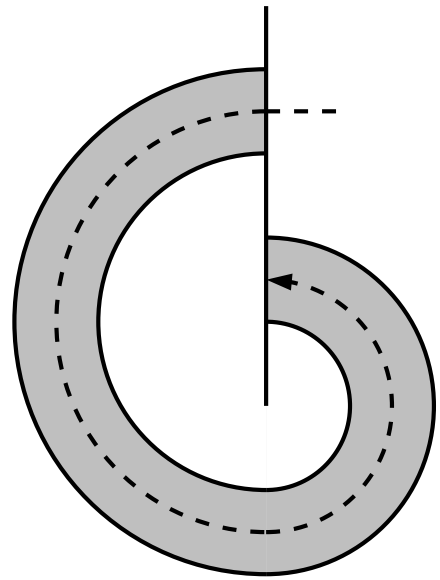


C-curve, S-curve, G-curves

Return region consists of paths “parallel” to one of these.

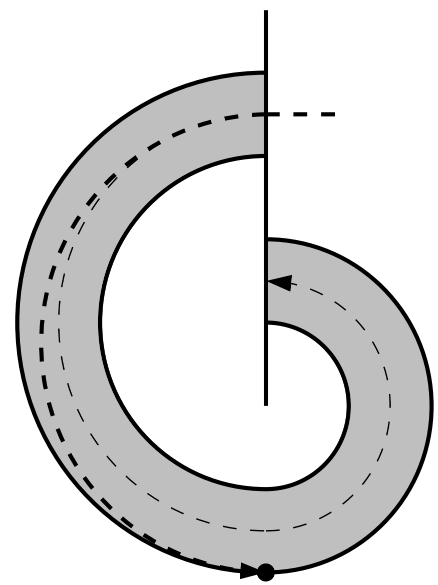


Return region consists of paths "parallel" to one of these.



There are $O(n)$ return regions and every propagation path enters one after crossing at most $O(n)$ thin parts.

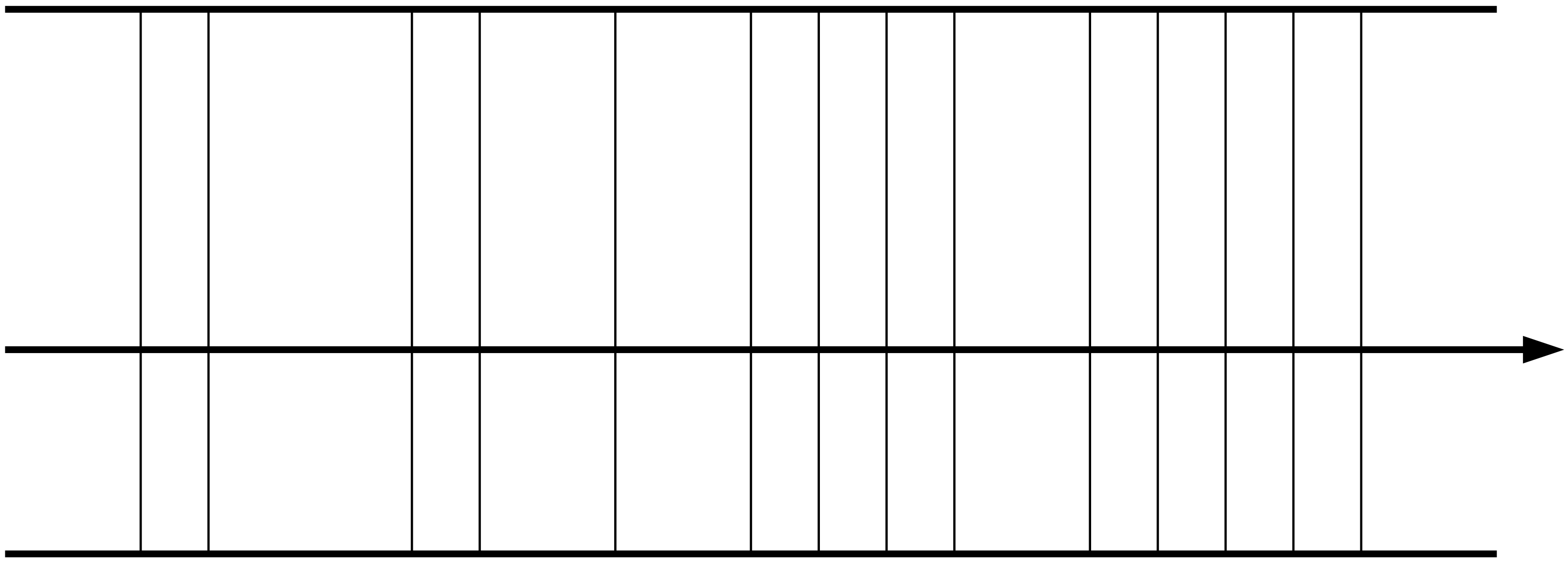
Return region consists of paths “parallel” to one of these.



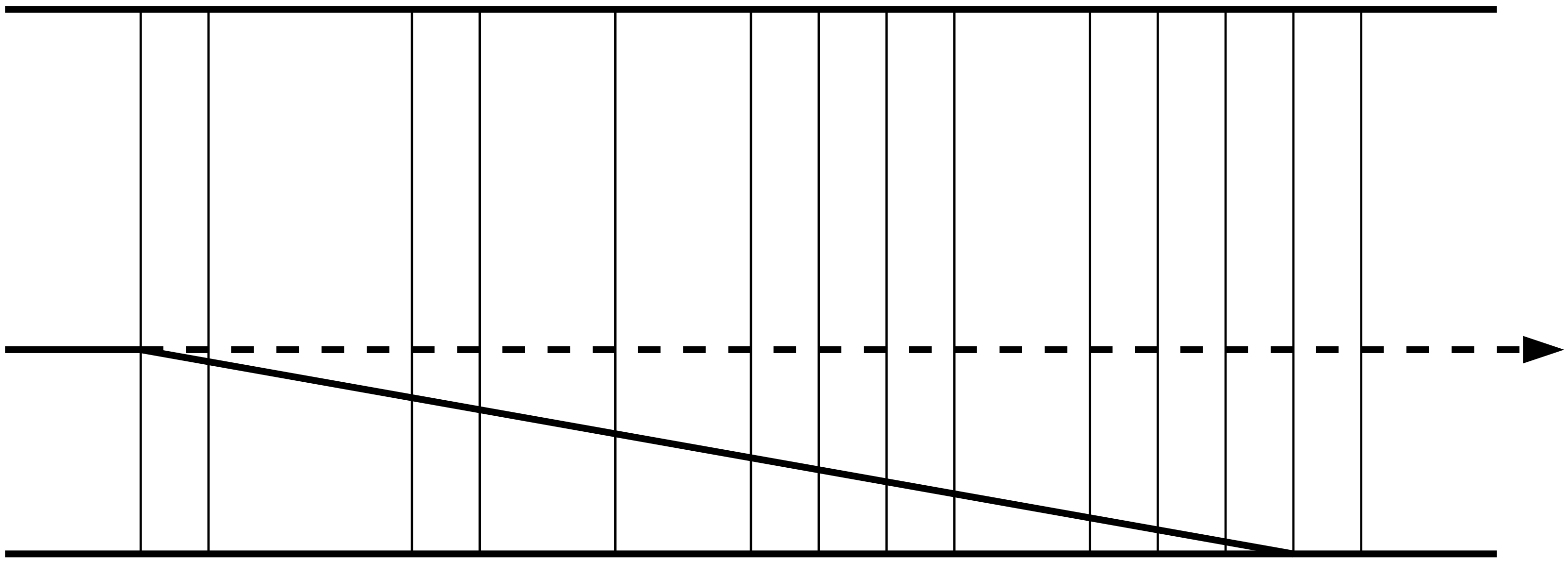
There are $O(n)$ return regions and every propagation path enters one after crossing at most $O(n)$ thin parts.

We want to bend paths to terminate before they exit.

For simplicity, "straighten" region to rectangle.

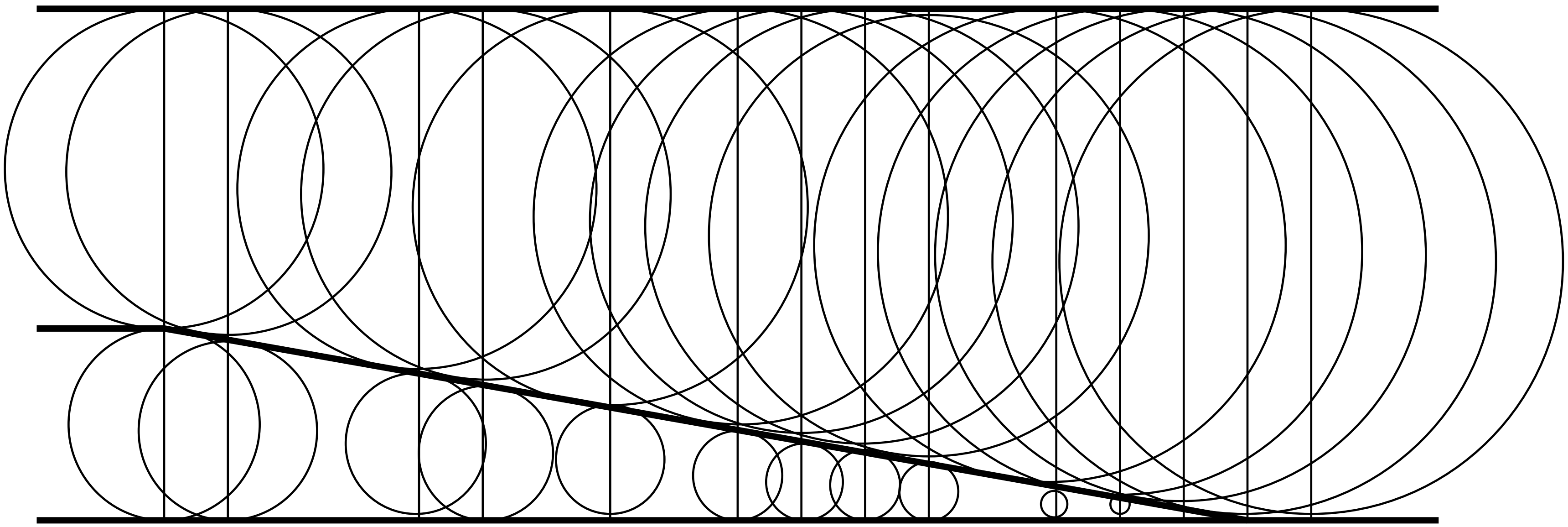


For simplicity, “straighten” region to rectangle.



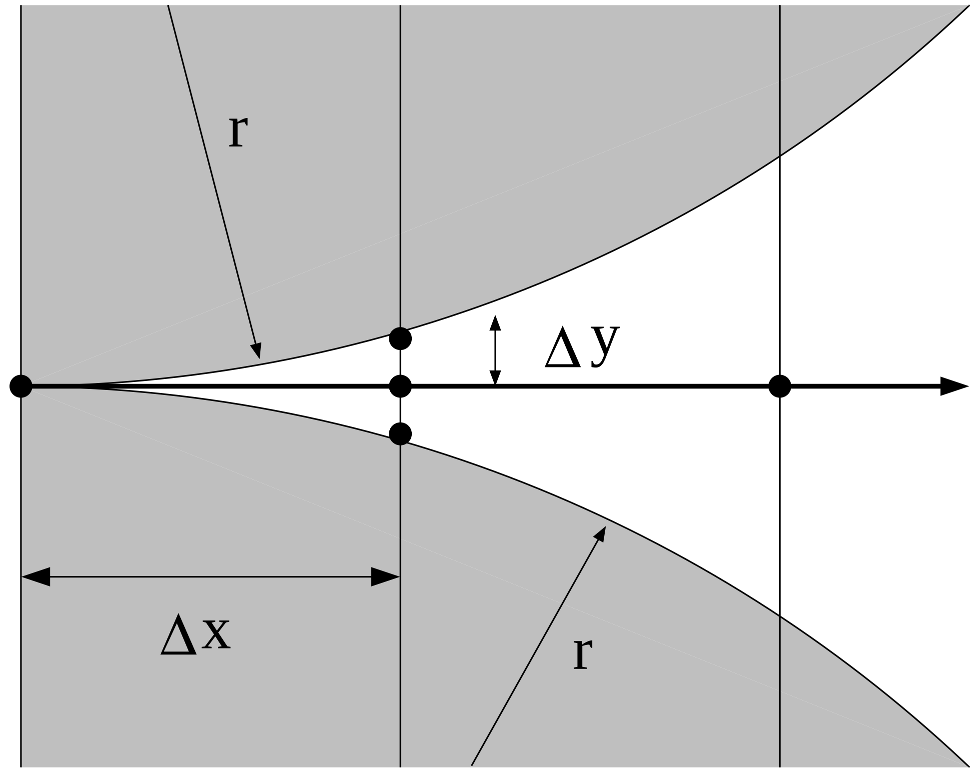
We want to bend path to hit side of tube. If it hits existing vertex, then path ends.

For simplicity, “straighten” region to rectangle.

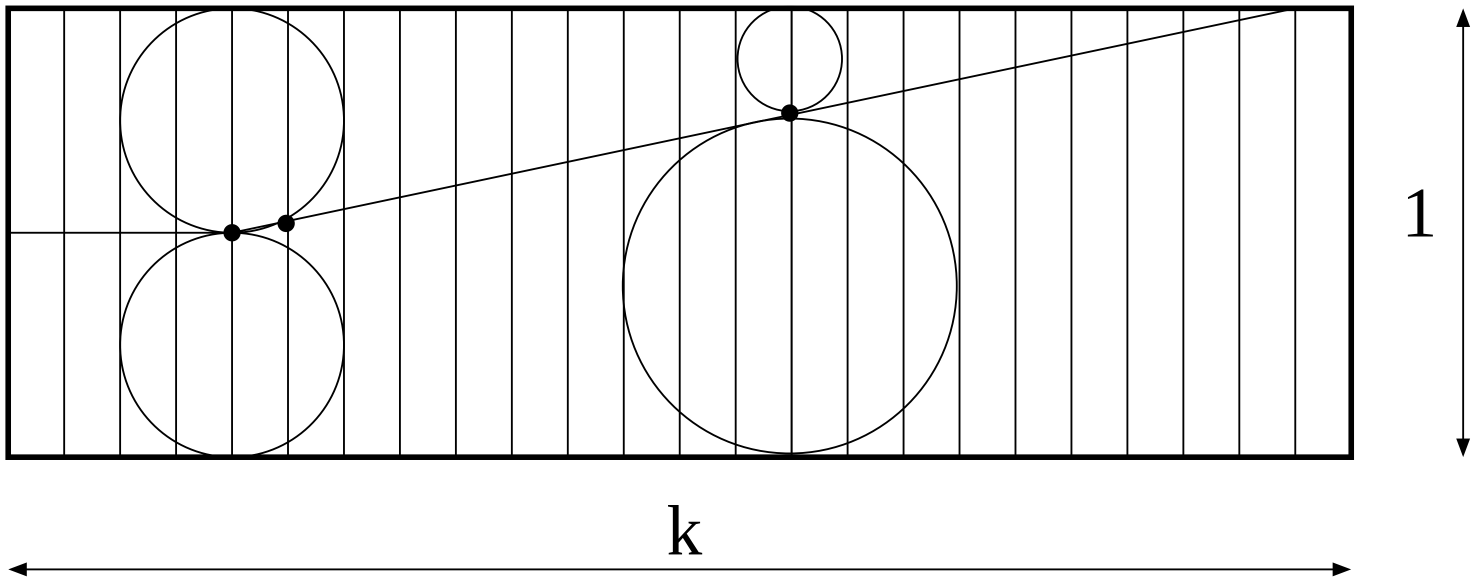


Bending must satisfy Gabriel condition. Diameter disks must not contain any path vertices.

How far can we bend?



Answer: $\Delta y \approx (\Delta x/r)^2 r = (\Delta x)^2/r.$



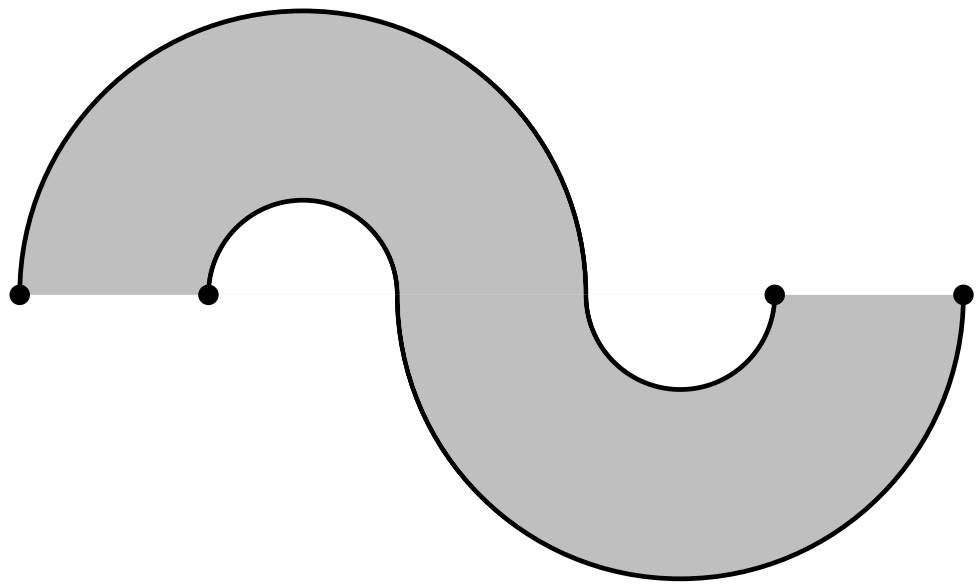
In $1 \times k$ tube crossing n (equally spaced) thin parts,

$$r \approx 1, \quad \Delta x \approx k/n, \quad \Delta y \approx k^2/n^2$$

Need $1 \approx \sum \Delta y = n\Delta y = n(k/n)^2 = k^2/n$.

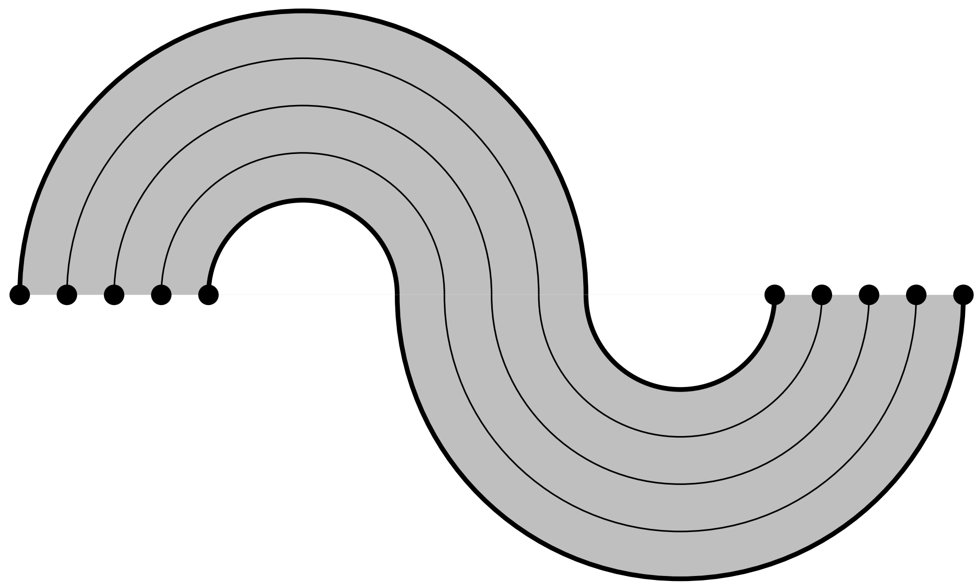
Can terminate path if $k \gg \sqrt{n}$.

Easy case: return region length > width.



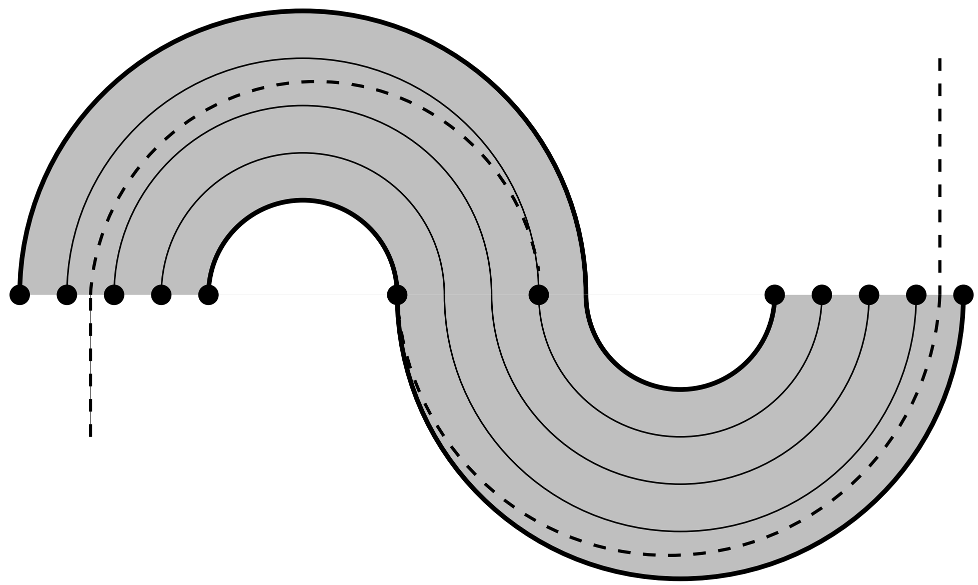
- Show there are $O(n)$ return regions.

