Experimental analysis of Delaunay flip algorithms on genus two hyperbolic surfaces

Vincent Despré, Loïc Dubois, Benedikt Kolbe, Monique Teillaud

Experimental analysis of Delaunay flip algorithms on genus two hyperbolic surfaces

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

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

Benedikt Kolbe
Hausdorff Center for Mathematics, University of Bonn, Germany

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

Abstract

Guided by insights on the mapping class group of a surface, we give experimental evidence that the upper bound recently proven on the diameter of the flip graph of a surface by Despré, Schlenker, and Teillaud (SoCG'20) is largely overestimated.

To obtain this result, we propose a set of techniques allowing us to actually perform experiments. We solve arithmetic issues by proving a density result on rationally described genus two hyperbolic surfaces, and we rely on a description of surfaces allowing us to propose a data structure on which flips can be efficiently implemented.

2012 ACM Subject Classification Mathematics of computing → Geometric topology; Theory of computation → Computational geometry

Keywords and phrases Topology, Mapping class group, Dehn twist, Combinatorial map, Hyperbolic surface, Delaunay Triangulation, Flip graph, Implementation

Funding This work was partially supported by grant ANR-17-CE40-0033 of the French National Research Agency ANR (project SoS) `https://SoS.loria.fr/`.
Benedikt Kolbe: This work was done while this author was working at Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France

Acknowledgements The authors want to thank Vincent Delecroix, Matthijs Ebbens, Hugo Parlier, Jean-Marc Schlenker, and Gert Vegter for helpful discussions over many years.

The source code for the experiments can be downloaded at

`https://members.loria.fr/Monique.Teillaud/Exp-hyperb-flips/`

1 Introduction

It was recently proven that the geometric flip graph of a closed connected hyperbolic surface without boundary is connected [7]. A Delaunay flip algorithm can thus transform any input geometric triangulation T, i.e., a triangulation whose edges are embedded as geodesic segments only intersecting at common endpoints, into a Delaunay triangulation. This is particularly useful in practice as a crucial preprocessing step to computing Delaunay triangulations on a surface: it transforms a "bad" representation of a surface, e.g., by a very elongated fundamental domain, to a "nice" representation by a Delaunay triangulation with only one vertex. Inserting a lot of points would rather be done by Bowyer's incremental algorithm [12, 6], inspired from previous work in the flat case [16].

The authors prove an upper bound on the number of flips: $C_h \cdot \Delta(T)^{6g-4} \cdot n^2$, where C_h is a constant, $\Delta(T)$ is the diameter of T, g is the genus of the surface, and n is the number of vertices [7]. The diameter $\Delta(T)$ is the smallest diameter of a fundamental domain that is the union of lifts of the triangles of T in $\mathbb{H}$ (note that this is not the diameter of the surface, which is independent of the representation). Computing it algorithmically looks quite difficult, however for a triangulation with only one vertex (thus with $4g-2$ triangles) some bounds can easily be given: $L_T \leq \Delta(T) \leq \Delta(F) \leq L_T.(4g-2)$, where L_T denotes the maximal length of edges of T and $F \subset \mathbb{H}$ is any fundamental domain made of lifts of the triangles of T. From these bounds, it turns out that $\Delta(T)$ cannot be too far from the diameter of any such F: they only differ by a factor of at most six in the case of a genus two surface. In the experiments, we will thus use the domain that naturally appears.

In this paper, we experimentally study the dependence of the number of flips on $\Delta(T)$ (Section 7), for surfaces of genus two. We suspect that the factor $\Delta(T)^{6g-4}$ is largely overestimated. It comes from the number of paths of bounded length on a surface. Intuitively, for a length bounded by L, this number roughly amounts to the volume of the ball of diameter L, so, it is exponential in L; if only simple paths are considered, this number reduces to L^{6g-4} [7], but there is no reason why the flip algorithm would use all the simple paths shorter than L instead of going straight. More formally, our expectation on the dependence in $\Delta(T)$ relates to insights on the structure of the mapping class group (Section 2.3).

To perform experiments, we set up a framework consisting of various tools. In Section 3, we present a data structure for triangulations of surfaces, which is able to support flips; it relies on the representation of genus two surfaces by octagons in $\mathbb{H}$ (Section 3.1). Not surprisingly, arithmetic issues quickly arise, as algebraic numbers are involved in the description of the octagons. We overcome them by proving a density result on rationally described octagons (Section 4.2), which allows us to restrict to rational numbers in our experiments.

Generating input surfaces and triangulations is far from trivial; it is a non-negligible part of our work (Section 5). We manage to obtain surfaces with a large diameter by twisting the abovementioned octagons (Section 3.2).

In Section 6, we run experiments comparing strategies on the order of flips, and conclude that the naive strategy is close to being the best one in practice. We adhere to it for our main experiments that study the dependence of the number of flips on $\Delta(T)$.

The way we conduct these experiments in Section 7 is inspired by previous work by Mark Bell [2] who studied flips in a topological setting. We focus on triangulations having only one vertex, both because the dependence on the number of vertices is clear, and because we are

motivated by the abovementioned preprocessing aspect of the algorithm. Quite surprisingly, in practice, we observe behavior that is only expected asymptotically.

2 Background

2.1 Hyperbolic surfaces

Consider a closed hyperbolic surface $\mathcal{S}$ (i.e., a compact oriented surface without boundary) of genus $g \geq 2$ and the underlying topological surface S_g. Given a hyperbolic structure h on $\mathcal{S}$, associated to a metric of constant curvature -1, the surface $\mathcal{S} = (S_g, h)$ is isometric to the quotient $\mathbb{H}/G$, where $\mathbb{H}$ is the hyperbolic plane and G is a (non-Abelian) discrete subgroup of the isometry group of $\mathbb{H}$ isomorphic to the fundamental group $\pi_1(S_g)$.

The universal cover of $\mathcal{S}$ is isometric to $\mathbb{H}$ equipped with a projection $\rho : \mathbb{H} \to \mathcal{S}$ that is a local isometry. The group G acts on $\mathbb{H}$, so that for any $p \in \mathcal{S}$, $\rho^{-1}(p)$ is an orbit under the action of G. A lift $\widetilde{p}$ of a point $p \in \mathcal{S}$ is one of the elements of the orbit $\rho^{-1}(p)$.

We use the Poincaré disk model of $\mathbb{H}$, in which $\mathbb{H}$ is represented as the open unit disk $\mathbb{D}$ of the complex plane $\mathbb{C}$. Every orientation preserving isometry $f : \mathbb{D} \to \mathbb{D}$ can be represented by a matrix $\begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathbb{C}^{2\times 2}$ such that $f(z) = \dfrac{az+b}{cz+d}$ for any $z \in \mathbb{D}$. Remark that with this definition the matrix is not unique and some matrices do not represent an isometry. Given two orientation preserving isometries f and g respectively represented by matrices A and B, the product $A \cdot B$ represents $f \circ g$.

