Computing an ε-net of a closed hyperbolic surface

Vincent Despré, Camille Lanuel, and Monique Teillaud

Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France

Abstract

An ε-net of a metric space X is a set of points P of X such that the balls of radius ε centered at points of P cover X, and two distinct points of P are at least ε apart. We present an algorithm to compute an ε-net of a closed hyperbolic surface and analyze its complexity.

Related Version `https://hal.science/hal-04466350`

1 Introduction

This paper focuses on hyperbolic surfaces, i.e., surfaces with a metric of constant negative curvature. These surfaces have been extensively examined from a mathematical perspective, due to their generic nature: any Riemannian surface of genus at least two can be conformally mapped to a unique hyperbolic surface [16, Section IV.8].

Hyperbolic geometry also plays a key role in computer science. One of the most famous examples is found in the analysis of rotation distance in binary trees [19]. Hyperbolic geometry naturally emerges as a valuable tool for graph representation [14, 15]. The hyperbolic plane also serves as the preferred model for illustrating the universal cover of surfaces with genus at least 2, which has proven to be crucial in the proof of purely topological results [11, 6].

Delaunay triangulations of hyperbolic spaces and surfaces have been studied in the computational geometry community [2, 17, 12, 13]. In this line, we adapt Shewchuk's Delaunay refinement algorithm [18] to construct ε-nets of hyperbolic surfaces, opening the door to the design of efficient approximation algorithms. To the best of our knowledge, this is the first result of this kind.

Let us recall definitions [5]. Let (X, d) be a metric space and $\varepsilon > 0$. A set $P \subset X$ is an *ε-covering* if $\forall x \in X, d(x, P) \leqslant \varepsilon$, i.e., if the closed balls of radius ε centered at each $p \in P$ cover X. It is an *ε-packing* if $\forall p \neq q \in P, d(p, q) \geqslant \varepsilon$, i.e., if the open balls of radius $\varepsilon/2$ centered at each $p \in P$ are pairwise disjoint. An *ε-net* is both an ε-covering and an ε-packing. In this paper, we prove:

▶ **Proposition 1.** *Any ε-packing of a closed hyperbolic surface S of genus g and systole σ contains $N \leqslant 16(g-1)\left(1/\varepsilon^2 + 1/\sigma^2\right)$ points. If $\varepsilon < \sigma$, then $N \leqslant 16(g-1)/\varepsilon^2$.*

The case when $\varepsilon < \sigma$ corresponds to the situation when the surface has no ε-thin part (see Section 2.2).

▶ **Proposition 2.** *The Delaunay refinement algorithm computes an ε-net using at most $\left(10 + C'_h \operatorname{Diam}(S)^{6g-4}\right) N^2 + (N-1)(144g^2 - 104g + 35) - 10$ elementary operations, where C'_h is a constant depending on the metric h of S, and $\operatorname{Diam}(S)$ is the diameter of S.*

For a fixed surface, the complexity is then $O(1/\varepsilon^4)$.

The first result can be regarded as folklore. We prove it in Section 3 for completeness. The second proposition rises interesting obstacles to deal with. In particular, Shewchuk's refinement adds circumcenters of some triangles, which is not straightforward in our context, as locating a new point requires to construct a portion of the universal cover of the surface. We manage to bound the size of this portion.

40th European Workshop on Computational Geometry, Ioannina, Greece, March 13–15, 2024.
This is an extended abstract of a presentation given at EuroCG'24. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear eventually in more final form at a conference with formal proceedings and/or in a journal.

2 Background on hyperbolic surfaces and notation

We refer the reader to textbooks for more details, e.g. [4, 1].

A closed hyperbolic surface can be seen as the quotient $\mathbb{H}^2/\Gamma$ of the hyperbolic plane $\mathbb{H}^2$ under the action of a group Γ of orientation-preserving isometries. Throughout the paper, objects in $\mathbb{H}^2$ are denoted with a tilde $\widetilde{\cdot}$, while objects on S are denoted without. In particular, for an object o on S, $\widetilde{o}$ denotes any of its lifts in $\mathbb{H}^2$. To simplify the language, we often use the term *copy* to refer to an image of an object in $\mathbb{H}^2$ by an element of Γ.