Easy case: return region length > width.



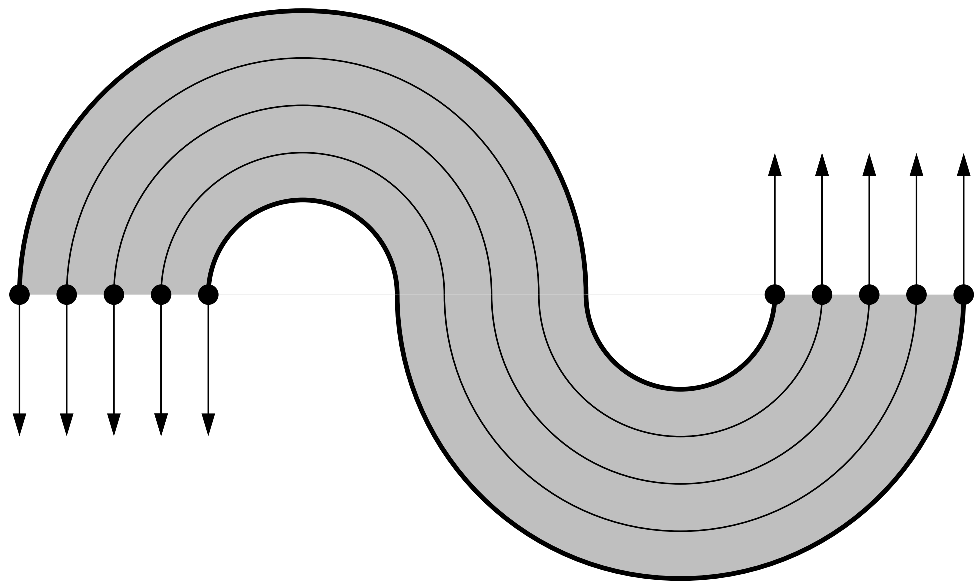
- Show there are $O(n)$ return regions.
- Divide each region into $O(\sqrt{n})$ long parallel tubes.

Easy case: return region length > width.



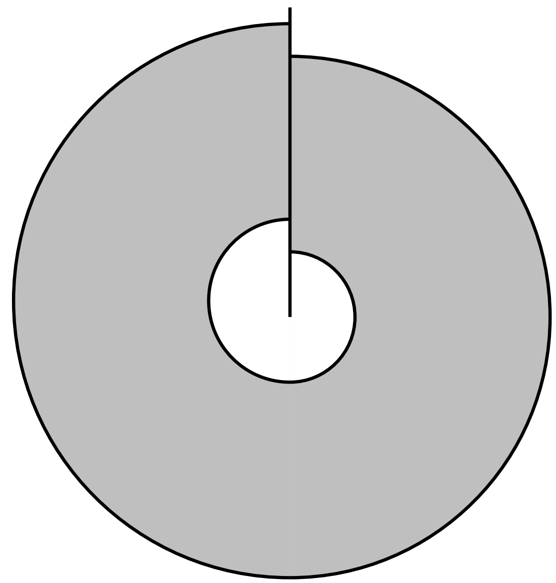
- Show there are $O(n)$ return regions.
- Divide each region into $O(\sqrt{n})$ long parallel tubes.
- Entering paths can be bent and terminated.
  Total vertices created $= O(n^2)$, but …

Easy case: return region length > width.

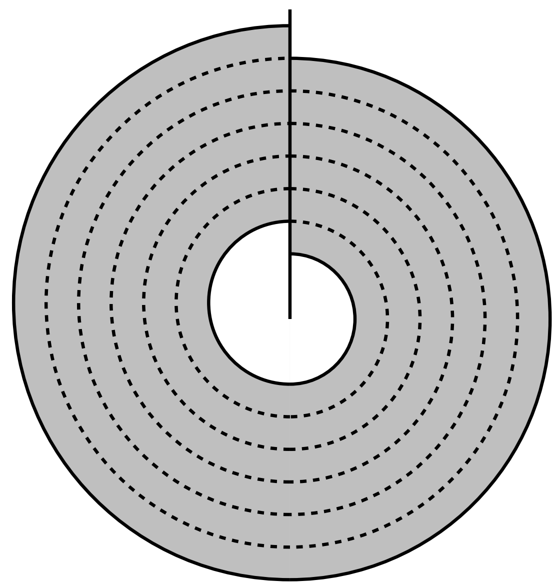


- Show there are $O(n)$ return regions.
- Divide each region into $O(\sqrt{n})$ long parallel tubes.
- Entering paths can be bent and terminated.
  Total vertices created $= O(n^2)$, but $\ldots$
- Each region has $O(\sqrt{n})$ new vertices to propagate.
  Vertices created is $O(n \cdot \sqrt{n} \cdot n) = O(n^{2.5})$.

Hard case is spirals:



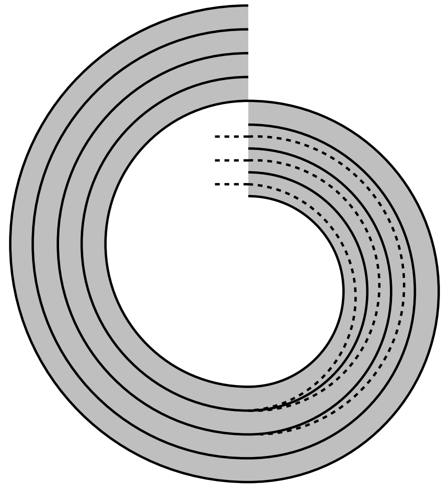
Hard case is spirals:



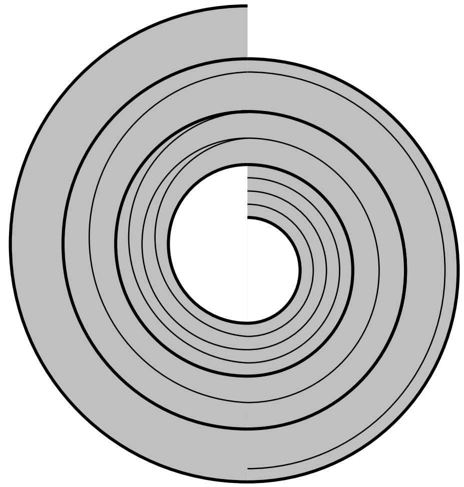
Curves may spiral arbitrarily often.

No curve can be allowed to pass all the way through the spiral. We stop them in a multi-stage construction.

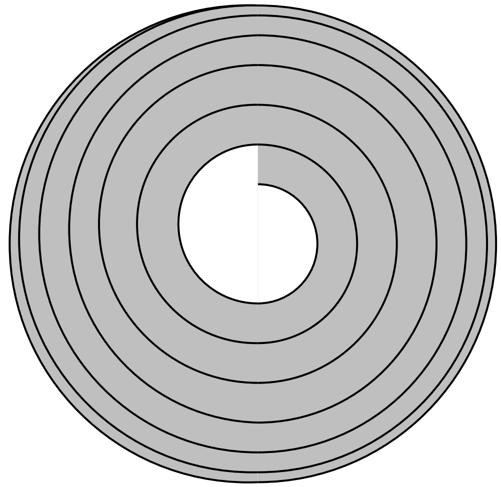
Normalize so "entrance" is unit width.



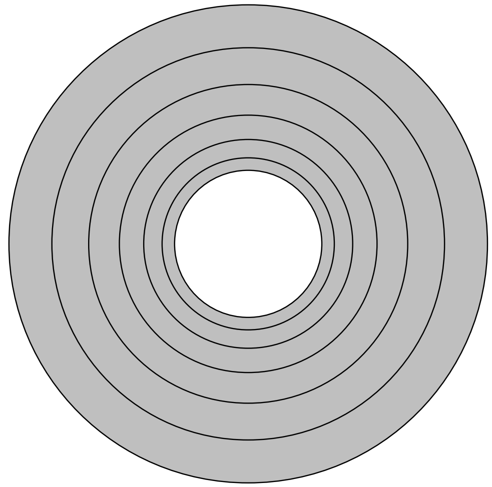
- Start with $\sqrt{n}$ parallel tubes at entrance of sprial. Terminate entering paths (1 spiral).



- Start with $\sqrt{n}$ parallel tubes at entrance of sprial. Terminate entering paths (1 spiral).
- Merge $\sqrt{n}$ tubes to single tube ($n^{1/3}$ spirals). (get longer as we move out.)



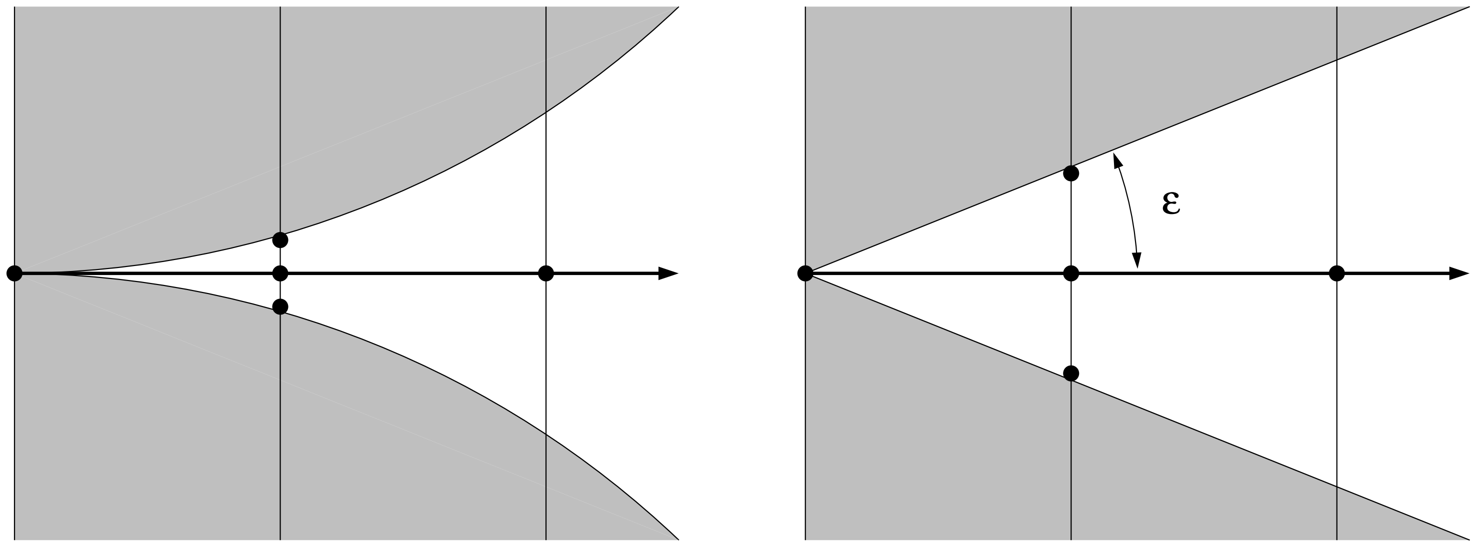
- Start with $\sqrt{n}$ parallel tubes at entrance of sprial.
  Terminate entering paths (1 spiral).
- Merge $\sqrt{n}$ tubes to single tube ($n^{1/3}$ spirals).
  (spirals get longer as we move out.)
- Make tube edge self-intersect ($n^{1/2}$ spirals)



- Start with $\sqrt{n}$ parallel tubes at entrance of sprial. Terminate entering paths (1 spiral).
- Merge $\sqrt{n}$ tubes to single tube ($n^{1/3}$ spirals). (spirals get longer as we move out.)
- Make tube edge self-intersect ($n^{1/2}$ spirals)
- Loops with increasing gaps ($n^{1/2}$ loops, n spirals)
- Beyond radius n spiral is empty.

Careful estimates needed to get $O(n^{2.5})$ vertices.

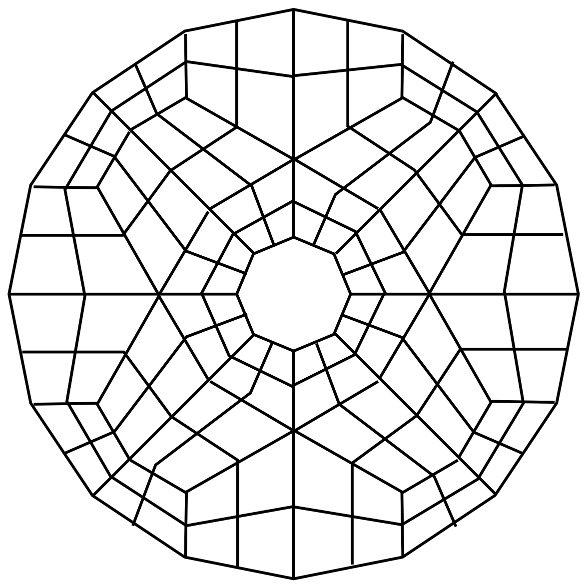
Almost Nonobtuse Triangulation: Replace cusps by cones of angle ϵ. Same construction in thick parts.

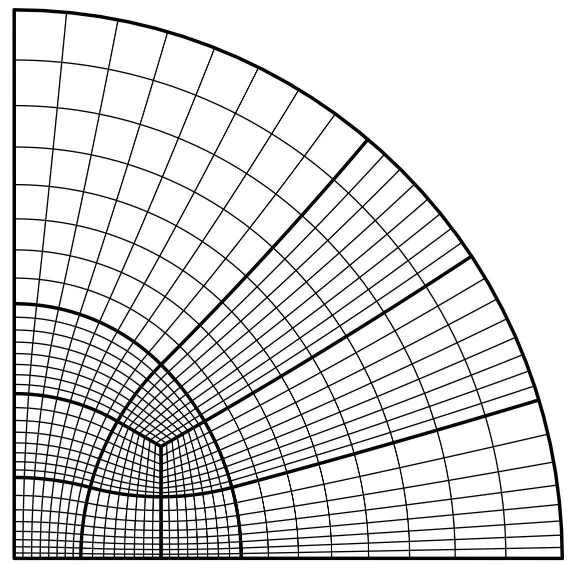


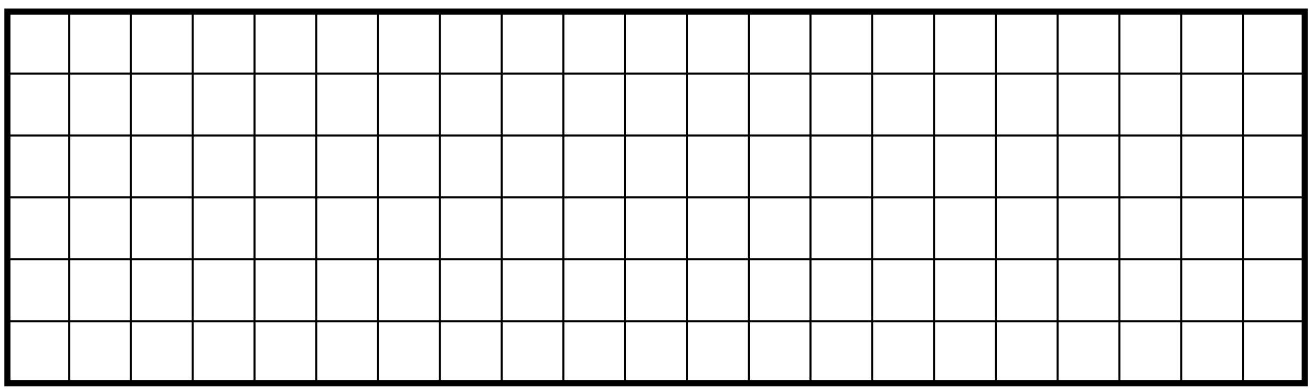
Paths can be terminated inside a $1 \times \frac{1}{\epsilon}$ tube.

Thm: Uses angles $\leq 90^\circ + \epsilon$ and $O(\frac{n^2}{\epsilon^2})$ triangles.

Quadrilateral meshes:







Some results

• Every n-gon has $O(n)$ quad mesh with angles $\leq 120^\circ$. Bern and Eppstein, 2000. $O(n \log n)$ work.

• They showed any quad mesh of regular hexagon has at least one angle $\geq 120^\circ$.

Some results

- Every n-gon has $O(n)$ quad mesh with angles $\leq 120^\circ$. Bern and Eppstein, 2000. $O(n \log n)$ work.

- They showed any quad mesh of regular hexagon has at least one angle $\geq 120^\circ$.

Theorem, B, 2008: Every n-gon has $O(n)$ quad mesh with angles $\leq 120^\circ$ and every new angle $\geq 60^\circ$. Takes $O(n)$ work.

Theorem, B, 2010: Every PSLG has a $O(n^2)$ quad mesh with all angles $\leq 120^\circ$ and all new angles $\geq 60^\circ$.

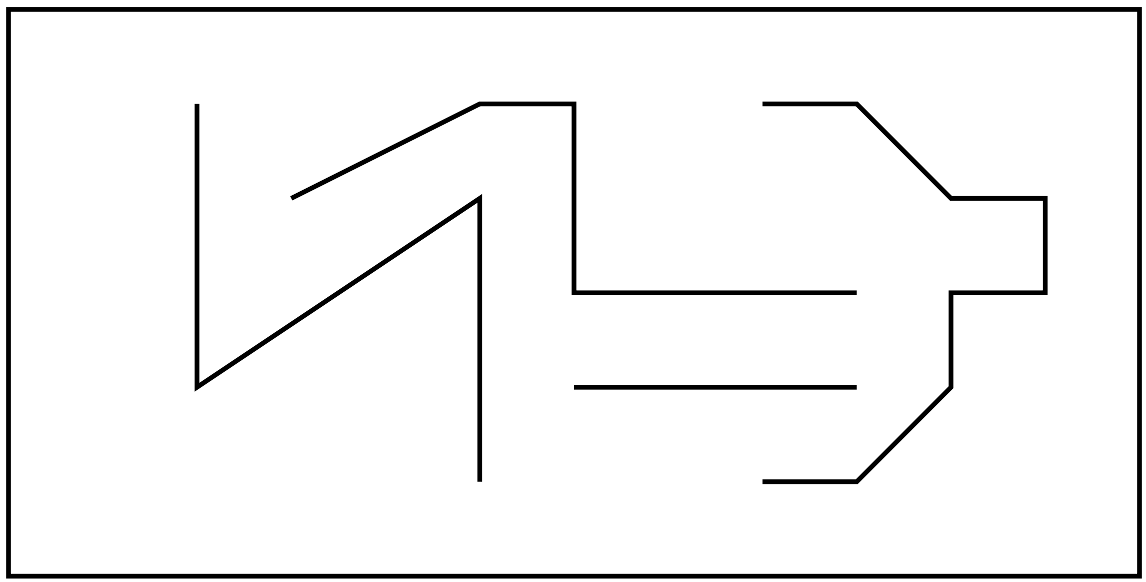
Angles bounds and complexity are sharp.

At most $O(\frac{n}{\epsilon})$ angles outside $[90^\circ - \epsilon, 90^\circ + \epsilon]$.

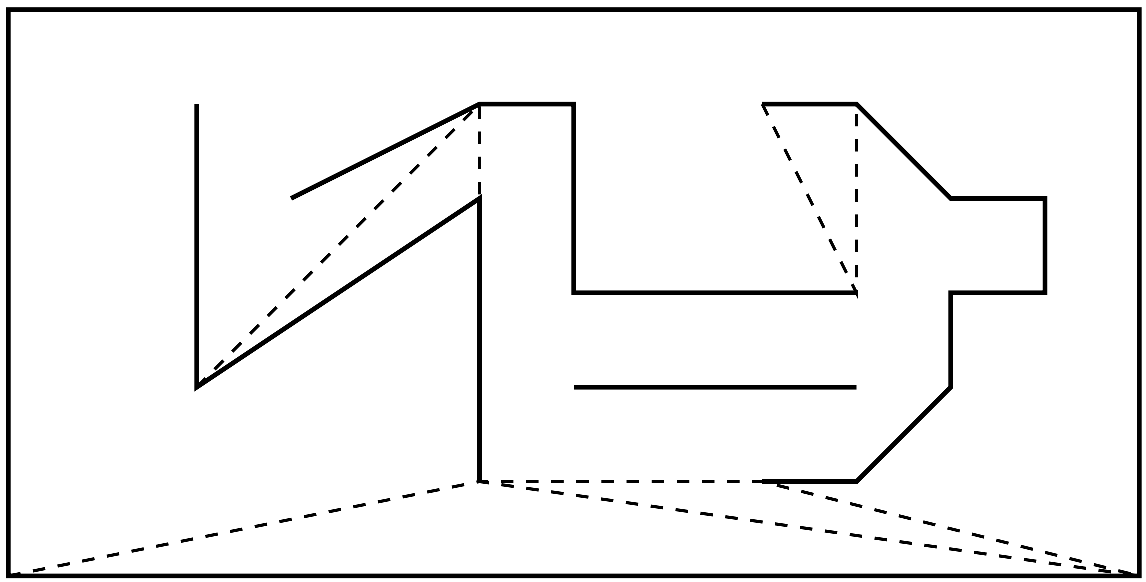
Idea of proof:

- Connect Γ. Components now polygons, not triangles.
- Define thick/thin pieces.
- Mesh thin parts using propogation paths as before.
- Mesh thick parts using hyperbolic geometric in disk and conformal map to polygon.
- Insure consistency between thick and thin parts using special meshes called “sinks”.

