Computing an ε-net of a hyperbolic surface

Camille Lanuel
Université de Lorraine, CNRS, Inria, LORIA, Nancy, France

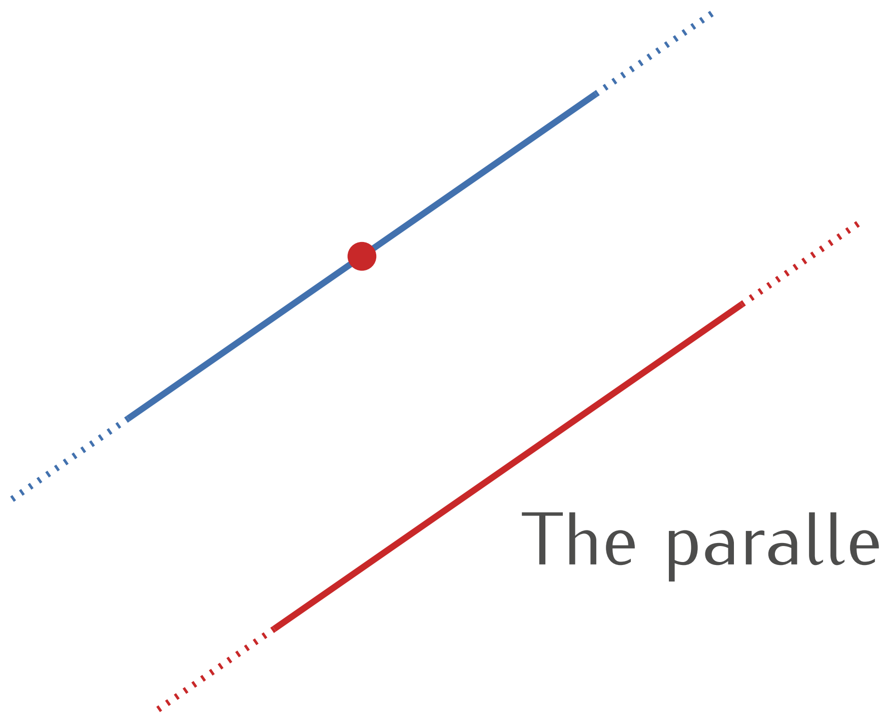
26 novembre 2025

1. **Introduction**
2. The ε-net algorithm
3. Implementation
4. Conclusion

Hyperbolic geometry

Axioms of Euclidean geometry

1. There is one and only one line segment between any two given points;
2. Any line segment can be extended continuously to a line;
3. There is one and only one circle with any given center and any given radius;
4. All right angles are congruent to one another;
5. **(Parallel postulate)** Given a line and a point not on the line, there is *exactly one* line through the point that is parallel to the given line.



The parallel postulate

Hyperbolic geometry

Axioms of ~~Euclidean~~ hyperbolic geometry

1. There is one and only one line segment between any two given points;
2. Any line segment can be extended continuously to a line;
3. There is one and only one circle with any given center and any given radius;
4. All right angles are congruent to one another;
5. **(Parallel postulate)** Given a line and a point not on the line, there is ~~*exactly one*~~ *infinitely many* line through the point that is parallel to the given line.

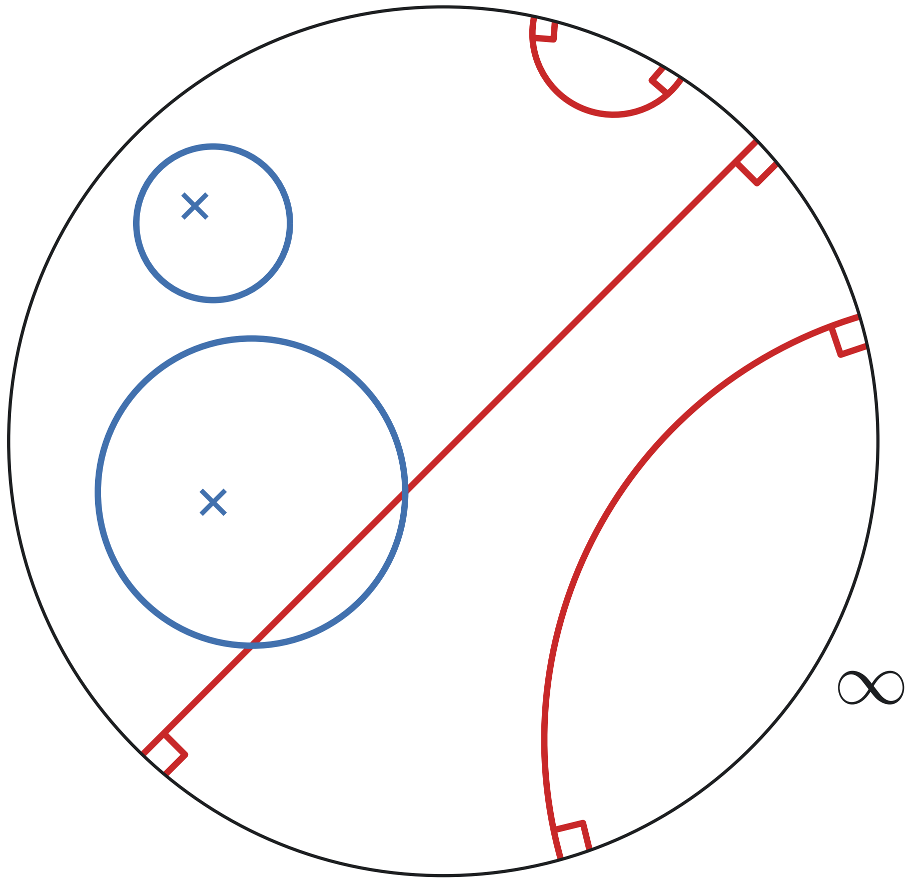
Hyperbolic geometry

Axioms of ~~Euclidean~~ hyperbolic geometry

1. There is one and only one line segment between any two given points;
2. Any line segment can be extended continuously to a line;
3. There is one and only one circle with any given center and any given radius;
4. All right angles are congruent to one another;
5. **(Parallel postulate)** Given a line and a point not on the line, there is ~~*exactly one*~~ *infinitely many* line through the point that is parallel to the given line.

Model of the hyperbolic plane
Simply connected 2-manifold equipped with a metric s.t. the 5 axioms of hyperbolic geometry are satisfied.

The Poincaré disk $\mathbb{H}^2$
open unit disk of $\mathbb{C}$ + metric



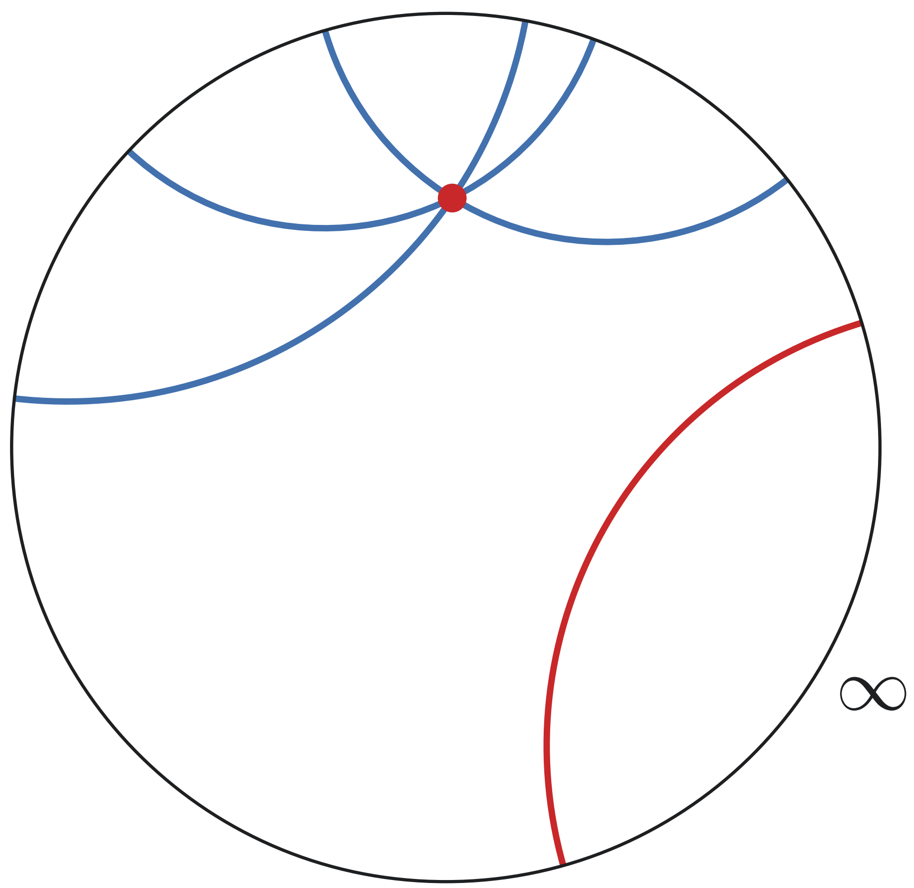
Hyperbolic geometry

Axioms of ~~Euclidean~~ hyperbolic geometry

1. There is one and only one line segment between any two given points;
2. Any line segment can be extended continuously to a line;
3. There is one and only one circle with any given center and any given radius;
4. All right angles are congruent to one another;
5. **(Parallel postulate)** Given a line and a point not on the line, there is ~~*exactly one*~~ *infinitely many* line through the point that is parallel to the given line.

Model of the hyperbolic plane
Simply connected 2-manifold equipped with a metric s.t. the 5 axioms of hyperbolic geometry are satisfied.

The parallel postulate in the Poincaré disk



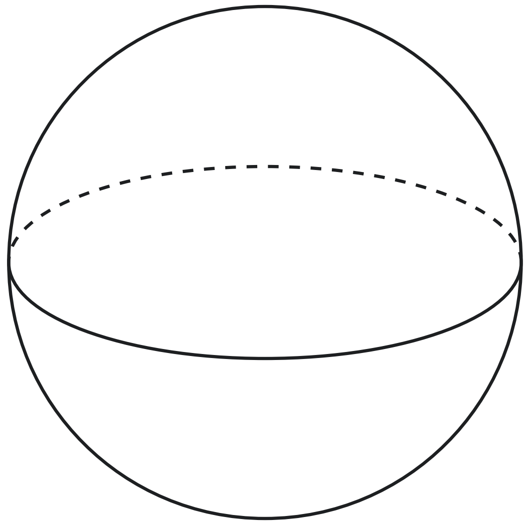
Surface

Surface
Connected 2-manifold.

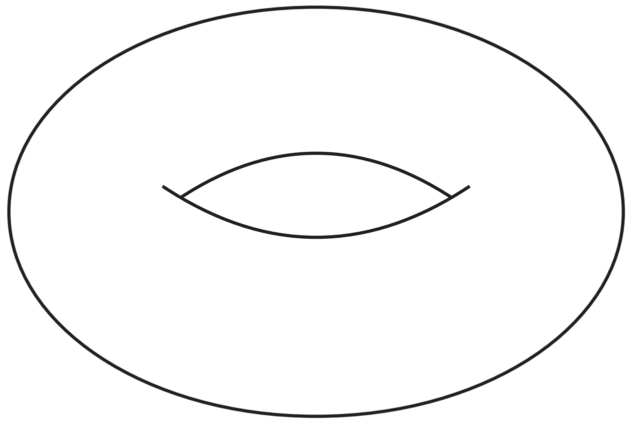
(+ compact, oriented, without boundary)

TOPOLOGY

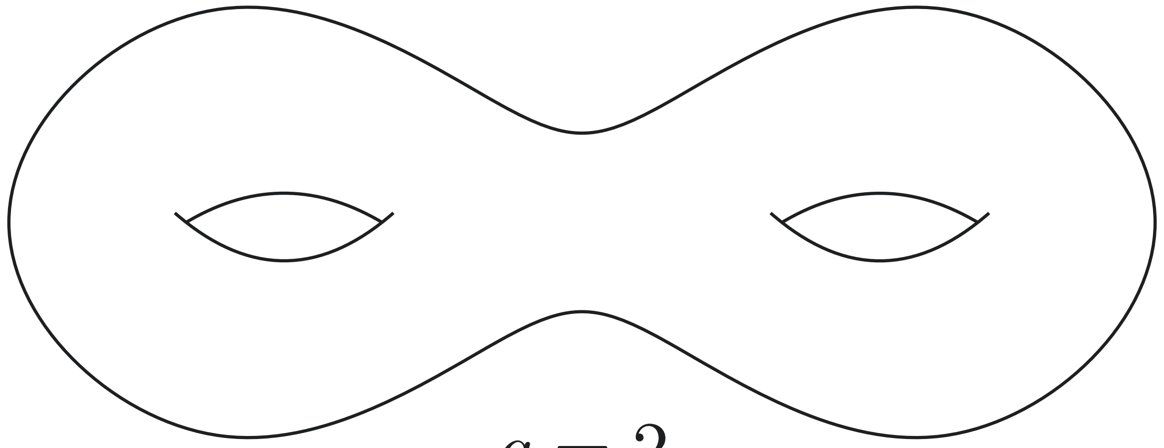
Genus g of a surface
Number of handles (”doughnut holes”).



$g = 0$



$g = 1$



$g = 2$

Surface

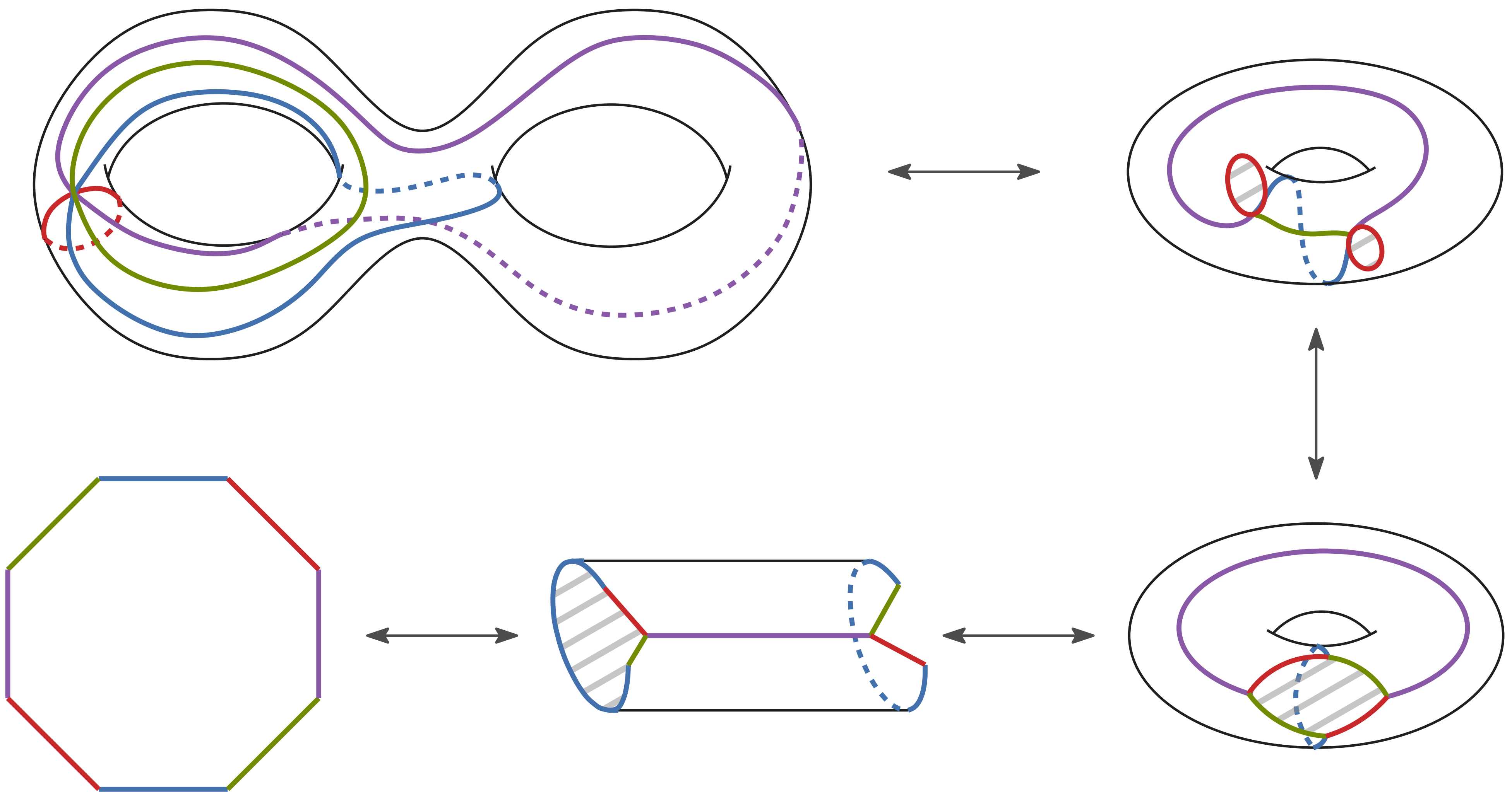
Surface
Connected 2-manifold.

TOPOLOGY

Genus g of a surface
Number of handles ("doughnut holes").

(+ compact, oriented, without boundary)

Every surface can be cut to obtain an oriented polygon called **fundamental polygon**.



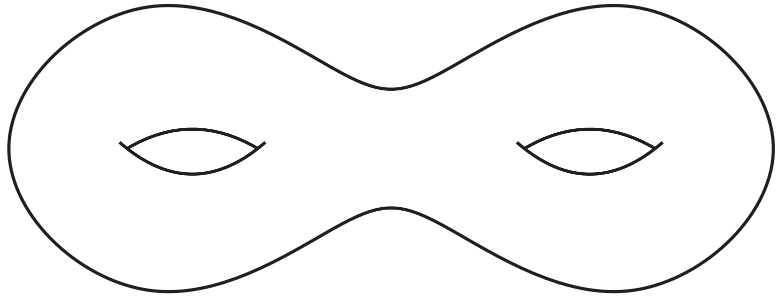
Hyperbolic surface

GEOMETRY

Hyperbolic surface
Surface equipped with a metric s.t. it is locally isometric to $\mathbb{H}^2$ (**hyperbolic metric**).

- Any surface with genus $g \geqslant 2$ admits a hyperbolic metric.
- Impossible to smoothly represent a hyperbolic surface in $\mathbb{R}^3$ while preserving its geometry.

[Hilbert, 1901]



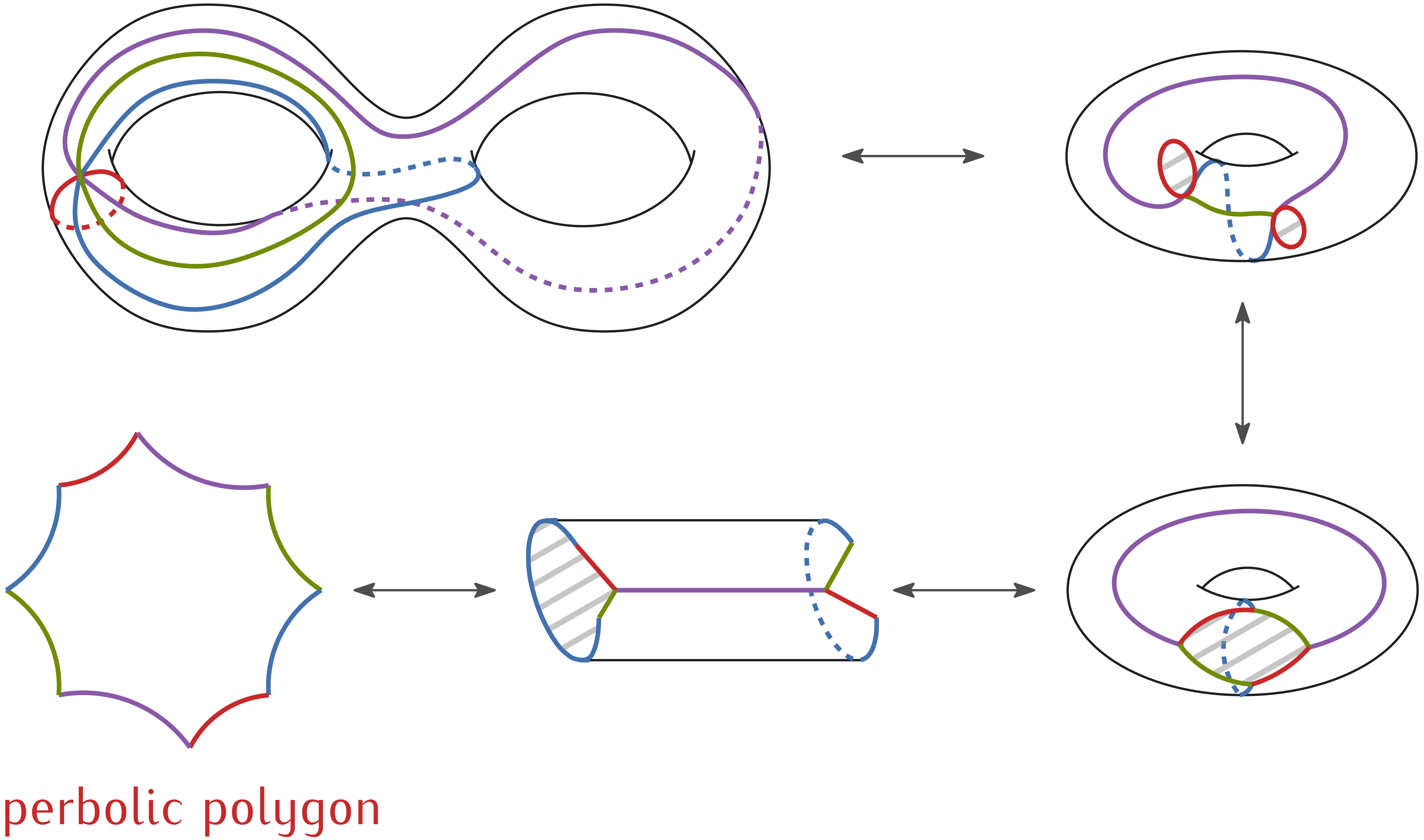
Hyperbolic surface

GEOMETRY

Hyperbolic surface
Surface equipped with a metric s.t. it is locally isometric to $\mathbb{H}^2$ (**hyperbolic metric**).

- Any surface with genus $g \geqslant 2$ admits a hyperbolic metric.
- Impossible to smoothly represent a hyperbolic surface in $\mathbb{R}^3$ while preserving its geometry.

[Hilbert, 1901]

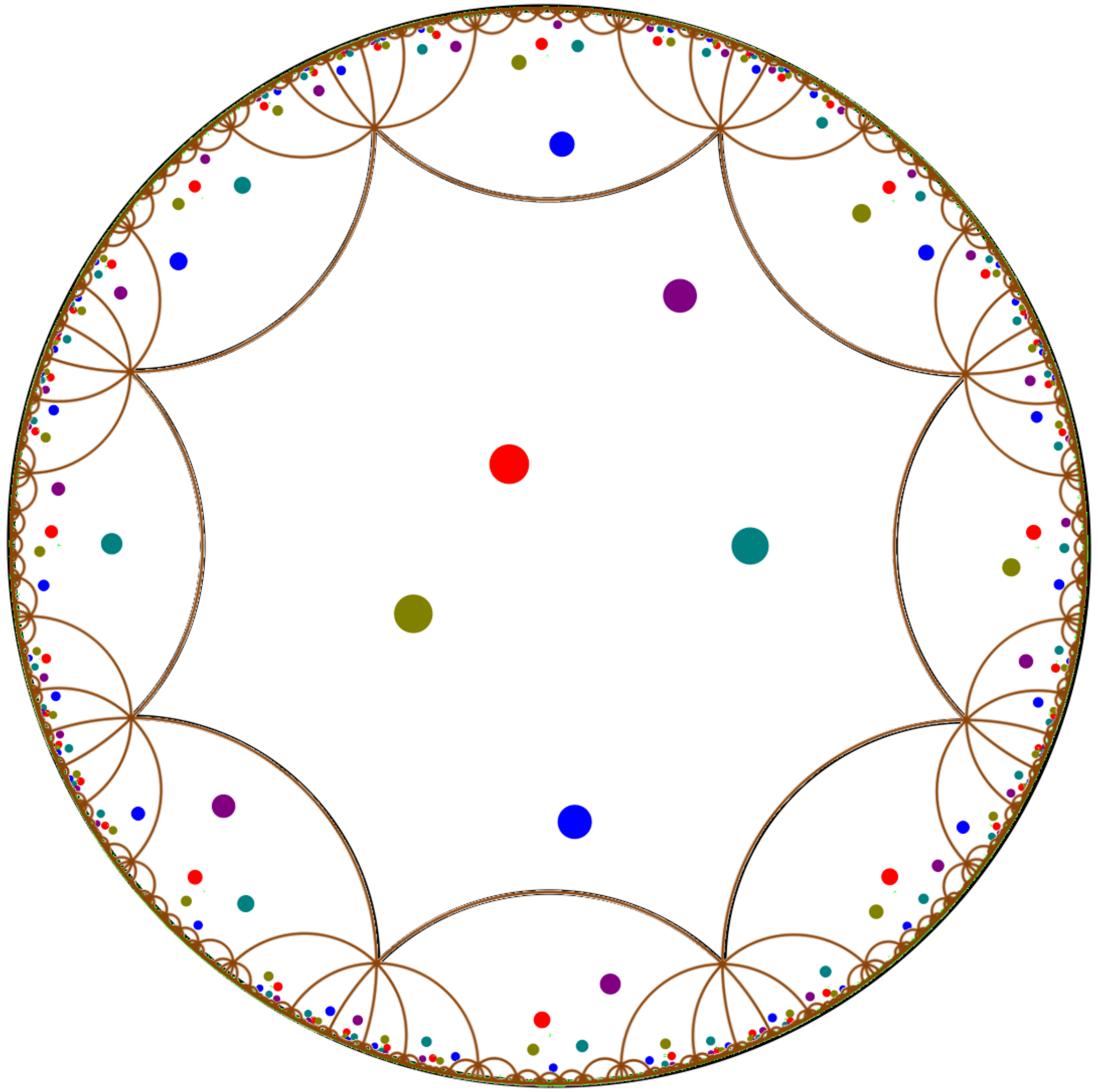


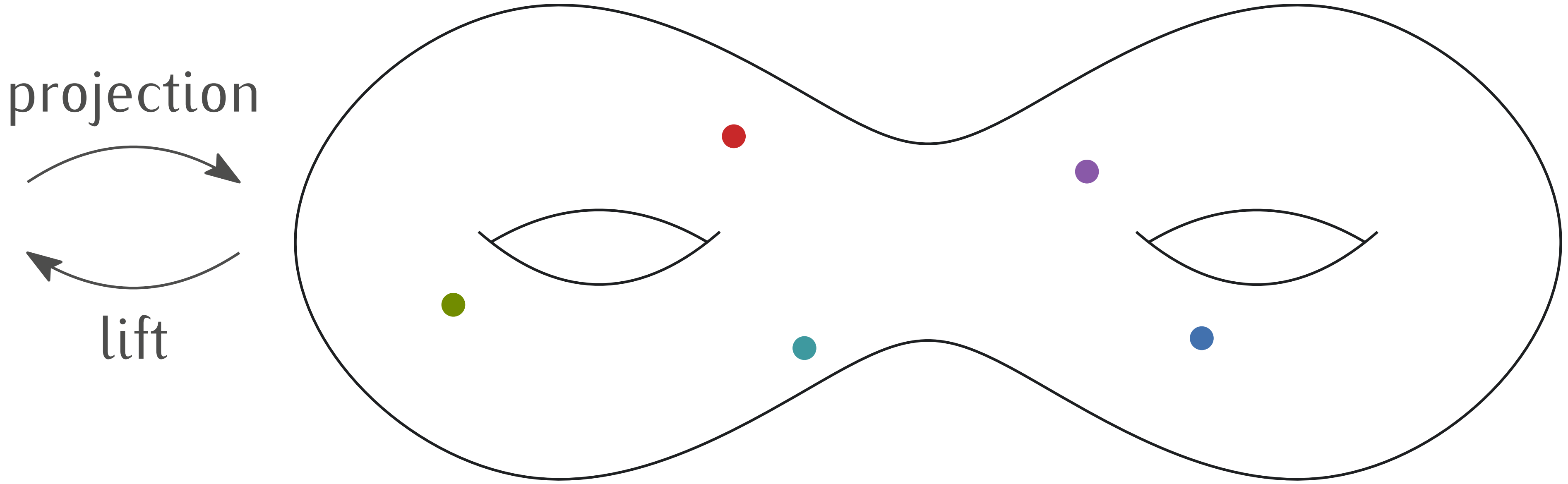
Hyperbolic surface

GEOMETRY

Hyperbolic surface
Surface equipped with a metric s.t. it is locally isometric to $\mathbb{H}^2$ (**hyperbolic metric**).