We work with the Poincaré disk model in which the hyperbolic plane $\mathbb{H}^2$ is represented as the unit disk of the complex plane $\mathbb{C}$. The unit circle consists of points at infinity. The geodesics are either diameters of the unit disk, or circular arcs that meet the boundary circle orthogonally. The hyperbolic circles are Euclidean circles (but their hyperbolic and Euclidean centers differ). Orientation-preserving isometries are represented as matrices in $\mathbb{C}^{2\times 2}$.

2.1 Delaunay triangulation and Dirichlet domain

A triangulation T of S is a partition of S into triangles; note that edges may be loops. A triangulation of S is a Delaunay triangulation if for each triangle t of T and any of its lifts $\widetilde{t}$ in $\mathbb{H}^2$, the open disk circumscribing $\widetilde{t}$ contains no vertex of the (infinite) lift of T in $\mathbb{H}^2$ [12]. The Voronoi diagram is the dual of the Delaunay triangulation. The Dirichlet domain $\mathcal{D}_{\widetilde{x}}$ of a point $\widetilde{x} \in \mathbb{H}^2$ is the (closed) cell of $\widetilde{x}$ in the Voronoi diagram of its (infinite) orbit $\Gamma\widetilde{x}$. Unlike the Euclidean case, Γ is non-commutative, and the combinatorics of a Dirichlet domain depends on the point x (Figure 1). The number k of sides of $\mathcal{D}_{\widetilde{x}}$ satisfies $4g \leqslant k \leqslant 12g-6$ (see, e.g., [8]).

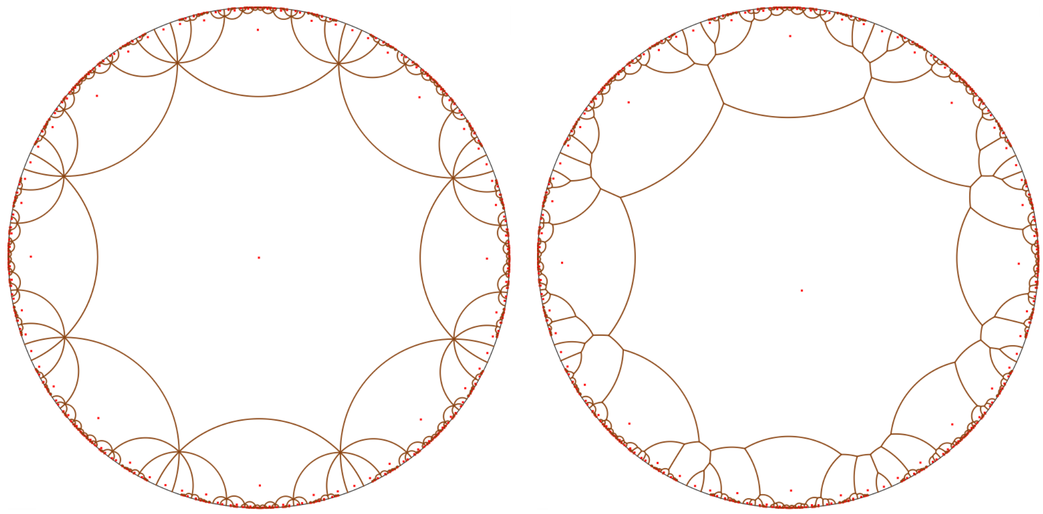

Figure 1 Dirichlet domains for the Bolza surface ($g = 2$). The domain on the left has $4g = 8$ sides and the one on the right has $12g - 6 = 18$ sides. Figure from [3].