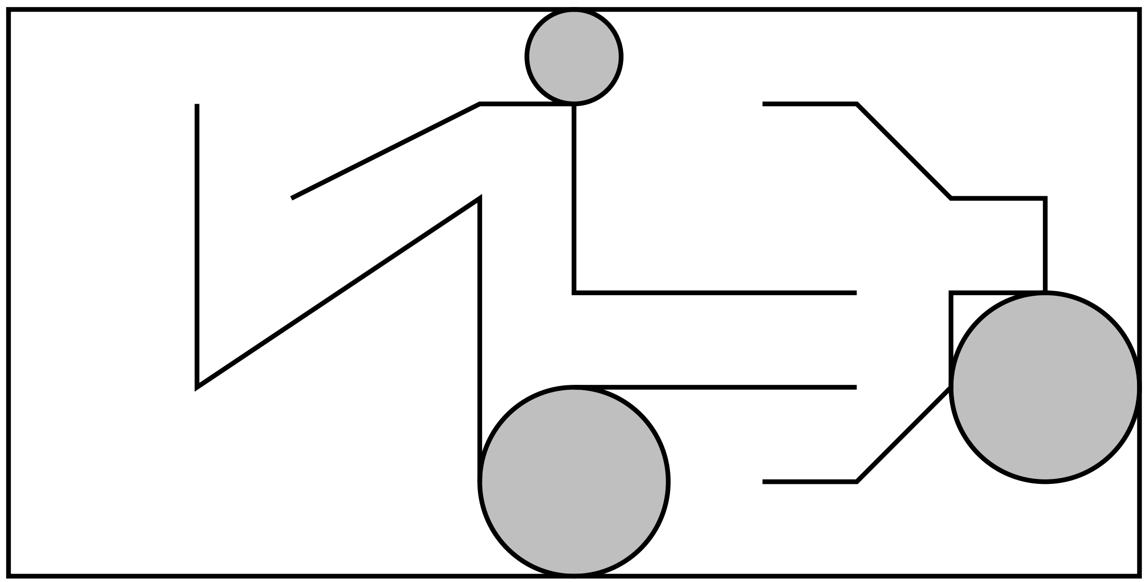
Triangulating Γ may give small angles. Must connect without small angles.



Triangulating Γ may give small angles. Must connect without small angles.

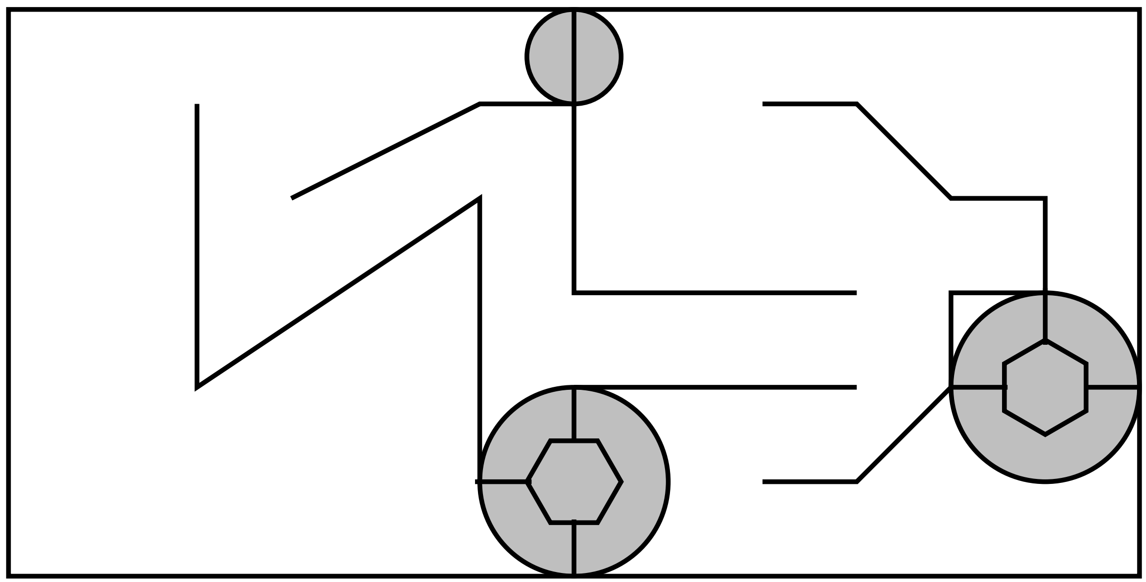


Triangulating Γ may give small angles. Must connect without small angles.



First connect Γ using disjoint disks.

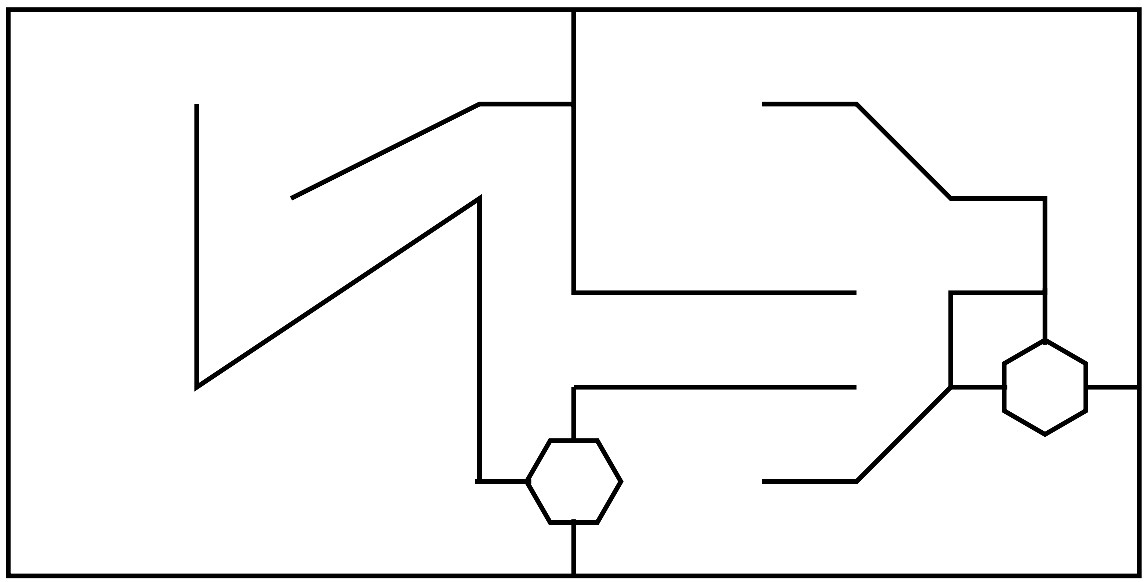
Triangulating Γ may give small angles. Must connect without small angles.



First connect Γ using disjoint disks.

Connect contact points using angles in $[60^\circ, 120^\circ]$

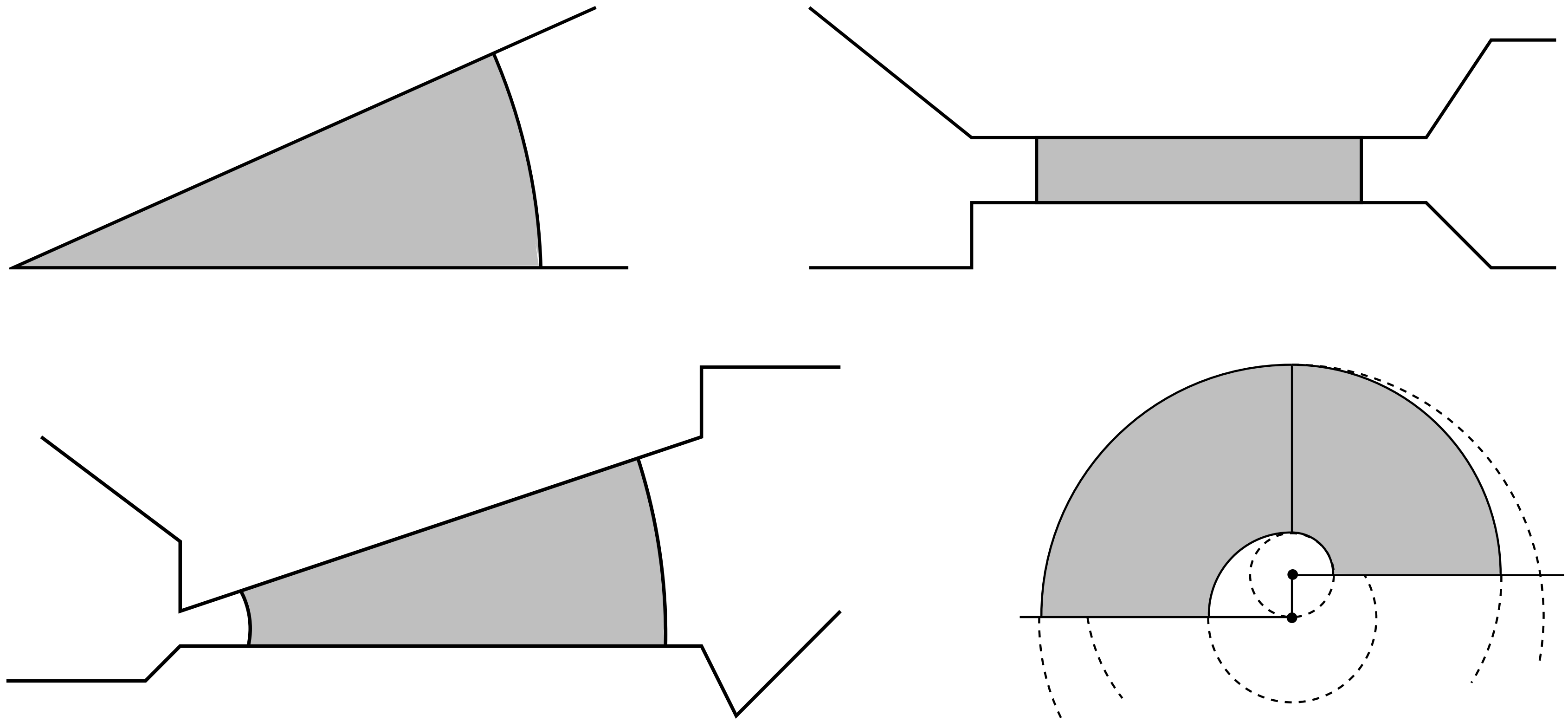
Triangulating Γ may give small angles. Must connect without small angles.



First connect γ using disjoint disks.

Connect contact points using angles in $[60^\circ, 120^\circ]$

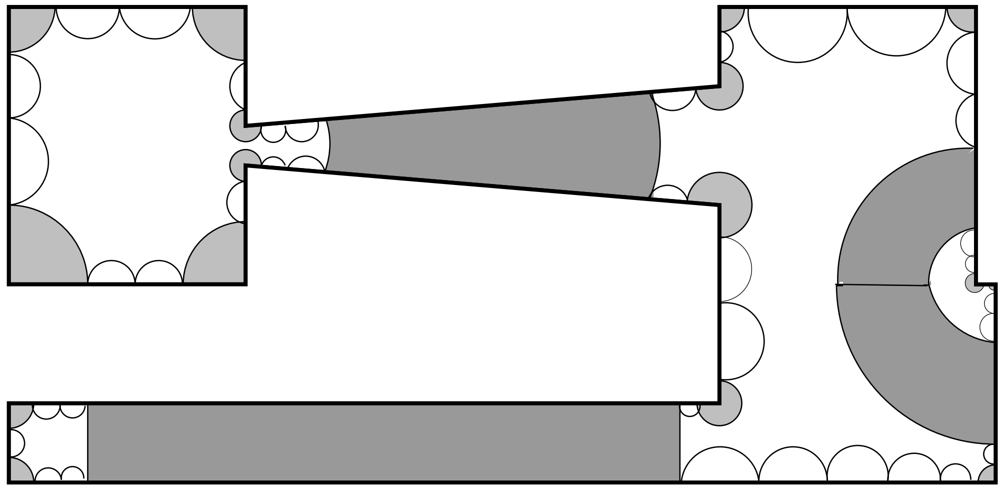
Thick/thin pieces of polygons.



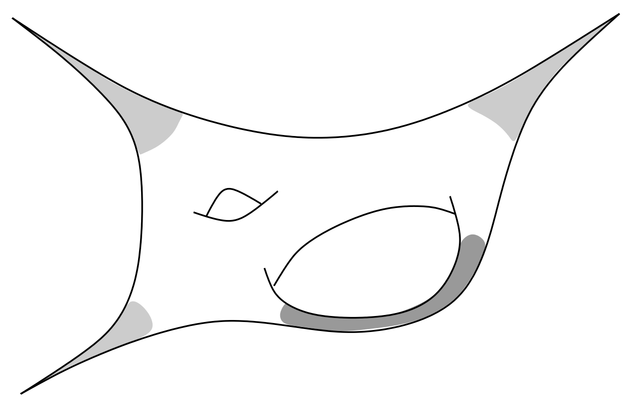
Thin piece is a sector whose two straight sides satisfy

$$\mathrm{dist}(I, J) \ll \min(|I|, |J|).$$

Conformally equivalent to $1 \times R$ rectangle, $R \gg 1$.



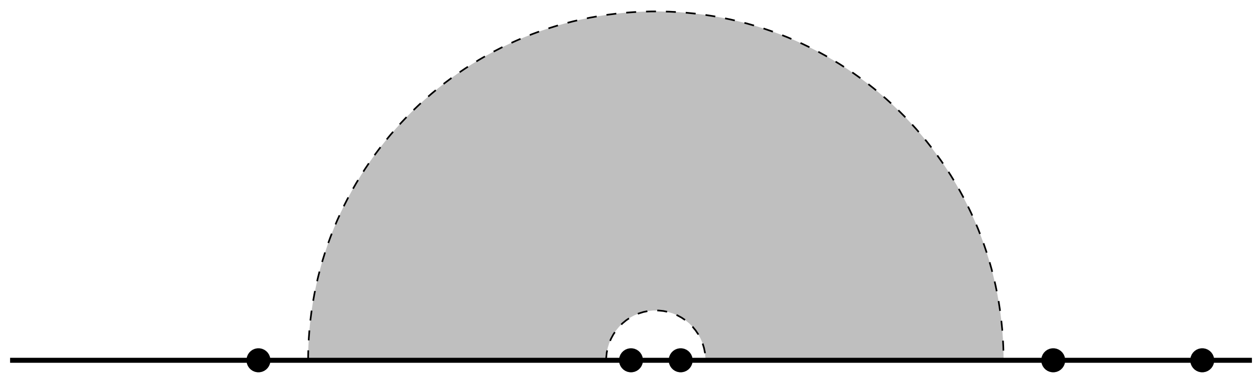
Parabolic = adjacent edges $\approx$ cusps
Hyperbolic = non-adjacent edges $\approx$ short geodesics.
Thick sides covered by $O(n)$ interior half-disks.



Thin parts computable in $O(n)$ using conformal map.

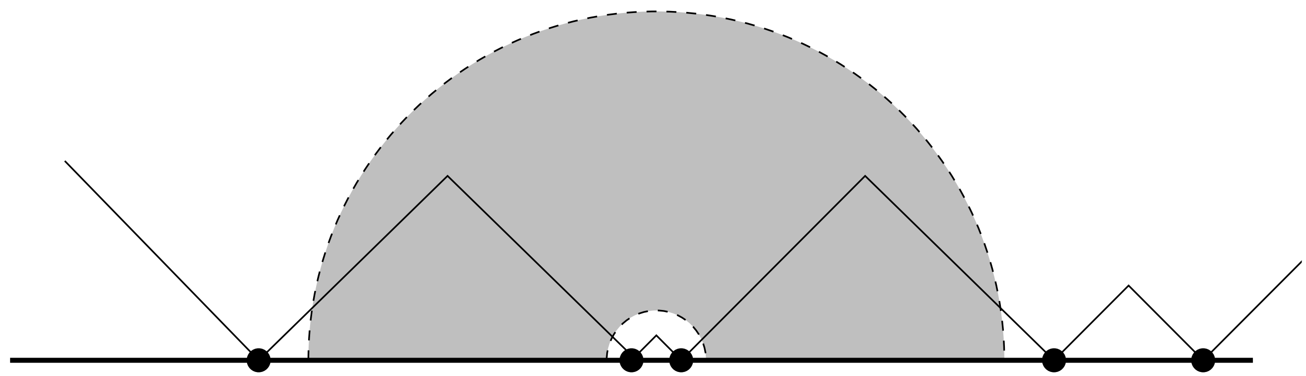
- Map polygon conformally to half-plane.
  Vertices map to points on line.

Thin parts computable in $O(n)$ using conformal map.



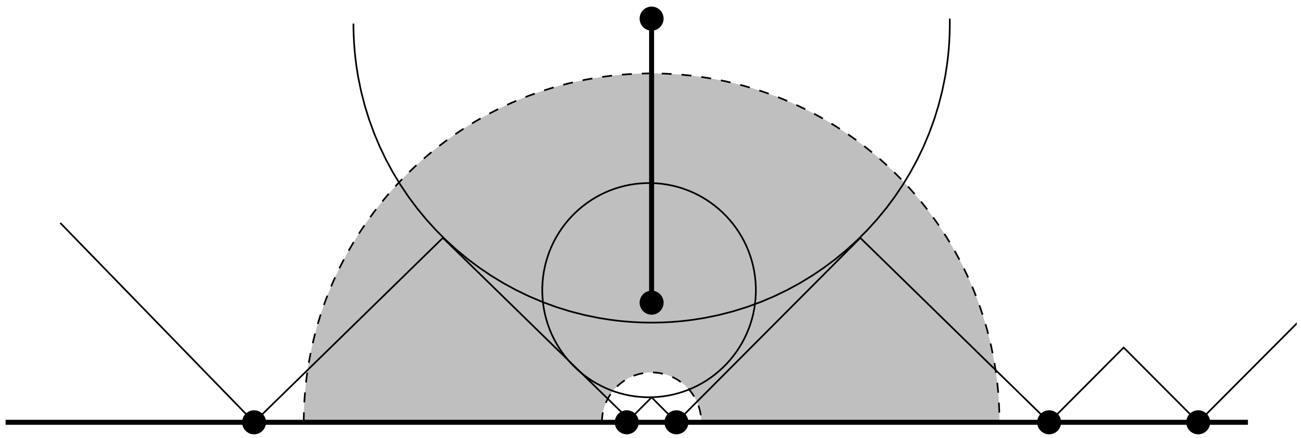
- Map polygon conformally to half-plane.
  Vertices map to points on line.
- Thin parts = wide annuli separating vertices.

Thin parts computable in $O(n)$ using conformal map.



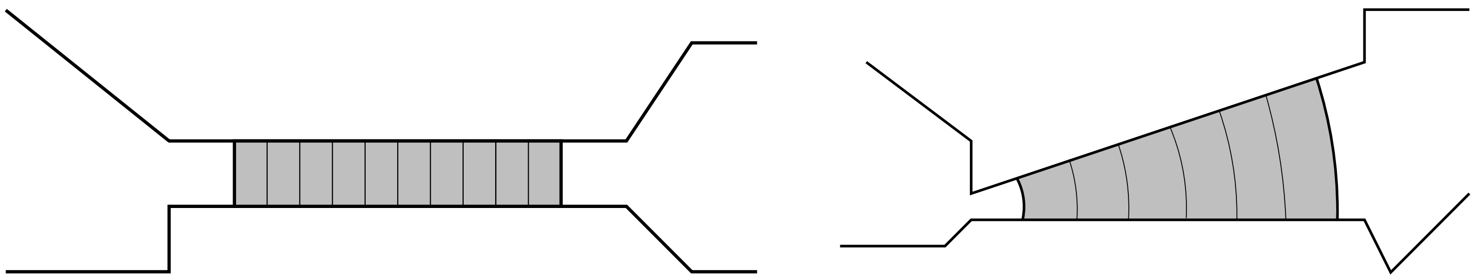
- Map polygon conformally to half-plane.
  Vertices map to points on line.
- Thin parts = wide annuli separating vertices.
- Draw sawtooth domain, valleys at vertices.

Thin parts computable in $O(n)$ using conformal map.



- Map polygon conformally to half-plane. Vertices map to points on line.
- Thin parts = wide annuli separating vertices.
- Draw sawtooth domain, valleys at vertices.
- Compute medial axis (internal maximal disks). Find long (hyperbolically) vertical edges.