Algebraic point of view
Hyperbolic surface: $S = \mathbb{H}^2/\Gamma$.
Γ: group of orientation-preserving isometries of $\mathbb{H}^2$.
Γ: generated by the side-pairings of the fundamental polygon.



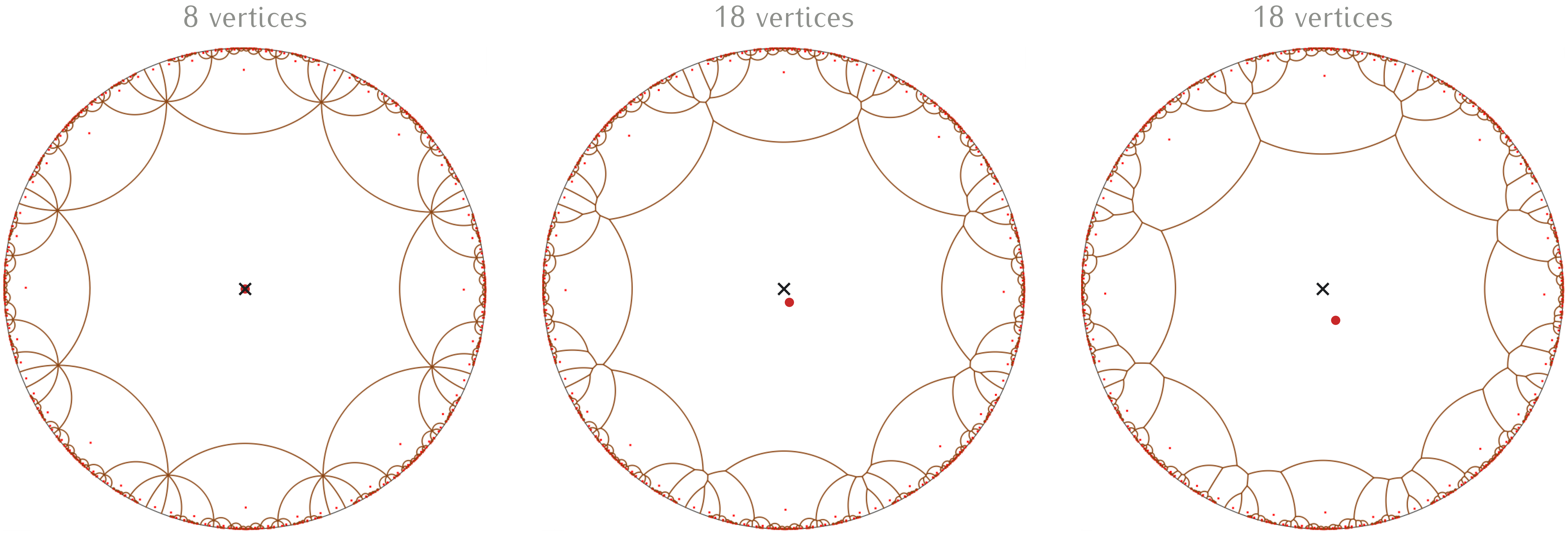


The Bolza surface (fig: Iordanov, Teillaud)

Dirichlet domain

Dirichlet domain of a lift $\widetilde{x}$ ($x \in S$)
Closed Voronoi cell of $\widetilde{x}$ in the orbit $\Gamma\widetilde{x}$.
$\mathcal{D}(\widetilde{x}) = \{\widetilde{y} \in \mathbb{H}^2 : d_{\mathbb{H}^2}(\widetilde{x}, \widetilde{y}) \leqslant d_{\mathbb{H}^2}(\widetilde{x}, \gamma\widetilde{y}) \forall \gamma \in \Gamma\}$

Dirichlet domains for the Bolza surface (fig: Bogdanov, Teillaud, Vegter)



Problem statement

- **The Bolza surface**: well-studied; ⟶ diameter, systole, ...
- **Generic hyperbolic surfaces**: harder to study.

  ⤷ Dirichlet domain, Delaunay triangulation

Problem statement

- **The Bolza surface**: well-studied; ⟶ diameter, systole, ...
- **Generic hyperbolic surfaces**: harder to study.

  ↪ Dirichlet domain, Delaunay triangulation

Motivation: design & implement approximation algorithms to help studying generic hyperbolic surfaces.

Problem statement

- **The Bolza surface**: well-studied; → diameter, systole, ...
- **Generic hyperbolic surfaces**: harder to study.

  → Dirichlet domain, Delaunay triangulation

Motivation: design & implement approximation algorithms to help studying generic hyperbolic surfaces.

First step

Approximate the geometry of the surface with a set of *well-distributed* points.

→ this thesis

ε-nets

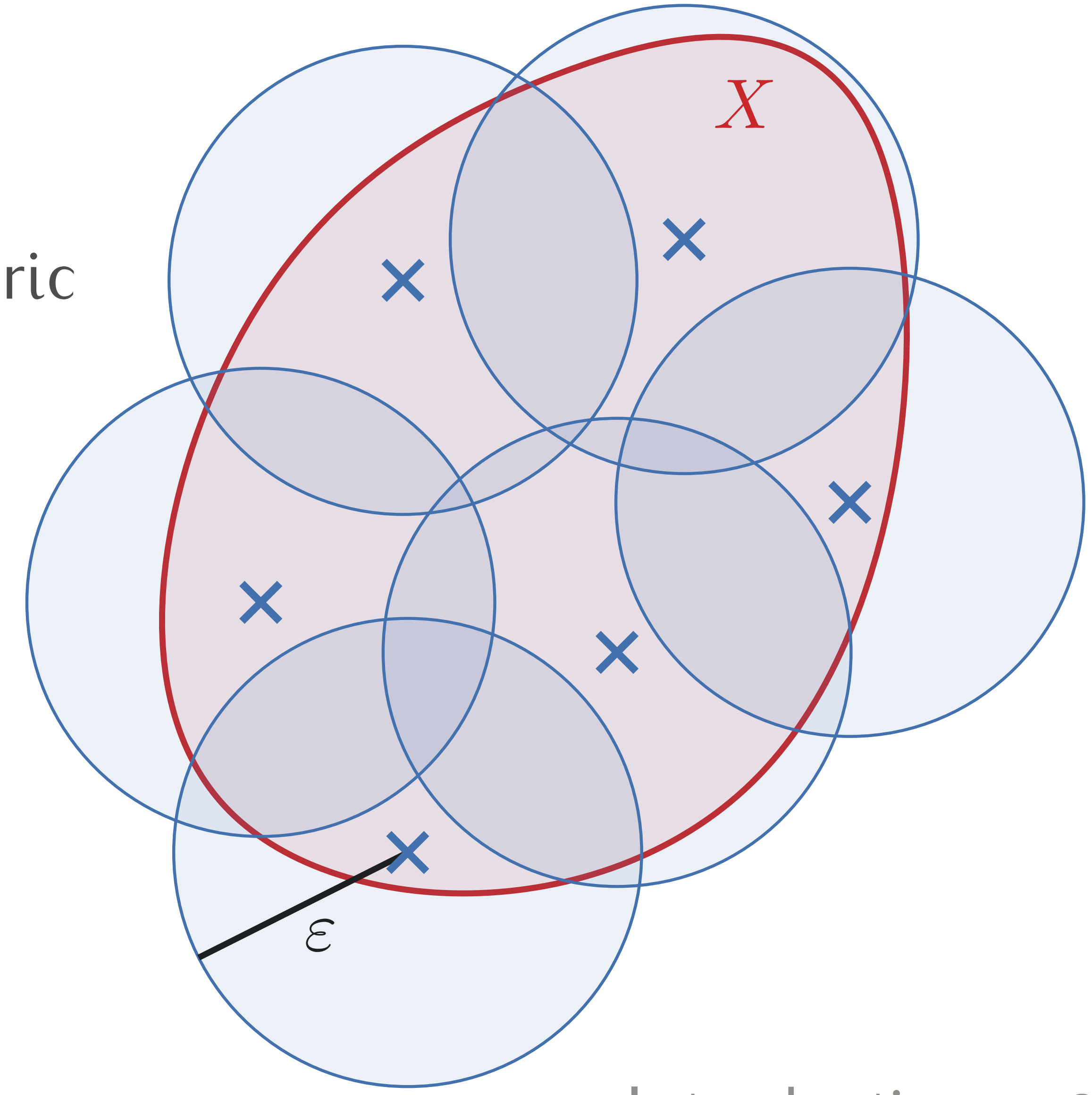
ε-net [Clarkson, 2006]

(X, d) metric space, $\varepsilon > 0$.

$P \subset X$ is an ε-net if:

1. the closed balls $\{x \in X \mid d(x,p) \leqslant \varepsilon\}_{p \in P}$ cover X (**ε-covering**), and
2. for all $p \neq q \in P$, $d(p,q) \geqslant \varepsilon$ (**ε-packing**).

An ε-net of a metric space X



ε-nets

ε-net [Clarkson, 2006]

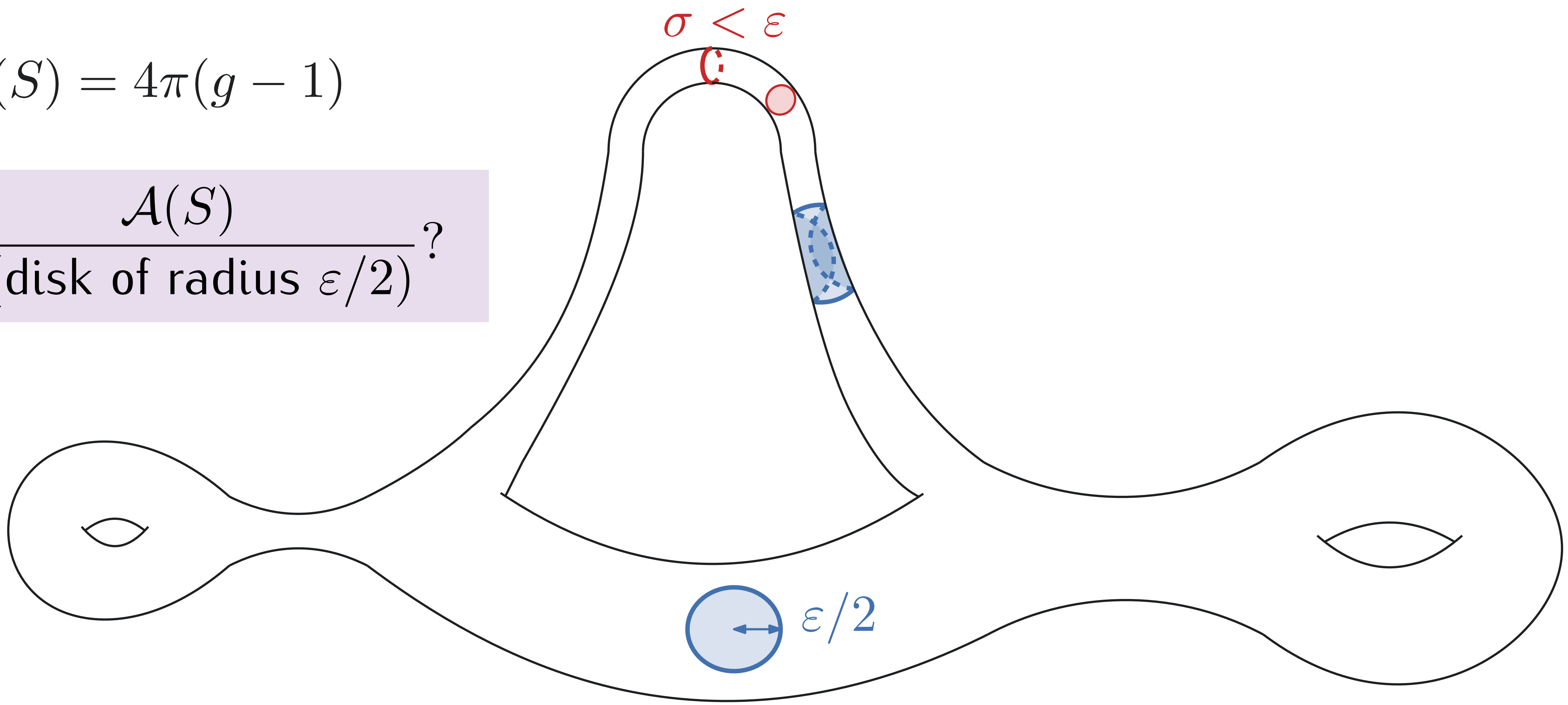
(X, d) metric space, $\varepsilon > 0$.
$P \subset X$ is an ε-net if:

1. the closed balls $\{x \in X \mid d(x,p) \leqslant \varepsilon\}_{p \in P}$ cover X (**ε-covering**), and
2. for all $p \neq q \in P$, $d(p,q) \geqslant \varepsilon$ (**ε-packing**).

Area of a hyperbolic surface

$$\mathcal{A}(S) = 4\pi(g-1)$$

$$N \leqslant \frac{\mathcal{A}(S)}{\mathcal{A}(\text{disk of radius } \varepsilon/2)}?$$



ε-nets

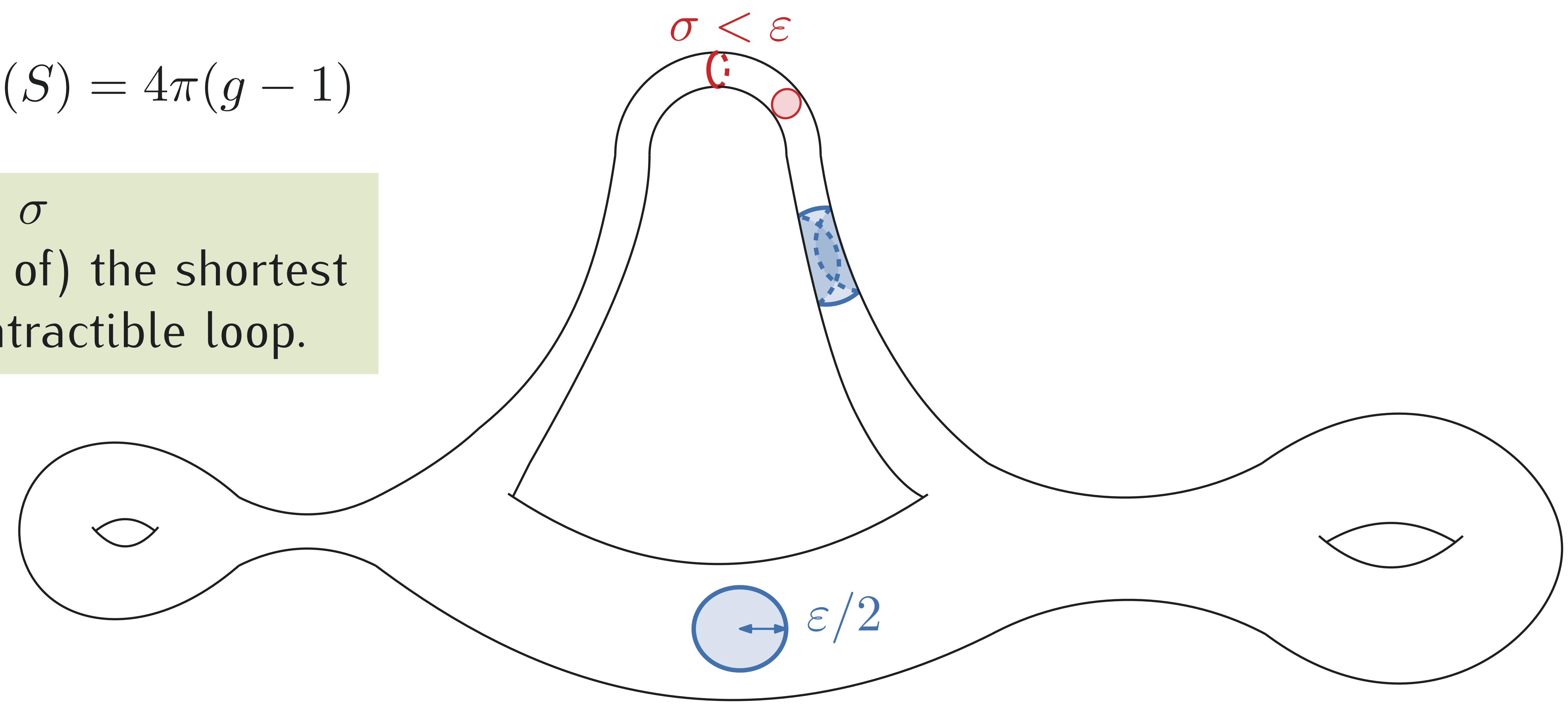
ε-net [Clarkson, 2006]
(X, d) metric space, $\varepsilon > 0$.
$P \subset X$ is an ε-net if:

1. the closed balls $\{x \in X \mid d(x, p) \leqslant \varepsilon\}_{p \in P}$ cover X (ε-**covering**), and
2. for all $p \neq q \in P$, $d(p, q) \geqslant \varepsilon$ (ε-**packing**).

Area of a hyperbolic surface

$$\mathcal{A}(S) = 4\pi(g - 1)$$

Systole σ
(Length of) the shortest non-contractible loop.



ε-nets

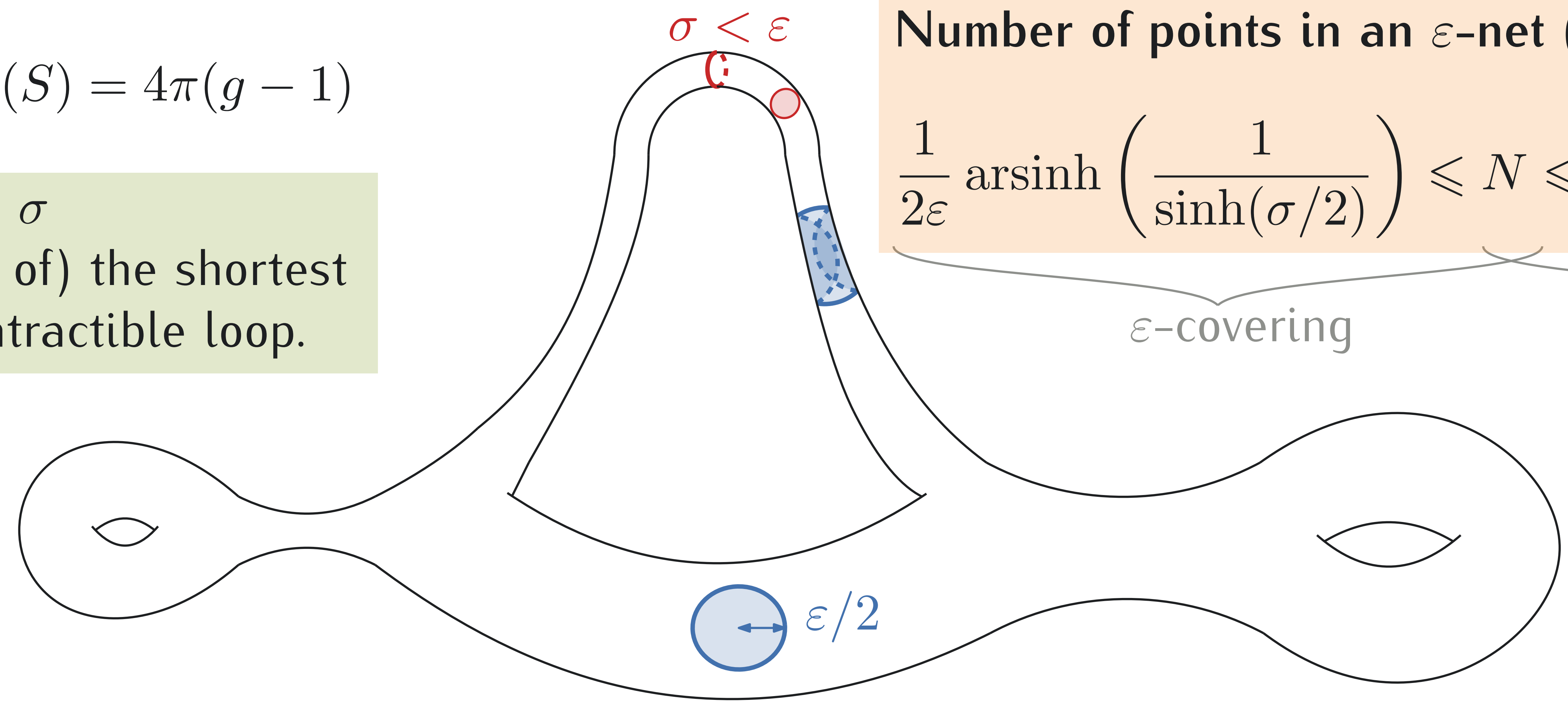
ε-net [Clarkson, 2006]
(X, d) metric space, $\varepsilon > 0$.
$P \subset X$ is an ε-net if:

1. the closed balls $\{x \in X \mid d(x,p) \leqslant \varepsilon\}_{p \in P}$ cover X (**ε-covering**), and
2. for all $p \neq q \in P$, $d(p,q) \geqslant \varepsilon$ (**ε-packing**).

Area of a hyperbolic surface

$$\mathcal{A}(S) = 4\pi(g-1)$$

Systole σ
(Length of) the shortest non-contractible loop.