Given $z \in \mathbb{C}$, we denote by $\mathrm{Re}\,[z]$ and $\mathrm{Im}\,[z]$ its real and imaginary parts, respectively, by $\overline{z}$ its conjugate, and by $|z|$ its modulus; i denotes a root of -1. Given a compact subset $X \subset \mathbb{D}$, $\mathcal{A}(X)$ is the hyperbolic area of X.

2.2 Triangulations and flips on hyperbolic surfaces

A topological triangulation of a hyperbolic surface $\mathcal{S}$ is any embedding of an undirected graph with a finite number of vertices onto $\mathcal{S}$ such that each resulting face is homeomorphic to an open disk and is bounded by exactly three distinct edge-embeddings. Remark that this underlying graph may have loops or multiple edges, and recall that the terms *embedding* and *embedded* subsume that any two edges don't intersect except at common vertices. A *geometric triangulation* is a topological triangulation of $\mathcal{S}$ whose edges are embedded as geodesic segments [7]. All triangulations considered in this paper will be geometric, so, we will just use the term *triangulation*. For any triangulation T of $\mathcal{S}$, the lift $\widetilde{T}$ of T is the (infinite) triangulation of $\mathbb{H}$ whose vertices and edges are the lifts of the vertices and the edges of T. A *Delaunay triangulation* T of $\mathcal{S}$ is a triangulation whose lift $\widetilde{T}$ is a Delaunay triangulation in $\mathbb{H}$. In other words, for each face t of T and any of its lifts $\widetilde{t}$, the open disk in $\mathbb{H}$ circumscribing $\widetilde{t}$ is empty, i.e., it contains no vertex of $\widetilde{T}$. Recall that circles in the Poincaré disk model correspond to circles in the complex plane $\mathbb{C}$.

A *Delaunay flip* is defined in a natural way: Consider an edge e of a (geometric) triangulation T and one of its lifts $\widetilde{e}$, together with the two triangles incident to $\widetilde{e}$ in the lifted triangulation $\widetilde{T}$ in $\mathbb{H}$. The edge e is *Delaunay-flippable* if the open disks of these triangles contain the fourth vertex of the quadrilateral that they form. In this case, the geodesic segment $\widetilde{e}'$ that is the other diagonal of the quadrilateral is contained in it. The Delaunay flip consists in replacing $\widetilde{e}$ by $\widetilde{e}'$ and projecting the two new triangles to $\mathcal{S}$ by ρ.

A *Delaunay flip algorithm* takes as input a triangulation of $\mathcal{S}$ and flips Delaunay-flippable edges (in any order) until there is none left. Such an algorithm terminates and outputs a

Delaunay triangulation [7].

2.3 Mapping class group

We use the same notation as Maher [14] and refer to his paper for details.

The set $\mathrm{Mod}(S_g)$ of all homeomorphims (up to isotopy) of a topological surface S_g is called the mapping class group of S_g. Following Thurston's classification [10], $\mathrm{Mod}(S_g)$ contains three kinds of elements: the periodic homeomorphims, which are of finite order and are not useful for our purposes; the reducible ones, for which at least one curve on S_g is fixed; and the so-called *pseudo-Anosov* homeomorphims, also known as the *hyperbolic* elements of $\mathrm{Mod}(S_g)$.

Dehn twists (see Figure 1) are typical reducible elements, as they fix all the curves that do not intersect the one used for twisting. A Dehn twist by a curve c at most adds to the length of a curve a constant that depends on the number of times the curve intersects c. A pseudo-Anosov element at most multiplies the length of a curve by a constant factor.

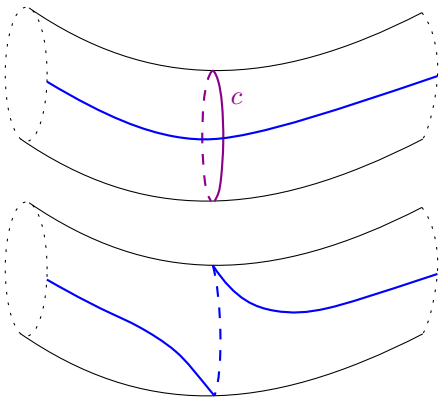

Figure 1 A Dehn twist along the curve c modifies the blue curve as shown.

$\mathrm{Mod}(S_g)$ can be generated by a finite set of Dehn twists [9]. Composing generators or their inverses in a random order gives a notion of random walk in $\mathrm{Mod}(S_g)$: such a walk reaches pseudo-Anosov elements with asymptotic probability 1 [14]. However, this asymptotic result does not provide a good idea of the local structure of $\mathrm{Mod}(S_g)$.

3 Representation of genus two surfaces and triangulations

The Teichmüller space $\mathcal{TM}_2$ of the topological surface S_2 is the set of all the hyperbolic structures (up to isotopy) that can be associated to S_2. It admits various parametrizations. The most commonly used is the set of Fenchel-Nielsen coordinates, but it is not well adapted to our needs. Here, we use a less usual set of parameters introduced by Aigon-Dupuy, Buser *et al.* [1], who proved that any such surface has a fundamental domain that is an octagon in $\mathbb{D}$. This versatile representation allows us to easily construct and manipulate surfaces in our experiments. In the following section, we recall and slightly extend the definitions of the original paper [1], following its notation. Then we show how we can twist octagons (Section 3.2). Finally, we describe our data structure for triangulations in Section 3.3 and we sketch in Section 3.4 how it is maintained through flips.

3.1 Admissible symmetric octagons

Given $j \geq 3$ and complex numbers $z_1, \ldots, z_j \in \mathbb{D}$ in geodesically convex position, $G[z_1, \ldots, z_j]$ denotes the hyperbolic polygon whose vertices are $z_1, \ldots, z_j$.

Let $\arg z \in [0, 2\pi[$ denote the argument of a point $z \neq 0_{\mathbb{C}}$. Given $z_0, z_1, z_2, z_3 \in \mathbb{D} \setminus \{0_{\mathbb{C}}\}$, the 4-tuple (z_0, z_1, z_2, z_3) is *valid* if $0 = \arg z_0 < \arg z_1 < \arg z_2 < \arg z_3 < \pi$; then the hyperbolic octagon $\mathcal{P}[z_0, z_1, z_2, z_3]$ is defined as $G[z_0, z_1, z_2, z_3, -z_0, -z_1, -z_2, -z_3]$. Such an hyperbolic octagon is a *symmetric octagon*. The interior angles of a symmetric octagon cannot be greater than π. If moreover $\mathcal{A}(\mathcal{P}[z_0, z_1, z_2, z_3]) = 4\pi$, then $\mathcal{P}[z_0, z_1, z_2, z_3]$ and the 4-tuple (z_0, z_1, z_2, z_3) are said to be *admissible*. Each closed hyperbolic surface of genus 2 can be obtained by identifying the opposite sides of an admissible symmetric octagon [1].