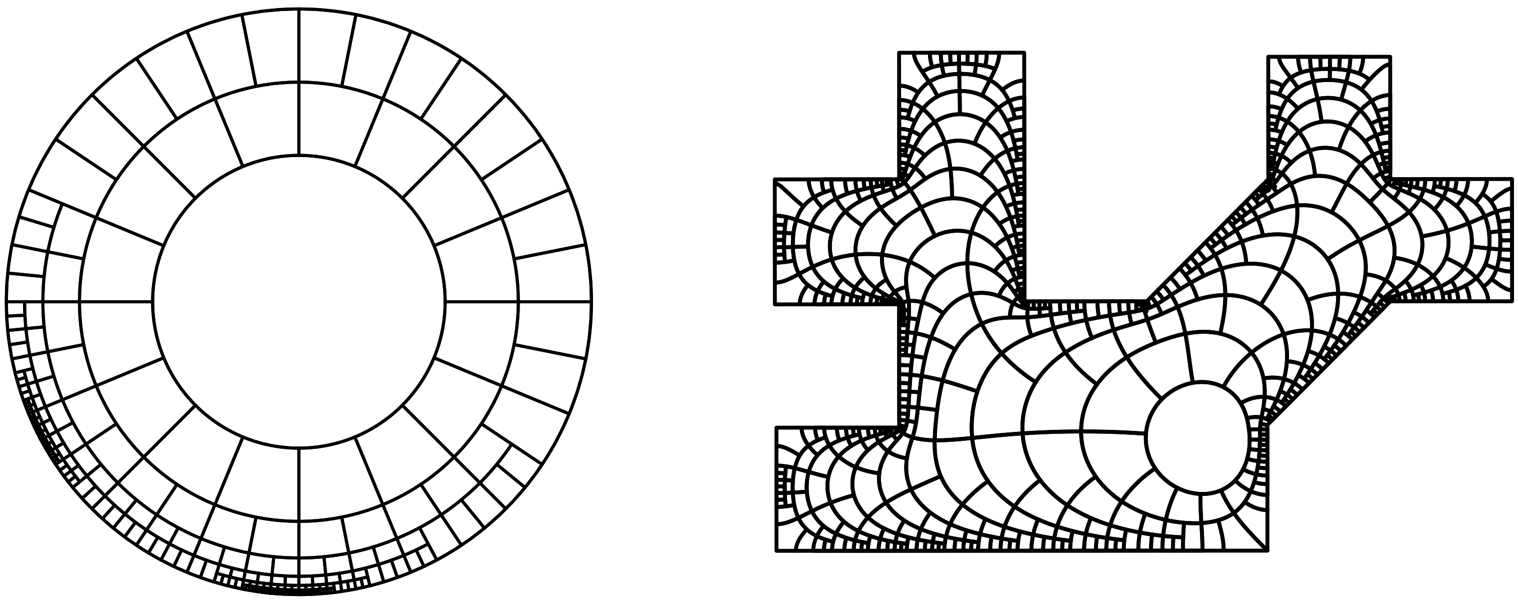
Thin parts have foliations.



Quad mesh in thin parts is just like $90^\circ + \epsilon$ triangulation.

Harder part is to mesh the thick parts.

Basic idea for thick parts: Conformal map from disk preserves angles except near vertices.



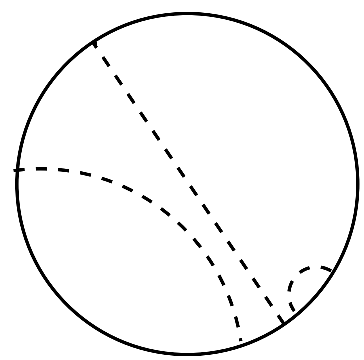
Transfer mesh on disk to mesh of polygon.

Need to be careful with tiles and timing.

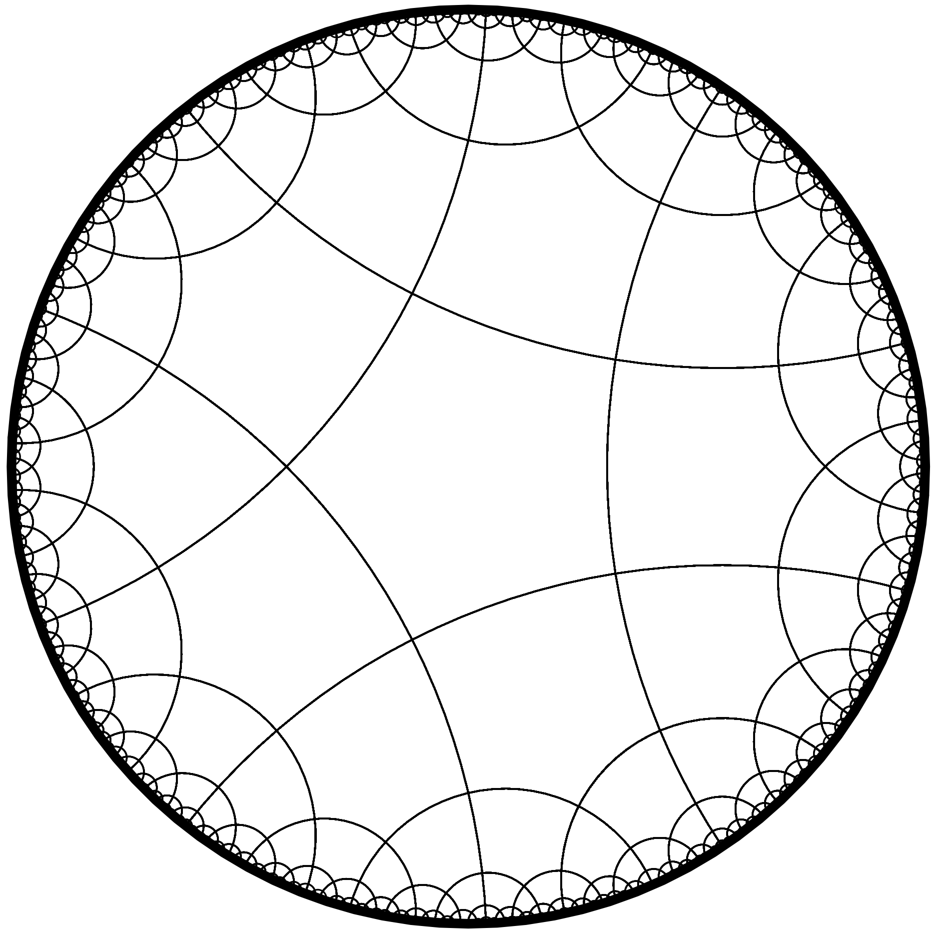
Hyperbolic metric on disk given by

$$d\rho = \frac{ds}{1-|z|^2} \simeq \frac{ds}{\mathrm{dist}(z, \partial D)}.$$

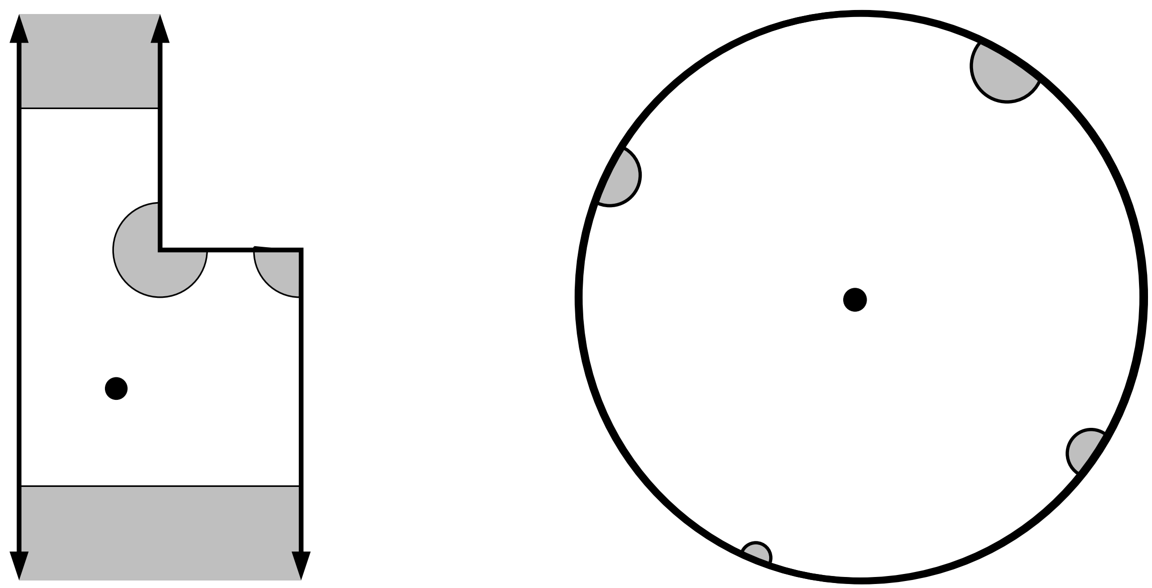
- Isometries are the Möbius transformations.
- Geodesics are circles perpendicular to boundary.



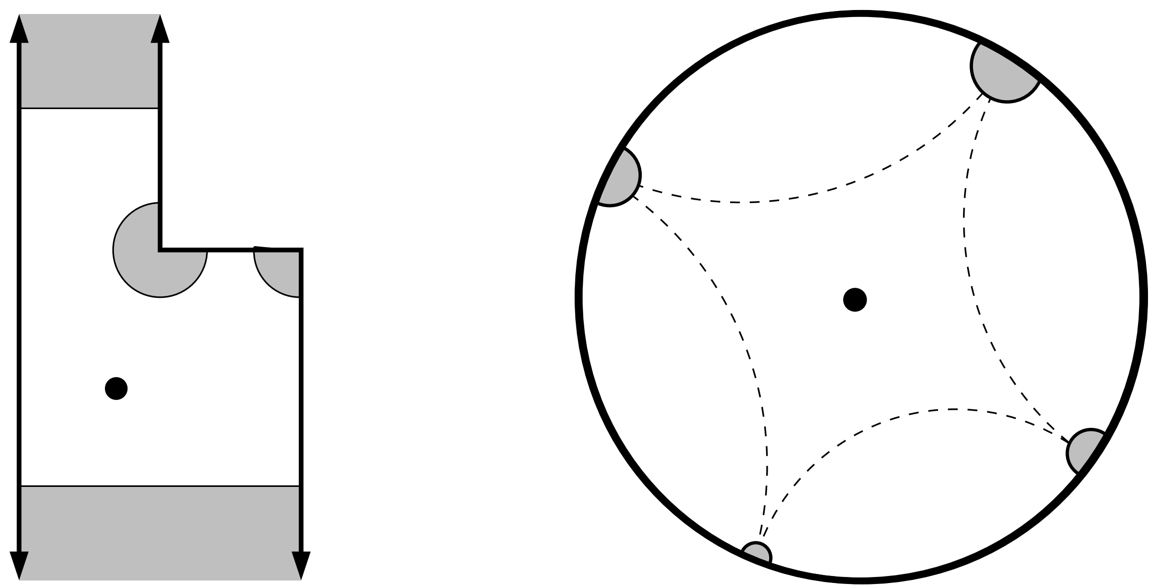
Hyperbolic space can be tesselated by right pentagons.



Conformal map from polygon to disk takes thick part to region as shown.

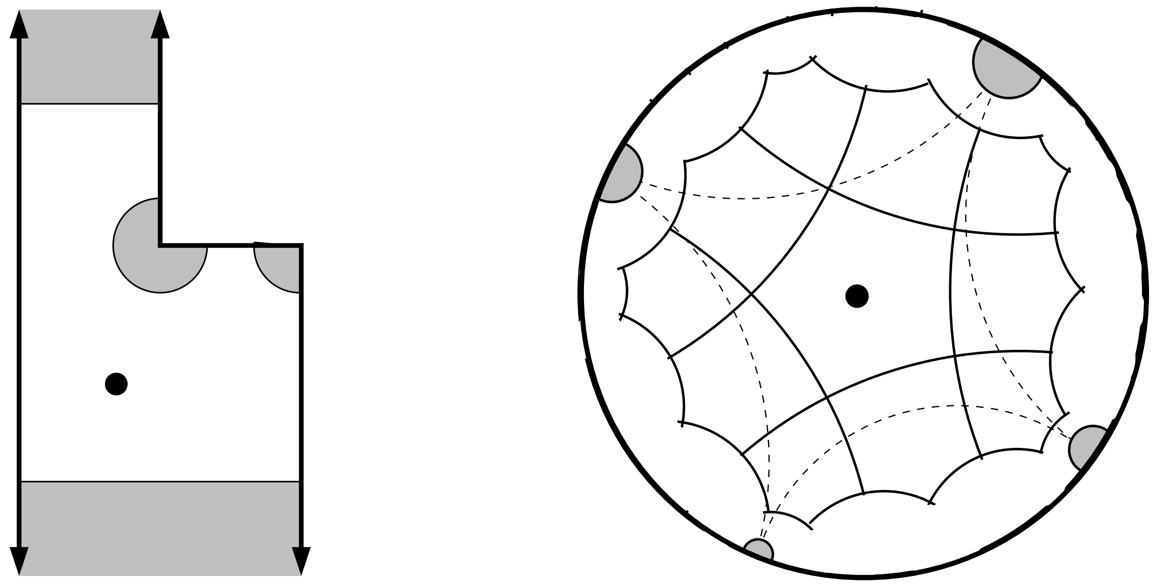


Conformal map from polygon to disk takes thick part to region as shown.



Draw (hyperbolic) convex hull of thin regions.

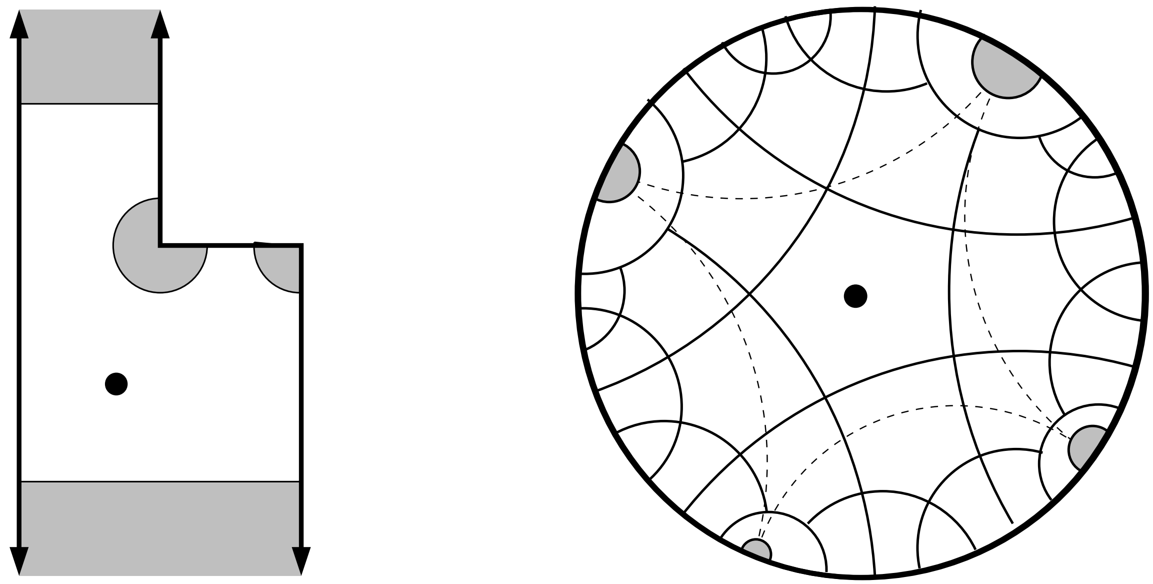
Conformal map from polygon to disk takes thick part to region as shown.



Draw (hyperbolic) convex hull of thin regions.

Take pentagons from tesselation hitting convex hull but missing thin parts.

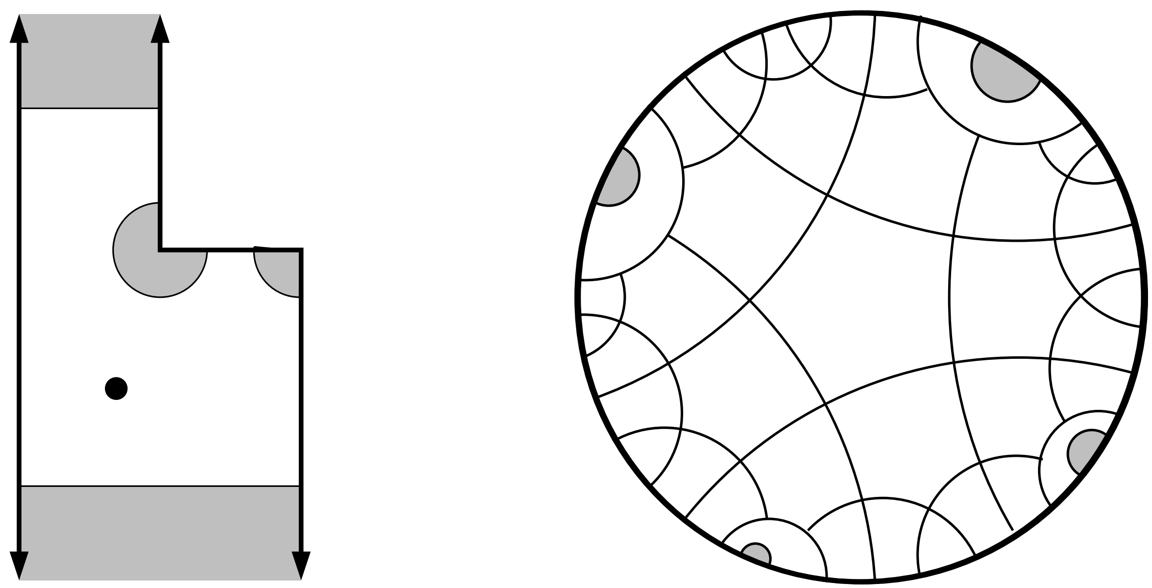
Conformal map from polygon to disk takes thick part to region as shown.



Draw (hyperbolic) convex hull of thin regions.

Take pentagons from tesselation hitting convex hull but missing thin parts. Extend pentagon edges to boundary.

Conformal map from polygon to disk takes thick part to region as shown.

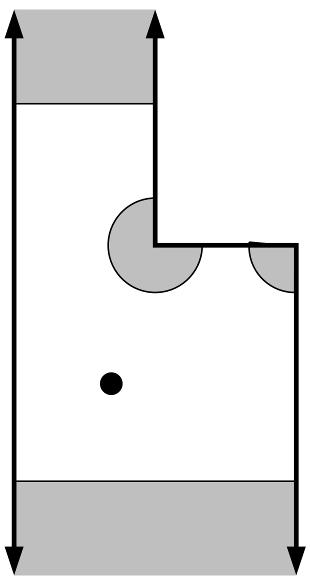


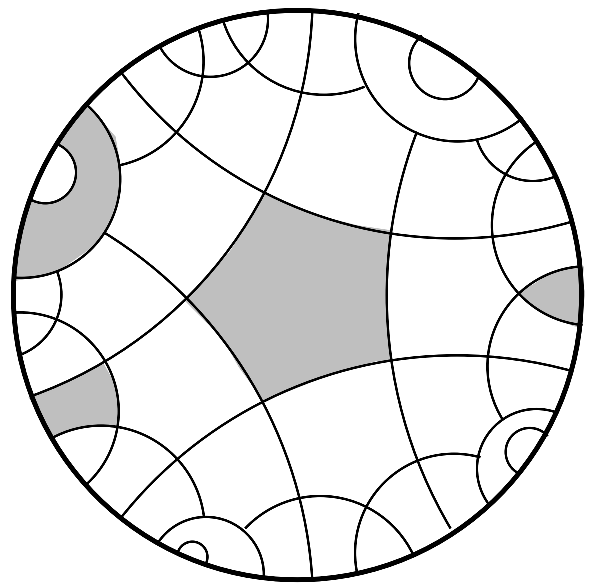
Draw (hyperbolic) convex hull of thin regions.

Take pentagons from tesselation hitting convex hull but missing thin parts. Extend pentagon edges to boundary.

Analog of Whitney decomposition or quadtreee construction).

Conformal map from polygon to disk takes thick part to region as shown.

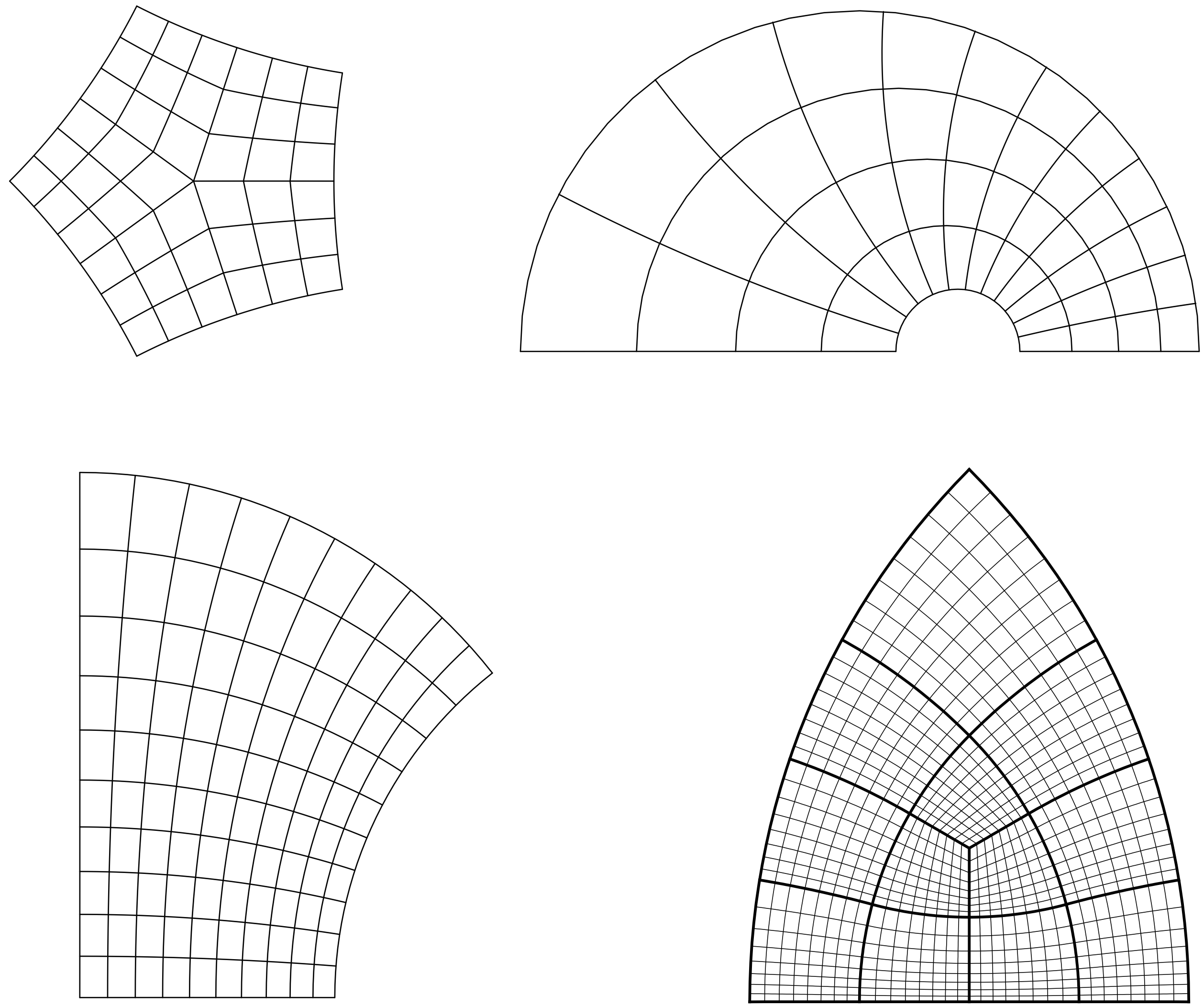




Draw (hyperbolic) convex hull of thin regions.