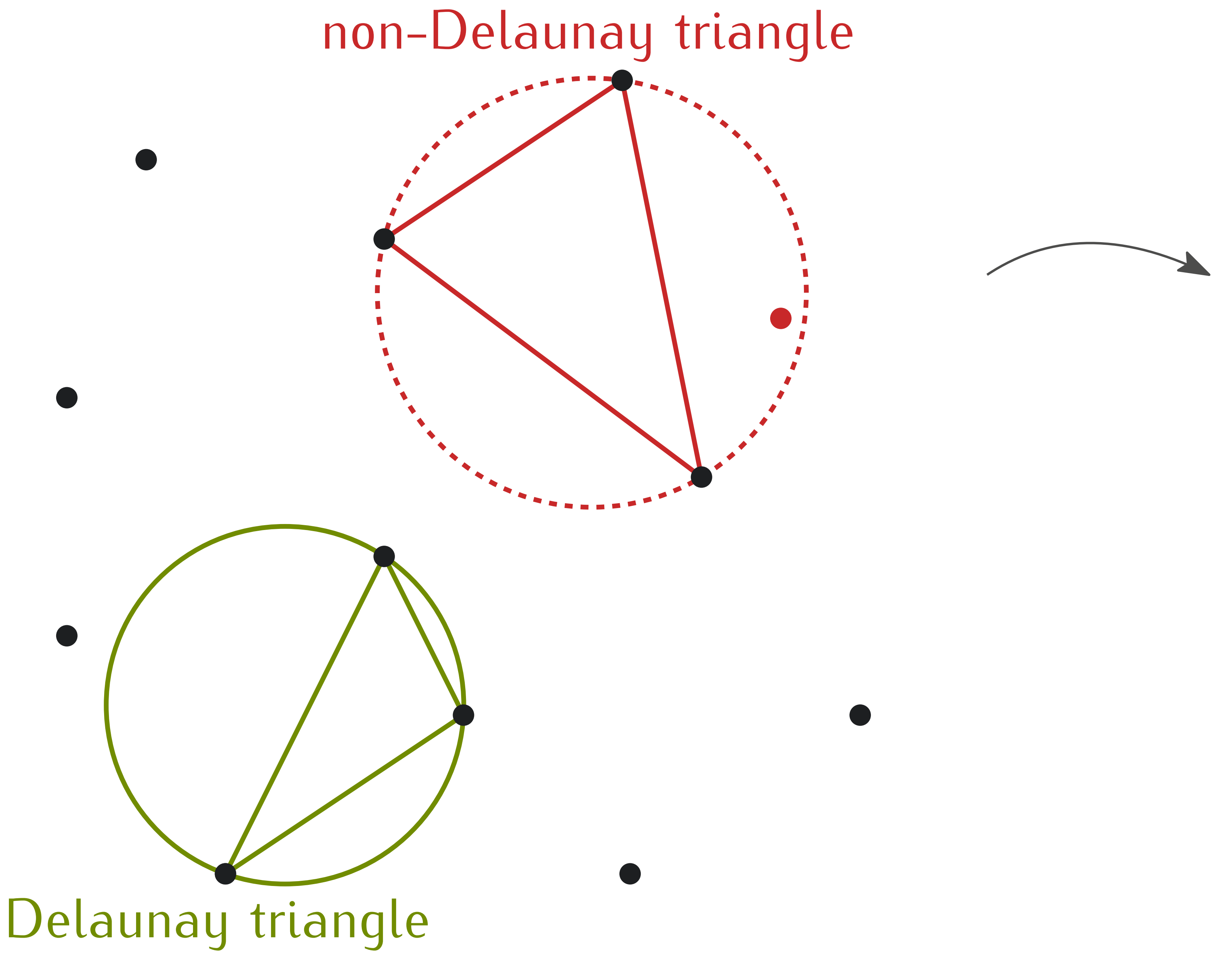


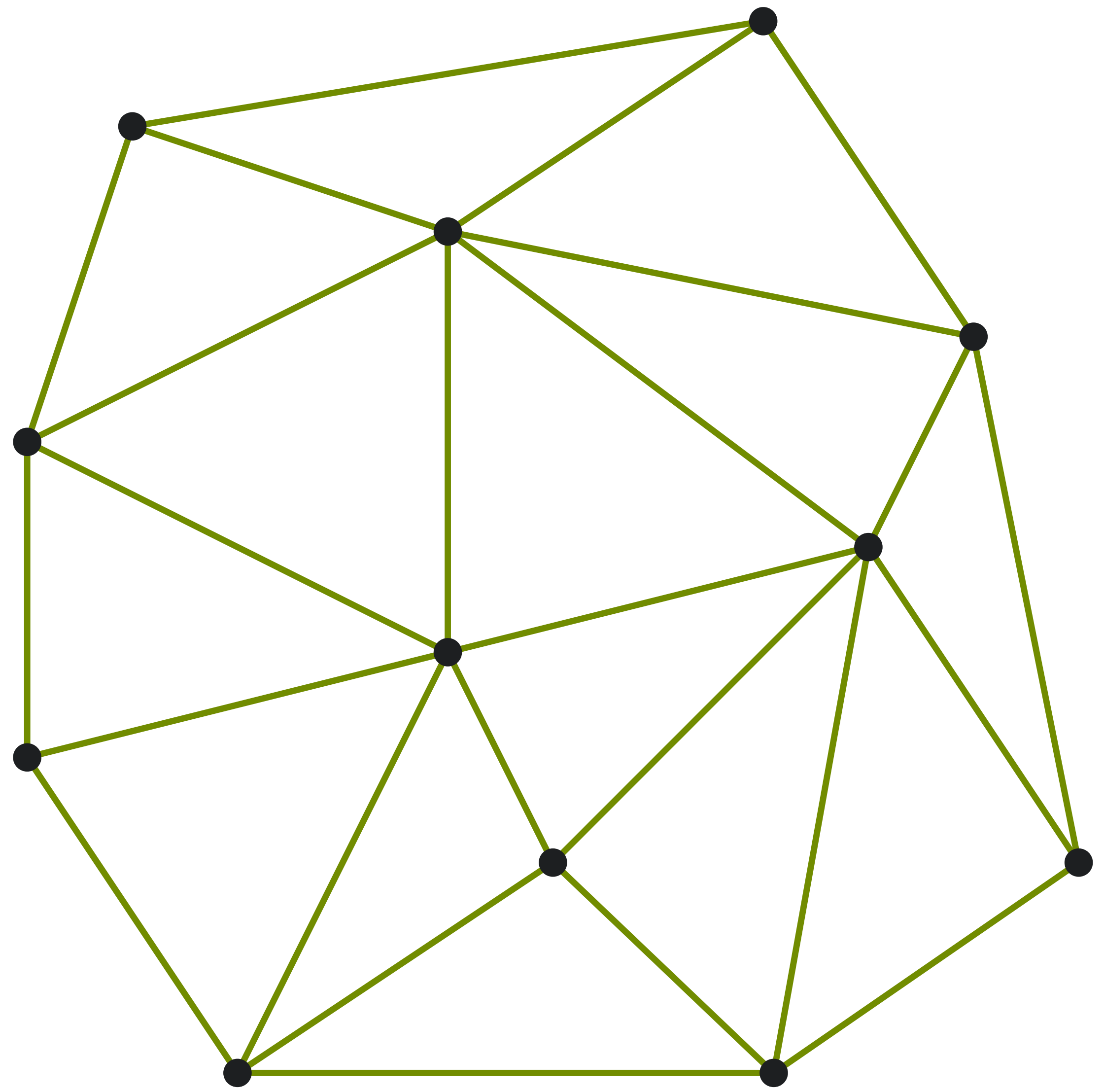
Number of points in an ε-net ($\sigma < \varepsilon$)

$$\underbrace{\frac{1}{2\varepsilon}\operatorname{arsinh}\left(\frac{1}{\sinh(\sigma/2)}\right) \leqslant N}_{\varepsilon\text{-covering}} \leqslant \underbrace{16(g-1)\left(\frac{1}{\varepsilon^2}+\frac{1}{\sigma^2}\right)}_{\varepsilon\text{-packing}}$$

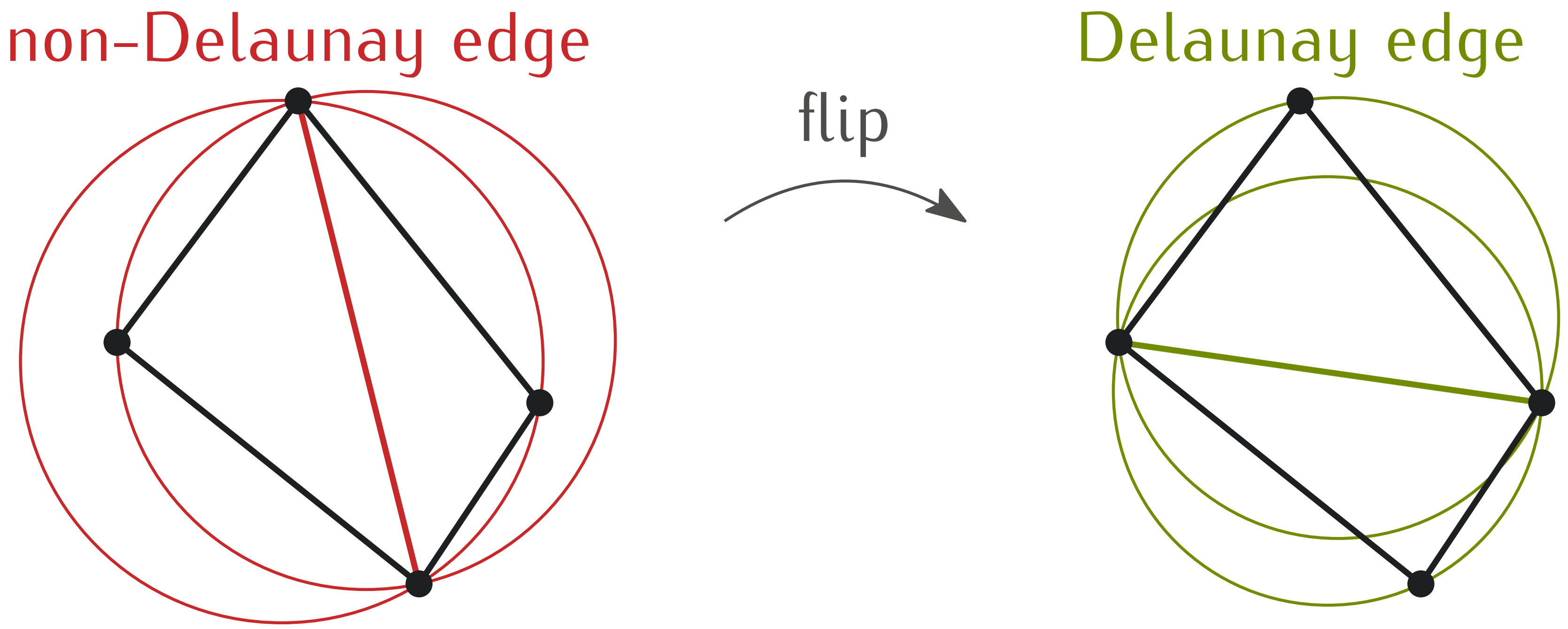
Delaunay triangulation

in $\mathbb{R}^2$



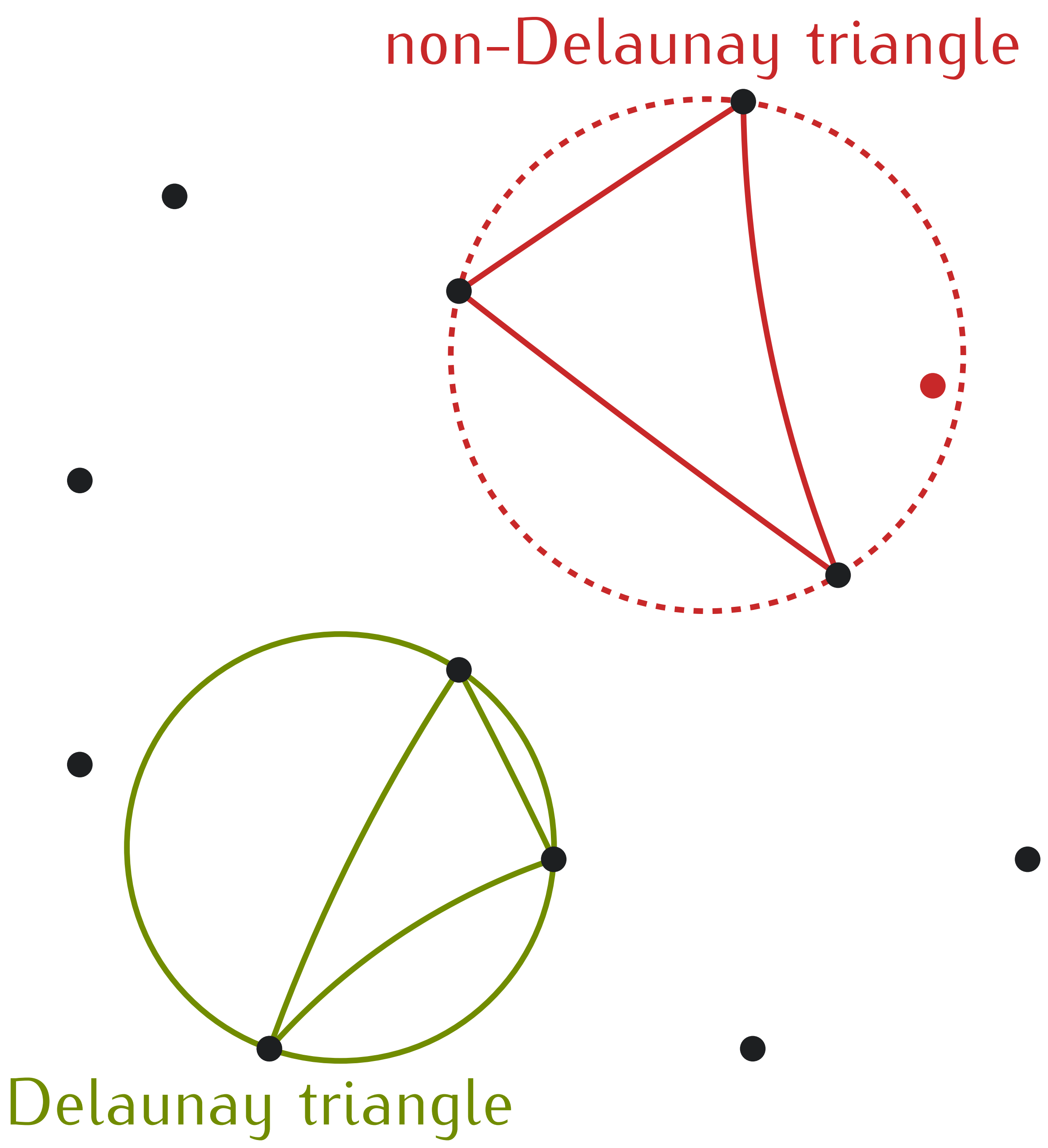


Delaunay triangulation

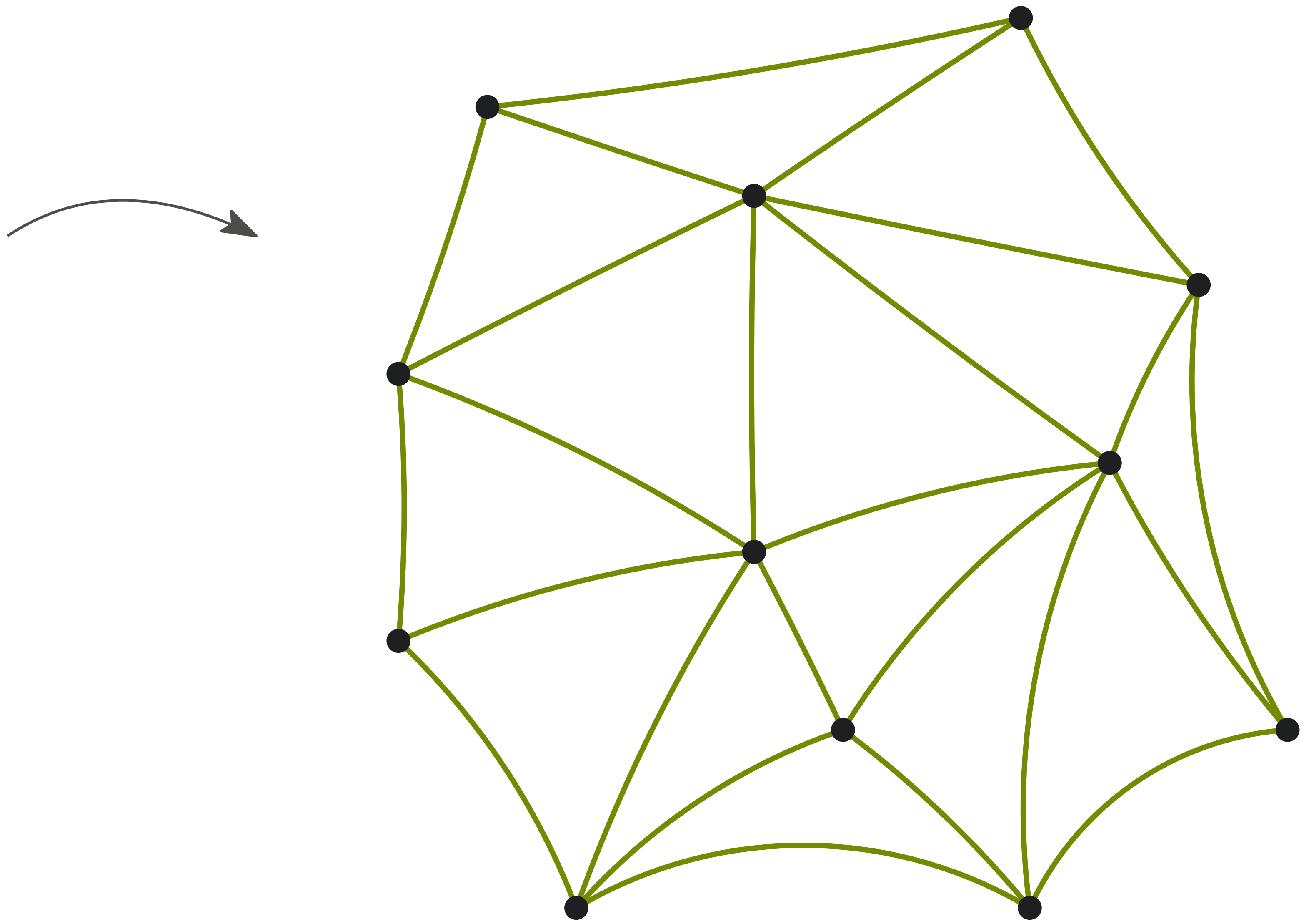


The flip algorithm

Delaunay triangulation



in $\mathbb{H}^2$ [Bogdanov, Devillers, Teillaud, 2014]



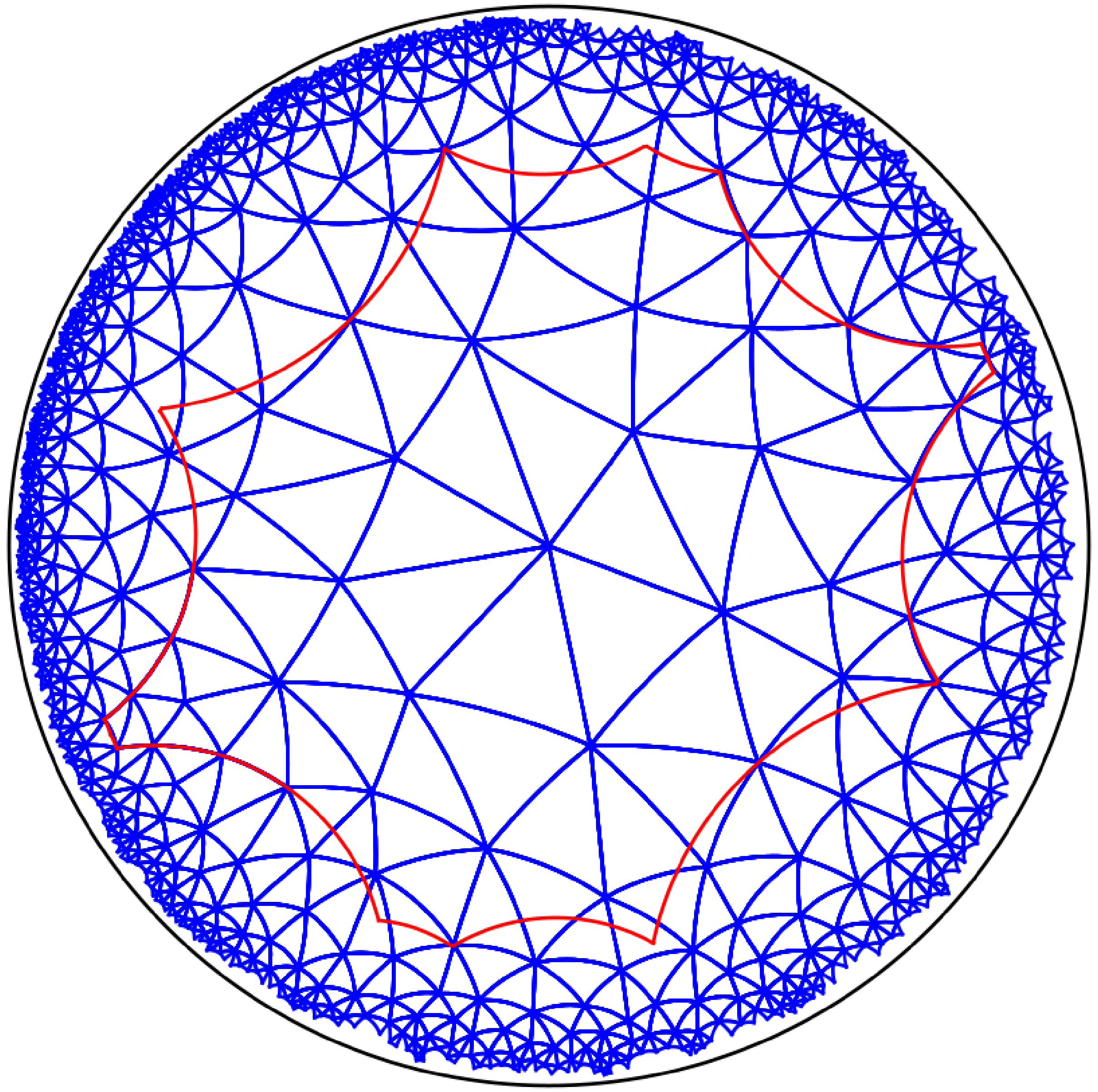
Delaunay triangulation

Generalization on hyperbolic surfaces

$V \subset S$. $\widetilde{V} \in \mathbb{H}^2$:= set of all the lifts of all the points of V.

$DT(V)$:= projection on S of $DT(\widetilde{V})$.

infinite and periodic



50 lifts of every triangle

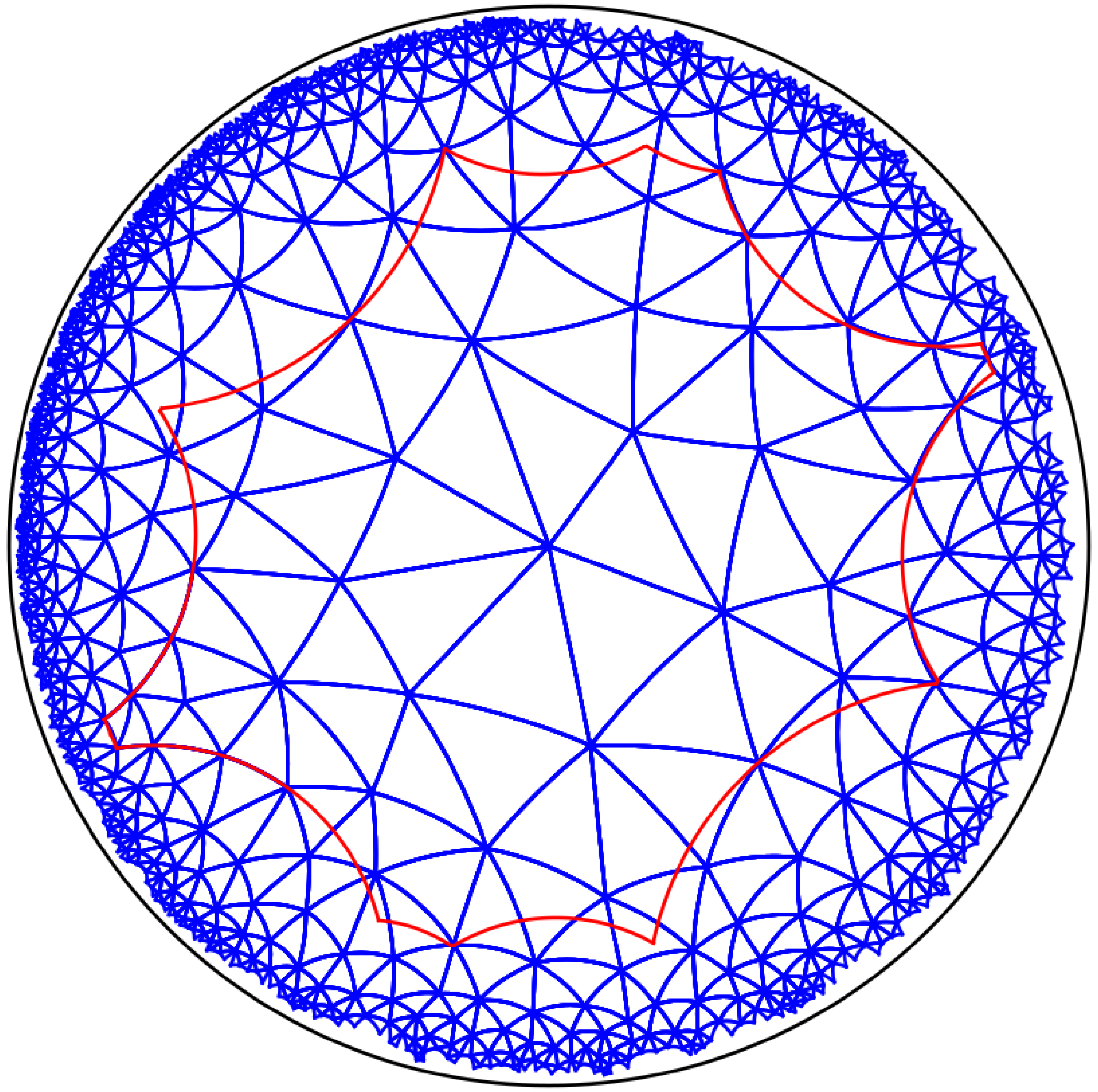
Delaunay triangulation

Generalization on hyperbolic surfaces
$V \subset S$. $\widetilde{V} \in \mathbb{H}^2$:= set of all the lifts of all the points of V.
$DT(V)$:= projection on S of $DT(\widetilde{V})$.

infinite and periodic

Complexity of the flip algorithm
With $n := |V|$. Starting from any triangulation T of S, the flip algorithm computes $DT(V)$ in $O(\Lambda(T)^{6g-4}n^2)$ flips.

[Despré, Schlenker, Teillaud, 2024]



50 lifts of every triangle

Overview of the ε-net algorithm

Input: DT of hyperbolic surface S with a single vertex.

Output: ε-net of S and its DT.