Remark that the eight vertices of the octagon correspond to the same point on the surface.

A valid 4-tuple (z_0, z_1, z_2, z_3) is admissible if and only if [1, Lemma 3.2]

$$\operatorname{Im}\left[\prod_{k=0}^{3}(1 - z_k\overline{z_{k+1}})\right] = 0. \tag{1}$$

The authors establish this condition after proving a preliminary result that we will reuse: for any two points $z, z' \in \mathbb{D} \setminus \{0_{\mathbb{C}}\}$ if $0 \leq \arg z \leq \arg z' \leq \pi$ then [1, Appendix (A7)]

$$2\arg(1 - z\overline{z'}) = \mathcal{A}(G[0_{\mathbb{C}}, z, z']). \tag{2}$$

An admissible 4-tuple can be constructed as follows [1, Section 3]. Start with $z_1, z_2, z_3 \in \mathbb{D}$ satisfying $0 < \arg(z_1) < \arg(z_2) < \arg(z_3) < \pi$. Abbreviate $u = (1 - z_1\overline{z_2})(1 - z_2\overline{z_3})$, $a = \operatorname{Im}[-u\overline{z_1}z_3]$, $b = \operatorname{Im}[u(z_3 - \overline{z_1})]$, $c = \operatorname{Im}[u]$. If

$$a + b + c < 0 \tag{3}$$

then (z_0, z_1, z_2, z_3) is an admissible 4-tuple if we set

$$z_0 = \frac{2c}{-b + \sqrt{b^2 - 4ac}}. \tag{4}$$

From now on indices are modulo 8. Let us consider an admissible 4-tuple (z_0, z_1, z_2, z_3) and define $z_{l+4} = -z_l$ for every $l \in \{0, 1, 2, 3\}$. For $k \in \{0, \ldots, 7\}$, there is a unique orientation preserving isometry $\tau_k : \mathbb{D} \to \mathbb{D}$ satisfying $\tau_k(z_{k+5}) = z_k$ and $\tau_k(z_{k+4}) = z_{k+1}$: the isometry τ_k maps a side of $\mathcal{P}[z_0, z_1, z_2, z_3]$ to the opposite side of $\mathcal{P}[z_0, z_1, z_2, z_3]$. Remark that $\tau_{k+4} = \tau_k^{-1}$. Define $\omega_k = \dfrac{z_k(1 - |z_{k+1}|^2) + z_{k+1}(1 - |z_k|^2)}{1 - |z_k z_{k+1}|^2}$ and note that $|\omega_k| < 1$; the isometry τ_k is given by $\tau_k(z) = (z + \omega_k)/(\overline{\omega_k} + 1)$ for every $z \in \mathbb{D}$ [1, Lemma 4.1].

3.2 Twisting admissible loosely-symmetric octagons

We say that a hyperbolic octagon P is *loosely-symmetric* if the opposite sides of P are isometric and the opposite interior angles of P are equal. If moreover $\mathcal{A}(P) = 4\pi$ then P is *admissible*. Clearly, symmetric octagons are loosely-symmetric octagons. Identifying the opposite sides of an admissible loosely-symmetric octagon gives a closed hyperbolic surface of genus 2 [4, Theorem 1.3.5].

Let $G[z_0, \ldots, z_7]$ be an admissible lossely-symmetric octagon. We will consider the Dehn twists along the axes of its side-pairings. These twists generate a subgroup of $\mathrm{Mod}(S_2)$ (see Section 2.3), which clearly contains non-reducible elements of $\mathrm{Mod}(S_2)$ as the generators do not all fix a common curve. Thus, this subgroup contains pseudo-Anosov elements [13].

For every $k \in \{0, \dots, 7\}$ we denote by τ_k the orientation preserving isometry of $\mathbb{D}$ satisfying $\tau_k(z_{k+5}) = z_k$ and $\tau_k(z_{k+4}) = z_{k+1}$. For $t, k \in \{0, \dots, 7\}$ we set

$$z'_k = \begin{cases} \tau_t(z_k) & \text{if } k - t \in \{1, 2, 3, 4\} \mod 8, \\ z_k & \text{otherwise.} \end{cases}$$

By the Gauss-Bonnet formula, the interior angles of $G[z_0, \dots, z_7]$ sum up to 2π; since opposite interior angles are equal, each interior angle is at most π. So, the geodesic segment between z_{t+1} and z_{t+5} is contained in $G[z_0, \dots, z_7]$ and cuts the polygon into the two interior disjoint pentagons $P_1 = G[z_{t+1}, z_{t+2}, z_{t+3}, z_{t+4}, z_{t+5}]$ and $P_2 = G[z_{t+5}, z_{t+6}, z_{t+7}, z_t, z_{t+1}]$; the intersection of P_1 and P_2 is the segment between z_{t+5} and z_{t+1}. One easily proves that the interior angles of P_1 are the interior angles of P_2. Similarly, $\tau_t(P_1)$ and P_2 are interior disjoint, they intersect on the segment between $z'_t = z_t$ and $z'_{t+4} = z_{t+1}$ and their union is $G[z'_0, \dots, z'_7]$. It follows that $G[z'_0, \dots, z'_7]$ is an admissible loosely-symmetric octagon; the surface that it defines is isometric to the surface defined by $G[z_0, \dots, z_7]$ as both surfaces can be obtained by the same identification of the sides of P_1 and P_2 (P_1 and $\tau_t(P_1)$ being isometric). We say that $(z'_0, \dots, z'_7)$ is obtained by *t-twisting* $(z_0, \dots, z_7)$. For every point z in the closure of $G[z_0, \dots, z_7]$ at least z or $\tau_t(z)$ lies in the closure of $G[z'_0, \dots, z'_7]$.

Let us denote by $(\tau'_k)_{0 \le k \le 7}$ the isometries defined for $z'_0, \dots, z'_7$, in the same way as $(\tau_k)_{0 \le k \le 7}$ above. By definition of the t-twist, the following holds for every $k \in \{0, \dots, 7\}$

$$\tau'_k = \begin{cases} \tau_t \circ \tau_k & \text{if } k - t \in \{1, 2, 3\} \mod 8, \\ \tau_k \circ \tau_t^{-1} & \text{if } t - k \in \{1, 2, 3\} \mod 8, \\ \tau_k & \text{if } k = t \mod 4. \end{cases}$$

More generally, for a word $t = t_1 \dots t_m$, we define the t-twist as the composition of the t_k-twists, $k = 1, \dots, m$, in this order. We choose to pick $t_1, \dots, t_m$ in $\{0, \dots, 3\}^m$ instead of $\{0, \dots, 7\}^m$ to only consider the generators without their inverses, so as to obtain large diameters as quickly as possible.