In this paper, we assume that the input surface S is given by a Delaunay triangulation having a single vertex b, i.e., all Delaunay edges are loops based in b. The point b is arbitrary. This introduces no restriction, as such a representation can be computed for any closed hyperbolic surface, starting from a standard representation by a fundamental domain and side pairings [8].[1] The Dirichlet domain $\mathcal{D}_{\widetilde{b}}$ of some lift $\widetilde{b}$ of b can be computed together with the corresponding side pairings, which are generating the group Γ. The sides of $\mathcal{D}_{\widetilde{b}}$ are denoted as $s_i, i = 0, \dots, k-1$ and the corresponding side pairings as $\gamma_i, i = 0, \dots, k-1$ (here, side pairings are pairwise inverses).

2.2 Thin and thick parts

The *injectivity radius* $r_x(S)$ of S at a point x is the supremum of all $r > 0$ such that the open ball of radius r centered at x, $B(x, r) = \{y \in S \mid \delta_S(x, y) < r\}$, where δ_S is the distance on S, is isometric to a disk in $\mathbb{H}^2$. In particular, $B(x, r)$ is a topologically embedded disk on S for all $r \leqslant r_x(S)$. The *systole* σ of a surface is the length of its shortest non-contractible curve, which we also denote by σ. The systole is related to the injectivity radius: $\sigma = 2 \cdot \inf \{r_x(S) \mid x \in S\}$.

For any $\varepsilon > 0$, the ε*-thin part* of S is $S^t_\varepsilon = \{x \in S \mid r_x(S) \leqslant \varepsilon/2\}$, and its ε*-thick part* is $S^T_\varepsilon = S \setminus S^t_\varepsilon$. Observe that if $\varepsilon < \sigma$, then there is no ε-thin part.

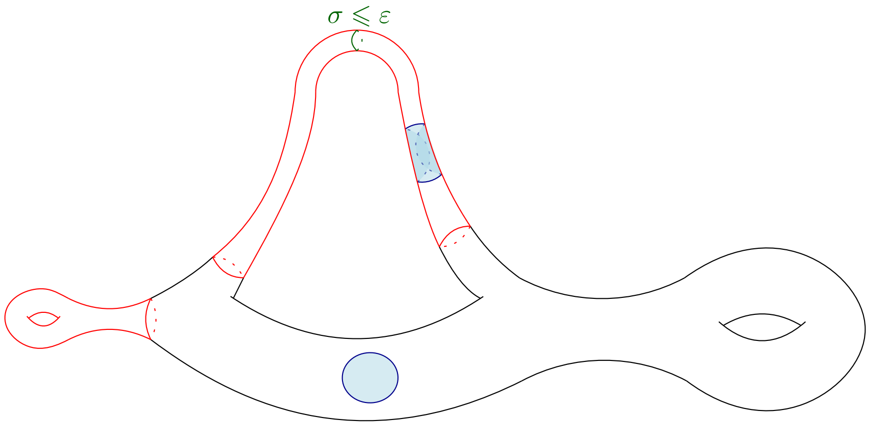

Figure 2 Thick and thin (red) parts of a hyperbolic surface. Disks of radius ε are shown in blue.

3 Proof of proposition 1

Let P be an ε-packing of S. The open balls of radius $\varepsilon/2$ centered at the points of P on the ε-thick part S^T_ε are isometric to disks in $\mathbb{H}^2$ and are pairwise disjoint. The area of such a disk centered at a point p is $\mathcal{A}\left(B\left(p, \frac{\varepsilon}{2}\right)\right) = 4\pi \sinh^2\left(\frac{\varepsilon}{4}\right)$ [1, Theorem 7.2.2]. Since $\sinh(x) \geqslant x$ for all $x \geqslant 0$, we have $\mathcal{A}\left(B\left(p, \frac{\varepsilon}{2}\right)\right) \geqslant \pi\varepsilon^2/4$.