Take pentagons from tesselation hitting convex hull but missing thin parts. Extend pentagon edges to boundary.

Pentagons, quadrilaterals, triangles and half-annuli.



Shapes can be meshed to match along common edges.

How much time to create conformal map?

Theorem (B, 2008): We can compute a ϵ-conformal map onto n-gon in $O(n \log \frac{1}{\epsilon} \log\log \frac{1}{\epsilon})$.

ϵ-conformal $=$ ϵ-distortion of angles.

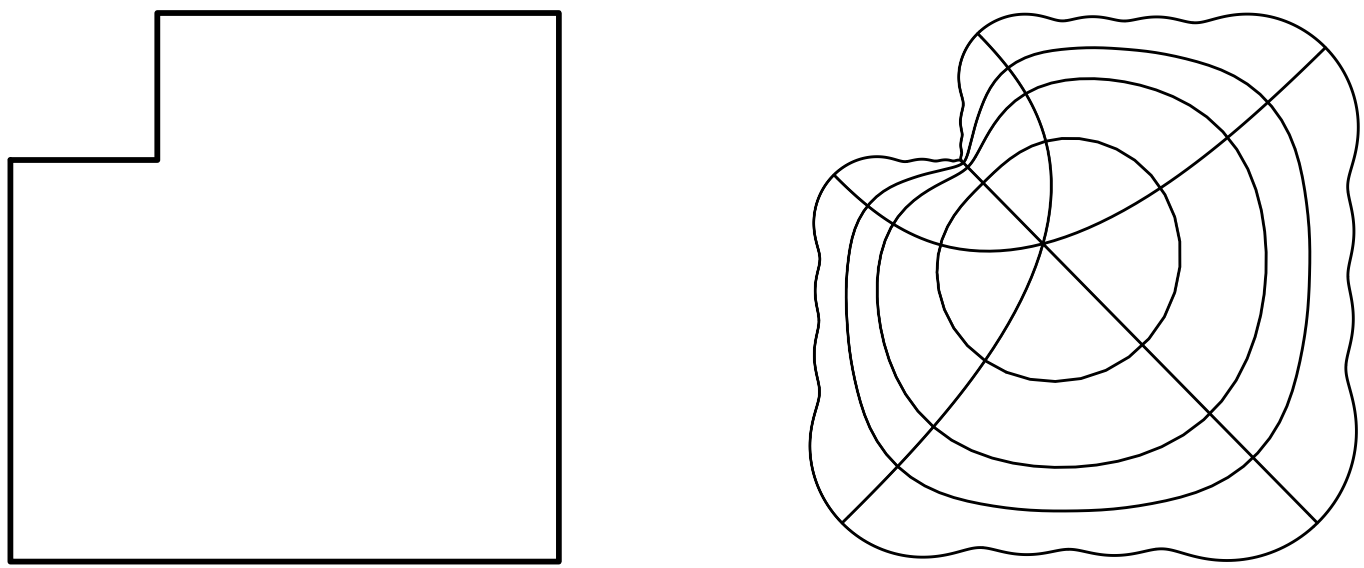
First method to give guarenteed sucess in linear time, e.g., CRDT algorithm of Driscoll and Vavsis is $O(n^3)$ and not proven to converge.

For our application we want to approximate conformal map onto n-gon at $O(n)$ points in $O(n)$ time.

Proof of the fast mapping theorem:

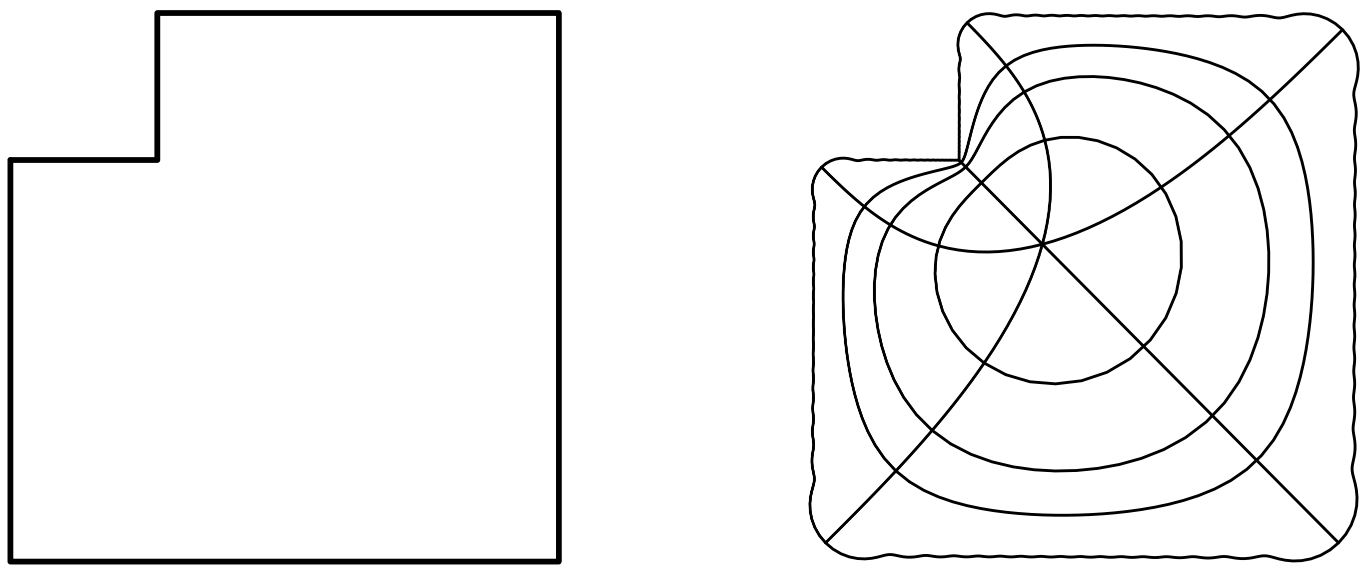
- Local representation of maps
- Newton's method for Beltrami's equation
- The iota map for initial guess

Conformal maps have power series, but corners of polygon create singularites on circle. Convergence is slow.



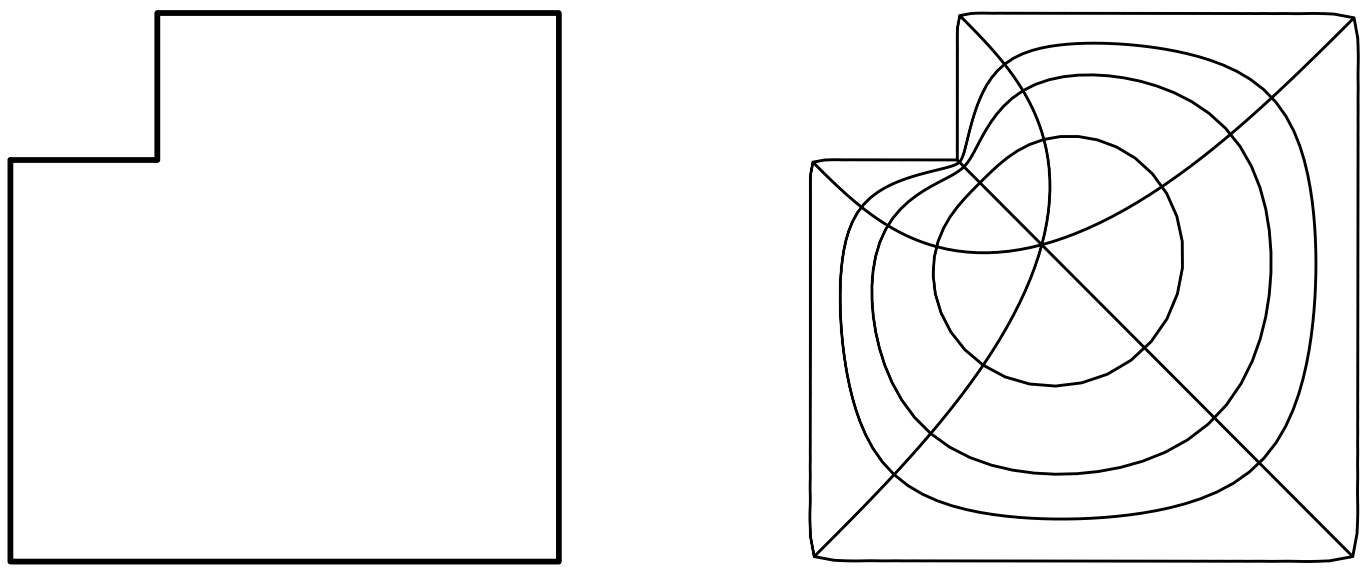
20 terms

Conformal maps have power series, but corners of polygon create singularites on circle. Convergence is slow.



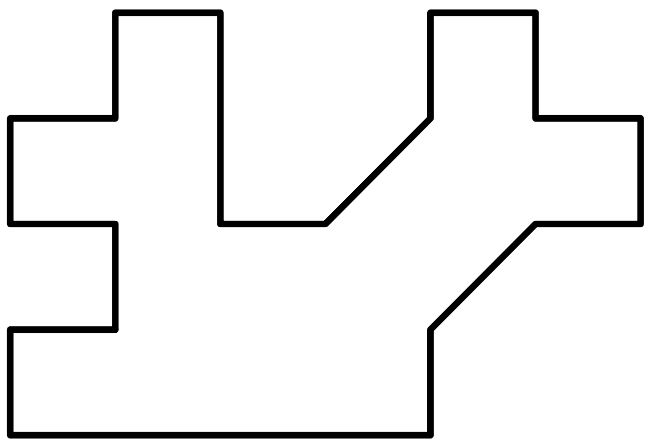
100 terms

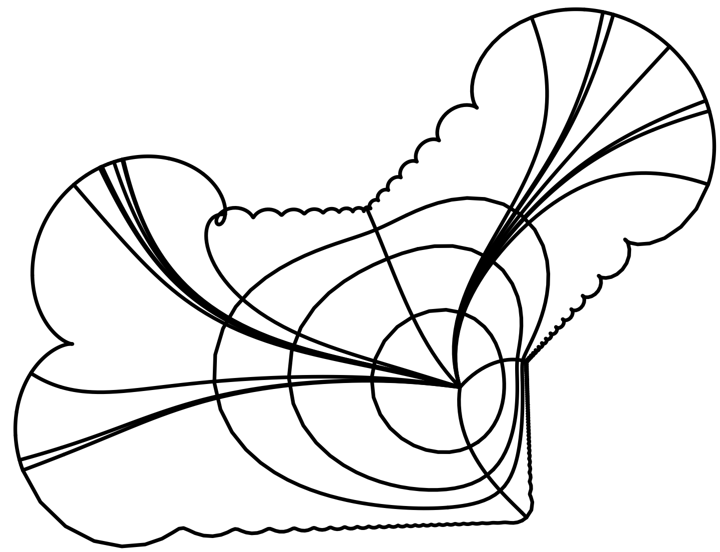
Conformal maps have power series, but corners of polygon create singularites on circle. Convergence is slow.



1000 terms

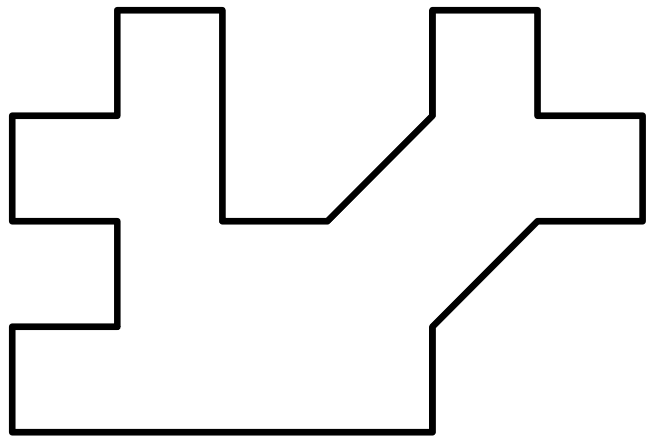
Conformal maps have power series, but corners of polygon create singularites on circle. Convergence is slow.

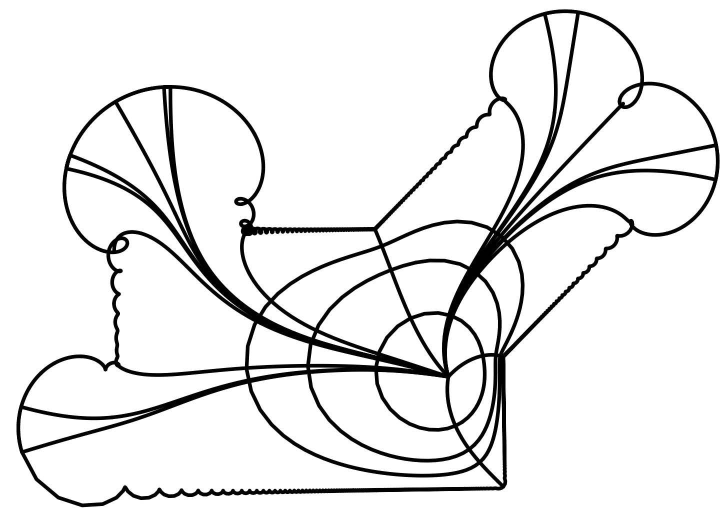




100 terms

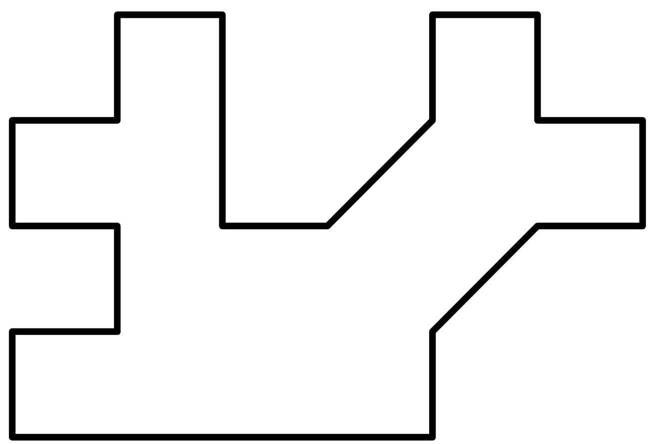
Conformal maps have power series, but corners of polygon create singularites on circle. Convergence is slow.

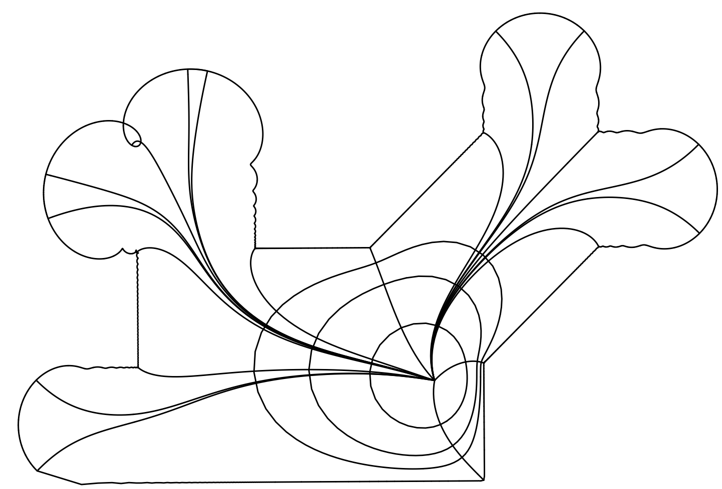




500 terms

Conformal maps have power series, but corners of polygon create singularites on circle. Convergence is slow.





2500 terms

1×20 rectangle would require about 10^{15} terms.

Schwarz-Christoffel representation:

$$f(z) = A + C \int^{z} \prod_{k=1}^{n} (1 - \frac{w}{z_k})^{\alpha_k - 1} dw,$$

$\{\alpha_1 \pi, \ldots, \alpha_n \pi\}$, are interior angles of polygon.
$\{z_1, \ldots, z_n\}$ are points on circle mapping to vertices.

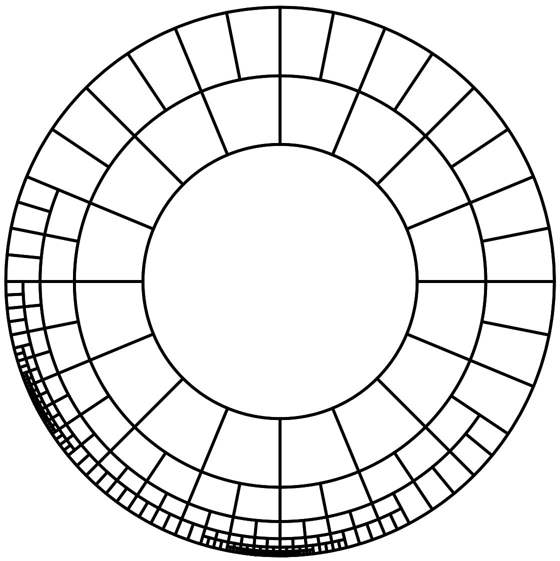
α's are known.

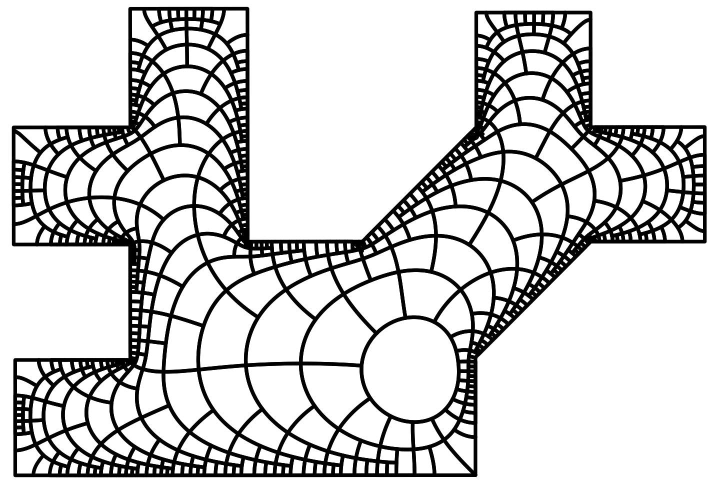
z's must be solved for.

Each evaluation of integrand is n-fold product. How many evaluations to compute integral acurately?

Recall we have a $O(n)$ time bound.

Local series representation: We cut disk into $O(n)$ regions and use a p-term series on each piece to approximate map with accuracy $\epsilon \approx 2^{-p}$ in hyperbolic metric.





Use partition of unity to get global map.

Easy to evaluate; just plug in.

Schwarz-Christoffel always gives conformal map, but onto wrong polygon if z-parameters are only approximate.

Hard to understand relationship between parameters and image domain, so hard to update parameters in provably correct way (many unproven heuristics which seem to work in practice, e.g., Davis, CRDT.)

Local series give map onto correct domain, but are not conformal if series are approximate.

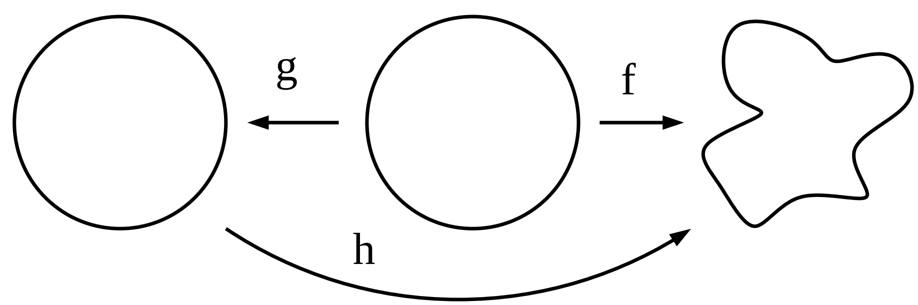
Easy to make a map conformal and preserve image.

$$\partial f = \frac{1}{2}(f_x - i f_y), \quad \overline{\partial} f = \frac{1}{2i}(f_x + i f_y).$$

We want $f : \mathbb{D} \to \Omega$ with $\overline{\partial} f = 0$.
We measure distance to conformality by

$$\|f\| = \sup |\mu_f| = \sup |\overline{\partial} f / \partial f|.$$

Main point: If $f : \mathbb{D} \to \Omega$, $g : \mathbb{D} \to \mathbb{D}$ and $\mu_g = \mu_f$, then $h = f \circ g^{-1} : \mathbb{D} \to \Omega$ is conformal.