3.3 Data structure

Though an *ad hoc* data structure was previously proposed for flipping triangulations [7], we choose to use *combinatorial maps*, which are commonly used to represent graphs embedded on a surface. We refer the reader to the literature for a formal definition [15, Section 3.3]. The data structure in this paper offers a representation of the triangulation that intrinsically lies on the surface, while the earlier data structure stuck to specific representatives of all vertices and faces of the lifted triangulation in the universal cover.

For our experiments, we use the flexible implementation of combinatorial maps that is publicly available in CGAL [5]. The *dart*, also known as *flag*, is the central object in a combinatorial map: it gives access to all incidence relations of an edge of the graph (Figure 2).

The geometric information for the triangulation is stored by adding a cross-ratio on each edge. Recall that the cross-ratio of four pairwise-distinct points in $\mathbb{H}$ represented by $z_1, z_2, z_3, z_4 \in \mathbb{D}$ is the complex number $[z_1, z_2, z_3, z_4] = \dfrac{(z_4 - z_2)(z_3 - z_1)}{(z_4 - z_1)(z_3 - z_2)}$ [3]. Using cross-ratios is perfectly suitable for a flip algorithm, due to their well-known property: assuming that the four points are oriented counterclockwise, $\operatorname{Im}[z_1, z_2, z_3, z_4] > 0$ if and only if z_4 lies in the open disk circumscribing the triangle (z_1, z_2, z_3).

Given a triangulation T of $\mathcal{S}$, for each edge $e = (u_1, u_3)$ we consider a lift $\widetilde{e}$ of e in $\mathbb{D}$ and the other vertices $\widetilde{u_2}$ and $\widetilde{u_4}$ of the two faces incident to $\widetilde{e}$ in $\widetilde{T}$, numbering vertices counterclockwise. The cross-ratio $\mathcal{R}_T(e)$ is defined as $[\widetilde{u_1}, \widetilde{u_2}, \widetilde{u_3}, \widetilde{u_4}]$; it is independent of the

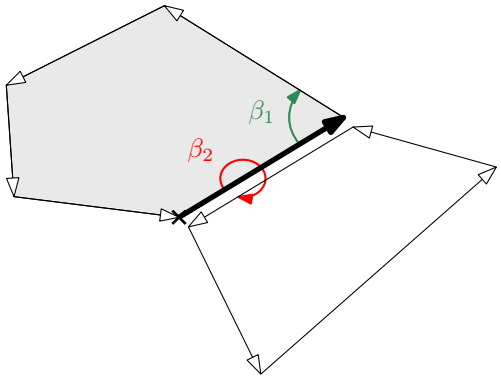

Figure 2 A dart in a combinatorial map (bold)

choice of the lift of e, as the cross-ratio is invariant under orientation preserving isometries of $\mathbb{D}$. An edge of e of T is Delaunay-flippable if and only if $\mathrm{Im}\,[\mathcal{R}_T(e)] > 0$.

Note that in our experiments, the lifts in $\mathbb{D}$ will only be used to initialize the cross-ratios of a given input triangulation T; they will be ignored during the flips, thus preserving the property that the data structure only considers the embedding of the triangulation on the surface. However, in order to be able to recover a lift in $\mathbb{D}$ in the end, e.g., for drawing a representation in $\mathbb{D}$ of the final Delaunay triangulation, we need to maintain an *anchor* during flips. The anchor $A = (\delta, a_1, a_2, a_3)$ consists in some dart δ, chosen arbitrarily, together with a triple (a_1, a_2, a_3) of points in $\mathbb{D}$ that are the vertices of a lift of the face containing δ.

A triangulation T is thus represented by (M, F, A), where M is the combinatorial map, F the map that associates a cross-ratio to each edge of F, and $A = (\delta, a_1, a_2, a_3)$ is the anchor.

3.4 Flipping an edge

In this section, we quickly sketch how the data structure is maintained through an edge flip. First we modify the combinatorial map, then we update the anchor, and we finally update the cross-ratios. Details and pseudo-code are given in Appendix B.

Performing a flip in the combinatorial map is a straightforward use of the functionalities given by the CGAL package [5]. The triangulation obtained from T after flipping edge e is denoted as $T^\star$. By definition, the dart δ of the anchor A belongs to the face t of T represented by a lift $\widetilde{t} = (a_1, a_2, a_3)$ in $\mathbb{D}$. If t is still a face of $T^\star$, then A is not modified by the flip. However, if e is an edge of t then t will not belong to $T^\star$ and we must update A. A lift $\widetilde{e}$ of e incident to $\widetilde{t}$ is replaced by $\widetilde{e^\star}$ when e is flipped. The anchor is updated so that it represents one of the two faces incident to $\widetilde{e^\star}$ in $T^\star$.

Finally, the cross-ratios must be retrieved. Only the cross-ratios of the at most 5 edges of the two triangles forming the quadrilateral whose diagonal is to be flipped must be updated. Their values after the flip are expressed in terms of their values before the flip (see Lemma 5 in Appendix A).

4 Solving arithmetic issues

The construction recalled in Section 3.1 shows that the real and imaginary parts of the complex numbers involved when defining surfaces are in general algebraic numbers. Efficiency issues when computing with algebraic numbers have been known for decades. More recently, they appeared when constructing Delaunay triangulations of hyperbolic surfaces [12, 8],

showing that the hope to get effective software was restricted to very simple cases. In Section 4.1 we describe a simple experiment on the Bolza surface illustrating that these arithmetic issues are actually prohibitive in practice for the Delaunay flip algorithm.

Then we show in Section 4.2 that any closed hyperbolic surface of genus 2 can be approximated by a surface represented by rational numbers. It is straightforward to check that the computations made during a Delaunay flip algorithm only use the four basic operations $+, -, \cdot, /$ (see Section 3.4), so, if the input surface is represented by rational numbers, then all numbers arising throughout the algorithm stay rational. This will allow us to run extensive experiments.