Key idea: Delaunay refinement. [Shewchuk, 2002]

Overview of the ε-net algorithm

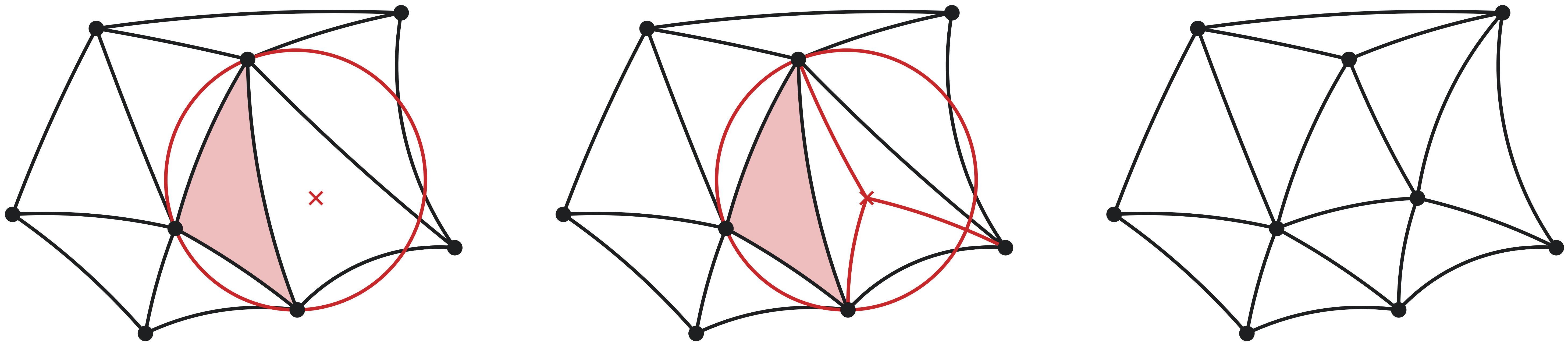
Input: DT of hyperbolic surface S with a single vertex.

Output: ε-net of S and its DT.

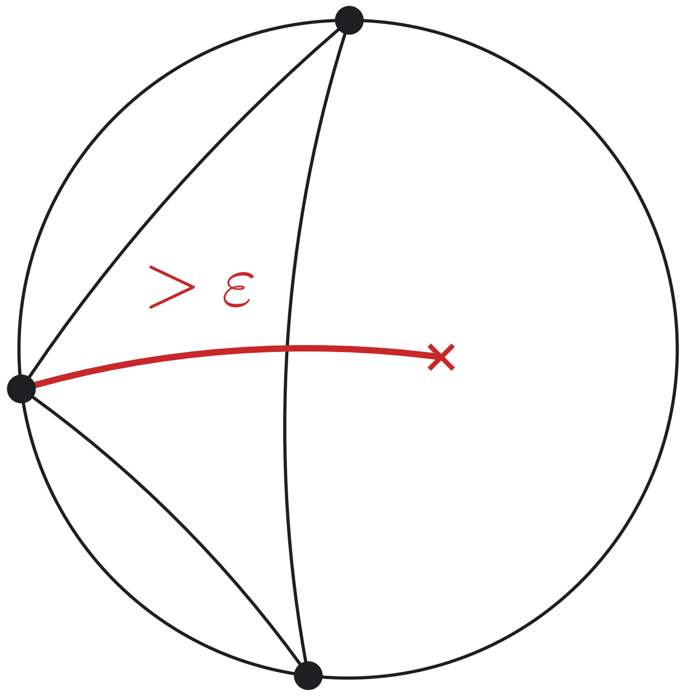
Key idea: Delaunay refinement. [Shewchuk, 2002]

- Insert circumcenter of a Delaunay triangle with circumradius $> \varepsilon$ (**large triangle**).
- Retrieve DT (with flip algo).
- Repeat until all triangles have circumradius $\leqslant \varepsilon$.

Schematic representation of the algorithm



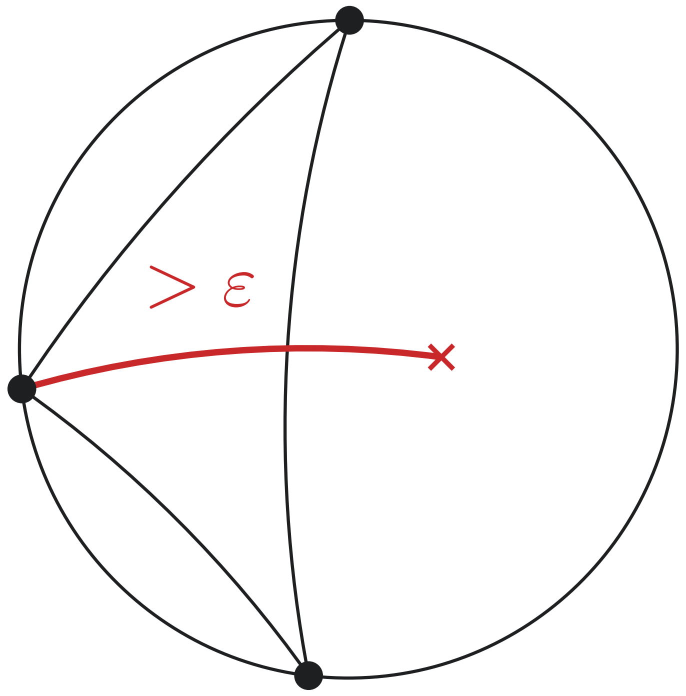
Why it works



Termination

- Each insertion maintains the ε-packing property.
- The number of points in an ε-packing is bounded.
- So the algorithm terminates after a finite number of insertions.

Why it works



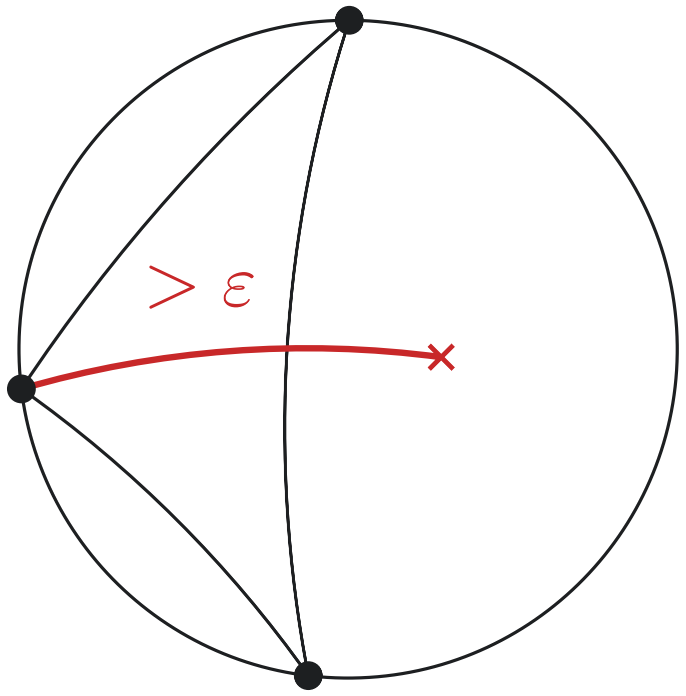
Termination

- Each insertion maintains the ε-packing property.
- The number of points in an ε-packing is bounded.
- So the algorithm terminates after a finite number of insertions.

Correctness

- Any point on S is in a triangle, which has circuradius $\leqslant \varepsilon$,
- So it is at distance $\leqslant \varepsilon$ from a vertex of this triangle (**lemma**).

Why it works



Termination

- Each insertion maintains the ε-packing property.
- The number of points in an ε-packing is bounded.
- So the algorithm terminates after a finite number of insertions.

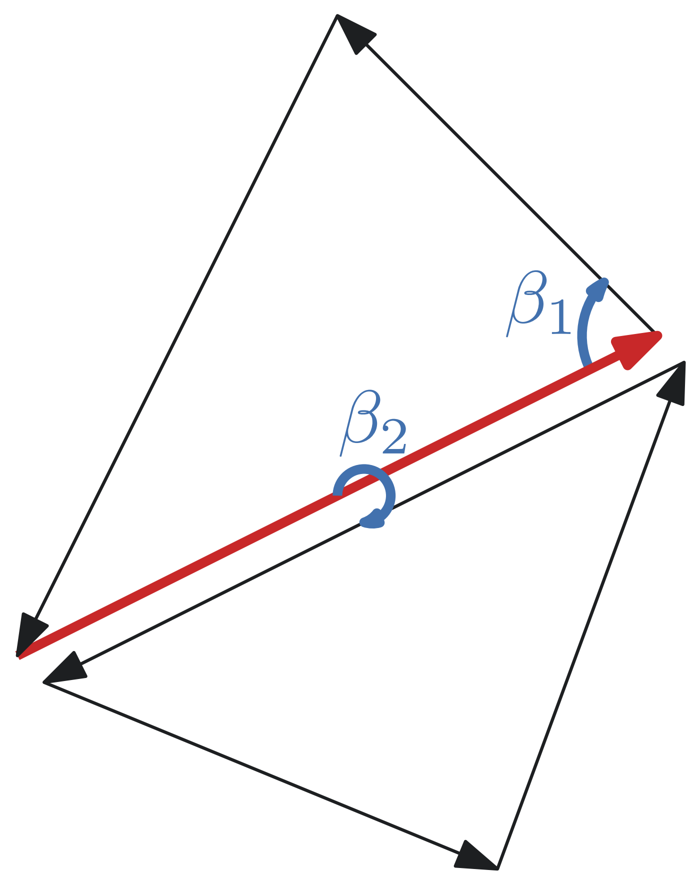
Correctness

- Any point on S is in a triangle, which has circuradius $\leqslant \varepsilon$,
- So it is at distance $\leqslant \varepsilon$ from a vertex of this triangle (**lemma**).

The algorithm terminates and ouputs an ε-net!

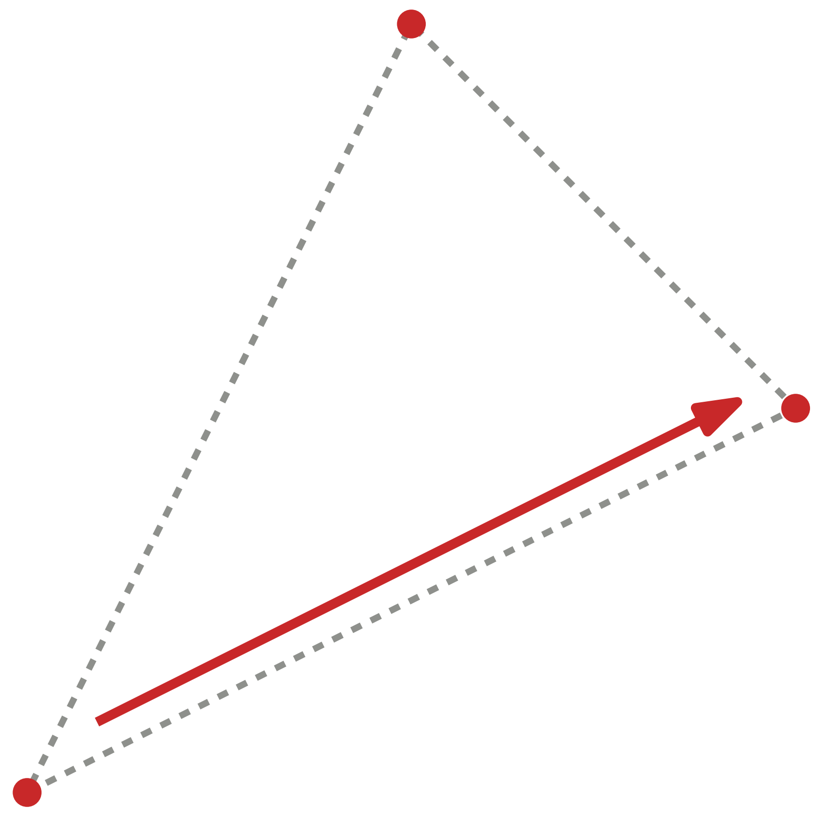
Data structure

- Combinatorial map: darts + pointers between darts;



Data structure

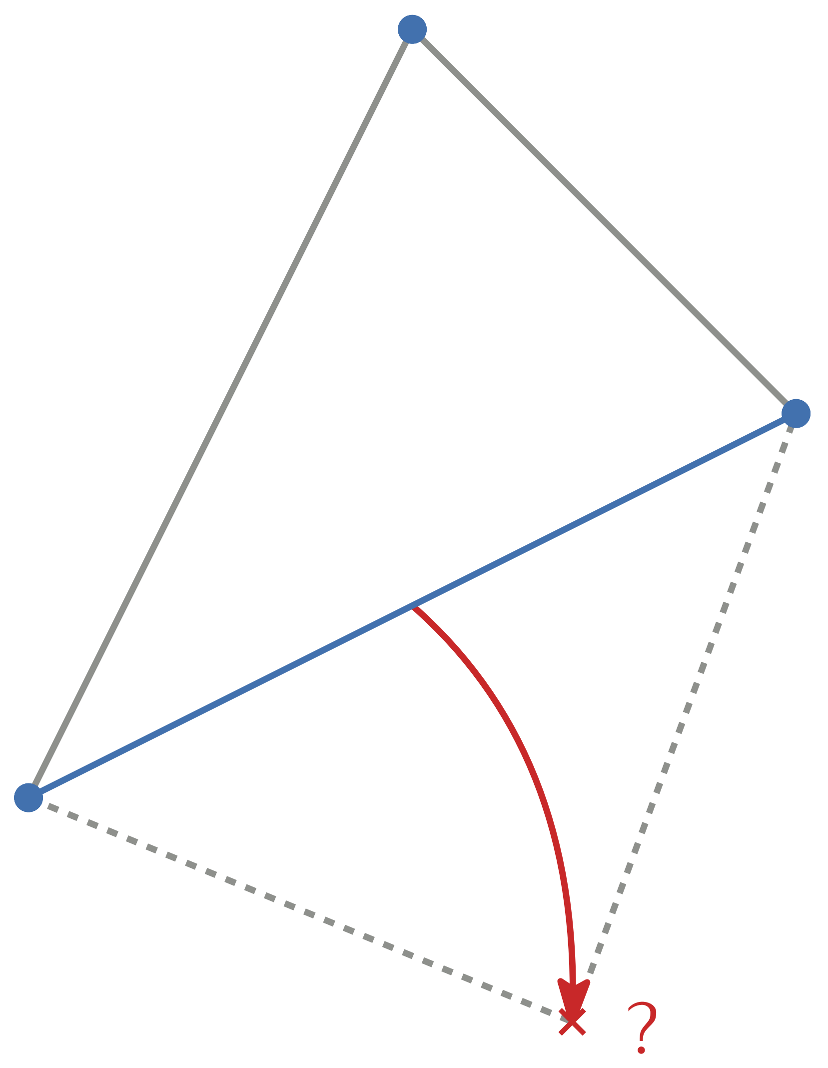
- Combinatorial map: darts + pointers between darts;
- An *anchor* for each face: 1 dart + 3 vertices in $\mathbb{H}^2$;

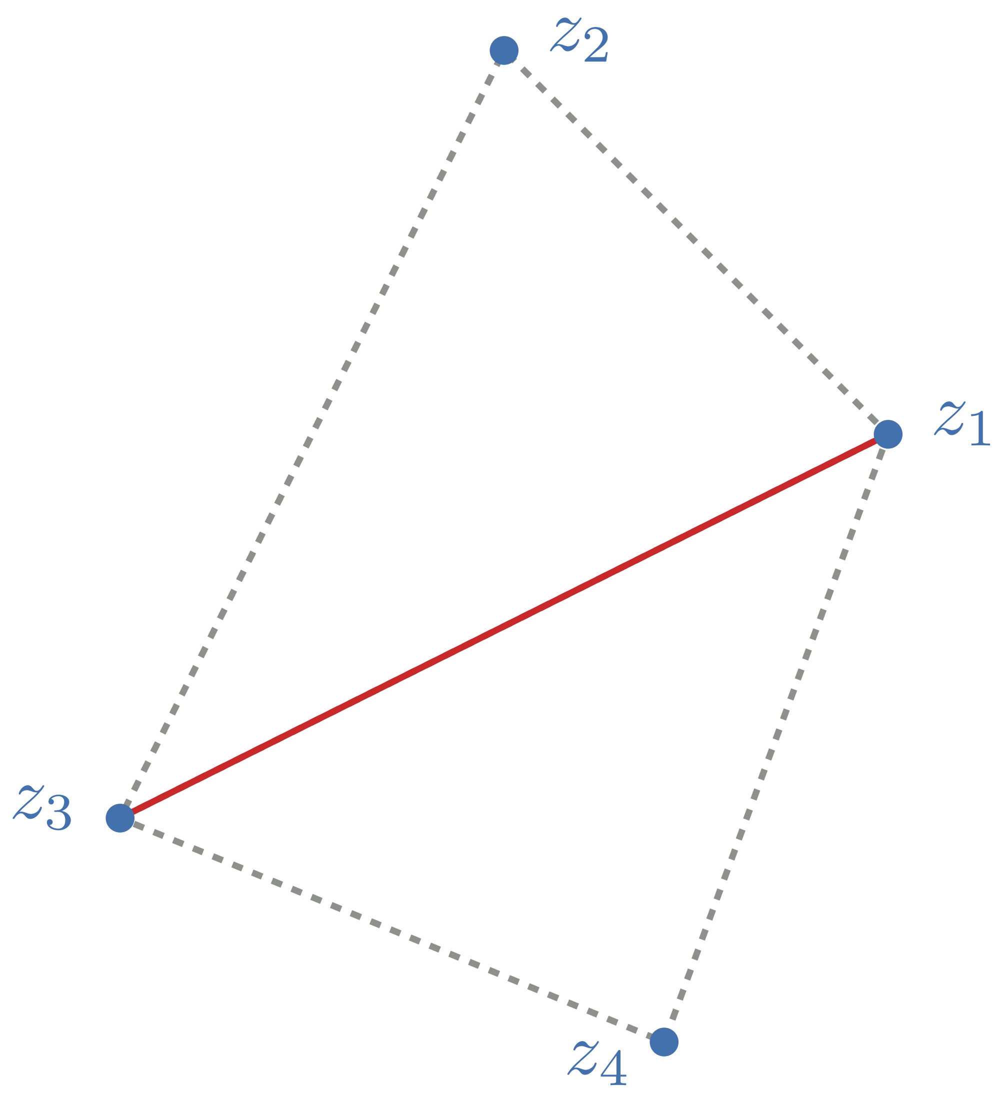


Data structure

- Combinatorial map: darts + pointers between darts;
- An *anchor* for each face: 1 dart + 3 vertices in $\mathbb{H}^2$;
- A *cross-ratio* for each edge: ratio between 4 vertices in $\mathbb{H}^2$.

(+ detects non-Delaunay edges)



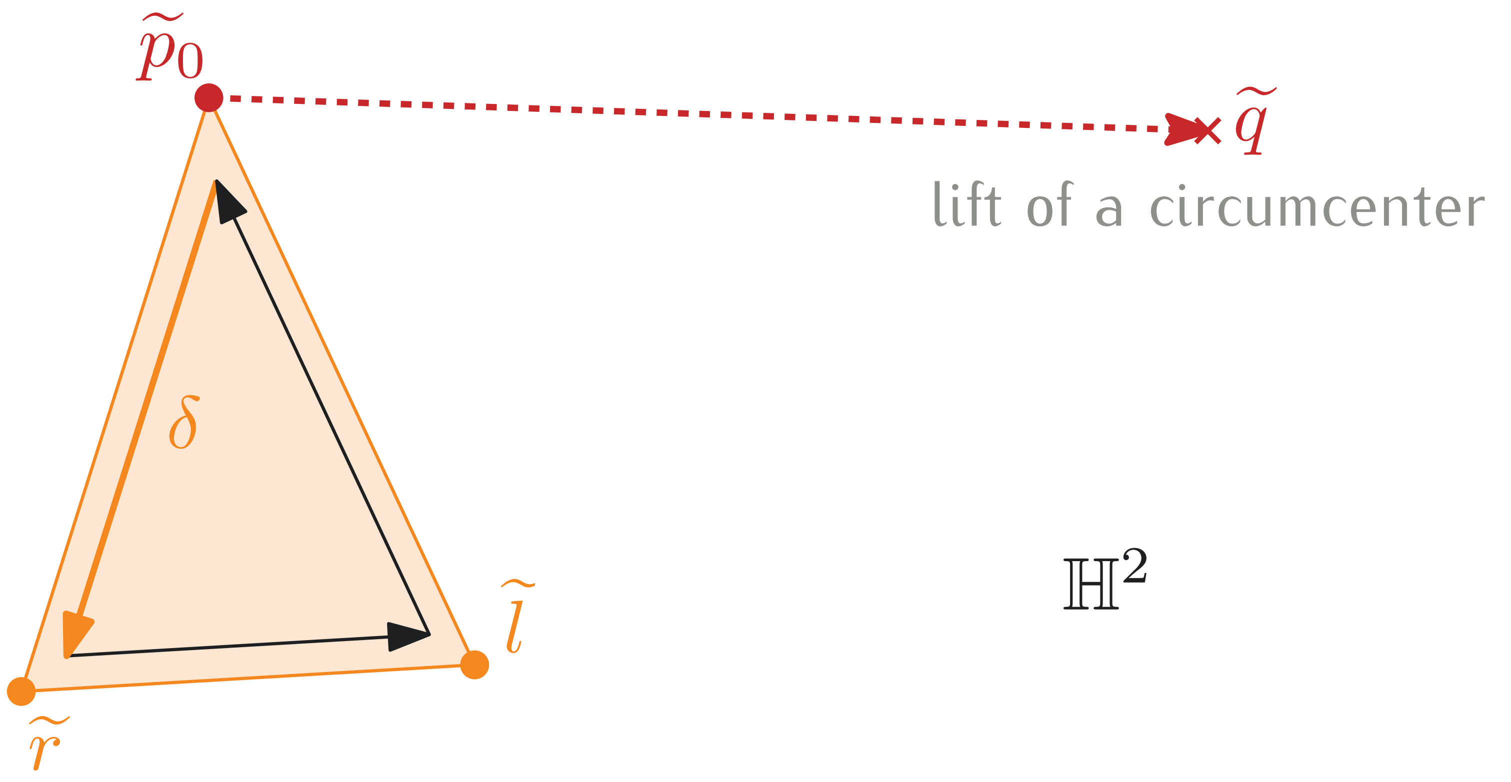


Cross-ratio of $z_1, z_2, z_3, z_4 \in \mathbb{C}$ (pairwise $\neq$).