Given f set $g = P[\mu(h+1)] + z$, where

$$h = T\mu + T\mu T\mu + T\mu T\mu T\mu + \ldots,$$

T is the Beurling transform

$$Th(w) = \lim_{r \to 0} \frac{1}{\pi} \iint_{|z-w|>r} \frac{h(z)}{(z-w)^2} dxdy,$$

P is the Cauchy integral

$$Ph(w) = -\frac{1}{\pi} \iint h(z) \left(\frac{1}{z-w} - \frac{1}{z}\right) dxdy.$$

Then $f \circ g^{-1} : \mathbb{D} \to \Omega$ is conformal.

If f has local representation by $O(n)$ p-term series, we can compute a g in time $O(np\log p)$ so that

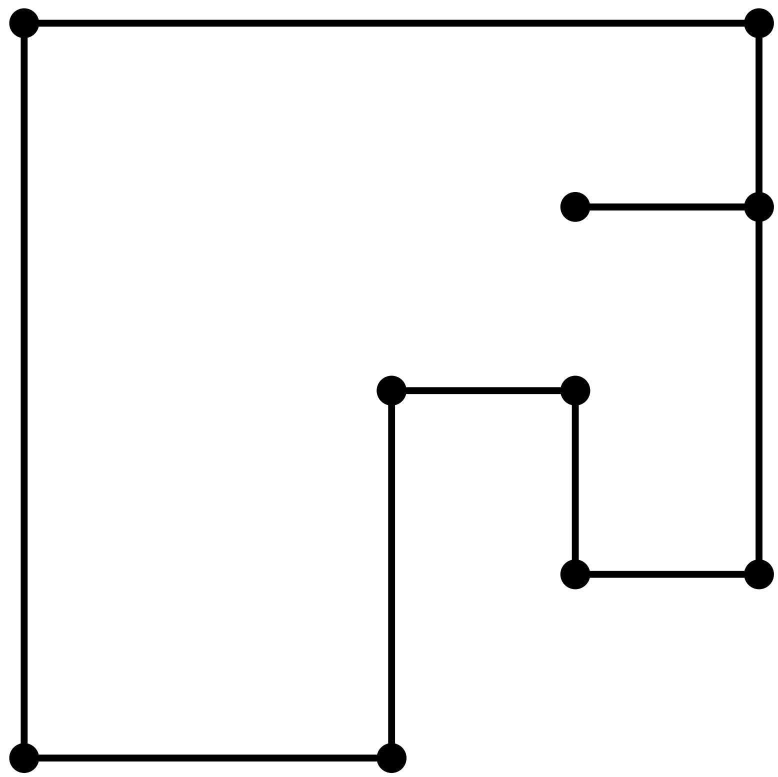
$$\|f \circ g^{-1}\| = O(\|f\|^2).$$

Iteration gives quadratic convergence to conformal map.

Uniform time bound needs good starting map.

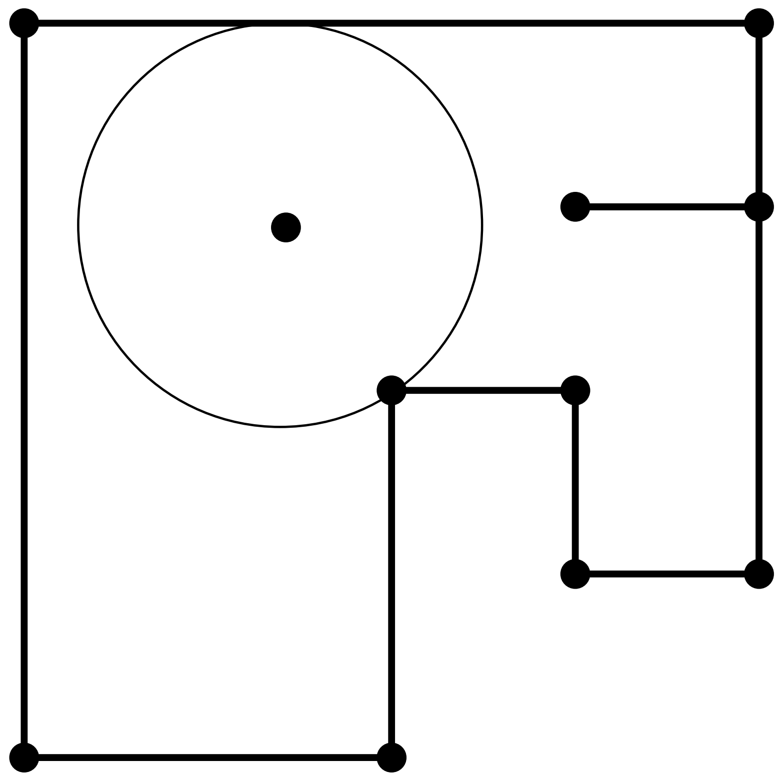
Starting map must be bounded distance to conformal.

Initial guess comes from **iota map**.



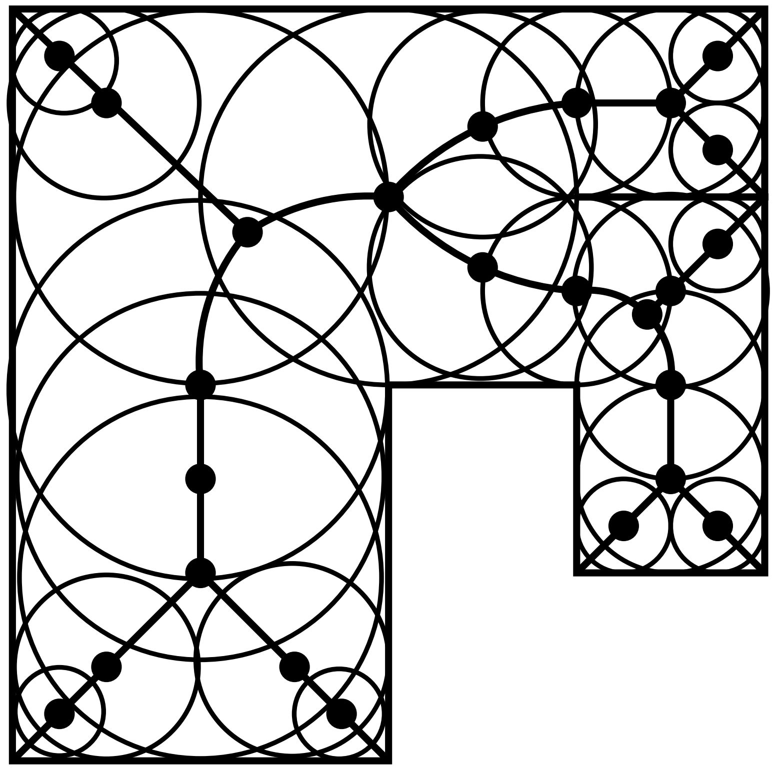
Start with a polygon.

Initial guess comes from **iota map**.



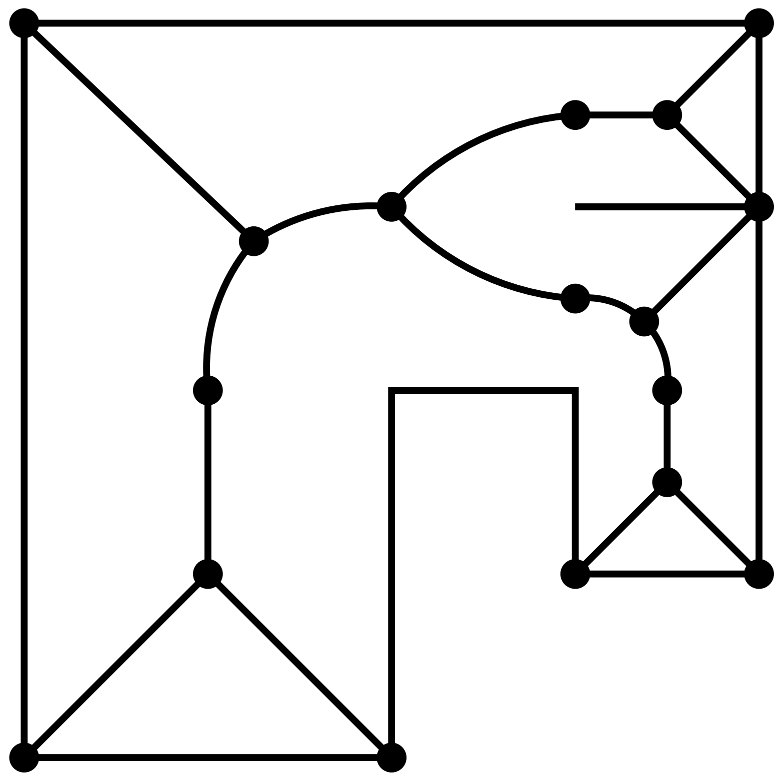
Consider interior disks with ≥ 2 contacts on boundary.

Initial guess comes from **iota map**.



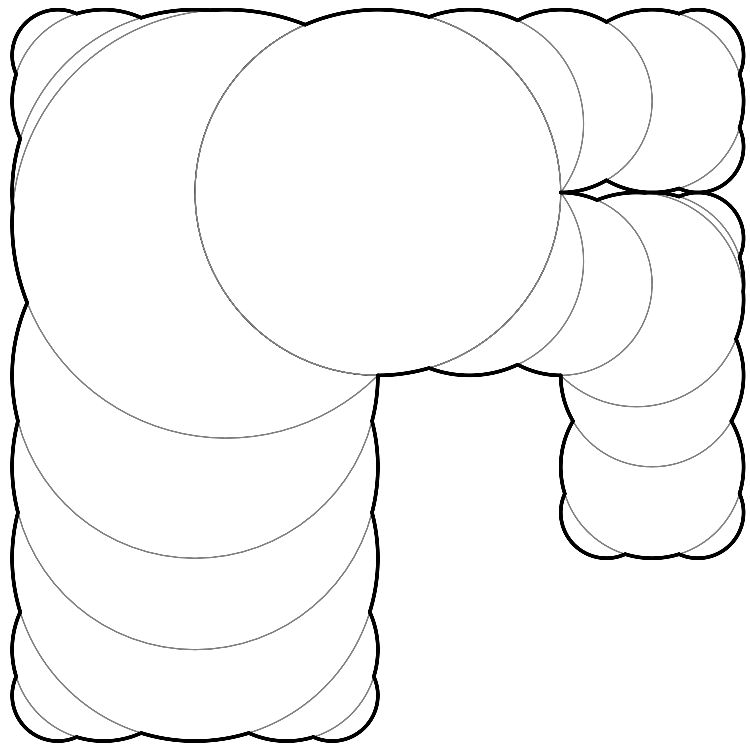
All such disks define **medial axis**.

Initial guess comes from **iota map**.



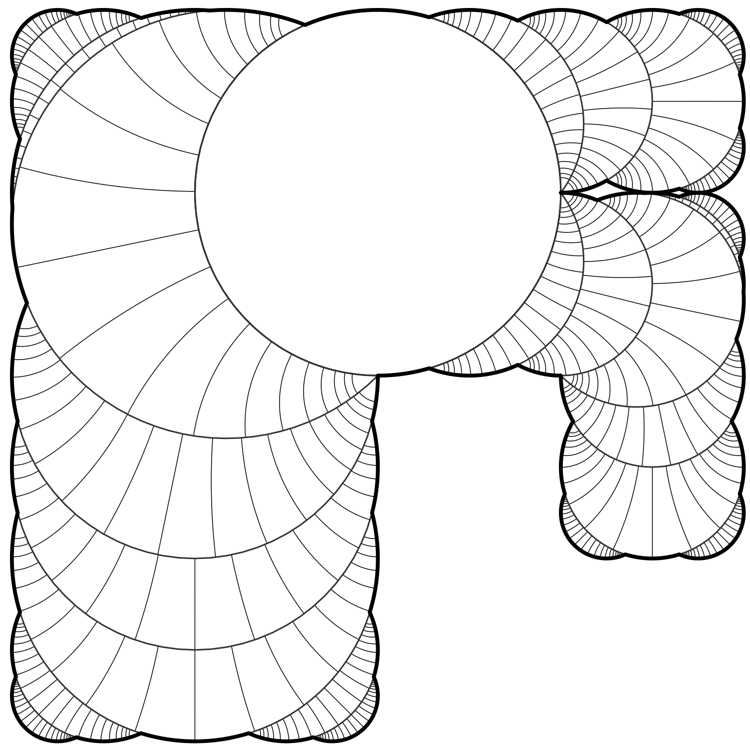
All such disks define **medial axis**.

Initial guess comes from **iota map**.



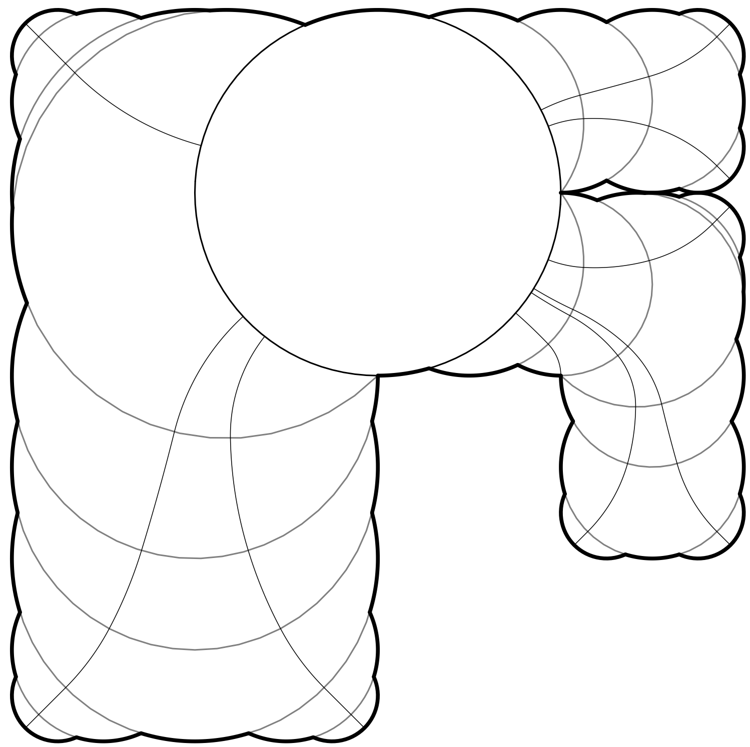
Take a finite set of medial axis disks.

Initial guess comes from **iota map**.



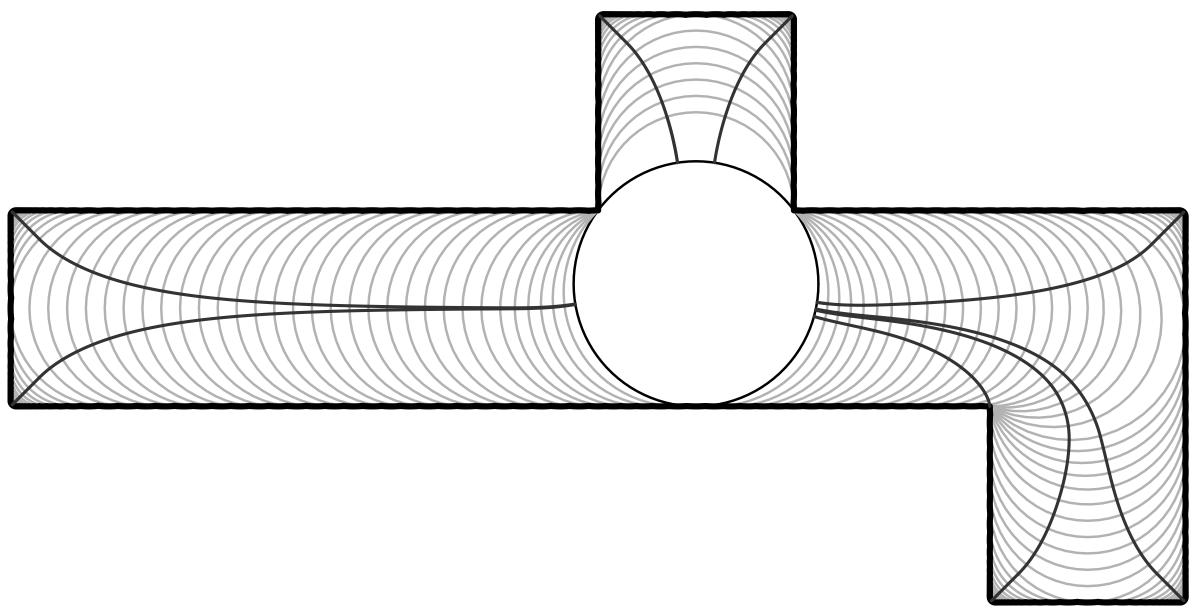
Foliate crescents by orthogonal arcs.

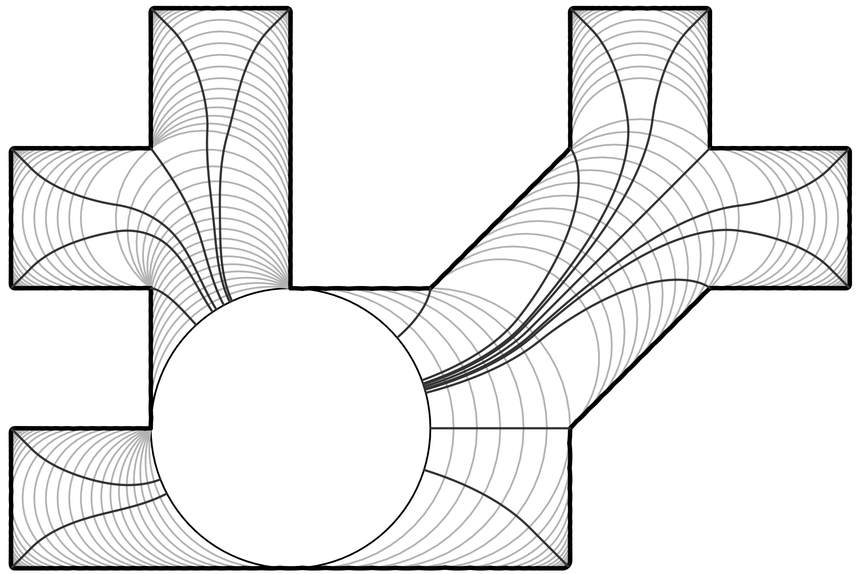
Initial guess comes from **iota map**.



Follow arcs to define map of boundary to circle.

Medial Axis Flow = iota map: Take limit to get explicit formulas for n-gons. Images of vertices computable in $O(n)$.



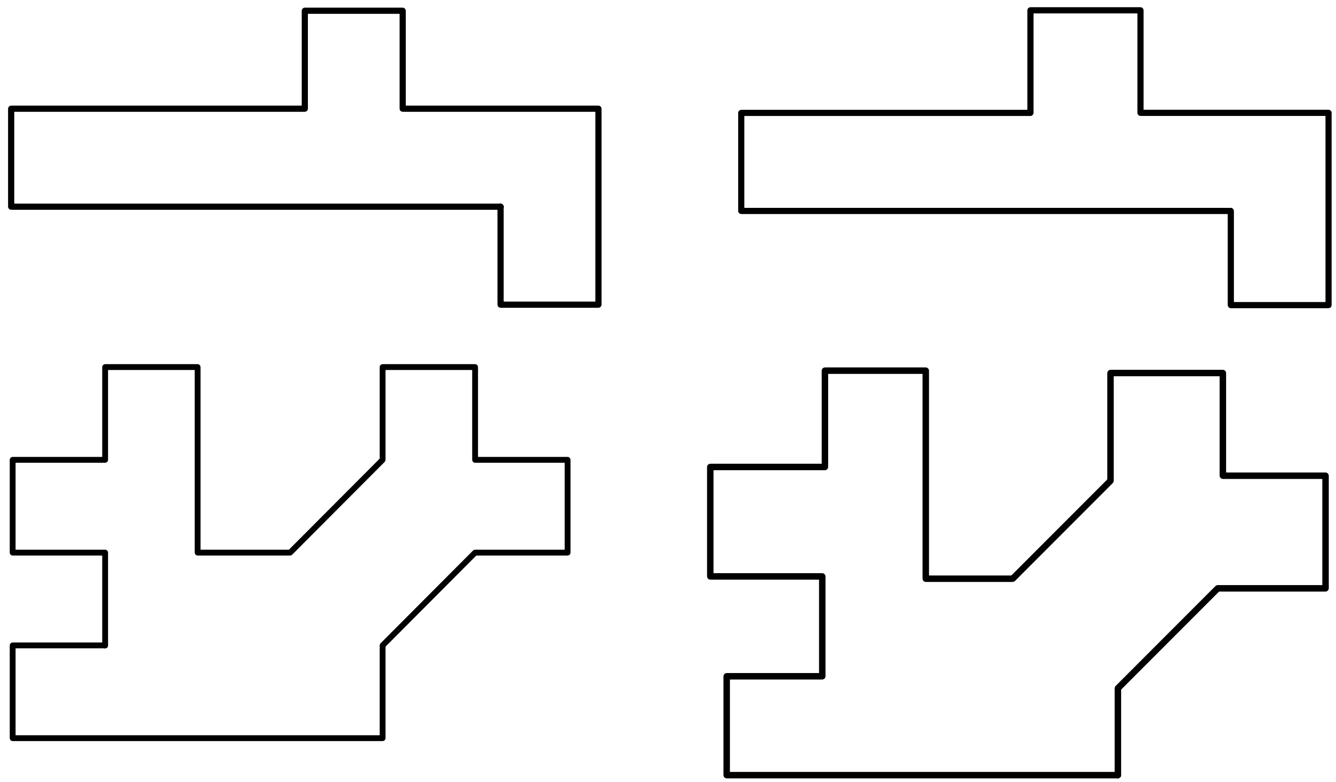


Theorem: Iota on boundary extends to interior map f with $|\mu_f| < k < 1$, universal k.