4.1 Issues when using algebraic numbers

Let $c_k = \frac{\exp\left(i\pi\frac{2k-1}{8}\right)}{\sqrt{2}}, k \in \{0,\dots,7\}$ be the vertices of a regular hyperbolic octagon in $\mathbb{D}$; identifying the opposite sides of this octagon gives a closed hyperbolic surface of genus 2 known as the Bolza surface. Consider the triangulation T_0 of the octagon shown in Figure 3. Identifying in T_0 the edges corresponding to opposite sides of the octagon yields a triangulation T of the Bolza surface. Let $e_0,\dots,e_4$ be the edges of T corresponding to the edges $e'_0,\dots,e'_4$ of T_0. The algebraic numbers $\cos\left(\frac{\pi}{8}\right) = \frac{\sqrt{2+\sqrt{2}}}{2}$ and $\sin\left(\frac{\pi}{8}\right) = \frac{\sqrt{2-\sqrt{2}}}{2}$ naturally

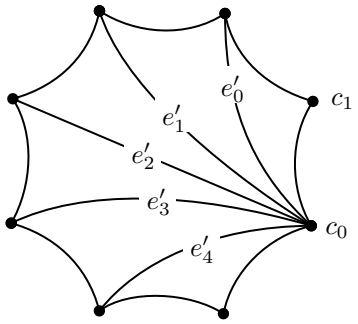

Figure 3 The triangulation T_0 and the edges $e'_0,\dots,e'_4$

appear when computing the cross-ratios $\mathcal{R}_T(e_l) = [c_0, c_{l+1}, c_{l+2}, c_{l+3}]$, $l \in \{0,\dots,4\}$.

As the points $c_k, k = 0,\dots,7$ are concyclic, the situation is degenerate and $e_0,\dots,e_4$ can be flipped though they are not strictly speaking Delaunay-flippable. The experiment consists in computing the cross-ratios that would appear during the flips of $e_0,\dots,e_4$ in this order (see Appendix C.1 for details). We used the CGAL wrapper `CORE::Expr` [11] for the algebraic numbers provided by the CORE library [17]. It took minutes to finish on an Intel Core i5-8250u cpu (1.6Ghz, 8 cores) and 16Gb of ram. Such a running time severely restricts the possibility to run heavy experiments with a Delaunay flip algorithm.

4.2 Density of the rationally described surfaces

For any $z \in \mathbb{D}$ and any $\varepsilon > 0$, $B(z,\varepsilon)$ denotes the open ball $\{z' \in \mathbb{D} \,:\, d(z,z') < \varepsilon\}$ where $d(\cdot,\cdot)$ is the hyperbolic distance in $\mathbb{D}$.

▶ **Definition 1.** *We say that a 4-tuple* (z_0,z_1,z_2,z_3) *is* rational *if* $z_k \in \mathbb{Q}+i\mathbb{Q}$ *for every* $k \in \{0,1,2,3\}$. *A* rationally described surface *is a surface obtained from a rational admissible 4-tuple* $(\mathbf{z_0},\mathbf{z_1},\mathbf{z_2},\mathbf{z_3})$ *by identifying the opposite sides of* $\mathcal{P}[\mathbf{z_0},\mathbf{z_1},\mathbf{z_2},\mathbf{z_3}]$.

▶ **Theorem 2.** *Let (z_0, z_1, z_2, z_3) be an admissible 4-tuple and $\varepsilon > 0$ be a real. There exists a rational admissible 4-tuple $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$ such that $\forall k \in \{0,1,2,3\}, \mathbf{z_k} \in B(z_k, \varepsilon)$.*

Proof. For two reals a and b, define $]a,b[= \{z \in \mathbb{C} \ : \ a < \mathrm{Re}\,[z] < b \text{ and } \mathrm{Im}\,[z] = 0\}$. We first choose for every $k \in \{0,1,2,3\}$ a point $\mathbf{z_k} \in B(z_k, \varepsilon) \cap (\mathbb{Q} + i\mathbb{Q})$, with the additional requirement that $\mathbf{z_0} \in]0,1[$, but without trying to satisfy the area condition $\mathcal{A}(G[-\mathbf{z_0}, \mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}]) = 2\pi$. Consider Figure 4. If ε is small enough, then $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$ is valid. We will now show that if each $\mathbf{z_k}$ is "close enough" to z_k for every k, then we can replace $\mathbf{z_3}$ by some U in $B(z_3, \varepsilon) \cap (\mathbb{Q} + i\mathbb{Q})$ so that the area condition is satisfied. More details on the construction can be found in Appendix C.2.

To do so we first define an isometry $\mathbf{f} : \mathbb{D} \to \mathbb{D}$ in the Poincaré disk: $\mathbf{f}(z) = \dfrac{z + \mathbf{z_0}}{\mathbf{z_0} z + 1}$. Remark that $\mathbf{f}(-\mathbf{z_0}) = 0_{\mathbb{C}}$. Since $\mathbf{f}$ and $\mathbf{f}^{-1}$ both map $\mathbb{D} \cap (\mathbb{Q} + i\mathbb{Q})$ to some subset of $\mathbb{D} \cap (\mathbb{Q} + i\mathbb{Q})$ we are reduced to replacing $\mathbf{f}(\mathbf{z_3})$ by some other element V of $B(\mathbf{f}(z_3), \varepsilon) \cap (\mathbb{Q} + i\mathbb{Q})$ satisfying $\mathcal{A}(G[0_{\mathbb{C}}, \mathbf{f}(\mathbf{z_0}), \mathbf{f}(\mathbf{z_1}), \mathbf{f}(\mathbf{z_2}), V]) = 2\pi$. Indeed by setting $U = \mathbf{f}^{-1}(V)$ we have $U \in B(z_3, \varepsilon) \cap (\mathbb{Q} + i\mathbb{Q})$ and $\mathcal{A}(G[-\mathbf{z_0}, \mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, U]) = \mathcal{A}(G[0_{\mathbb{C}}, \mathbf{f}(\mathbf{z_0}), \mathbf{f}(\mathbf{z_1}), \mathbf{f}(\mathbf{z_2}), V]) = 2\pi$.