Let N^T be the number of points of P on the ε-thick part S^T_ε. By the Gauss-Bonnet theorem, the area of the surface S is $\mathcal{A}(S) = 4\pi(g-1)$. Summing the above inequality over all the points in $P \cap S^T_\varepsilon$, we obtain $N^T \pi\varepsilon^2/4 \leqslant \sum_{p \in P \cap S^T_\varepsilon} \mathcal{A}\left(B\left(p, \frac{\varepsilon}{2}\right)\right) \leqslant 4\pi(g-1)$, thus

$$N^T \leqslant \frac{16(g-1)}{\varepsilon^2}. \tag{1}$$

The open balls of radius $\varepsilon/2$ in the ε-thin part S^t_ε, if it exists, that is if $\sigma \leqslant \varepsilon$, are also pairwise disjoint, but they are not isometric to disks in $\mathbb{H}^2$. However, by definition, the open balls of radius $\sigma/2$ are isometric to disks in $\mathbb{H}^2$. We can apply the reasoning that led to

[1] The common basepoint is denoted as b'' in [8].

inequality (1) for σ instead of ε, and obtain a bound on the number of points of P on the thin part S^t_ε: $N^t \leqslant 16(g-1)/\sigma^2$. The bound on the total number of points of P follows.

4 Construction of the ε-net

The input of the algorithm consists of the Delaunay triangulation of S with a single vertex b, together with the Dirichlet domain $\mathcal{D}_{\widetilde{b}}$ of a lift $\widetilde{b}$ and the group Γ generated by side-pairings. As mentioned in Section 2.1, this does not induce any loss of generality.

Our algorithm is inspired by Shewchuk's Delaunay refinement [18]. The general idea is to break each Delaunay triangle whose circumcircle has a radius greater than ε by inserting its circumscribing center in the triangulation.

We reuse the data structure proposed by Despré *et al.* for computing the Delaunay triangulation of a surface by edge flips [12]. A triangulation of S is represented by

- its vertices: a vertex p has constant-time access to its lift $\widetilde{p_b}$ in $\mathcal{D}_{\widetilde{b}}$ and one of its incident triangles;
- and its triangles: a triangle Δ has constant-time access to its three vertices $p_0^\Delta, p_1^\Delta, p_2^\Delta$, its three adjacent triangles, and three isometries $\gamma_0^\Delta = \mathbb{1}_\Gamma, \gamma_1^\Delta, \gamma_2^\Delta$ in Γ defined as follows.

A triangle $\Delta = (p_0^\Delta; p_1^\Delta; p_2^\Delta)$ does not always have a lift entirely included in $\mathcal{D}_{\widetilde{b}}$. However, it always has at least one lift with at least one vertex in $\mathcal{D}_{\widetilde{b}}$ (see Figure 3). Let us choose such a lift and denote it as $\widetilde{\Delta_0}$; up to a re-indexing of its vertices, $\widetilde{p_0^\Delta} \in \mathcal{D}_{\widetilde{b}}$. Then γ_1^Δ and γ_2^Δ are the isometries such that the other two vertices of $\widetilde{\Delta_0}$ are $\gamma_1^\Delta \widetilde{p_1^\Delta}$ and $\gamma_2^\Delta \widetilde{p_2^\Delta}$. Note that the other lifts of Δ having at least one vertex in $\mathcal{D}_{\widetilde{b}}$ can be retrieved by applying the inverses of these isometries to $\widetilde{\Delta_0}$. The union, on all triangles of the triangulation of S, of their lifts with at least one vertex in $\mathcal{D}_{\widetilde{b}}$ covers the fundamental domain $\mathcal{D}_{\widetilde{b}}$.