Iota gives good starting point for iteration. Only need $O(\log\log\frac{1}{\epsilon})$ iterations to attain accuracy ϵ.

Also "good enough" to compute thin parts in $O(n)$.

How close is iota map to conformal? Try using "iota parameters" in Schwarz-Christoffel formula.



Target Polygon Iota Parameters

Suffices to show nearest point map onto certain convex sets in hyperbolic 3-space is bi-Lipschitz at large scales.

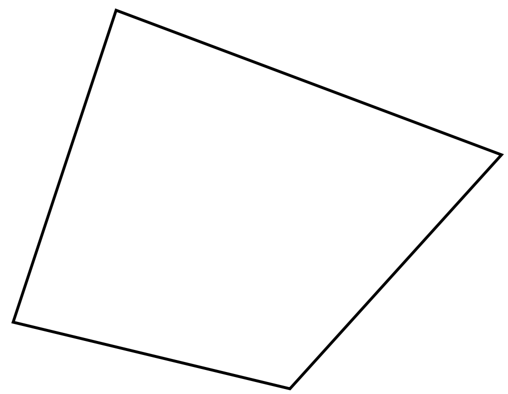
- short proof of special case, Sullivan, 1981
- long proof of general case, Epstein and Marden, 1985
- short proof of general case, B, 2001

I worked on this to compute fractal dimension of the set of directions in a hyperbolic manifold that corresponded to bounded geodesic rays.

Connection to conformal maps and meshing came later.

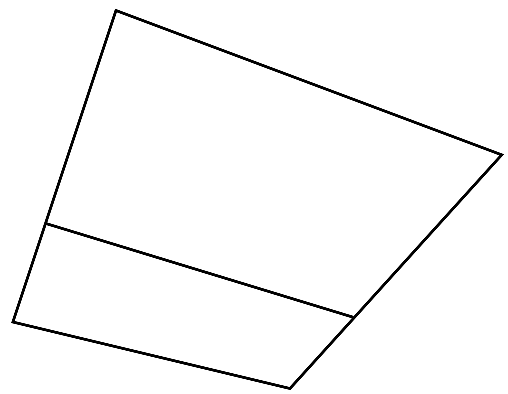
Some special constructions needed to merge thick and thin meshes.

Vertices propagate via linear interpolation.



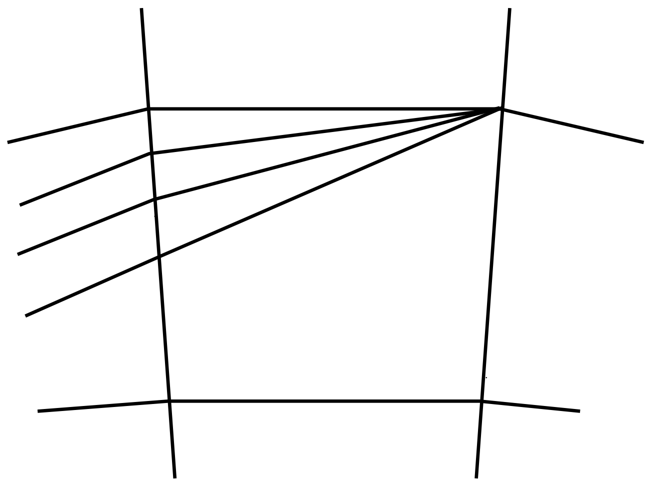
Some special constructions needed to merge thick and thin meshes.

Vertices propagate via linear interpolation.

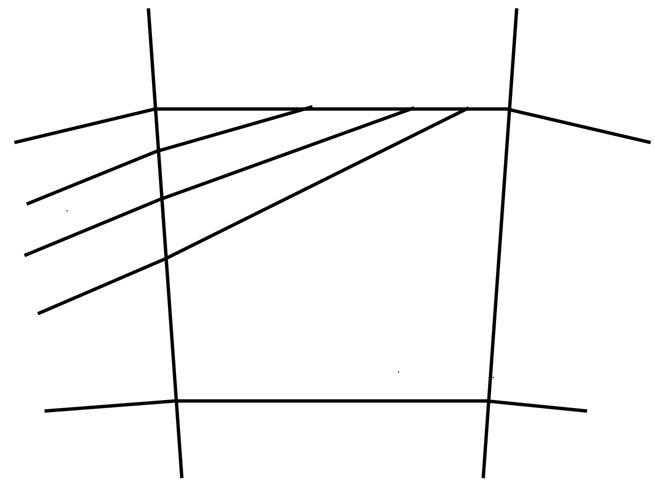


This preserves angle bounds.

In thin parts propagations paths are “bent” to hit tube wall instead of vertex.

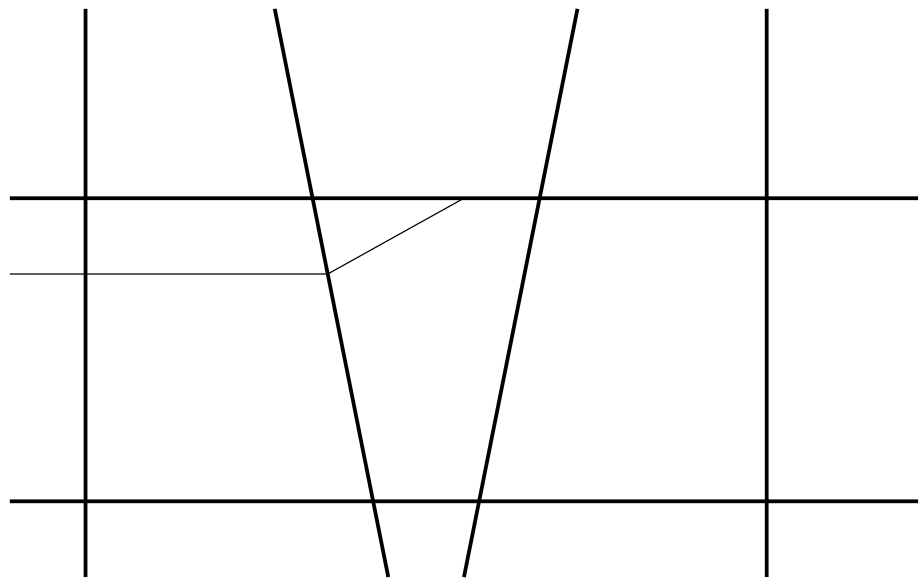


nonobtuse triangulation

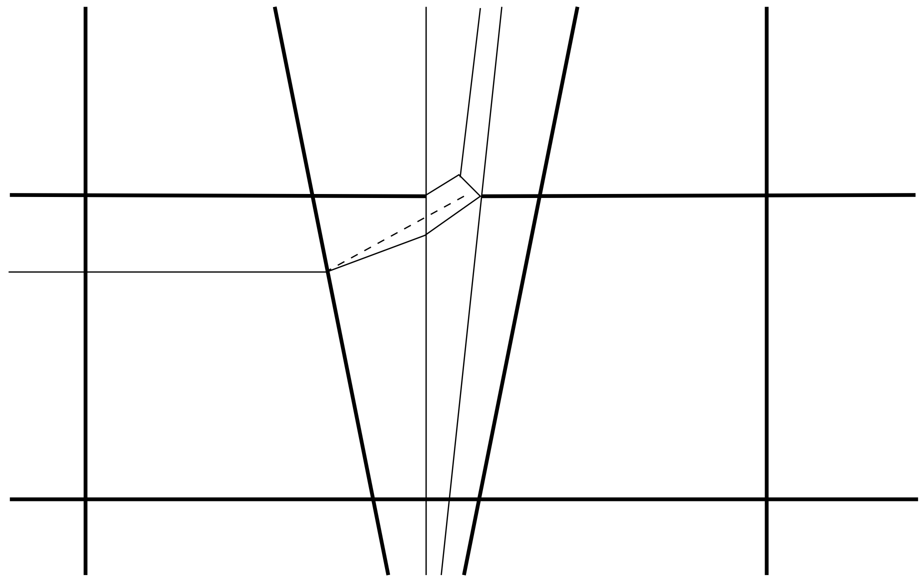


quadrilateral meshing

Consider one such line.

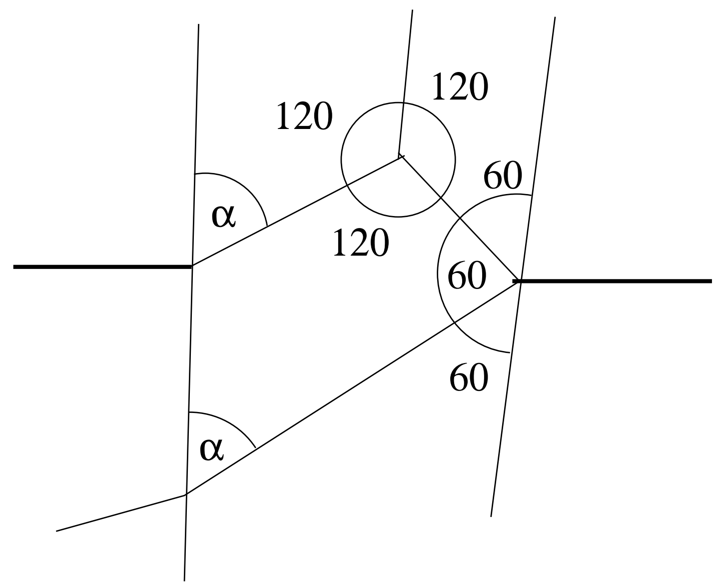


Consider one such line.



Construction "bends" line by about 90°. Path continues until it hits thick part.

Consider one such line.



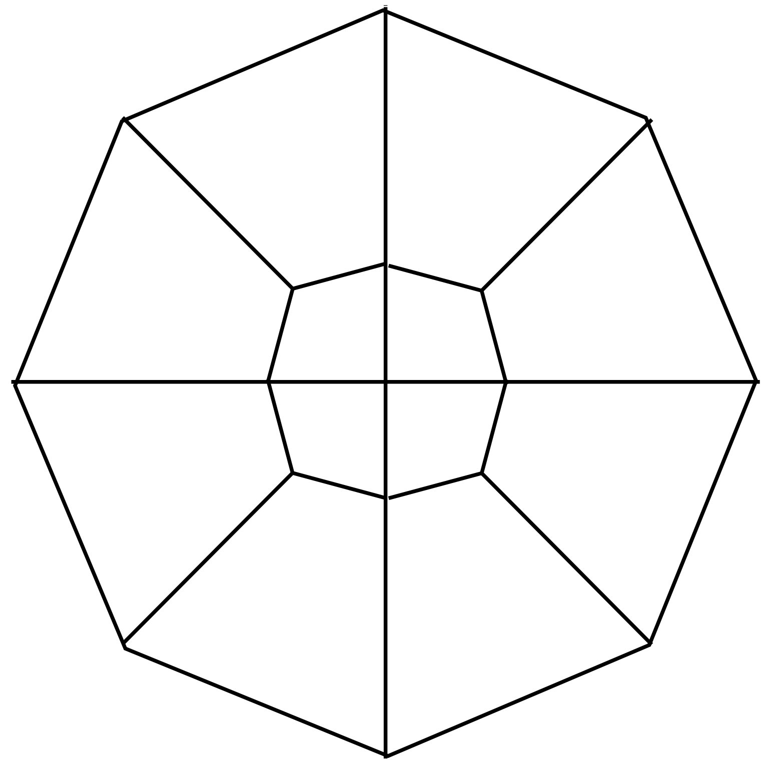
Enlargment of bending construction.

A **sink** is a polygon so that if we add an even number of vertices to boundary, it can quad meshed with angles in $[60^\circ, 120^\circ]$ and no new boundary vertices.

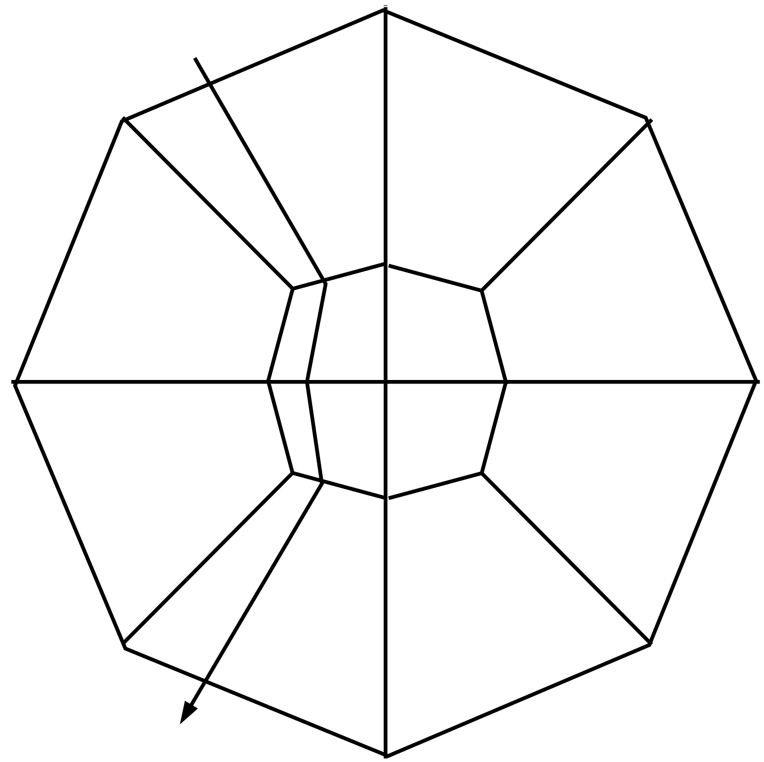
A **sink** is a polygon so that if we add an even number of vertices to boundary, it can quad meshed with angles in $[60^\circ, 120^\circ]$ and no new boundary vertices.

A simple polygon with a quad mesh must have an even number of boundary vertices.

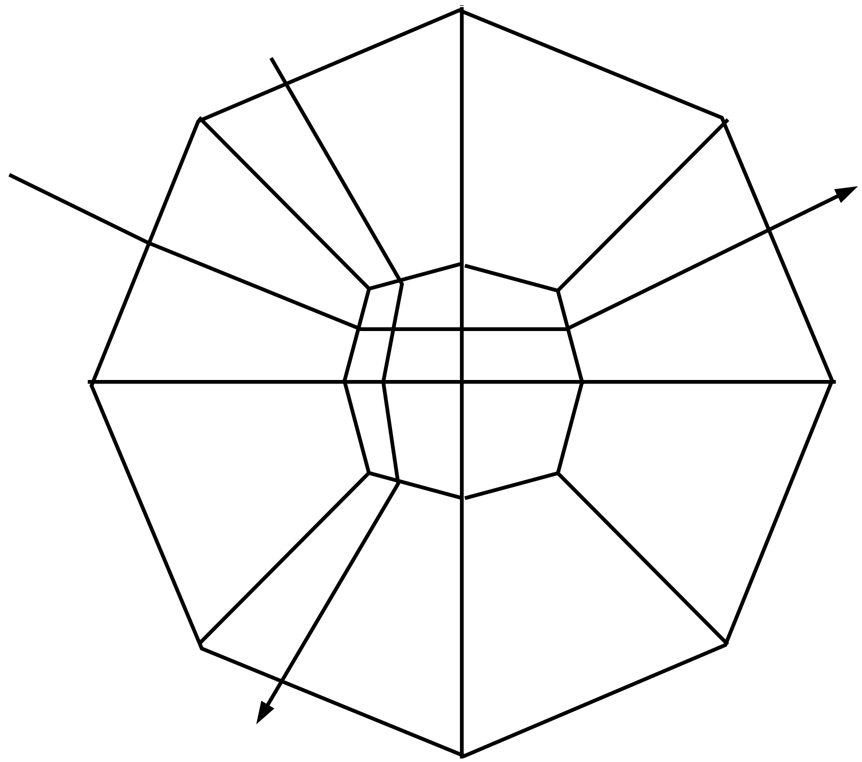
Lemma: An octagon is a sink.



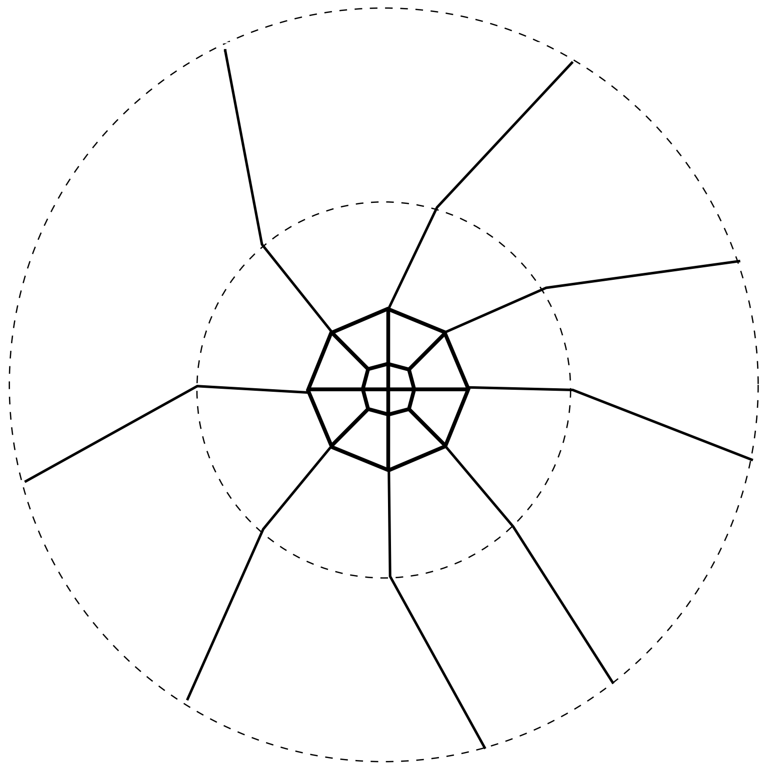
Lemma: An octagon is a sink.



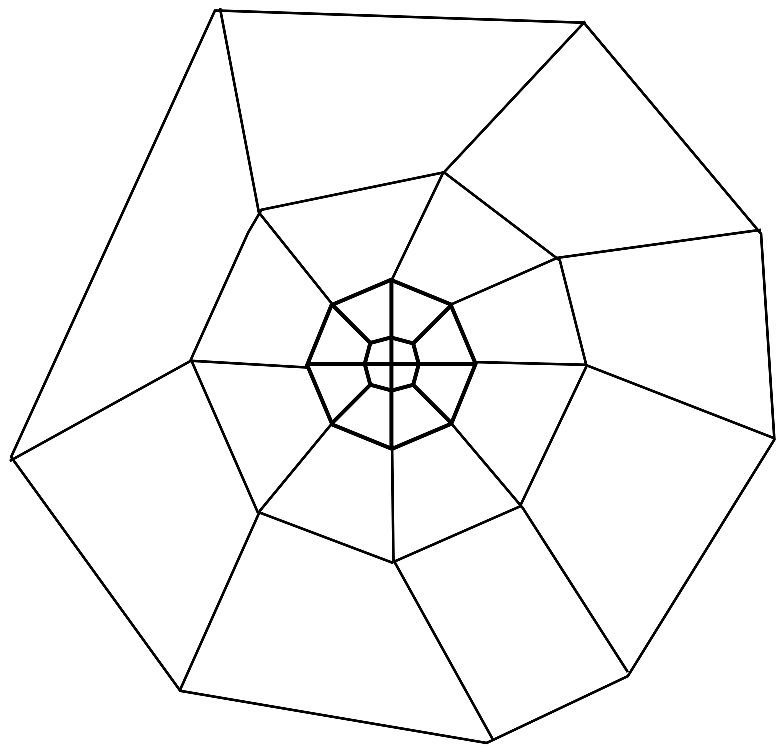
Lemma: An octagon is a sink.



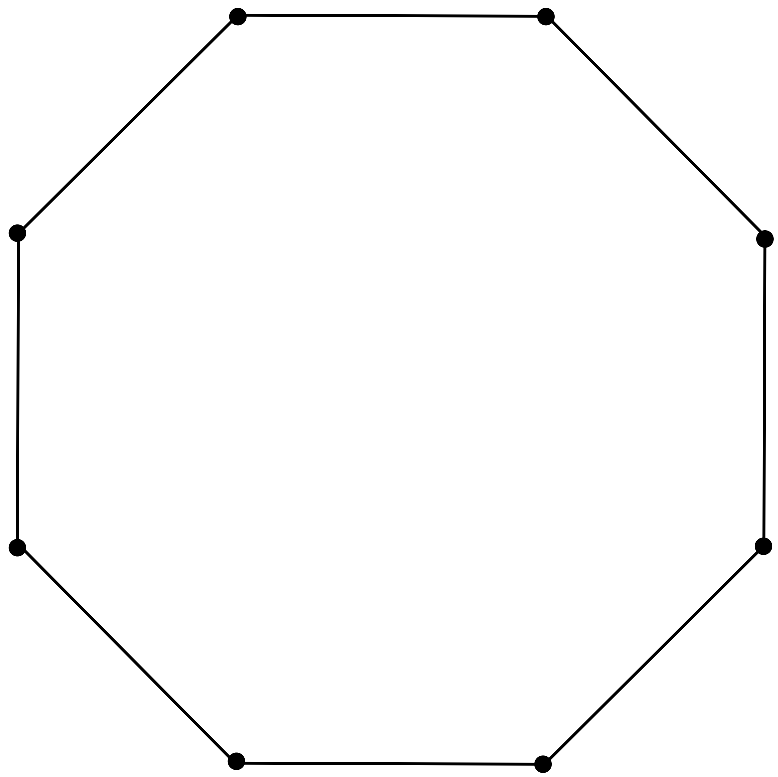
Lemma: An octagon is a sink.