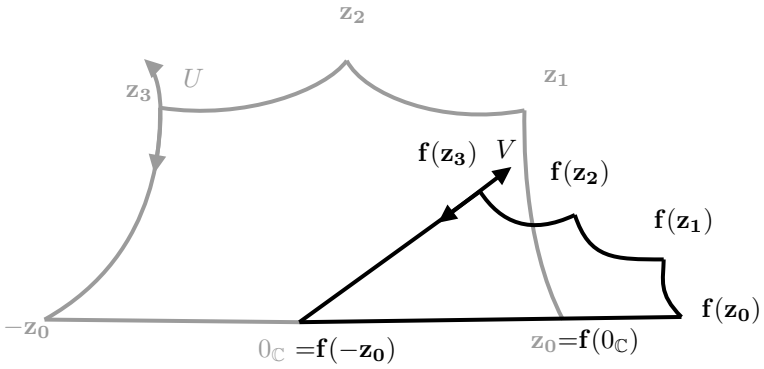

Figure 4 Illustration of the proof of Proposition 2

To find such a point V, we define a polynomial $P \in \mathbb{Q}[X]$ by setting

$$P(X) = \mathrm{Im}\left[(1 - \mathbf{f}(\mathbf{z_0})\overline{\mathbf{f}(\mathbf{z_1})})(1 - \mathbf{f}(\mathbf{z_1})\overline{\mathbf{f}(\mathbf{z_2})})(1 - X\mathbf{f}(\mathbf{z_2})\overline{\mathbf{f}(\mathbf{z_3})})\right].$$

Remark that the degree of P is at most 1 thus $P(X) = (P(1) - P(0))X + P(0)$. Let us first show that if we choose $\mathbf{z_k}$ close to z_k for every $k \in \{0,1,2,3\}$, $P(1)$ will be close to 0 and $P(0)$ close to $\kappa > 0$. Remark that $0 = \arg \mathbf{f}(\mathbf{z_0}) < \arg \mathbf{f}(\mathbf{z_1}) < \arg \mathbf{f}(\mathbf{z_2}) < \arg \mathbf{f}(\mathbf{z_3}) < \pi$. So we can apply Equality (2) and obtain

$$\begin{aligned}
&\arg\left[\left(1 - \mathbf{f}(\mathbf{z_0})\overline{\mathbf{f}(\mathbf{z_1})}\right)\left(1 - \mathbf{f}(\mathbf{z_1})\overline{\mathbf{f}(\mathbf{z_2})}\right)\left(1 - \mathbf{f}(\mathbf{z_2})\overline{\mathbf{f}(\mathbf{z_3})}\right)\right] \\
&= \arg\left(1 - \mathbf{f}(\mathbf{z_0})\overline{\mathbf{f}(\mathbf{z_1})}\right) + \arg\left(1 - \mathbf{f}(\mathbf{z_1})\overline{\mathbf{f}(\mathbf{z_2})}\right) + \arg\left(1 - \mathbf{f}(\mathbf{z_2})\overline{\mathbf{f}(\mathbf{z_3})}\right) \\
&= \frac{1}{2}\left[\mathcal{A}\left(G[0_{\mathbb{C}}, \mathbf{f}(\mathbf{z_0}), \mathbf{f}(\mathbf{z_1})]\right) + \mathcal{A}\left(G[0_{\mathbb{C}}, \mathbf{f}(\mathbf{z_1}), \mathbf{f}(\mathbf{z_2})]\right) + \mathcal{A}\left(G[0_{\mathbb{C}}, \mathbf{f}(\mathbf{z_2}), \mathbf{f}(\mathbf{z_3})]\right)\right] \\
&= \frac{1}{2}\mathcal{A}\left(G[0_{\mathbb{C}}, \mathbf{f}(\mathbf{z_0}), \mathbf{f}(\mathbf{z_1}), \mathbf{f}(\mathbf{z_2}), \mathbf{f}(\mathbf{z_3})]\right) = \frac{1}{2}\mathcal{A}\left(G[-\mathbf{z_0}, \mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}]\right).
\end{aligned}$$

By observing that every expression in between the equalities belongs to $[0, 2\pi[$ we see that they are indeed equalities and not only congruences modulo 2π. By choosing $\mathbf{z_k}$ close to z_k for every $k \in \{0,1,2,3\}$ we make the last expression approach $\frac{1}{2}\mathcal{A}(G[-z_0, z_0, z_1, z_2, z_3]) = \pi$,

which makes $P(1)$ tend to 0. Similarly, we obtain

$$\arg\left[\left(1-\mathbf{f}(\mathbf{z_0})\overline{\mathbf{f}(\mathbf{z_1})}\right)\left(1-\mathbf{f}(\mathbf{z_1})\overline{\mathbf{f}(\mathbf{z_2})}\right)\right] = \frac{1}{2}\mathcal{A}\left(G[-\mathbf{z_0},\mathbf{z_0},\mathbf{z_1},\mathbf{z_2}]\right).$$

By choosing $\mathbf{z_k}$ closer and closer to z_k, the last expression tends to $\frac{1}{2}\mathcal{A}(G[-z_0,z_0,z_1,z_2]) \not\equiv 0[\pi]$, so that $P(0)$ is close to some constant $\kappa > 0$, whence we can assume that $P(1) \neq P(0)$.

Now we can construct V. Set $\lambda = \frac{P(0)}{P(0)-P(1)}$ and $V = \lambda\mathbf{f}(\mathbf{z_3})$; we have both $V \in \mathbb{Q}+i\mathbb{Q}$ and $P(\lambda) = 0$. By making λ tend to 1 we can choose V as close as we want to $\mathbf{f}(\mathbf{z_3})$. Finally, remark that $P(\lambda) = 0$ implies $\mathcal{A}(G[0_{\mathbb{C}},\mathbf{f}(\mathbf{z_0}),\mathbf{f}(\mathbf{z_1}),\mathbf{f}(\mathbf{z_2}),V]) = 2\pi$ by Equality (2). ◀

▶ Remark 3. This theorem implies the density of the hyperbolic structures corresponding to rational admissible 4-tuple in $\mathcal{TM}_2$ with its canonical topology. However, a proof would go beyond the scope of this paper and would be quite technical.

5 Generation of input for the experiments

To conduct experiments, we need to generate a large number of triangulations having a large diameter. To this aim, we combine all the tools introduced so far: we start from admissible octagons and perform a large number of twists so as to increase the diameter; thanks to the results in Section 4, we can restrict to rational admissible octagons. In the end we triangulate the resulting octagons.

More specifically, input surfaces and triangulations are constructed following four steps.

[step 1] We construct an initial rational admissible 4-tuple $(\mathbf{z_0},\mathbf{z_1},\mathbf{z_2},\mathbf{z_3})$.

[step 2] We choose $n_p \geq 0$ and construct points $(\mathbf{p_1},\ldots,\mathbf{p_{n_p}}) \in (\mathbb{Q}+i\mathbb{Q})^n$ lying within the closure of $G[\mathbf{z_0},\mathbf{z_1},\mathbf{z_2},\mathbf{z_3}]$.

[step 3] We choose $m \geq 0$ and a sequence $(t_1,\ldots,t_m)$ of twists.

[step 4] Finally, from the 4-tuple $(\mathbf{z_0},\mathbf{z_1},\mathbf{z_2},\mathbf{z_3})$, the points $(\mathbf{p_1},\ldots,\mathbf{p_{n_p}})$, and the sequence $(t_1,\ldots,t_m)$, we construct a representation (M,F,A) of an input triangulation T. Together with the point corresponding to the vertices of the octagon, T has $n = n_p+1$ vertices.