$$[z_1, z_2, z_3, z_4] = \frac{(z_4 - z_2)(z_3 - z_1)}{(z_4 - z_1)(z_3 - z_2)}$$

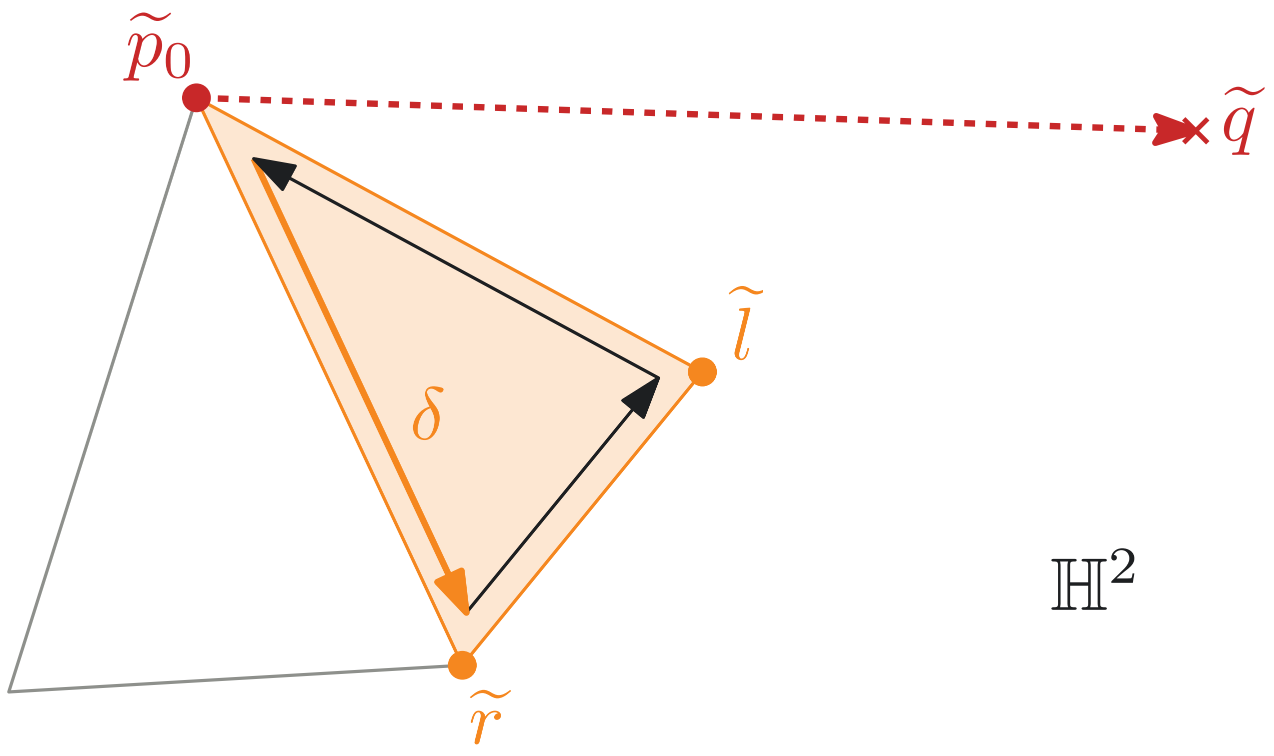
Key step: locate the circumcenter

The straight walk in our data structure
Inititalization phase



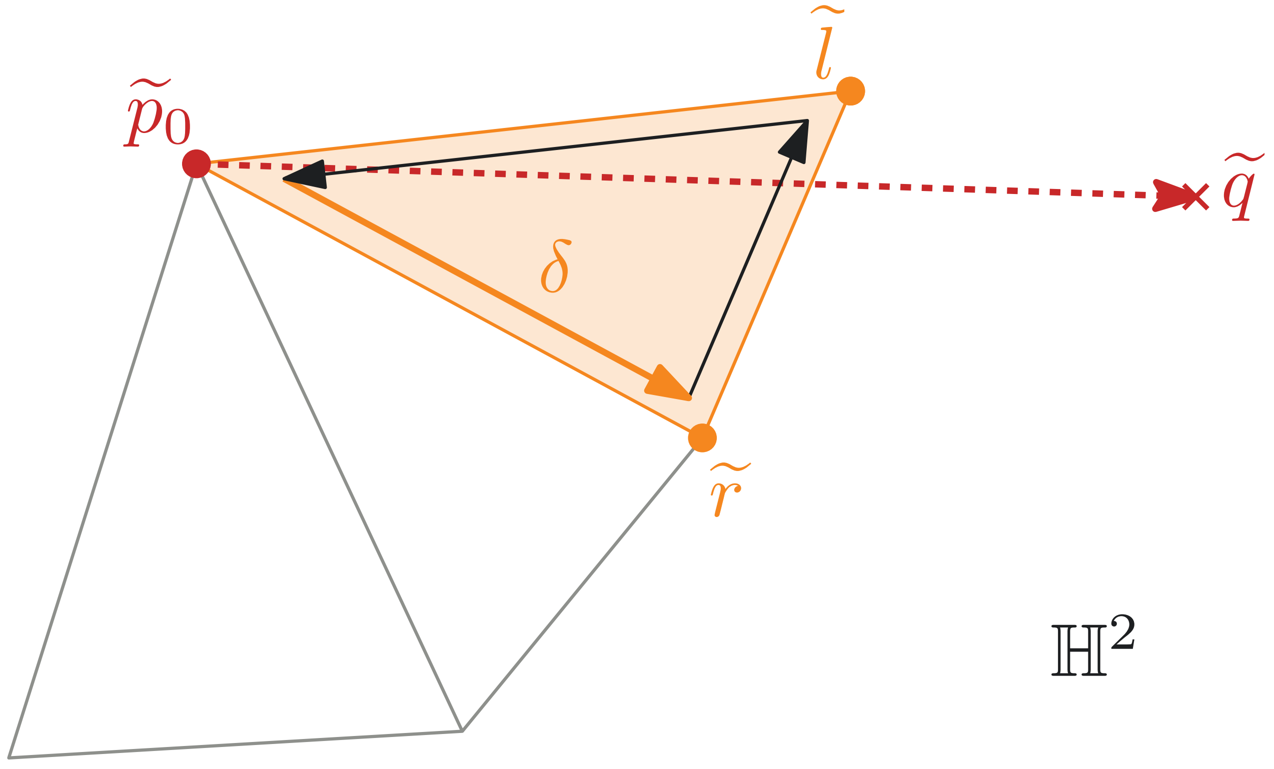
Key step: locate the circumcenter

The straight walk in our data structure
Inititalization phase



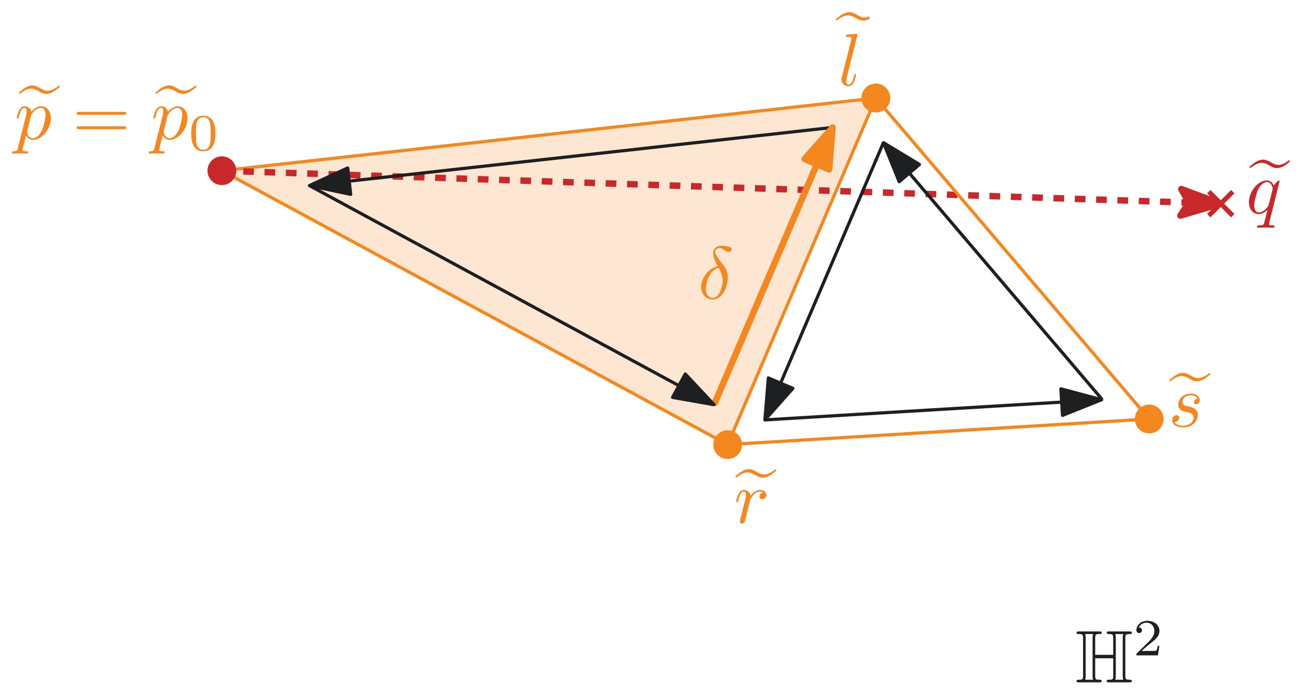
Key step: locate the circumcenter

The straight walk in our data structure
Inititalization phase



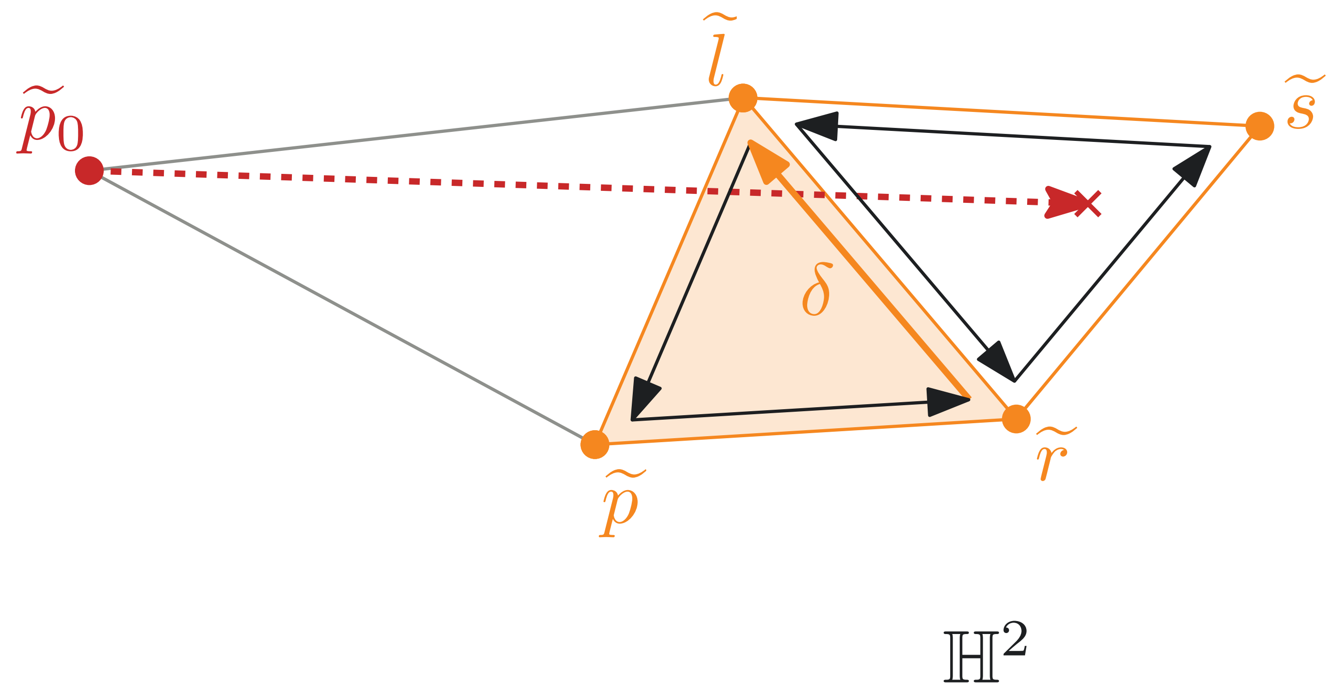
Key step: locate the circumcenter

The straight walk in our data structure



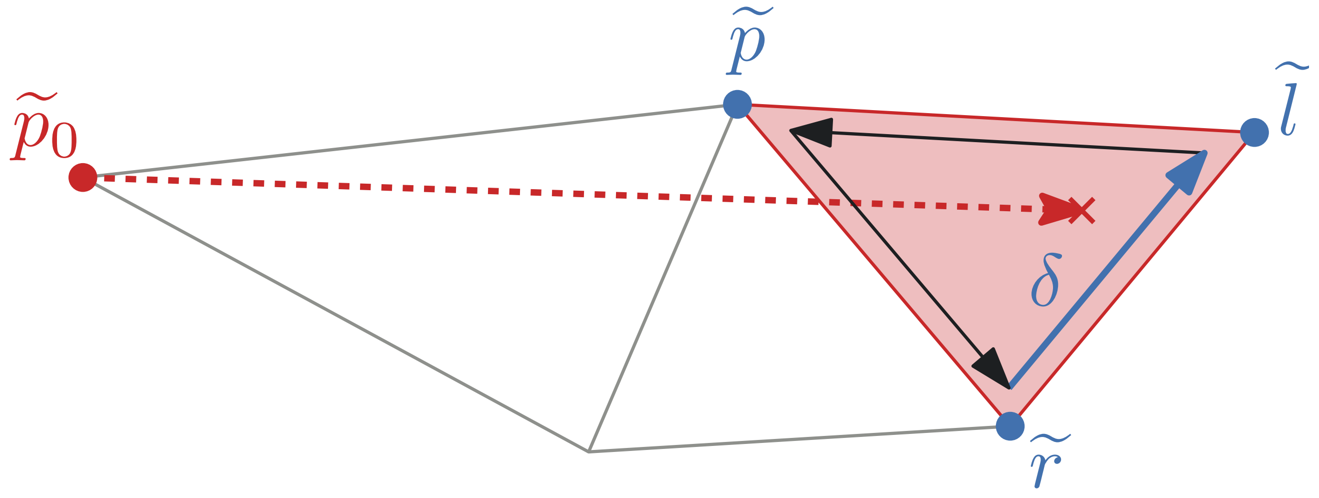
Key step: locate the circumcenter

The straight walk in our data structure



Key step: locate the circumcenter

The straight walk in our data structure



$\mathbb{H}^2$

Worst-case complexity

Reminder

Number of points in an ε-net: $N \leqslant 16(g-1)(1/\varepsilon^2 + 1/\sigma^2)$.

Complexity of the straight walk

- The geodesic $\widetilde{p_0}\widetilde{q}$ can intersect each edge of the triangulation at most twice;
- So the straight walk costs $O(i)$ at the insertion of the i-th point.
- Consequence: all the point locations cost $O(N^2)$ in total.

Worst-case complexity

Reminder

Number of points in an ε-net: $N \leqslant 16(g-1)(1/\varepsilon^2 + 1/\sigma^2)$.

Complexity of the straight walk

- The geodesic $\widetilde{p}_0\widetilde{q}$ can intersect each edge of the triangulation at most twice; → because it is a Delaunay triangulation [Despré, Kolbe, Teillaud, 2021]
- So the straight walk costs $O(i)$ at the insertion of the i-th point.
- Consequence: all the point locations cost $O(N^2)$ in total.

Worst-case complexity

Reminder

Number of points in an ε-net: $N \leqslant 16(g-1)(1/\varepsilon^2 + 1/\sigma^2)$.

Complexity of the straight walk

- The geodesic $\widetilde{p}_0\widetilde{q}$ can intersect each edge of the triangulation at most twice;
- So the straight walk costs $O(i)$ at the insertion of the i-th point.
- Consequence: all the point locations cost $O(N^2)$ in total.

→ because it is a Delaunay triangulation [Despré, Kolbe, Teillaud, 2021]

Complexity of the flip algorithm

The total cost of flips is $O(N^2)$.

→ adaptation of the proof of [Despré, Schlenker, Teillaud, 2024]

Worst-case complexity

Reminder

Number of points in an ε-net: $N \leqslant 16(g-1)(1/\varepsilon^2 + 1/\sigma^2)$.

Complexity of the straight walk

- The geodesic $\widetilde{p}_0\widetilde{q}$ can intersect each edge of the triangulation at most twice; → because it is a Delaunay triangulation [Despré, Kolbe, Teillaud, 2021]
- So the straight walk costs $O(i)$ at the insertion of the i-th point.
- Consequence: all the point locations cost $O(N^2)$ in total.

Complexity of the flip algorithm

The total cost of flips is $O(N^2)$.

→ adaptation of the proof of [Despré, Schlenker, Teillaud, 2024]

The ε-net algorithm computes an ε-net in $O(N^2)$ time.

[EuroCG 2024]

Pseudo ε-net

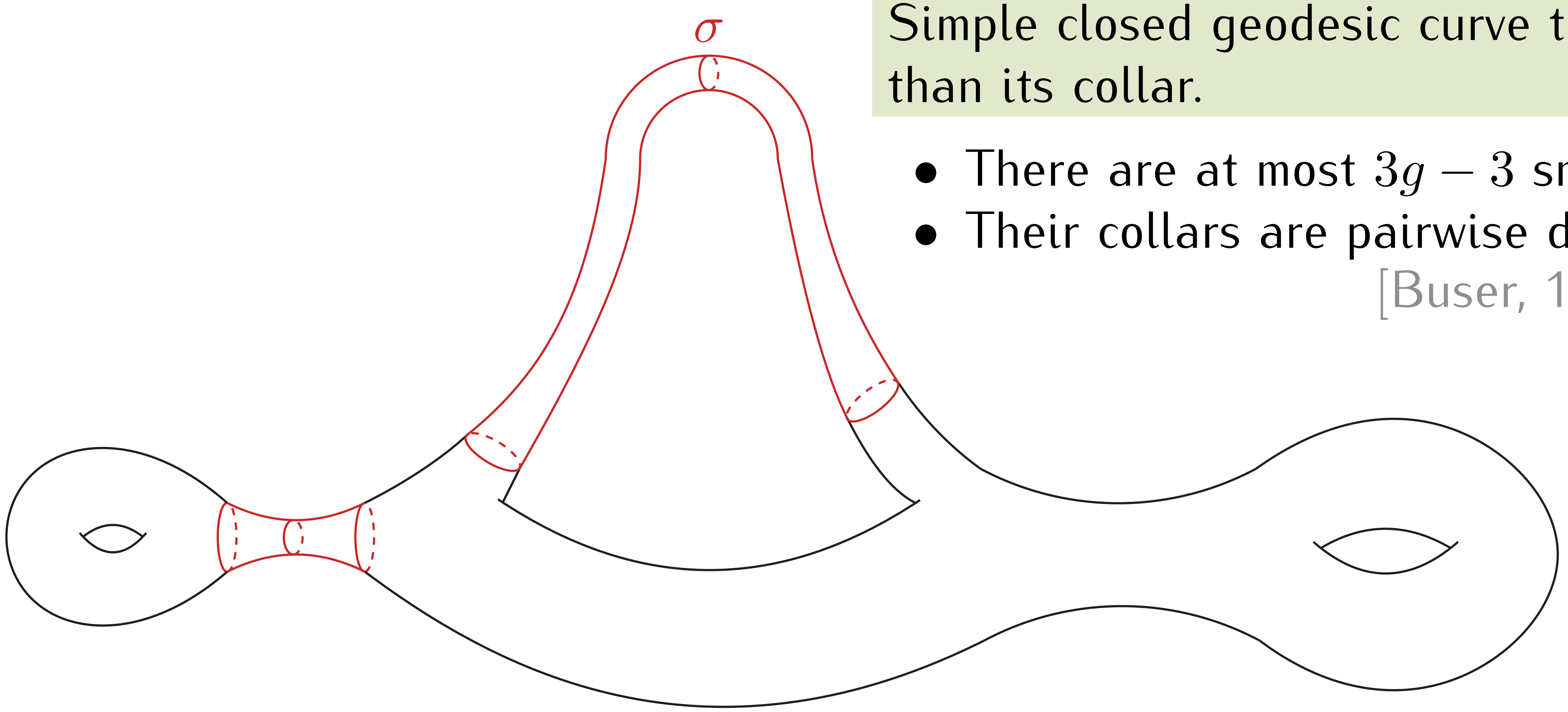
Reminder
$N \to +\infty$ when $\sigma \to 0$.

- A simple closed geodesic curve ρ yields a collar $\mathcal{C}(\rho)$ that is homeomorphic to a cylinder.
- The shorter the curve, the longer the collar.

Small curve
Simple closed geodesic curve that is shorter than its collar.

- There are at most $3g - 3$ small curves.
- Their collars are pairwise disjoint.

[Buser, 1992 (textbook)]



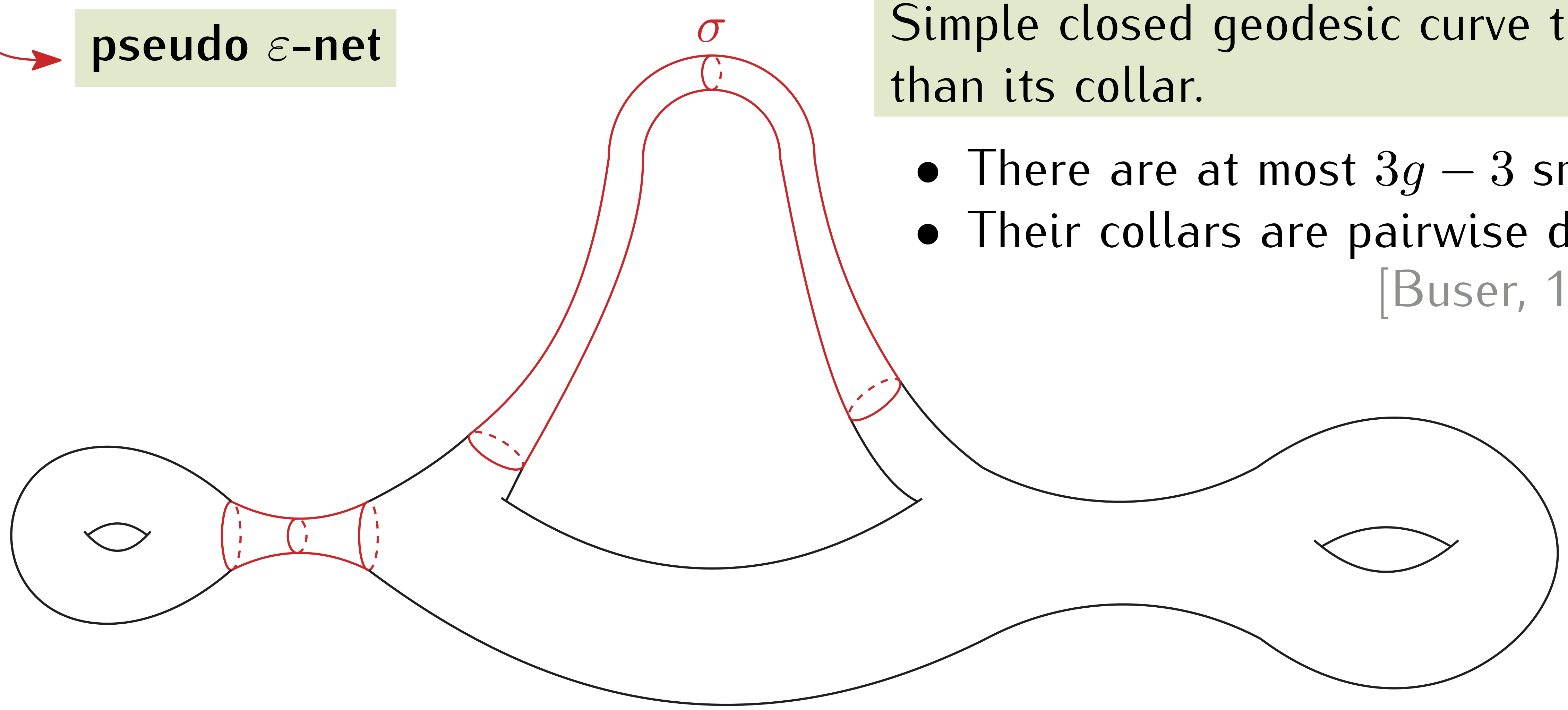
Pseudo ε-net

Reminder
$N \to +\infty$ when $\sigma \to 0$.

$(\varepsilon \leqslant \ln \sqrt{2} \approx 0.35)$

Idea: For every small curve shorter than ε, do not put points in an ε-collar around it.

→ **pseudo ε-net**



- A simple closed geodesic curve ρ yields a collar $\mathcal{C}(\rho)$ that is homeomorphic to a cylinder.
- The shorter the curve, the longer the collar.

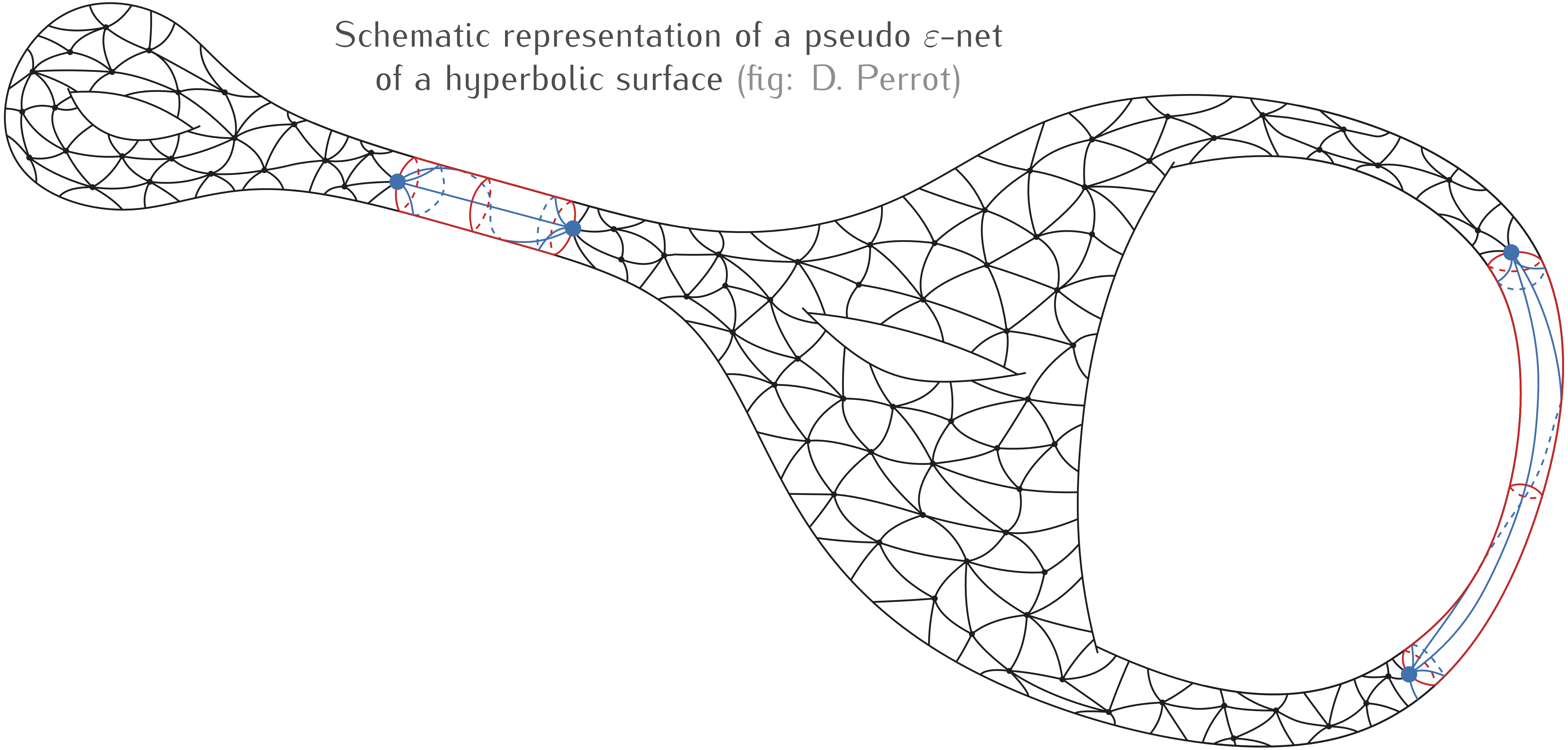
Small curve
Simple closed geodesic curve that is shorter than its collar.

- There are at most $3g - 3$ small curves.
- Their collars are pairwise disjoint.