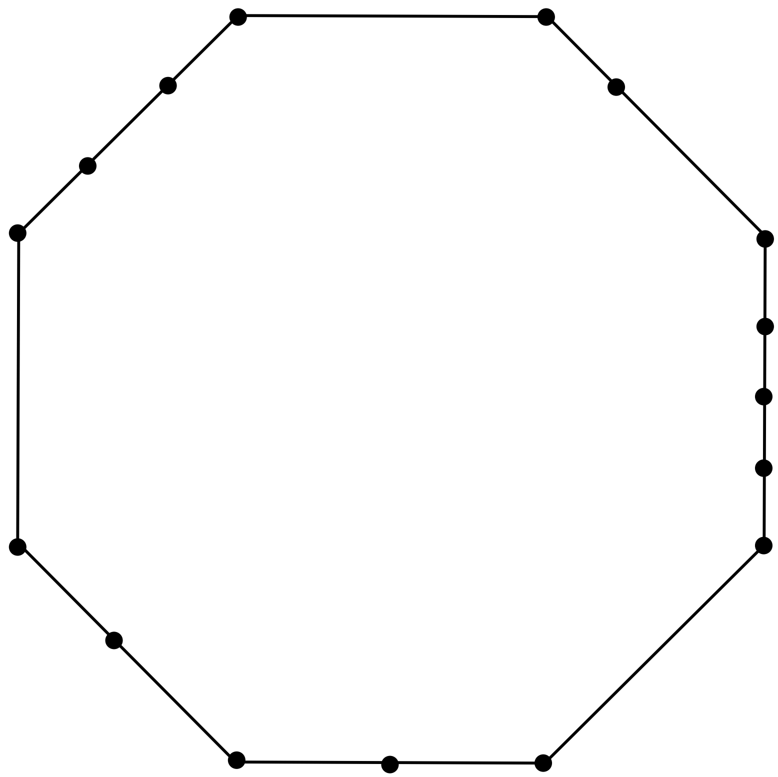
Lemma: An octagon is a sink.



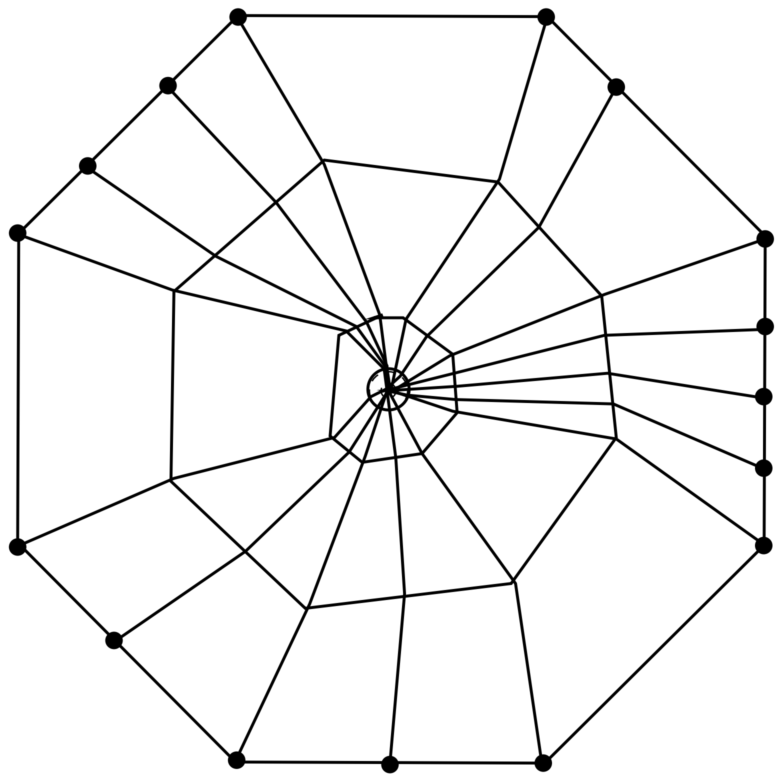
Lemma: An octagon is a sink.



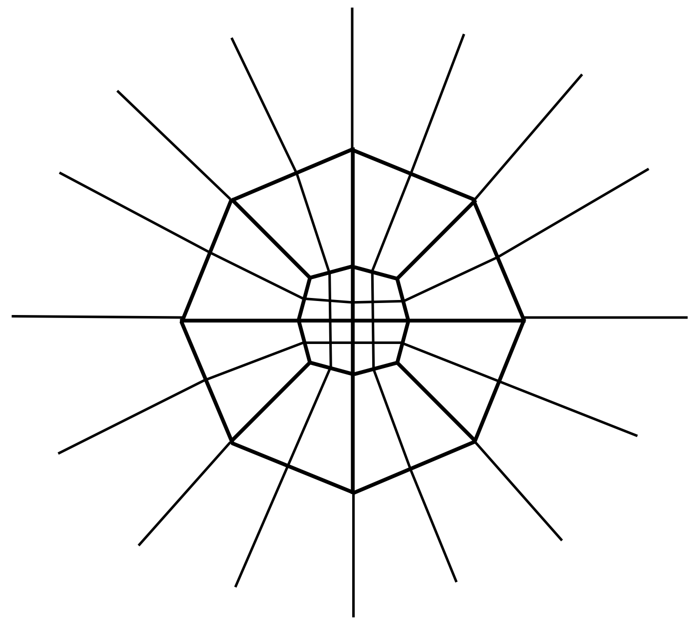
Lemma: An octagon is a sink.



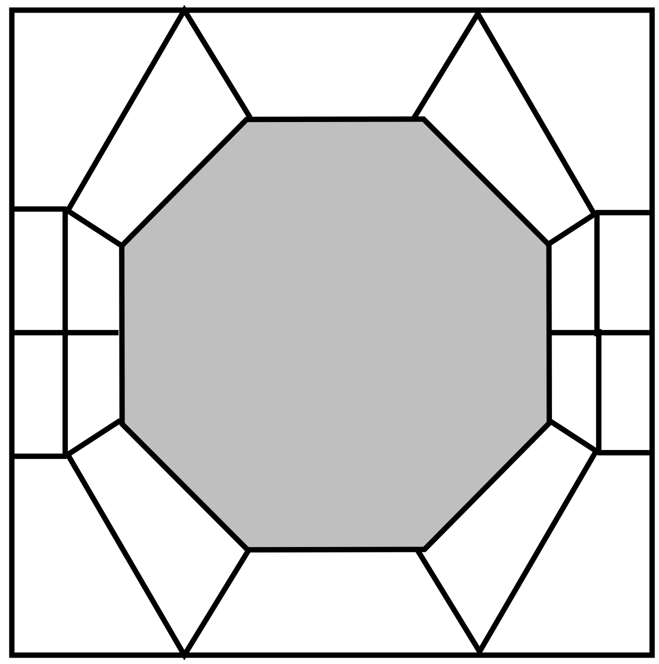
Lemma: An octagon is a sink.



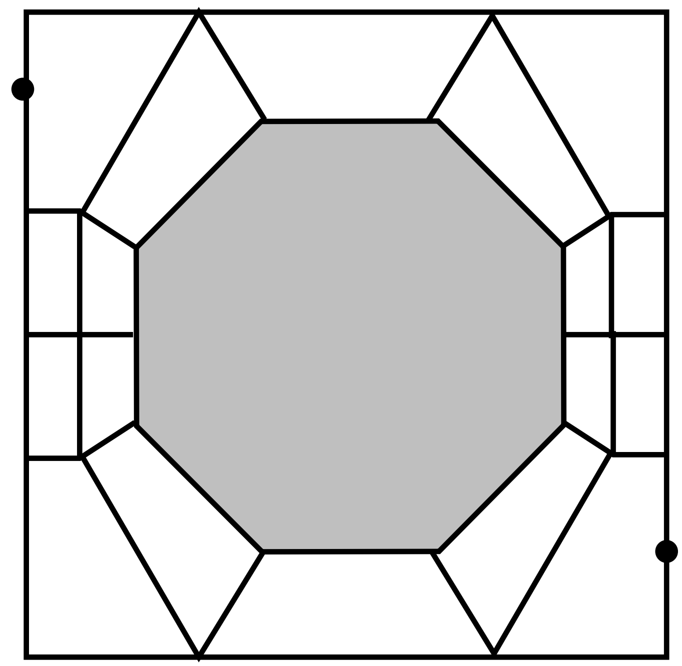
Lemma: An octagon is a sink.



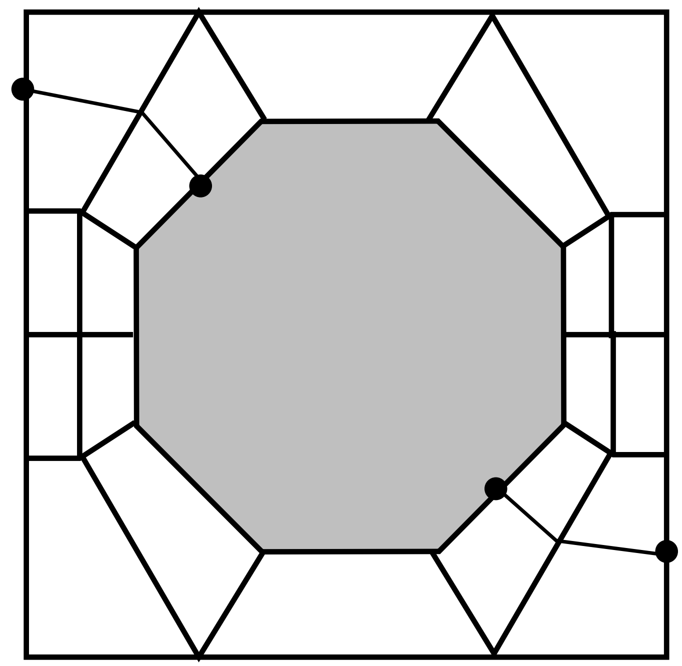
Lemma: Squares are sinks.



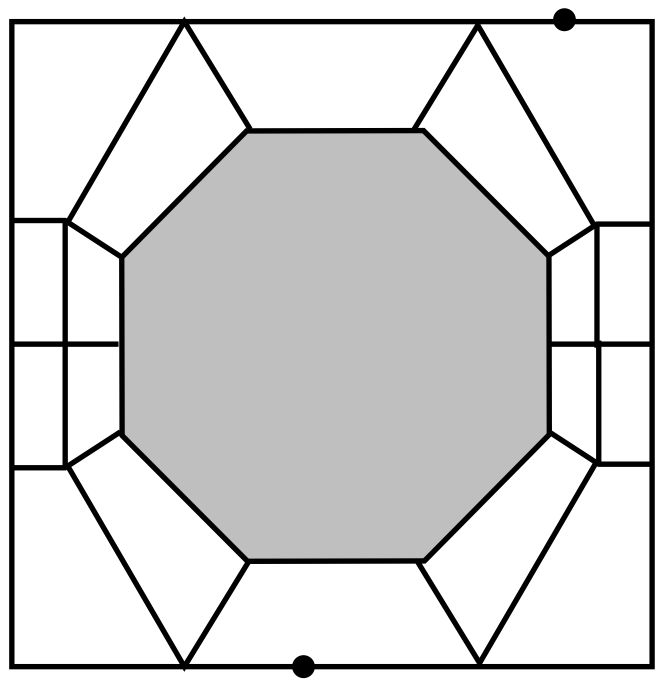
Lemma: Squares are sinks.



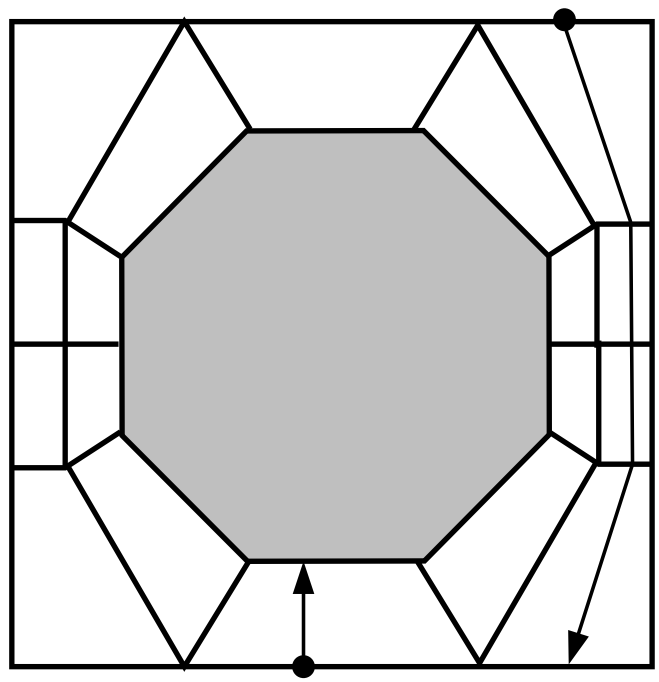
Lemma: Squares are sinks.



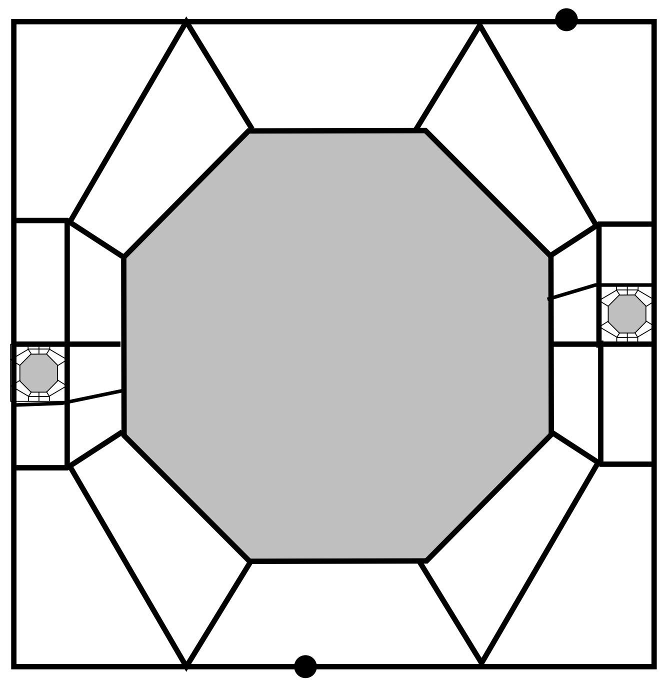
Lemma: Squares are sinks.



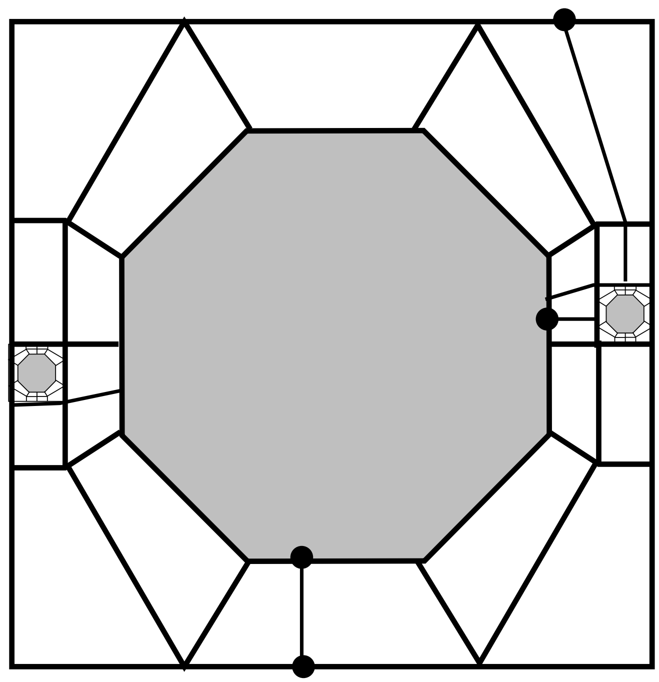
Lemma: Squares are sinks.



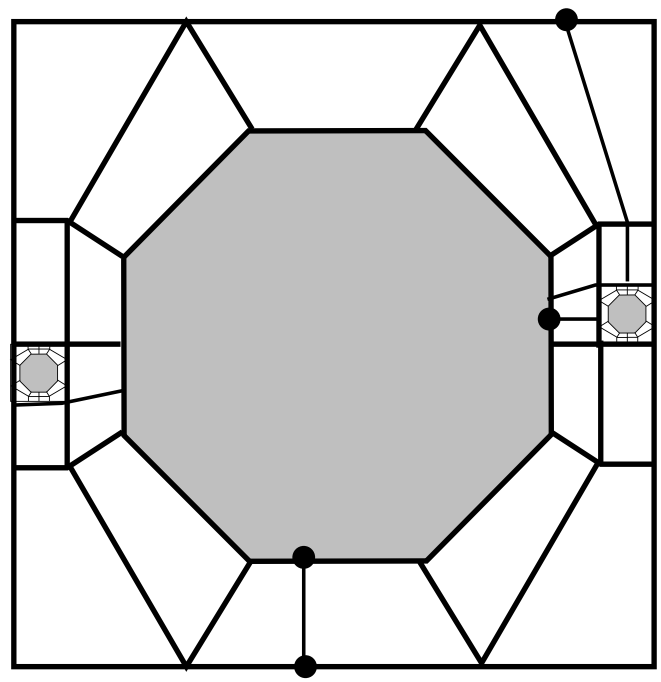
Lemma: Squares are sinks.



Lemma: Squares are sinks.



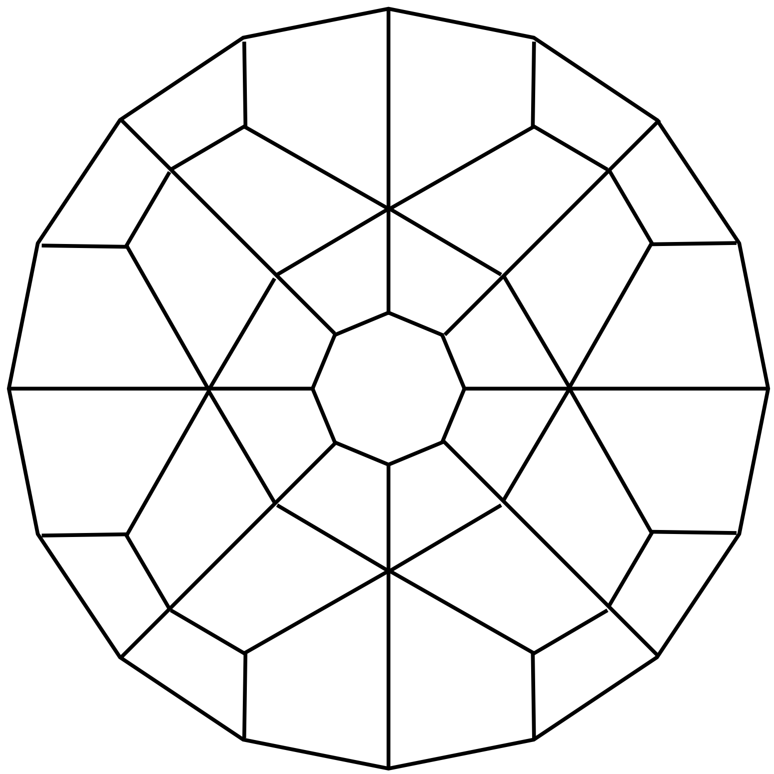
Lemma: Squares are sinks.



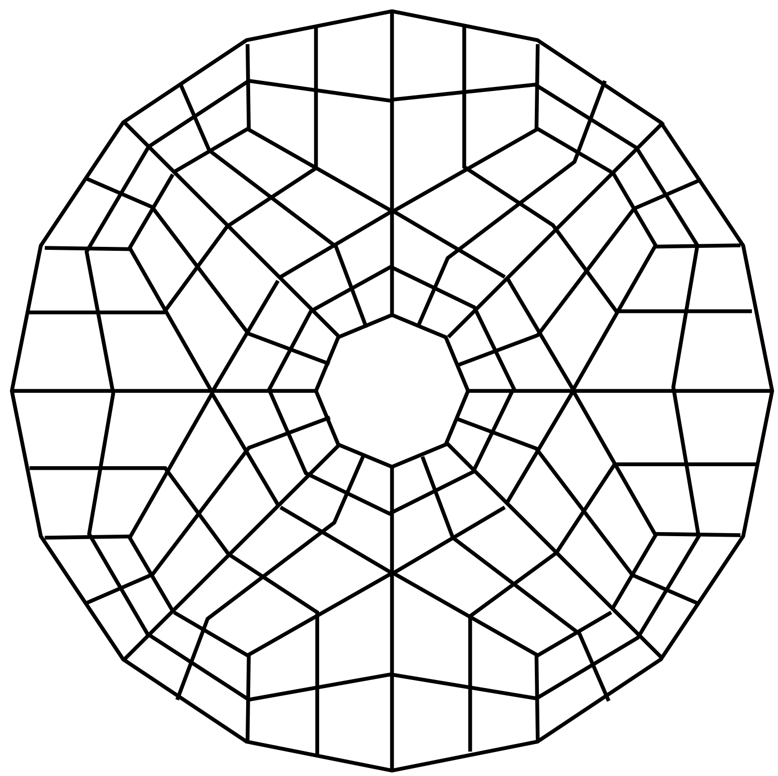
Lemma: Any quad with angles in $[60°, 120°]$ is a sink.

So after meshing thick parts, they can "absorb" propagation lines from thin parts, but need even number per quad.

Every quad mesh can be doubled by connecting midpoints.



Every quad mesh can be doubled by connecting mid-points.



Doubling allows us to insure every sink has a even number of boundary vertices.