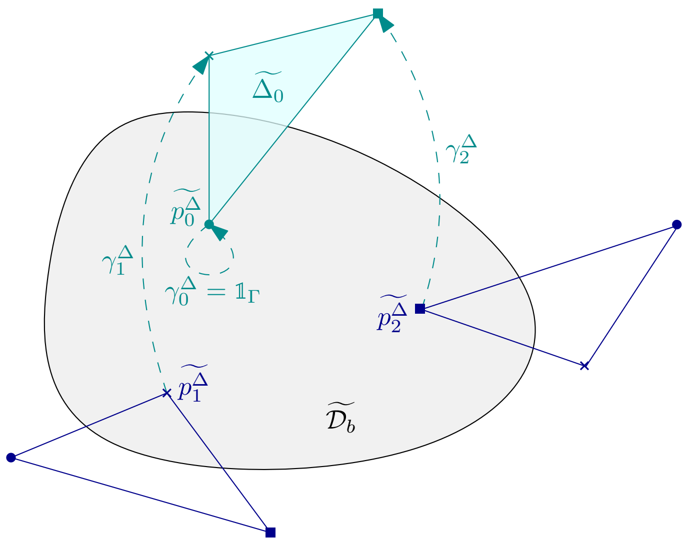

Figure 3 Example of a triangle Δ having three lifts with one vertex in $\mathcal{D}_{\widetilde{b}}$ (the hyperbolic triangles are schematically represented with straight edges).

We denote as $DT(\cdot)$ the Delaunay triangulation of a set of points on S.

Let us fix $\varepsilon > 0$. In a first step, the set of points is initialized as $P_1 = \{b\}$.

At each step $i \geqslant 2$, the algorithm inserts the circumscribing center c of a triangle Δ^ε whose radius is greater than ε. The set of points is updated as $P_i = P_{i-1} \cup \{c\}$ as well as the Delaunay triangulation $DT(P_i)$. To do so, several operations are needed.

We first compute the radius of $\widetilde{\Delta_0}$ for every triangle Δ of $DT(P_{i-1})$, until a triangle Δ^ε whose radius is at least ε is found.[2] The circumcenter $\widetilde{c}$ of the lift $\widetilde{\Delta_0^\varepsilon}$ is a lift of c, but it does not necessarily lie in $\mathcal{D}_{\widetilde{b}}$. This can be checked by testing whether $\widetilde{b}$ and $\widetilde{c}$ lie on the same side of the supporting line of each side of $\mathcal{D}_{\widetilde{b}}$.

To actually insert c into $DT(P_{i-1})$, we need to find the lift $\widetilde{c_b}$ of c that lies in $\mathcal{D}_{\widetilde{b}}$. If $\widetilde{c}$ lies in $\mathcal{D}_{\widetilde{b}}$, then $\widetilde{c_b} = \widetilde{c}$. Otherwise, the algorithm walks in the tiling $\{\gamma\mathcal{D}_{\widetilde{b}}\}_{\gamma\in\Gamma}$ of $\mathbb{H}^2$ along the geodesic segment $\widetilde{p_0^{\Delta^\varepsilon}}\widetilde{c}$. The first copy of $\mathcal{D}_{\widetilde{b}}$ traversed by $\widetilde{p_0^{\Delta^\varepsilon}}\widetilde{c}$ is found by looking for the side $s_{j_1}, j_1 \in \{0, \dots, k-1\}$ of $\mathcal{D}_{\widetilde{b}}$ intersecting it.[3] The walk along $\widetilde{p_0^{\Delta^\varepsilon}}\widetilde{c}$ continues in $\gamma_{j_1}\mathcal{D}_{\widetilde{b}}$, and so on, until the copy $\gamma_{j_n}\dots\gamma_{j_1}\mathcal{D}_{\widetilde{b}}$ containing $\widetilde{c}$ is found. Then $\widetilde{c_b} = \gamma_{j_1}^{-1}\dots\gamma_{j_n}^{-1}\widetilde{c}$. Note that the walk still works when $\widetilde{p_0^{\Delta^\varepsilon}}\widetilde{c}$ goes through a vertex of a copy of $\mathcal{D}_{\widetilde{b}}$.