Sections 6 and 7 will refer to these four steps. Step 1 was applied a thousand times to construct the 1,000 rational admissible 4-tuples $Q_1,\ldots,Q_{1,000}$; the experiments consider the first n_q 4-tuples. We also constructed in step 3 10,000 random sequences of twists noted $S_1,\ldots,S_{10,000}$, each of length 10, of which some of the experiments will use the first n_s sequences. The values of n_p, n_q, n_s will be specified in the description of each experiment.

Technicalities for steps 1, 2, and 4 are deferred to Appendix D. We only elaborate on step 3 here. Consider a sequence of m twists represented by the word $t = t_1 \ldots t_m$ (see Section 3.2). We will study two kinds of sequences in the experiments of Sections 6 and 7:

- A *power sequence* is represented by a word u^m for some $u \in \{0,\ldots,3\}$.
- In a *random sequence*, $t_1,\ldots,t_m$ are chosen uniformly and independently in $\{0,\ldots,3\}$.

It appears in practice that the length of a random sequence has a stronger impact on the computations than the length of a power sequence. When twisting, we update an 8-tuple $(\mathbf{z_0},\ldots,\mathbf{z_7}) \in (\mathbb{Q}+i\mathbb{Q})^8$ corresponding to an admissible loosely-symmetric octagon $G[\mathbf{z_0},\ldots,\mathbf{z_7}]$, together with the orientation preserving isometries identifying its opposite sides. Both the points and the isometries are represented by complex numbers: $8(m+1)$ complex numbers for the $8(m+1)$ points and $32(m+1)$ complex numbers for the $8(m+1)$ isometries. Each such complex number is represented by two rational numbers and each such rational number is represented by two integers: its numerator and its denominator. The

running time of a sequence of $m \geq 0$ twists depends on the sizes of these $160(m+1)$ integers; here the size of an integer is the number of digits of its decimal representation.

Twisting promptly gives rise to big numbers. As an example, take the rational admissible 4-tuple $\mathbf{z_0} = 10/11, \mathbf{z_1} = 1/2 + 1/2i, \mathbf{z_2} = -1/10 + 9/10i, \mathbf{z_3} = -3/5 + 3/5i$. Then twist $(\mathbf{z_0}, \ldots, \mathbf{z_3}, -\mathbf{z_0}, \ldots, -\mathbf{z_3})$ by m twists and get $(\mathbf{z}_{0,m}, \ldots, \mathbf{z}_{7,m}) \in (\mathbb{Q} + i\mathbb{Q})^8$ such that $G[\mathbf{z}_{0,m}, \ldots, \mathbf{z}_{7,m}]$ is an admissible loosely-symmetric octagon. Denote by $\tau_{k,m}$ the orientation preserving isometry of $\mathbb{D}$ mapping $\mathbf{z}_{k+5,m}$ to $\mathbf{z}_{k,m}$ and $\mathbf{z}_{k+4,m}$ to $\mathbf{z}_{k+1,m}$, for $k \in \{0, \ldots, 7\}$.

Consider first the power sequence of twists 0^m (the choice of 0 is without loss of generality) for $m \in \{0, \ldots, 3000\}$. A simple recursion gives the following for every m:

$$\tau_{k,m} = \begin{cases} (\tau_{0,0})^m \circ \tau_{k,0} & \text{if } k \in \{1,2,3\} \mod 8 \\ \tau_{k,0} \circ (\tau_{0,0})^{-m} & \text{if } k \in \{5,6,7\} \mod 8 \\ \tau_{k,0} & \text{if } k = 0 \mod 4 \end{cases}$$

$$\mathbf{z}_{k,m} = \begin{cases} (\tau_{0,0})^m(\mathbf{z}_{k,0}) & \text{if } k \in \{1,2,3\} \mod 8, \\ \mathbf{z}_{k,0} & \text{otherwise.} \end{cases}$$

From that it is easy to see that the sizes of the integers involved in the representations of $(\tau_{k,m})_{0 \leq k \leq 7}$ and $(\mathbf{z}_{k,m})_{0 \leq k \leq 7}$ grow at most linearly in m. See Figure 5.

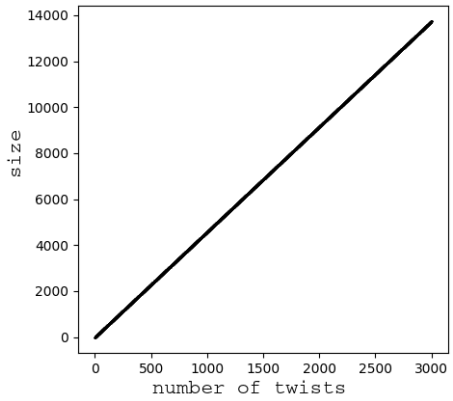

Figure 5 Size of the numerator of $\mathrm{Re}\,[z_{1,m}]$ as a function of the length m of a power sequence of twists.

When twisting with a random sequence, the growth in size of the integers involved does not appear to be linear in the number of twists. Examples are shown in Figure 6 for the random sequence 23330132013121032301 of $m = 20$ twists. Remark that the numerator of the real part of the top-left coefficient of the matrix representing $\tau_{0,20}$ contains more than $200,000$ digits in its decimal representation. In general the bottleneck for computations seems to be the sizes of such coefficients of the isometries $(\tau_{k,m})_{0 \leq k \leq 7}$.

6 Comparison of flip strategies

As recalled in Section 2.2, a Delaunay flip algorithm can flip Delaunay-flippable edges in any order. In this section, we consider six strategies:

- *naive* strategy: choose the first Delaunay-flippable edge given by the iterator `DartRange::iterator` on the CGAL combinatorial map.
- *random* strategy: choose uniformly among all the Delaunay-flippable edges.

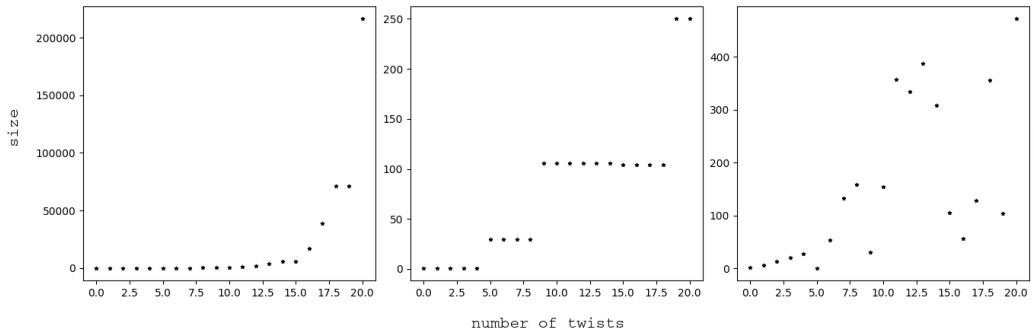

Figure 6 Sizes of integers as a function of the length m of a random sequence of twists. Left: the numerator of the real part of the top-left coefficient of the matrix representation of $\tau_{0,m}$. Middle: the numerator of $\mathrm{Re}\,[\mathbf{z}_{1,m}]$. Right: the numerator of $\mathrm{Re}\,[\mathbf{z}_{4,m}]$.