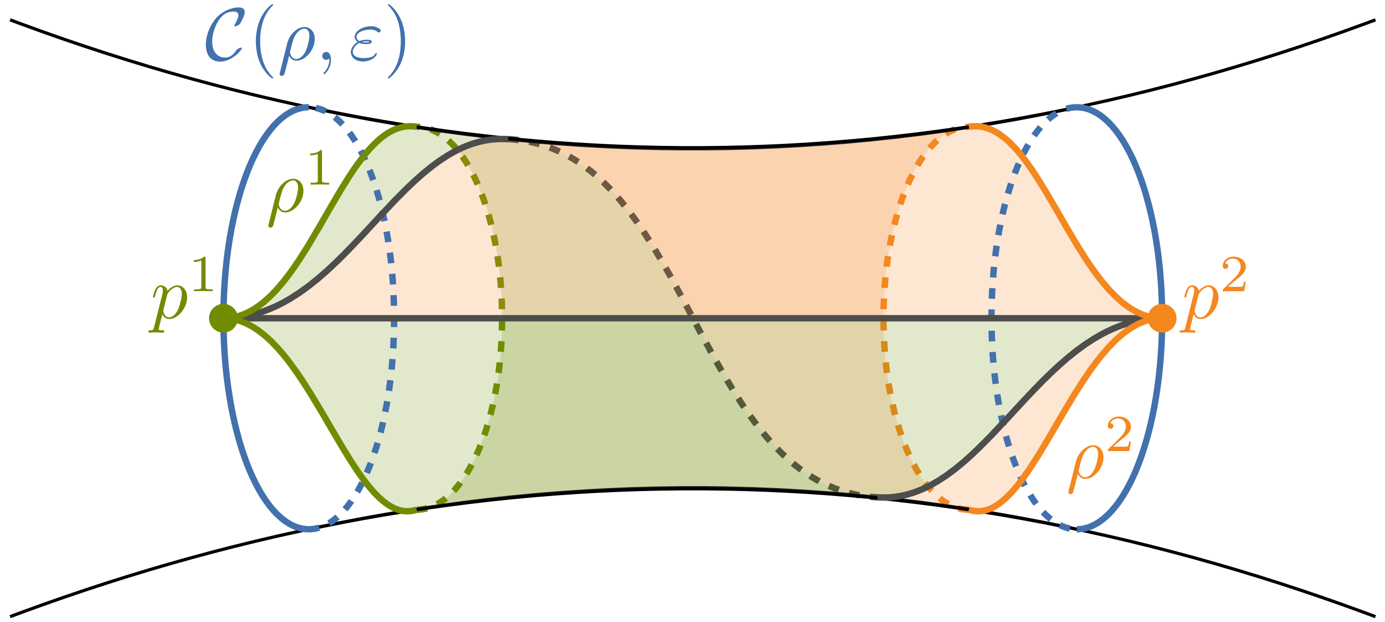
[Buser, 1992 (textbook)]

Pseudo ε-net

Schematic representation of a pseudo ε-net
of a hyperbolic surface (fig: D. Perrot)



Pseudo ε-net

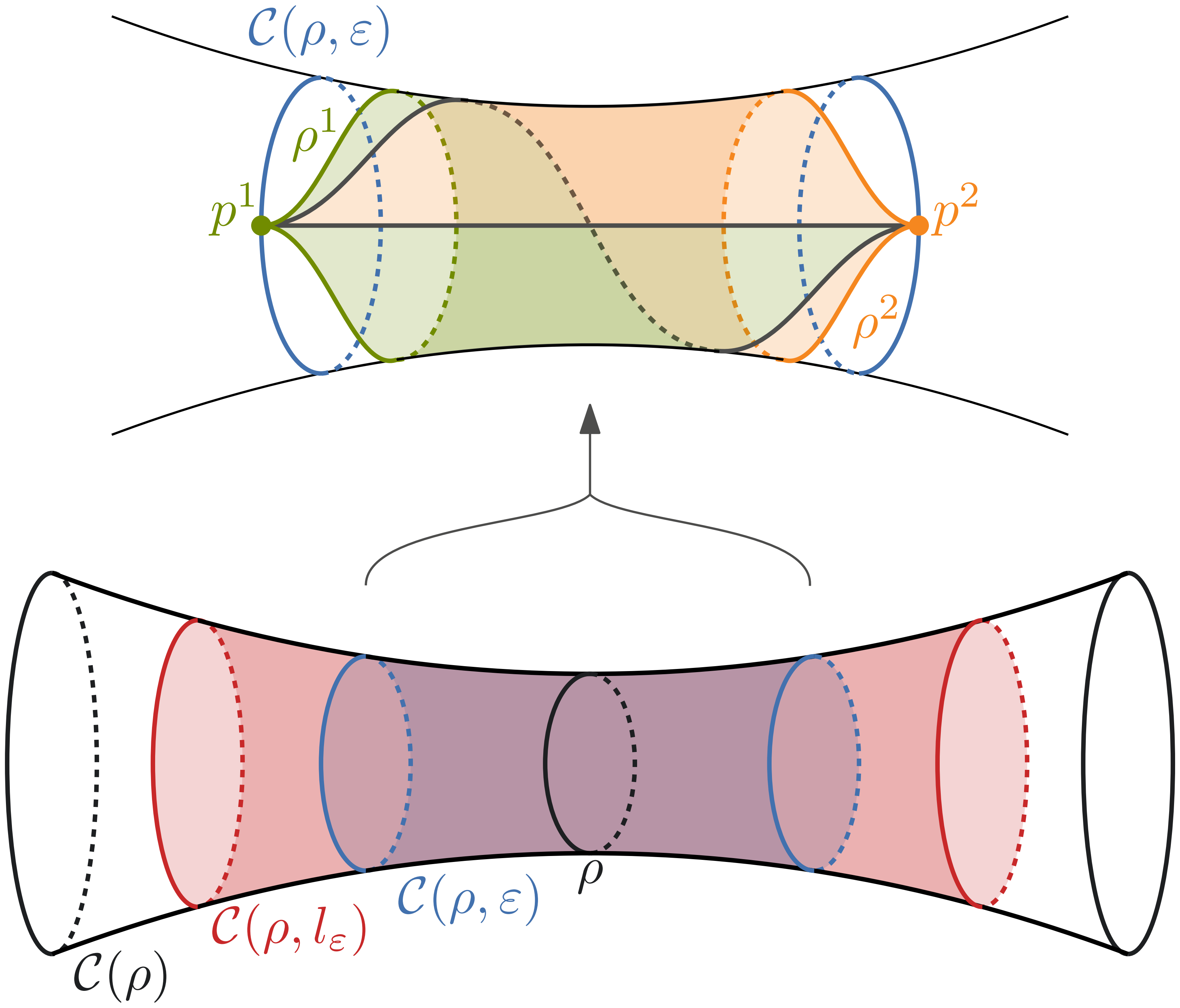


Pseudo ε-net

The first time that a point is inserted in an l_ε-collar:

1. Cancel the insertion;
2. Insert p^1 and p^2 on the boundary of the ε-collar;
3. Triangulate the ε-collar.

Keep executing the standard ε-net algorithm
Do not touch the triangles in the ε-collar.



Pseudo ε-net

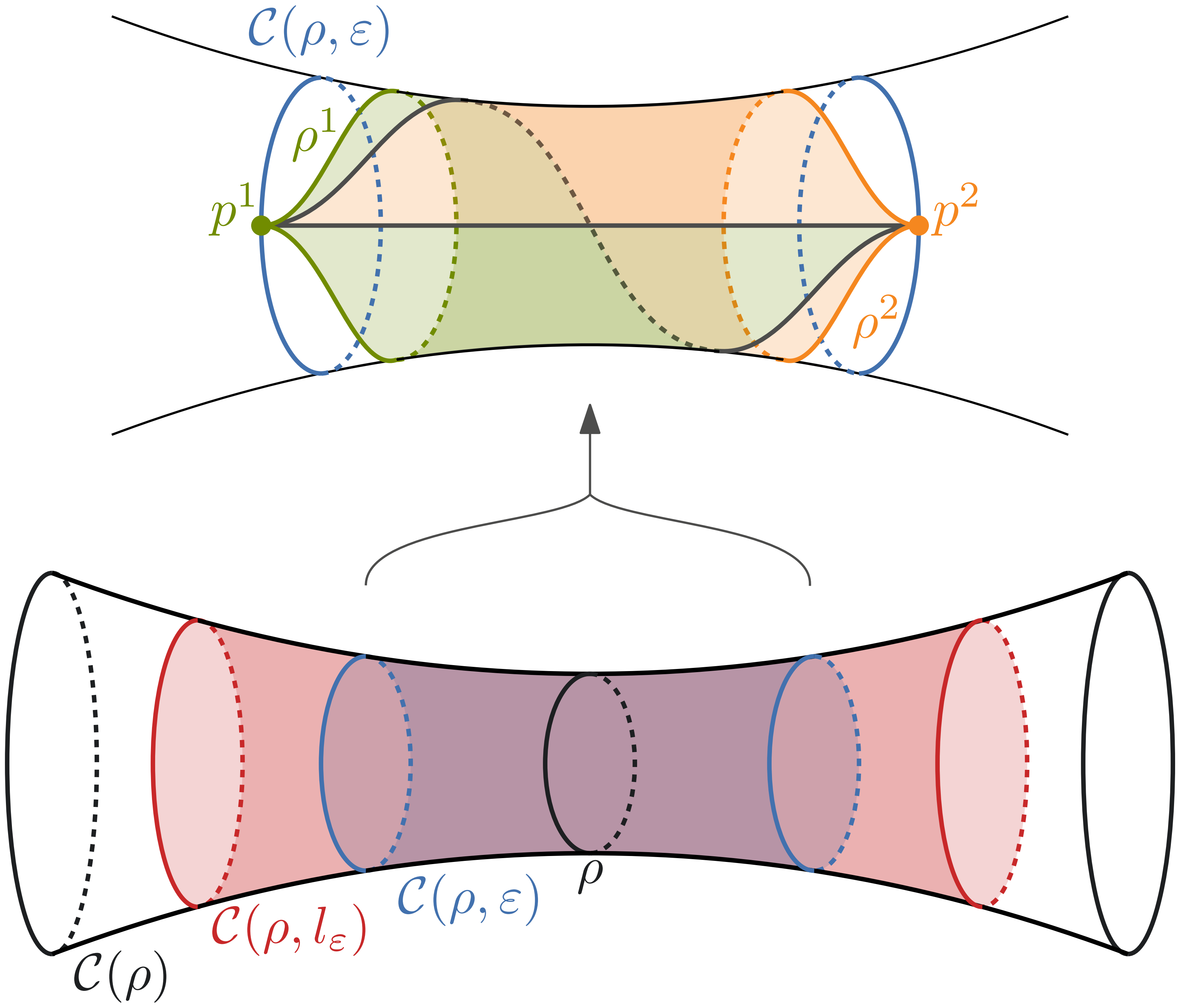
The first time that a point is inserted in an l_ε-collar:

1. Cancel the insertion;
2. Insert p^1 and p^2 on the boundary of the ε-collar;
3. Triangulate the ε-collar.

Keep executing the standard ε-net algorithm
Do not touch the triangles in the ε-collar.

Lemma
No point will be inserted in an ε-collar once it has been handled.

Worst-case complexity
Still $O(N^2)$ but with N independent from σ.



1. Introduction
2. The ε-net algorithm
3. **Implementation**
4. Conclusion

Code architecture

Implemented with the C++ CGAL library.
→ *The* open-source computational geometry library.

CGAL

industry
Geometry Factory

academia
Test of Time Award
(SoCG 2023)

Code architecture

Implemented with the C++ CGAL library.
→ *The* open-source computational geometry library.

Relies on the *Triangulation on hyperbolic surface* package. [Dubois et al.]

Triangulation_on_hyperbolic_surface_2 (cmap with cross-ratios + 1 anchor)

- flip(dart): flips the edge of the given dart
- lift(): computes a lift of every triangle in a connected way

Code architecture

Implemented with the C++ CGAL library.
→ *The* open-source computational geometry library.

Relies on the *Triangulation on hyperbolic surface* package. [Dubois et al.]

Triangulation_on_hyperbolic_surface_2 (cmap with cross-ratios + 1 anchor)

- flip(dart): flips the edge of the given dart
- lift(): computes a lift of every triangle in a connected way

↑

Delaunay_triangulation_on_hyperbolic_surface_2 (cmap with cross-ratios and anchors)

- locate(point, opt: anchor): locates the point in the triangulation starting from the given anchor
- insert(point, opt: anchor): inserts the point in the triangulation
- epsilon_net(epsilon): computes an ε-net of the surface

Robustness and efficiency

We need to avoid algebraic numbers and use rational numbers instead.

Generation of hyperbolic surfaces

Fundamental polygon with rational coordinates.

from the CGAL package ($g = 2$) or from Despré and Pouget's generation (any g)

Robustness and efficiency

We need to avoid algebraic numbers and use rational numbers instead.

Generation of hyperbolic surfaces

Fundamental polygon with rational coordinates. → dense in the set of hyperbolic surfaces

from the CGAL package ($g = 2$) or from Despré and Pouget's generation (any g)

Robustness and efficiency

We need to avoid algebraic numbers and use rational numbers instead.

Generation of hyperbolic surfaces

Fundamental polygon with rational coordinates. → dense in the set of hyperbolic surfaces

from the CGAL package ($g = 2$) or from Despré and Pouget's generation (any g)

...but circumcenters are still algebraic numbers. → round their coordinates to rational numbers

(ok if the ε-packing property is maintained, check at the end)

Naively

`CGAL::Sqrt_extension` → `double` → rational number type

Robustness and efficiency

We need to avoid algebraic numbers and use rational numbers instead.

Generation of hyperbolic surfaces

Fundamental polygon with rational coordinates. → dense in the set of hyperbolic surfaces

from the CGAL package ($g = 2$) or from Despré and Pouget's generation (any g)

...but circumcenters are still algebraic numbers. → round their coordinates to rational numbers

(ok if the ε-packing property is maintained, check at the end)

Naively

`CGAL::Sqrt_extension` → `double` → rational number type

generally works for $g = 2$ or 3 ...but does not output an ε-net when $g > 3$ or when σ is very small.

Increase precision

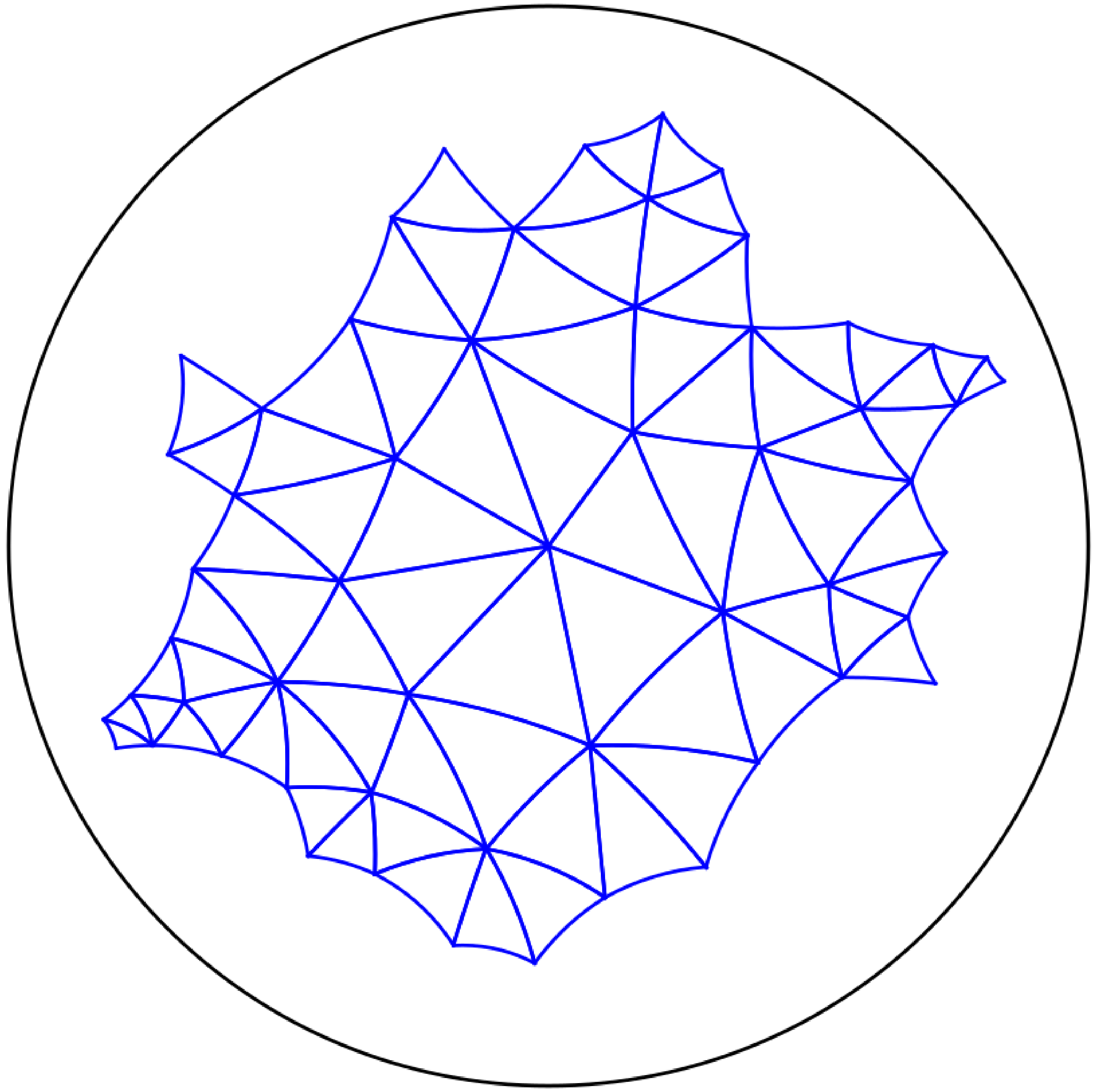
`CGAL::Sqrt_extension` → `CGAL::Gmpfr` → `CGAL::Gmpq`

precision given by the user

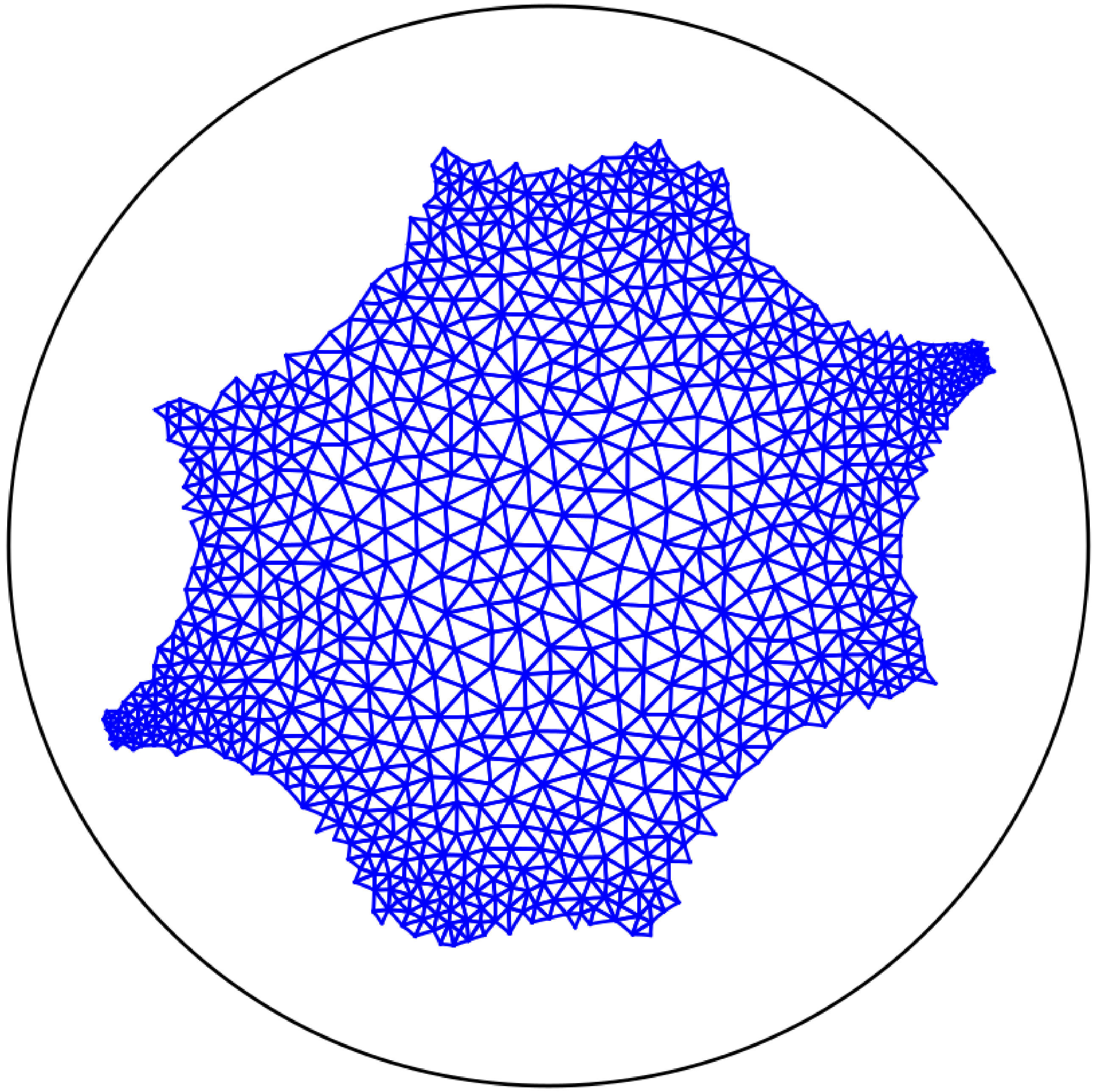
Produces valid ε-nets for any surface!