The Delaunay triangulation $DT(P_i)$ of $P_i = P_{i-1} \cup \{c\}$ can then be computed. First, the triangle Δ_c of $DT(P_{i-1})$ containing c is found by naively checking if $\widetilde{c_b}$ lies in one of the (at most three) lifts of each triangle Δ in $DT(P_{i-1})$ having a vertex in $\mathcal{D}_{\widetilde{b}}$. This can be done by testing, for each edge, whether $\widetilde{c_b}$ and the third vertex of the triangle lie on the same side of its supporting line. Then Δ_c is split into three by creating an edge between c and its three vertices. In the data structure, the three isometries stored in each new triangle are $\mathbb{1}_\Gamma$ for c, and the corresponding isometries in Δ_c for the other two vertices. Then $DT(P_i)$ is computed with a sequence of flips and the data structure is updated [12].

The termination of the algorithm is quite obvious. At step $i = 1$, the ε-packing P_1 consists of one point. At each step $i \geqslant 2$, the point added to P_i is the circumcenter of a Delaunay triangle whose radius is at least ε. Because no vertex lies in the interior of a Delaunay disk, the center added is at distance at least ε from any point of P_i. By induction, P_i is an ε-packing containing i points. By Proposition 1, the algorithm must terminate after a finite number $N-1$ of insertions. It returns an ε-packing P_N of cardinality N.

It remains to show that P_N is an ε-covering of S. Let x be a point on S. It lies in a triangle Δ of $DT(P_N)$. Let $\widetilde{\Delta}$ be a lift of Δ and $\widetilde{x}$ the lift of x lying in $\widetilde{\Delta}$. The circumdisk of $\widetilde{\Delta}$ has a radius $r \leqslant \varepsilon$. There is a vertex of $\widetilde{\Delta}$ whose distance to $\widetilde{x}$ is at most r (see [10, Lemma 2]). That vertex is a lift of a point of P_N by definition of Δ. It follows that $\delta_S(x, P_N) \leqslant \varepsilon$, therefore P_N is an ε-net. This establishes the first claim of Proposition 2.

5 Algorithm analysis

This section is devoted to proving the complexity announced in Proposition 2.

The following operations take $O(1)$ time in the real RAM model and we consider them as elementary operations:

- Computing $\widetilde{\Delta_0}$ from a triangle Δ of the data structure (see Section 4 for notation);
- Computing the radius or the center of the circumcircle of a triangle in $\mathbb{H}^2$;
- Deciding if a point lies on the right or the left side of an oriented geodesic segment in $\mathbb{H}^2$;
- Flipping an edge of a triangulation [12, Section 4.1].

At the beginning of a step $i \geqslant 2$, P_{i-1} contains $i-1$ points, the Euler characteristic shows that $DT(P_{i-1})$ has $2i + 4g - 2$ triangles, which gives the cost of finding Δ^ε.

---

[2] Of course a priority queue could be used to improve the complexity of this search. We accept a linear complexity for simplicity, as this is not the dominant operation in the algorithm.

[3] To check whether two geodesic segments $\widetilde{x}_1\widetilde{x}_2$ and $\widetilde{y}_1\widetilde{y}_2$ intersect, we check whether $\widetilde{x}_1$ and $\widetilde{x}_2$ lie on opposite sides of the supporting line of $\widetilde{y}_1\widetilde{y}_2$, and we run the same test, swapping the roles of x and y.

Recall that the number k of sides of $\mathcal{D}_{\widetilde{b}}$ is at most $12g-6$ (see Section 2). Determining whether $\widetilde{c}$ lies in (a given copy of) $\mathcal{D}_{\widetilde{b}}$ thus requires at most $12g-6$ elementary operations. The algorithm tests the copies of $\mathcal{D}_{\widetilde{b}}$ that intersect the geodesic segment $\widetilde{p_0^{\Delta^\varepsilon}}\widetilde{c}$. Since $\widetilde{\Delta_0^\varepsilon}$ is a triangle of $DT\left(\widetilde{P_{i-1}}\right)$, its circumcircle does not contain any other lift of $p_0^{\Delta^\varepsilon}$, so $\widetilde{p_0^{\Delta^\varepsilon}}$ is the closest lift of $p_0^{\Delta^\varepsilon}$ to $\widetilde{c}$. The geodesic segment $\widetilde{p_0^{\Delta^\varepsilon}}\widetilde{c}$ is thus a lift of a distance path[4] on S, what is called a *distance path* in $\mathbb{H}^2$. By [9, Proposition 14], every side of $\mathcal{D}_{\widetilde{b}}$ is either a distance path, or the concatenation of two distance paths. As two distance paths that do not have a subarc in common, which is the case here, can intersect at most once [9, Lemma 8], $\widetilde{p_0^{\Delta^\varepsilon}}\widetilde{c}$ traverses at most $2k$ sides of copies of $\mathcal{D}_{\widetilde{b}}$. If an intersection occurs at a vertex of degree d of a copy of $\mathcal{D}_{\widetilde{b}}$, then this counts for d intersections. Searching the copy of $\mathcal{D}_{\widetilde{b}}$ containing $\widetilde{c}$ hence requires $k^2 \leqslant (12g-6)^2$ elementary operations. Computing $\widetilde{c_b}$ costs 1 operation.

Finding Δ_c in $DT(P_{i-1})$ when $\widetilde{c_b}$ is known requires at most $9(2i+4g-2)$ elementary operations since it amounts to checking the three edges of at most three lifts of each triangle. The update of the data structure when splitting the triangle containing c into three is done in 8 elementary operations (deleting the triangle that contains c, adding c to the list of vertices, creating 3 triangles and 3 isometries).

Adding the above costs for step i, locating c in $DT(P_{i-1})$ and splitting the triangle containing it costs at most $10(2i+4g-2)+(12g-6)^2+9$ elementary operations.

The flips are counted globally for all steps, which concludes the proof of Proposition 2.

▶ **Lemma 5.1.** *The total number of flipped edges during the execution of the algorithm is at most* $C'_h \operatorname{Diam}(S)^{6g-4}N^2$*, where* C'_h *is a constant depending on the metric* h *of* S*, and* $\operatorname{Diam}(S)$ *is the diameter of* S*.*

The proof of this lemma mimicks the proofs in [12]. The situation is quite different here, as the points are inserted incrementally and the flips are done at each insertion, whereas all points are know in advance in [12], which requires to rewrite a complete proof. Due to lack of space, we refer the reader to [10, Lemma 1].

Note that the bound comes from the best upper bound $O(\operatorname{Diam}(S)^{6g-4})$ known so far for the flip algorithm [12]. The actual complexity of the flip algorithm may be much better [7].

Acknowledgements. The authors wish to thank Hugo Parlier for interesting discussions.

[4] A distance path on S is a shortest path between two points. It is necessarily a geodesic segment, but not all geodesic segments are distance paths since they only locally minimize distances.