Output

with the lift() method



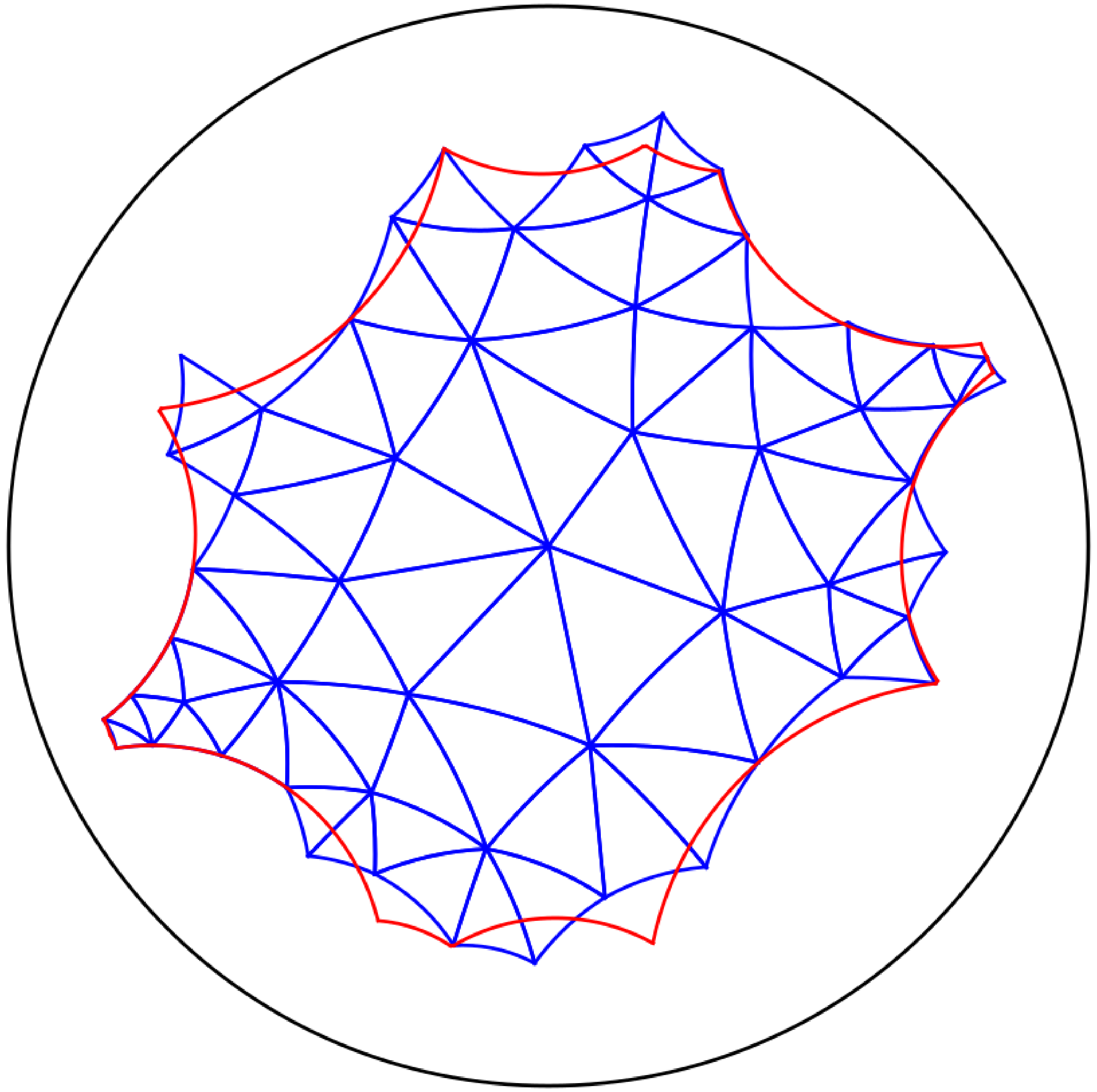
0.5-net of a genus 2 surface



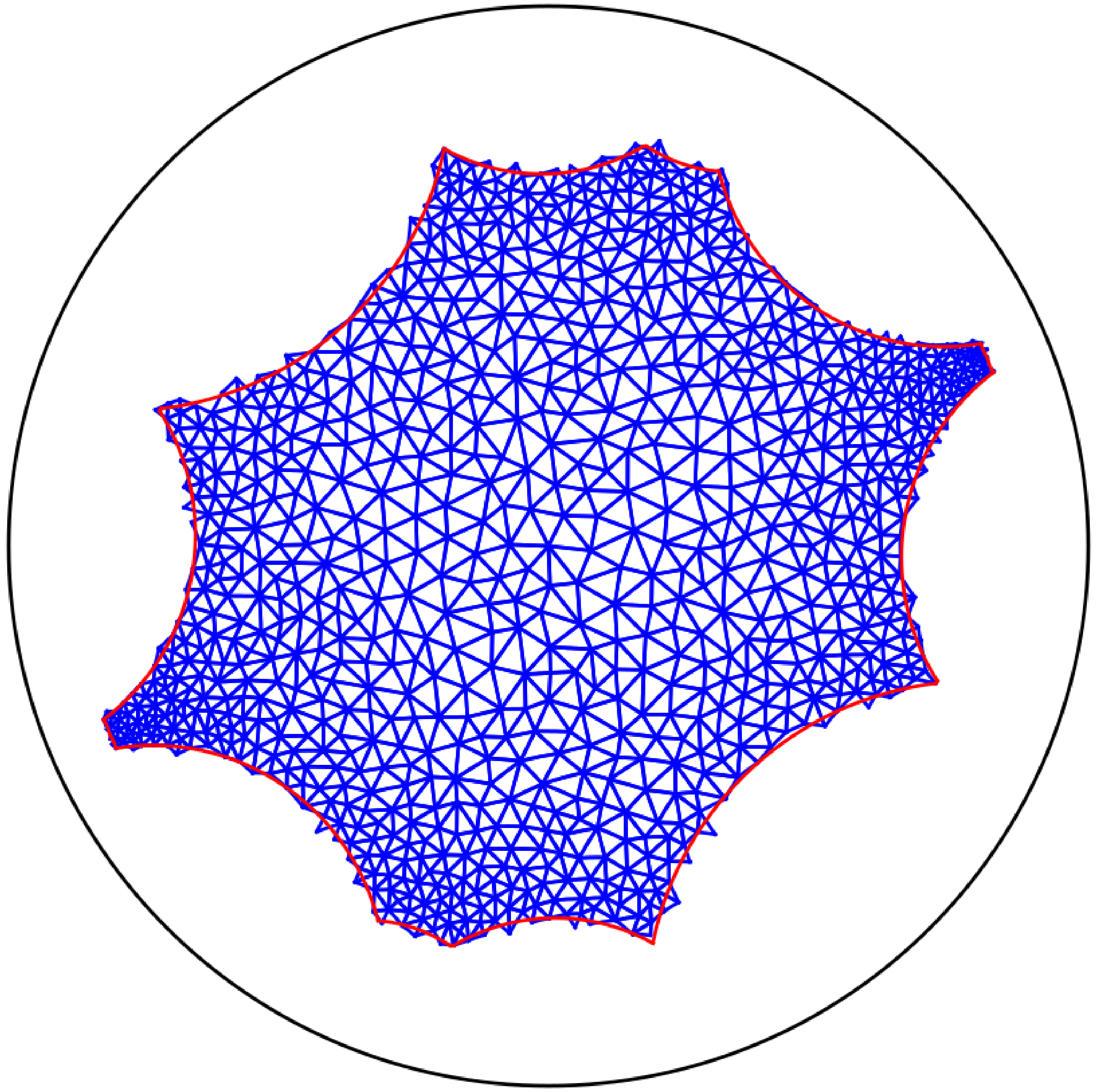
0.1-net of the same surface

Output

with the lift() method



0.5-net of a genus 2 surface

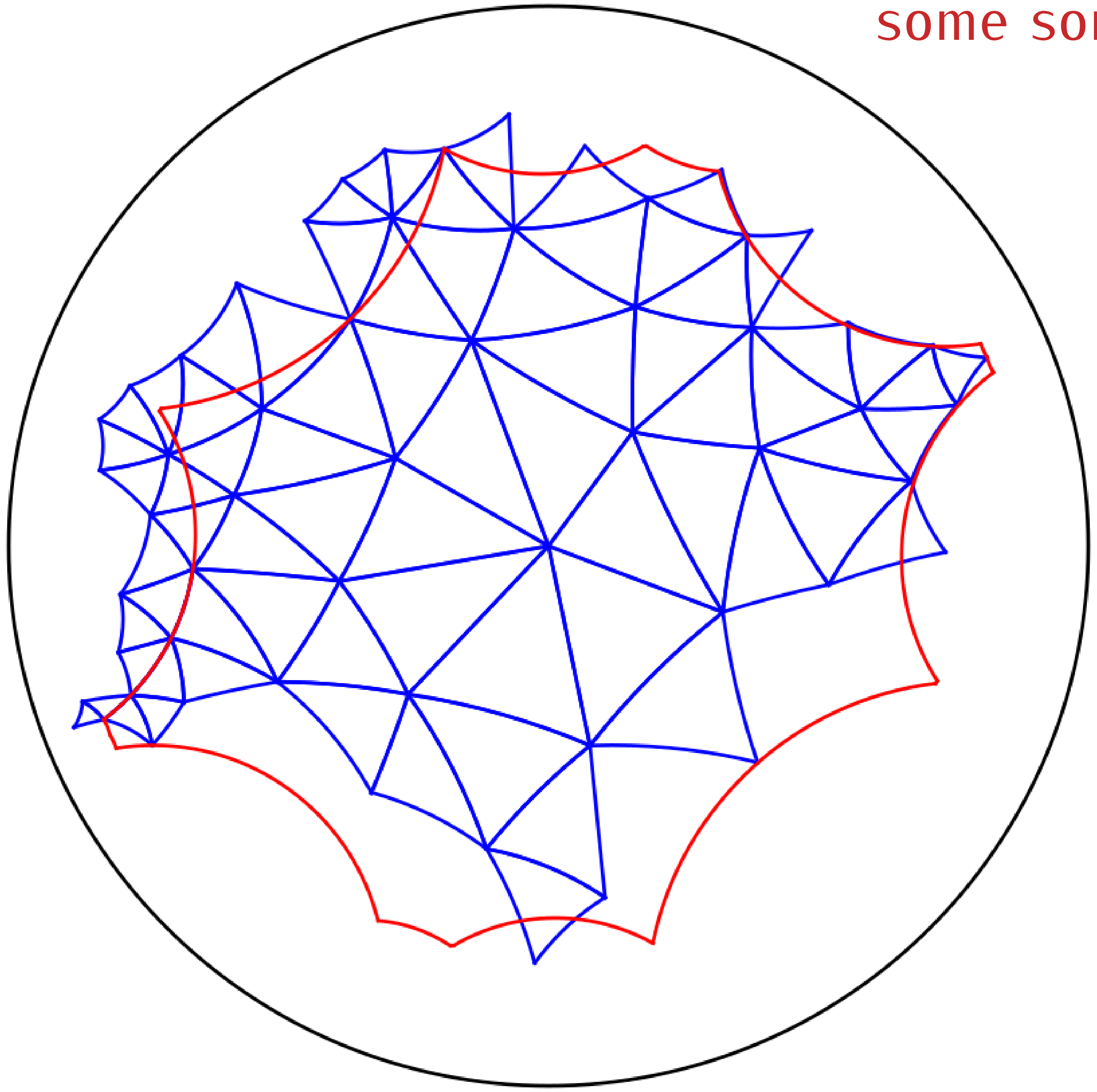


0.1-net of the same surface

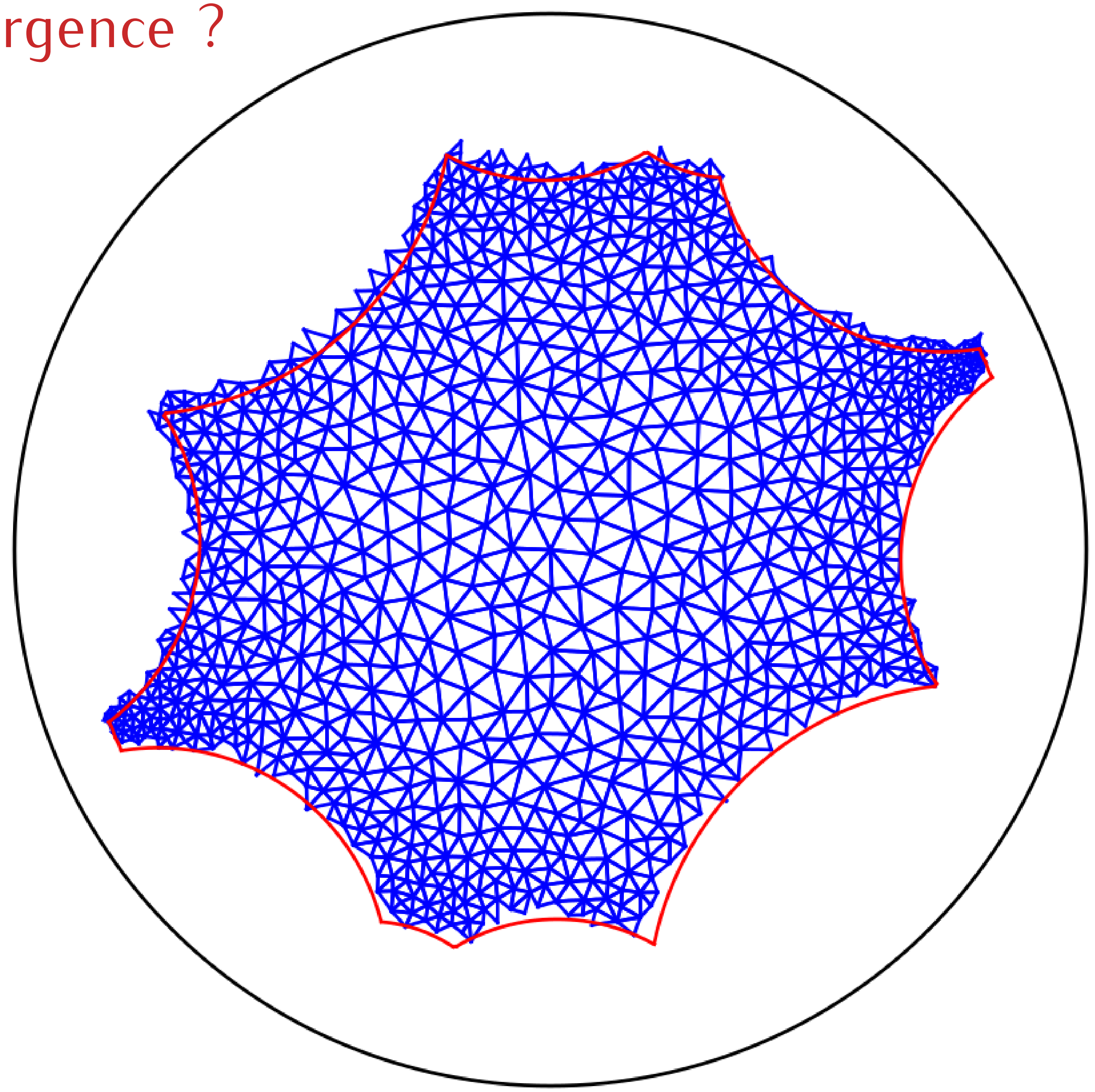
Output

with a BFS algorithm ("combinatorial Dirichlet domain")

some sort of convergence ?

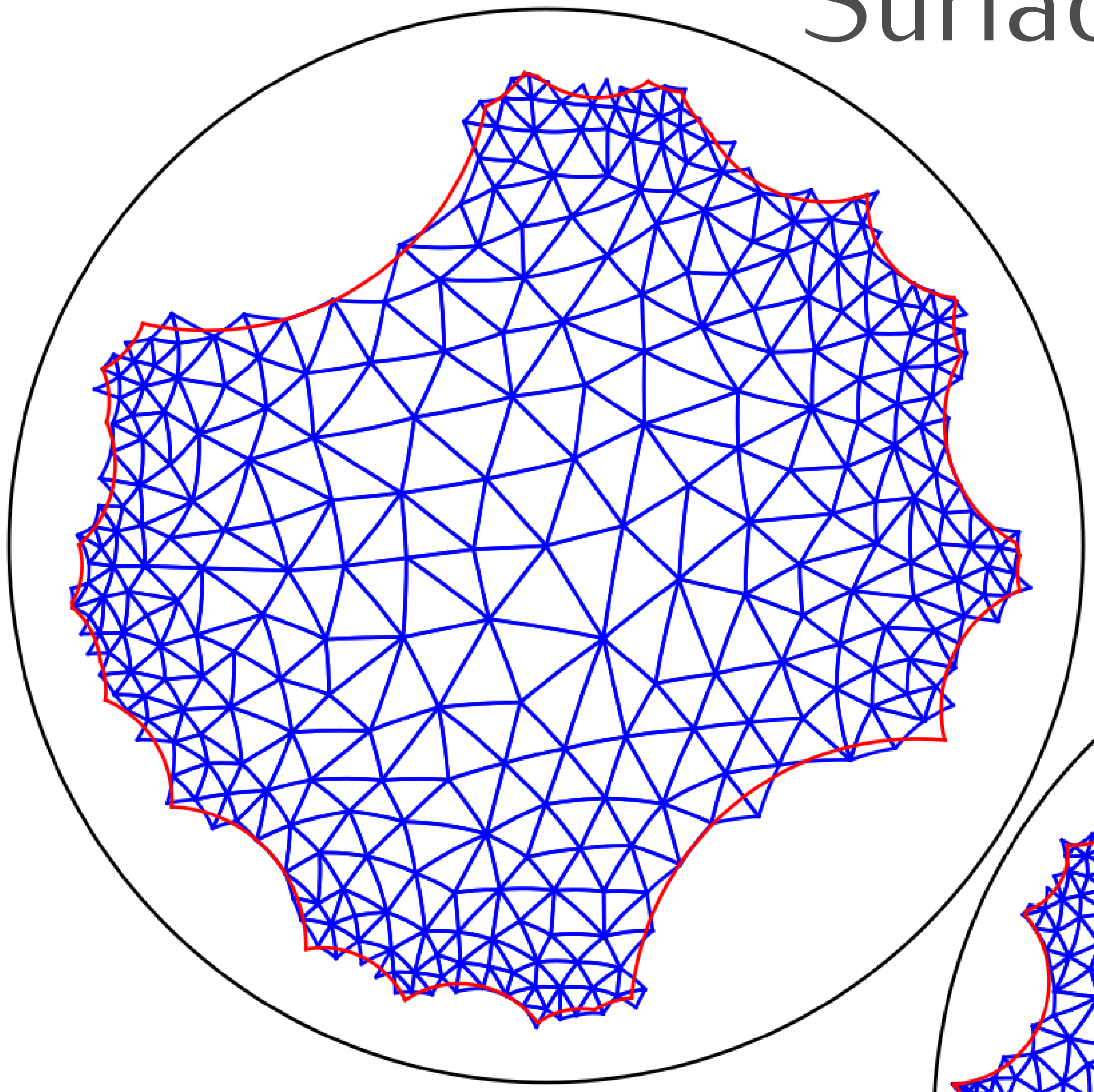


0.5-net of a genus 2 surface



0.1-net of the same surface

Surfaces with higher genus



$g = 3$

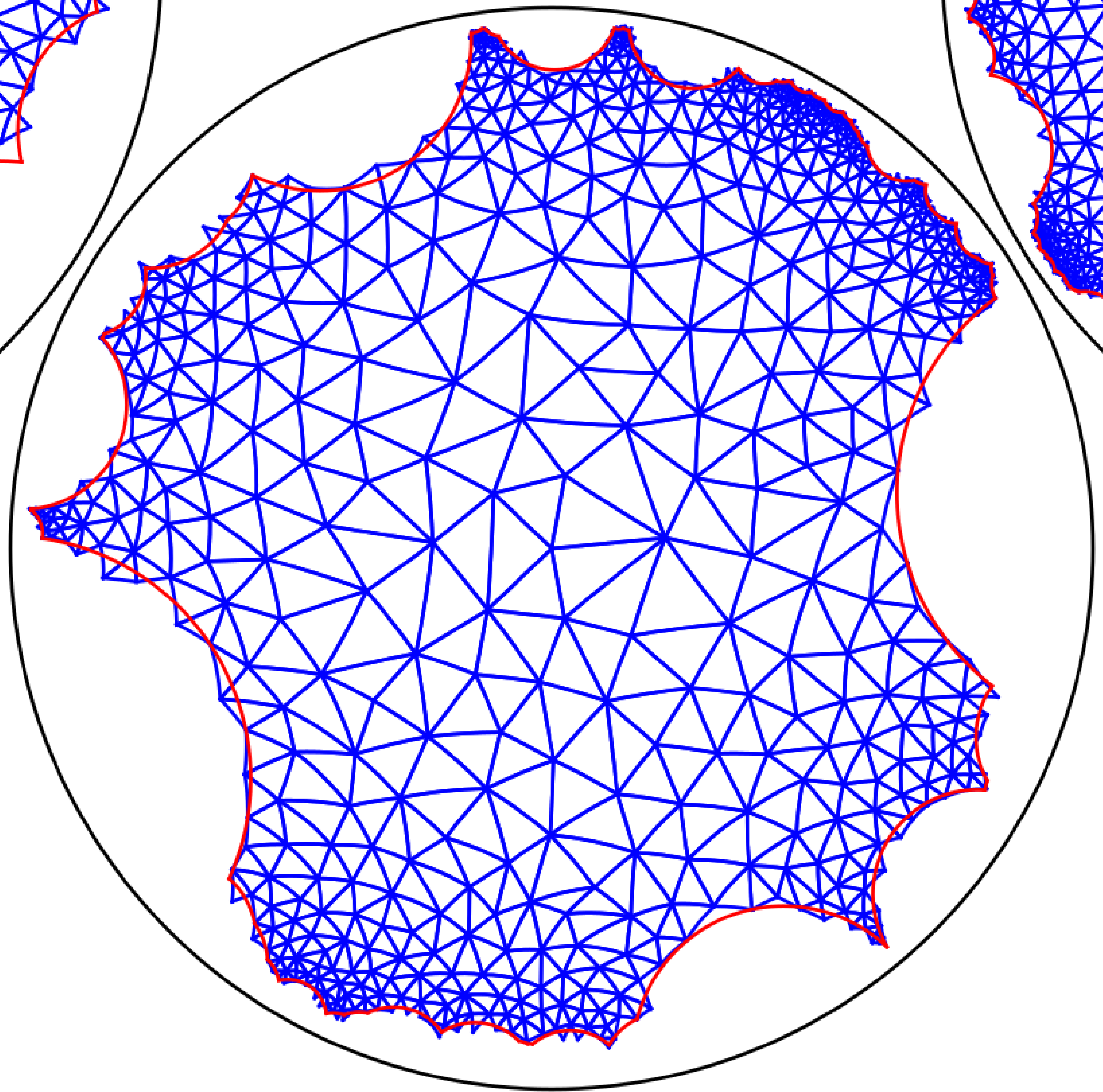
precision 53 (double)

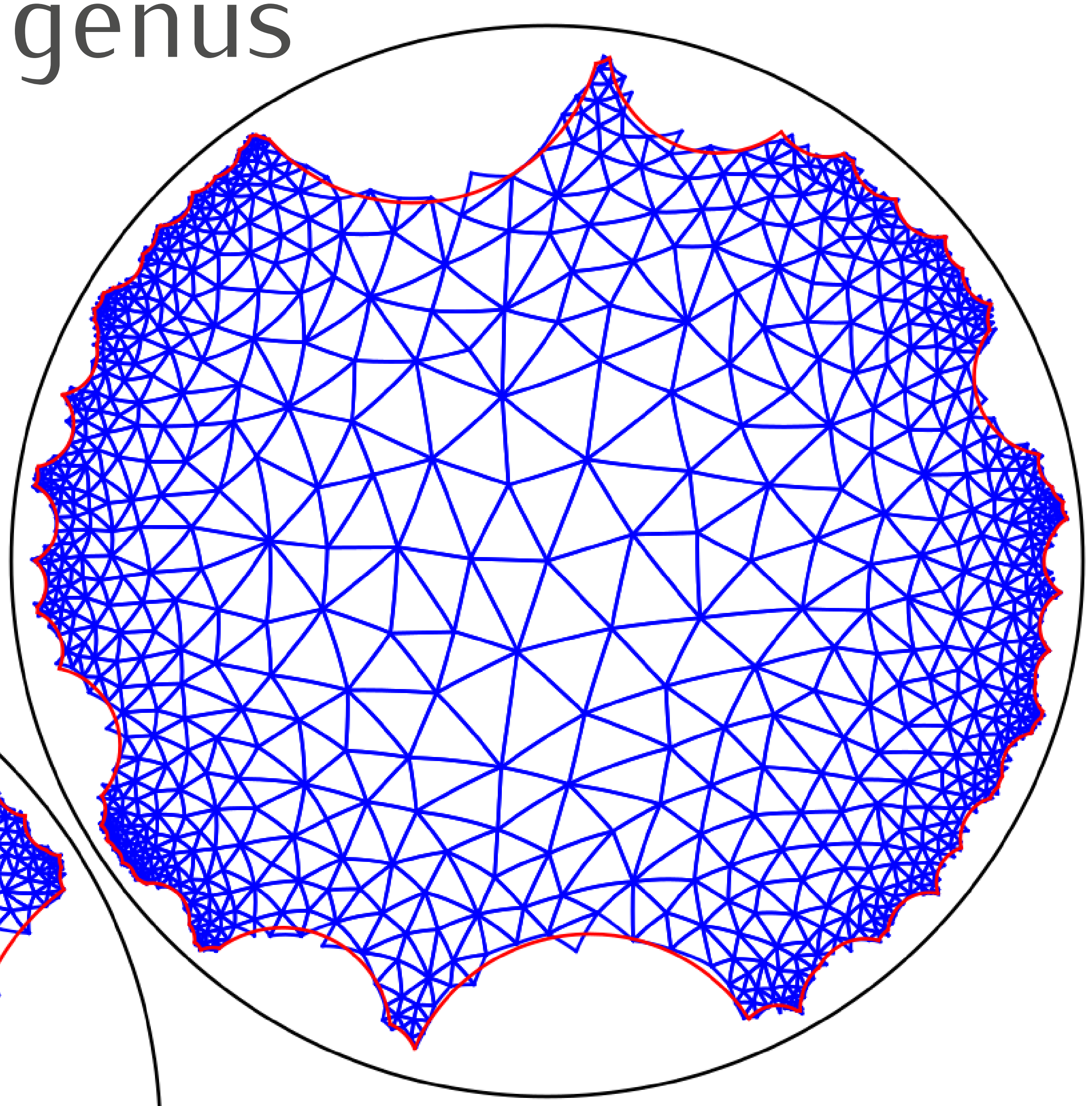
With the lift() method

$\varepsilon = 0.25$

precision 106 $(= 2 \times 53)$

$g = 5$





$g = 7$

precision 106 $(= 2 \times 53)$

Experiments

How does the algorithm behave in practice?

Experiments on 180 genus 2 surfaces (large systoles).

Experiments

How does the algorithm behave in practice?

Experiments on 180 genus 2 surfaces (large systoles).

Sanity check: number of vertices
$N = 54\%$ of the upper bound (on average).

ε	0.50	0.40	0.30	0.20	0.10	0.05	0.01
Average number of points	34	54	96	216	865	3,454	86,314
Upper bound ($16/\varepsilon^2$)	64	100	178	400	1,600	6,400	160,000

Average number of points in the obtained ε-nets

Experiments

How does the algorithm behave in practice?

Experiments on 180 genus 2 surfaces (large systoles).

Point location

How many triangles are visited when locating approximate circumcenters?

- In practice: small constant number (< 0.35);
- vs $O(i)$ at step i in theory.
- 68% of approximate circumcenters lie in their triangle (on average).

Experiments

How does the algorithm behave in practice?

Experiments on 180 genus 2 surfaces (large systoles).

Point location

How many triangles are visited when locating approximate circumcenters?

- In practice: small constant number (< 0.35);
- vs $O(i)$ at step i in theory.
- 68% of approximate circumcenters lie in their triangle (on average).

Number of flips

How many edges are flipped to maintain the DT after an insertion?

- In practice: small constant number (≈ 3, decreases when $\varepsilon \to 0$);
- Therefore, $O(N)$ flips overall vs $O(N^2)$ in theory.

Experiments

How does the algorithm behave in practice?

Experiments on 180 genus 2 surfaces (large systoles).

Point location

How many triangles are visited when locating approximate circumcenters?

- In practice: small constant number (< 0.35);
- vs $O(i)$ at step i in theory.
- 68% of approximate circumcenters lie in their triangle (on average).

Number of flips

How many edges are flipped to maintain the DT after an insertion?

- In practice: small constant number (≈ 3, decreases when $\varepsilon \to 0$);
- Therefore, $O(N)$ flips overall vs $O(N^2)$ in theory.

The algorithm runs in linear time in practice instead of $O(N^2) = O(1/\varepsilon^4)$ (worst-case).

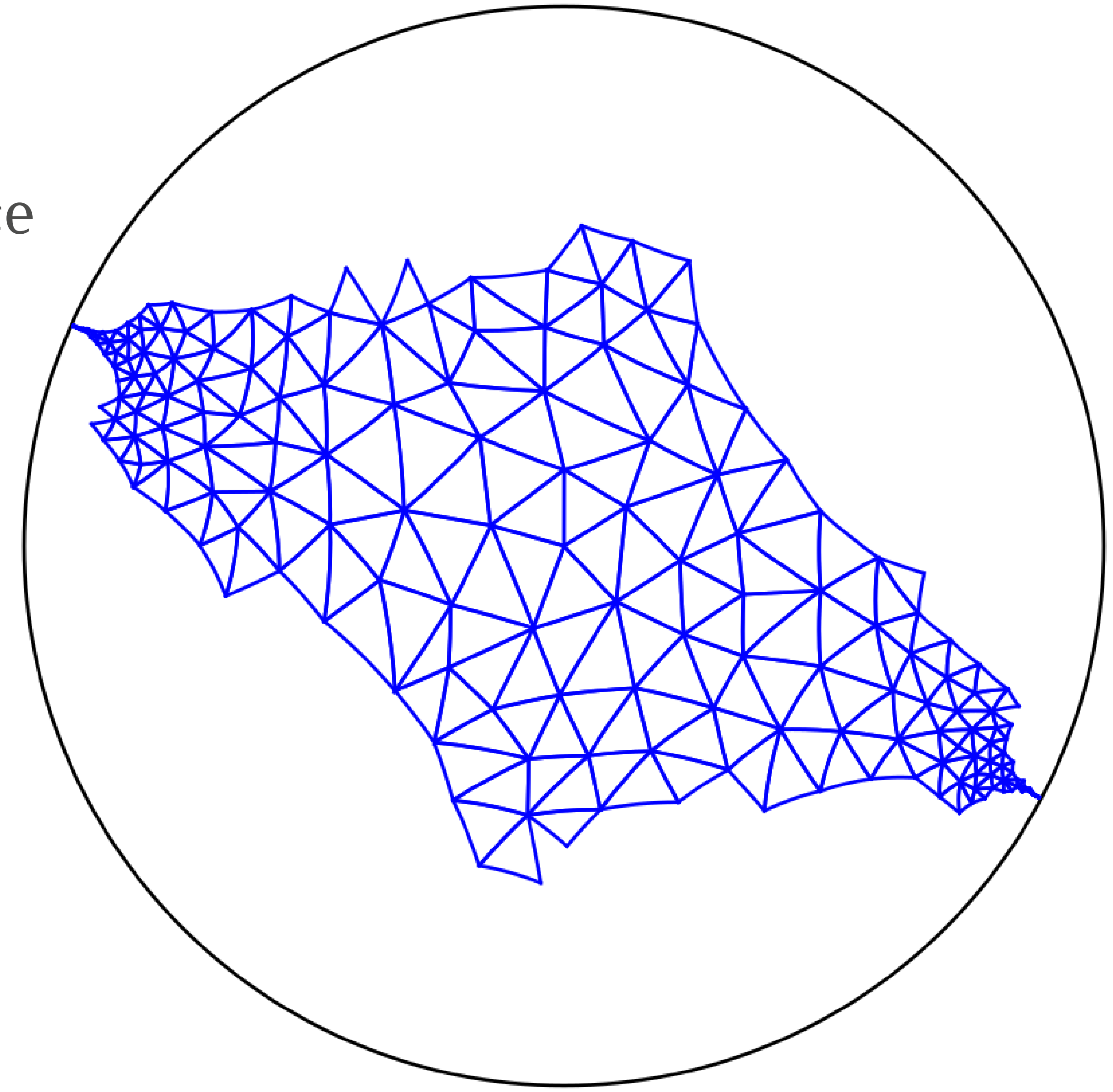
[ESA 2025]

Surface with a small systole

Hand-crafted a genus 2 surface with $\sigma < 10^{-4}$.

0.25-net of the surface with a small systole

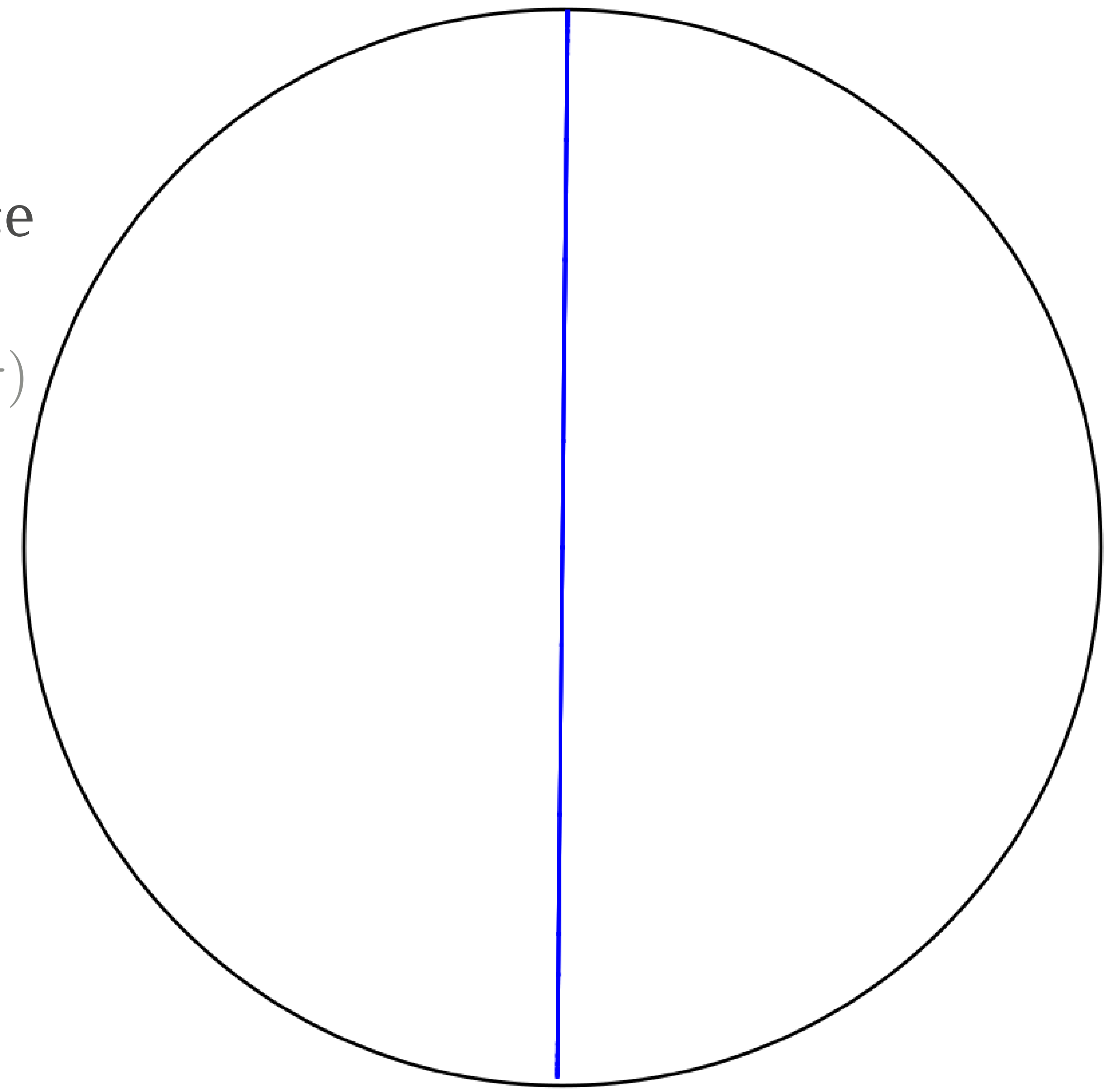
drawing centered on the thick part



Surface with a small systole

Hand-crafted a genus 2 surface with $\sigma < 10^{-4}$.

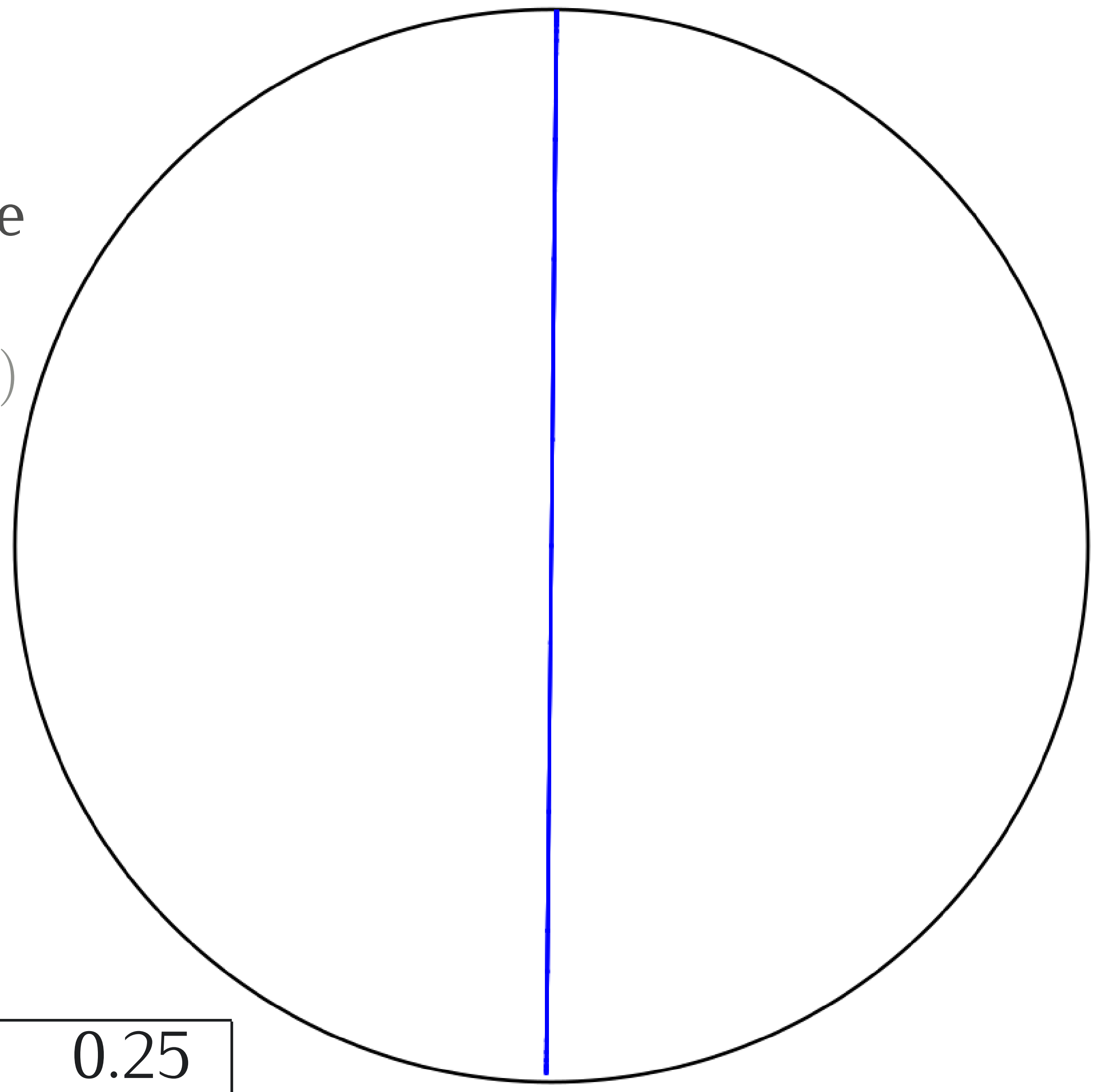
0.25-net of the surface
with a small systole
drawing centered on $\mathcal{C}(\sigma)$



Surface with a small systole

Hand-crafted a genus 2 surface with $\sigma < 10^{-4}$.

0.25-net of the surface
with a small systole
drawing centered on $\mathcal{C}(\sigma)$



Reminder
$N \leqslant 16(g-1)\left(\frac{1}{\varepsilon^2}+\frac{1}{\sigma^2}\right)$

Width of the ε-collar around σ

$$w(\sigma,\varepsilon) = 2\,\mathrm{arcosh}\left(\frac{\sinh(\varepsilon/2)}{\sinh(\sigma/2)}\right)$$

$w(10^{-4}, 0.5) \approx 18$

ε	0.50	0.45	0.40	0.35	0.30	0.25
Number of points	58	64	75	108	137	179
$0.54(16/\varepsilon^2 + w(10^{-4},\varepsilon)/\varepsilon)$	54	64	78	98	127	174

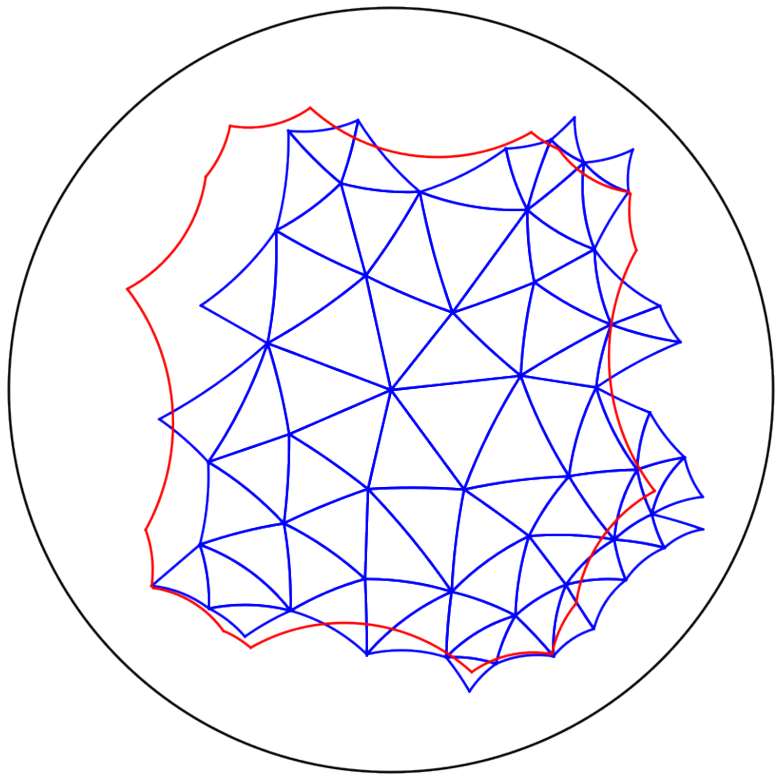
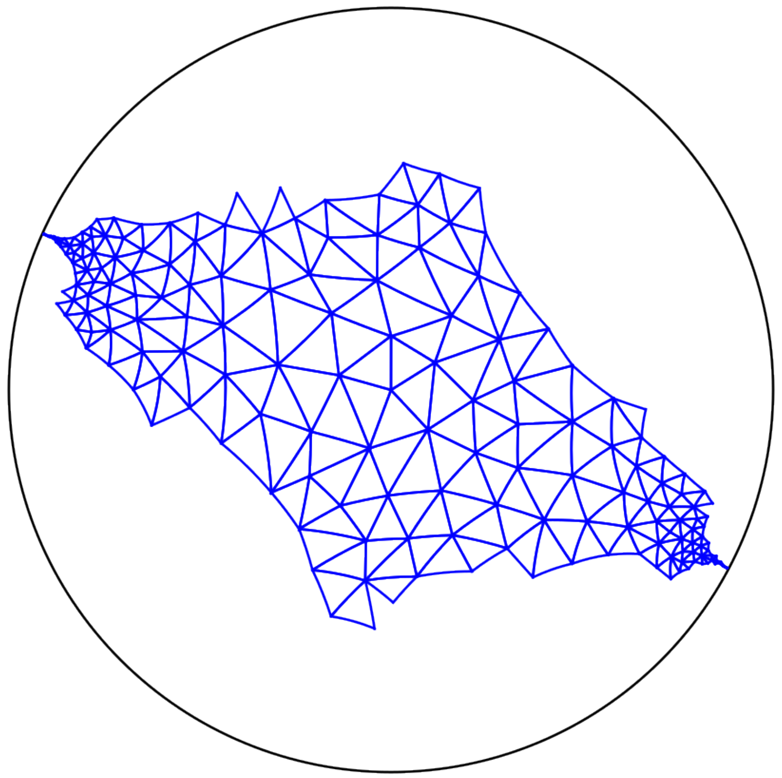
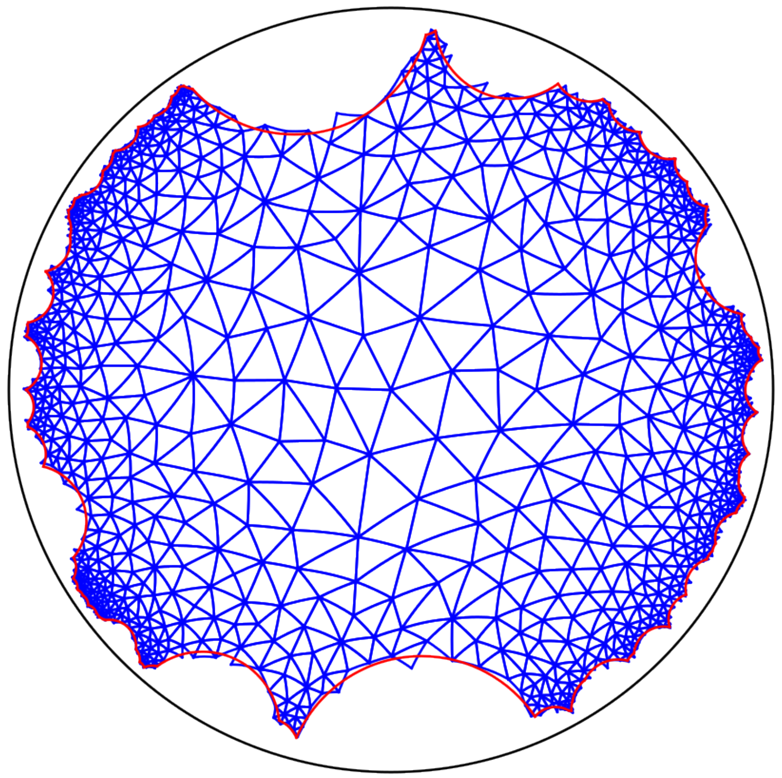
Number of points in an ε-net of this surface

1. Introduction
2. The ε-net algorithm
3. Implementation
4. **Conclusion**

Conclusion

In this thesis

- Designed and implemented a Delaunay refinement algorithm to compute ε-nets on hyperbolic surfaces;
- Managed to use rational numbers to ensure robustness and efficiency;
- Modified the algorithm to compute a pseudo ε-net (not implemented).

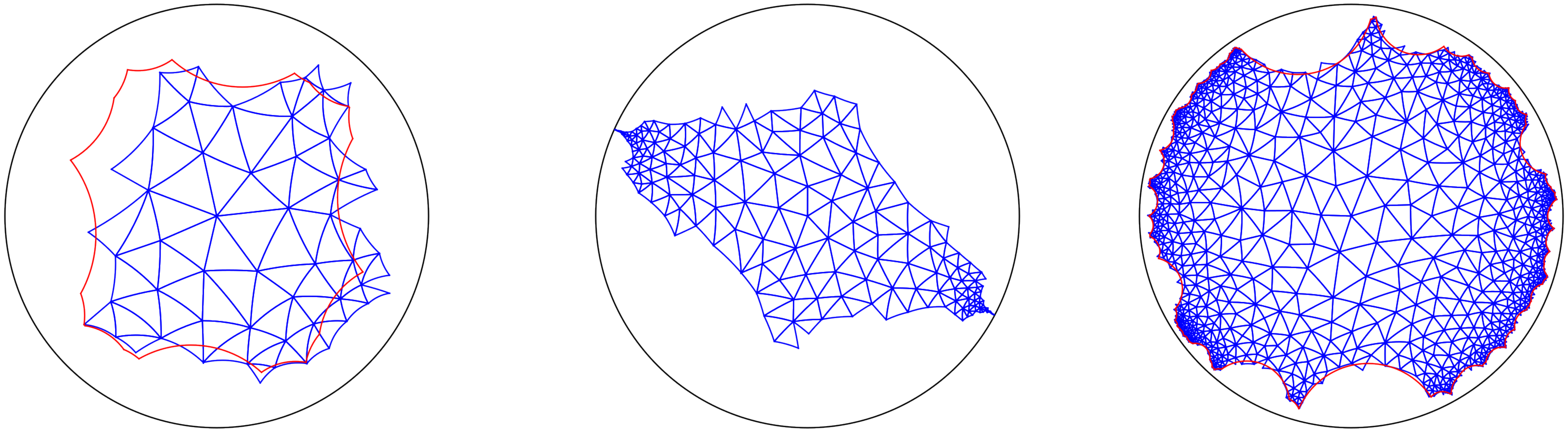
Conclusion

In this thesis

- Designed and implemented a Delaunay refinement algorithm to compute ε-nets on hyperbolic surfaces;
- Managed to use rational numbers to ensure robustness and efficiency;
- Modified the algorithm to compute a pseudo ε-net (not implemented).

Future work

- Integrate our code into CGAL (current work);
- Implement the pseudo ε-net algorithm.
- Design and implement approximation algorithms using an ε-net as input;
- Prove an average-time complexity for the ε-net algorithm;
- Formalize and prove the convergence of a combinatorial Dirichlet domain to a Dirichlet domain;



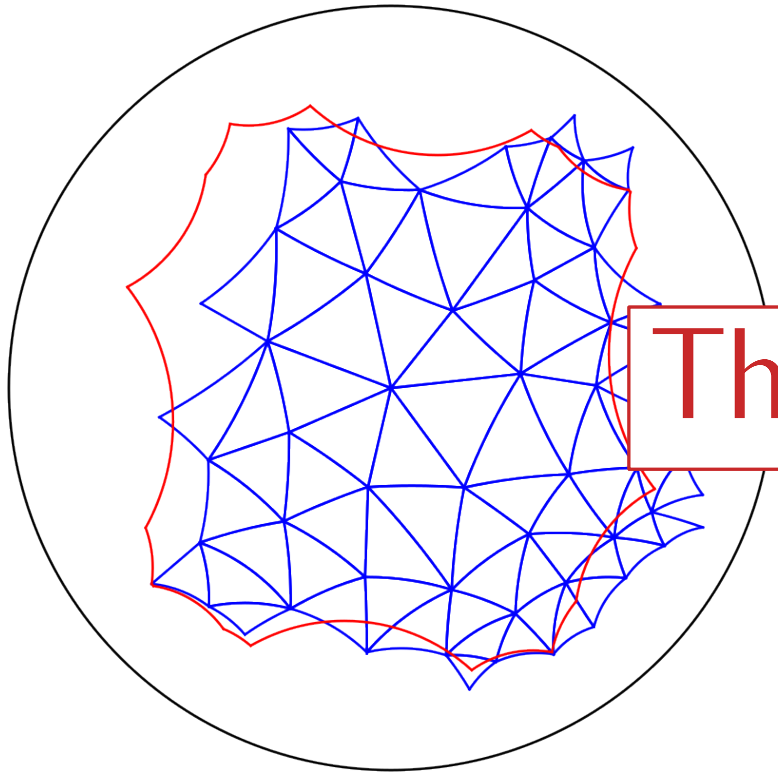
Conclusion

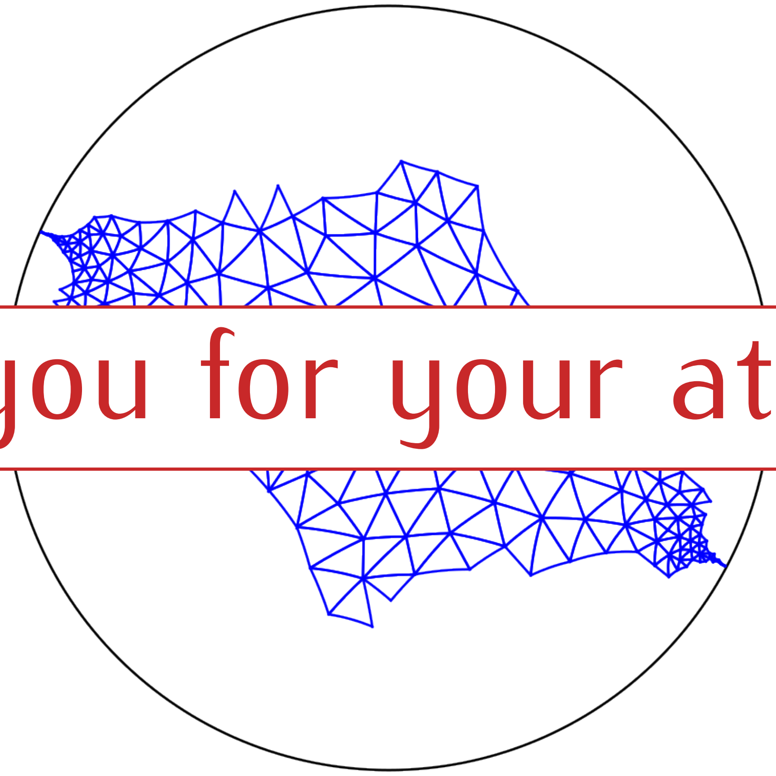
In this thesis

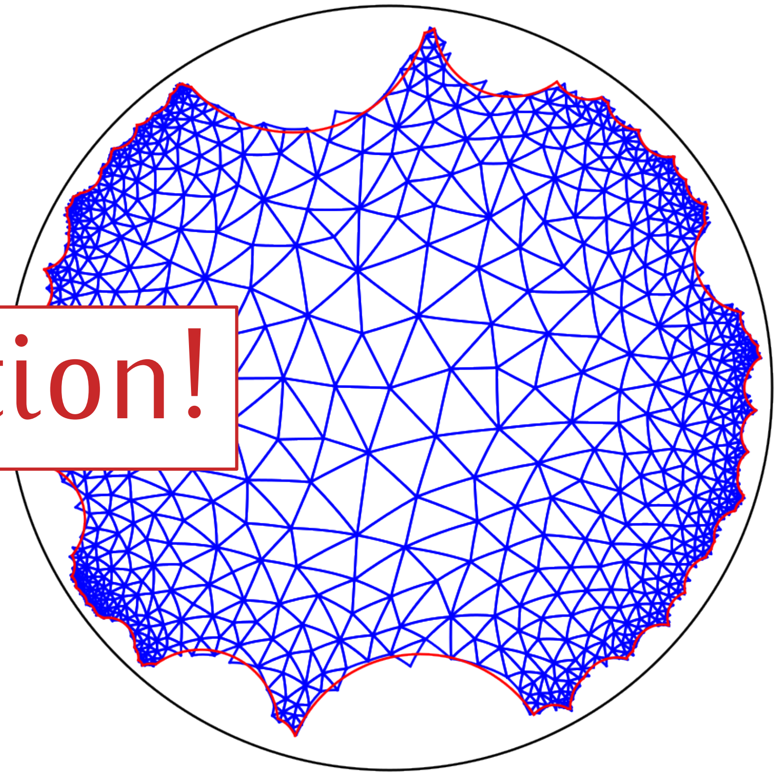
- Designed and implemented a Delaunay refinement algorithm to compute ε-nets on hyperbolic surfaces;
- Managed to use rational numbers to ensure robustness and efficiency;
- Modified the algorithm to compute a pseudo ε-net (not implemented).

Future work

- Integrate our code into CGAL (current work);
- Implement the pseudo ε-net algorithm.
- Design and implement approximation algorithms using an ε-net as input;
- Prove an average-time complexity for the ε-net algorithm;
- Formalize and prove the convergence of a combinatorial Dirichlet domain to a Dirichlet domain;







Thank you for your attention!