References

1. Alan F. Beardon. *The Geometry of Discrete Groups*. Graduate Texts in Mathematics. Springer New York, 1st edition, 1983. doi:10.1007/978-1-4612-1146-4.
2. Mikhail Bogdanov, Olivier Devillers, and Monique Teillaud. Hyperbolic Delaunay complexes and Voronoi diagrams made practical. *Journal of Computational Geometry*, 5(1):56–85, March 2014. doi:10.20382/jocg.v5i1a4.
3. Mikhail Bogdanov, Monique Teillaud, and Gert Vegter. Delaunay triangulations on orientable surfaces of low genus. In *32nd International Symposium on Computational Geometry (SoCG)*, 2016. doi:10.4230/LIPIcs.SoCG.2016.20.
4. Peter Buser. *Geometry and Spectra of Compact Riemann Surfaces*. Modern Birkhäuser Classics. Birkhäuser Boston, 1st edition, 2010. doi:10.1007/978-0-8176-4992-0.
5. Kenneth L. Clarkson. Building triangulations using ϵ-nets. In *38th annual ACM Symposium on Theory of Computing (STOC)*, pages 326–335. Association for Computing Machinery, May 2006. Extended abstract (long paper available at https://kenclarkson.org/enet_tris/p.pdf). doi:10.1145/1132516.1132564.
6. Éric Colin de Verdière, Vincent Despré, and Loïc Dubois. Untangling graphs on surfaces. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4909–4941, January 2024. doi:10.1137/1.9781611977912.175.
7. Vincent Despré, Loïc Dubois, Benedikt Kolbe, and Monique Teillaud. Experimental analysis of Delaunay flip algorithms on genus two hyperbolic surfaces, December 2021. URL: https://hal.inria.fr/hal-03462834.
8. Vincent Despré, Benedikt Kolbe, Hugo Parlier, and Monique Teillaud. Computing a Dirichlet domain for a hyperbolic surface. In *39th International Symposium on Computational Geometry (SoCG)*, volume 258, pages 27:1–27:15, 2023. doi:10.4230/LIPIcs.SoCG.2023.27.
9. Vincent Despré, Benedikt Kolbe, and Monique Teillaud. Representing infinite hyperbolic periodic Delaunay triangulations using finitely many Dirichlet domains, July 2021. Preprint. URL: https://hal.science/hal-03045921.
10. Vincent Despré, Camille Lanuel, and Monique Teillaud. Computing an ε-net of a closed hyperbolic surface. Preprint, 2024. URL: https://hal.science/hal-04466350.
11. Vincent Despré and Francis Lazarus. Computing the geometric intersection number of curves. *Journal of the ACM*, 66(6):45:1–45:49, November 2019. doi:10.1145/3363367.
12. Vincent Despré, Jean-Marc Schlenker, and Monique Teillaud. Flipping geometric triangulations on hyperbolic surfaces. In *36th International Symposium on Computational Geometry (SoCG)*, volume 164, pages 35:1–35:16, June 2020. doi:10.4230/LIPIcs.SoCG.2020.35.
13. Matthijs Ebbens, Hugo Parlier, and Gert Vegter. Minimal Delaunay triangulations of hyperbolic surfaces. *Discrete & Computational Geometry*, 69(2):568–592, February 2022. doi:10.1007/s00454-022-00373-0.
14. David Eppstein. Squarepants in a tree: Sum of subtree clustering and hyperbolic pants decomposition. *ACM Transactions on Algorithms*, 5(3):29:1–29:24, July 2009. doi:10.1145/1541885.1541890.
15. David Eppstein. Limitations on realistic hyperbolic graph drawing. In *International Symposium on Graph Drawing and Network Visualization (GD)*, Lecture Notes in Computer Science, pages 343–357. Springer International Publishing, 2021. URL: https://arxiv.org/abs/2108.07441, doi:10.1007/978-3-030-92931-2_25.
16. Hershel M. Farkas and Irwin Kra. *Riemann Surfaces*, volume 71 of *Graduate Texts in Mathematics*. Springer, New York, NY, 1992. doi:10.1007/978-1-4612-2034-3.
17. Iordan Iordanov and Monique Teillaud. Implementing Delaunay triangulations of the Bolza surface. In *33rd International Symposium on Computational Geometry (SoCG)*, pages 44:1–44:15, July 2017. doi:10.4230/LIPIcs.SoCG.2017.44.

18 Jonathan Richard Shewchuk. Delaunay refinement algorithms for triangular mesh generation. *Computational Geometry: Theory and Appplications*, 22(1):21–74, May 2002. doi:10.1016/S0925-7721(01)00047-5.

19 Daniel D. Sleator, Robert E. Tarjan, and William P. Thurston. Rotation distance, triangulations, and hyperbolic geometry. In *18th annual ACM symposium on Theory of computing (STOC 1986)*, pages 122–135. Association for Computing Machinery, November 1986. doi:10.1145/12130.12143.