- *minimag* and *maximag* strategies: choose the edge e whose cross-ratio $\mathcal{R}_T(e)$ minimizes (resp. maximizes) $\mathrm{Im}\,[\mathcal{R}_T(e)]$ among the Delaunay-flippable edges.
- *minratio* and *maxratio* strategies: choose the edge e whose cross-ratio $\mathcal{R}_T(e)$ minimizes (resp. maximizes) the quotient $|\mathrm{Im}\,[\mathcal{R}_T(e)]| \,/\, |\mathcal{R}_T(e)|$.

We present eight experiments A, B, C, D, E, F, G, and H, allowing us to compare the number of flips that the six strategies induce on a variety of inputs. The notation Q_k, S_k and the parameters n_q, n_s, n_p are defined in Section 5.

experiment	A	B	C	D
n_q	50	30	10	1
n_s	50	30	10	10
n_p	0	10	100	1,000

Table 1 Parameters for experiments A, B, C, and D

experiment	E	F	G	H
n_q	100	30	10	10
Ω	$0, 30, 60, 90, 120$	$0, 30, 60, 90, 120$	$0, 10, 20, 30$	$0, 5, 10$
n_p	0	10	100	1,000

Table 2 Parameters for experiments E, F, G, and H

Let us first check that the strategy actually has an influence on the number of flips. Experiments A, B, C, and D use random sequences of twists. The values of n_q, n_s, n_p are shown in Table 1. We first construct the set X containing the 11 prefixes of the sequence of twists S_k (including the empty sequence) for every $k \in \{1, \ldots, n_s\}$: X contains at most $10n_s + 1$ sequences whose sizes vary between 0 and 10. Then for every $k \in \{1, \ldots, n_q\}$ and every $s \in X$, we perform steps 2 and 4 with $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}) = Q_k$, n_p interior vertices, and $t_1 \ldots t_m = s$. In each case, we run the Delaunay flip algorithm six times: one for each strategy, and count the number of flips that were needed for the algorithm to terminate. Among those six integers we denote by $\alpha_{k,s}$ the minimum and by $\beta_{k,s}$ the maximum. Figure 7 shows that choosing a strategy has an impact on the number of flips. A point lying far from the line $y = x$ represents a computation where one of the strategies clearly requires more flips than

another, while a point lying close to this line represents a computation where the strategies were essentially equivalent in the number of flips they induced.

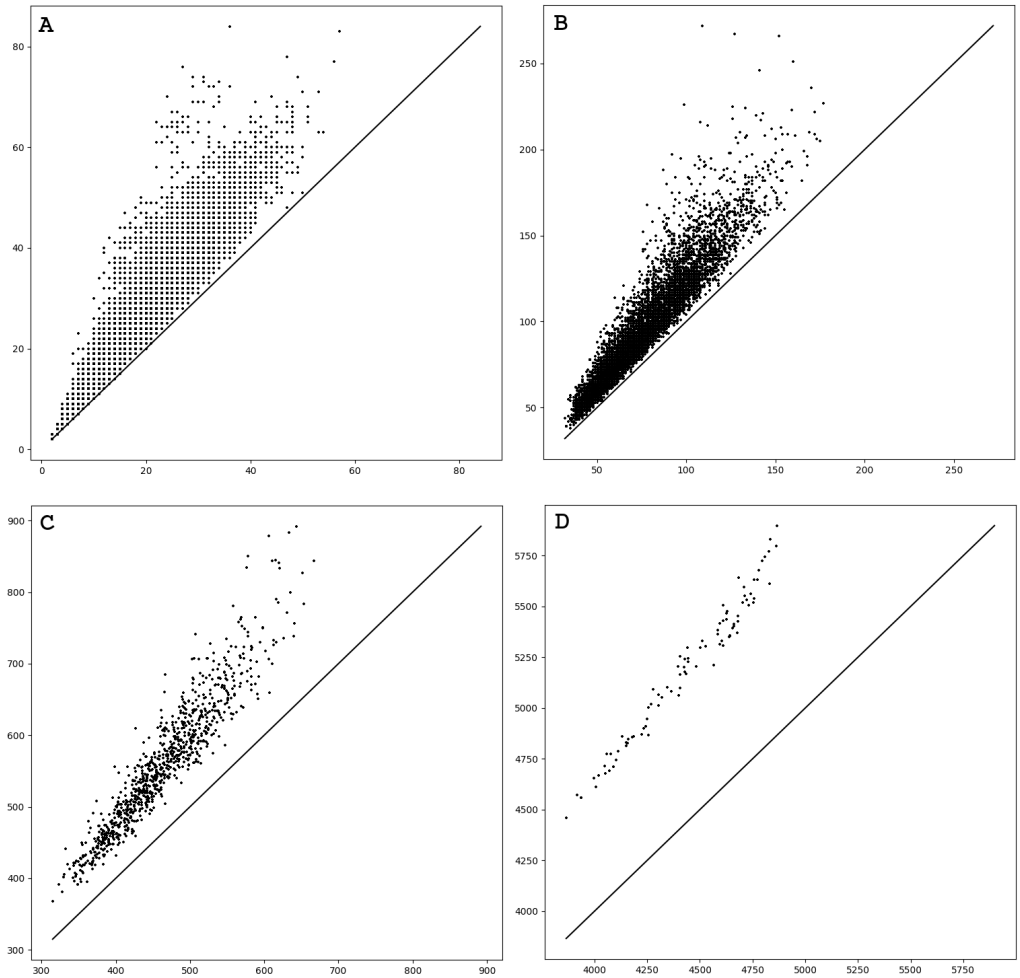

Figure 7 Experiments A, B, C, and D: points $(\alpha_{k,s}, \beta_{k,s})$ for $k \in \{1, \ldots, n_s\}$ and $s \in X$.

Experiments E, F, G, and H use power sequences of twists. They are parameterized by n_q, n_p, and a set Ω of integers giving the lengths of the considered twists, see Table 2. For every $k \in \{1, \ldots, n_q\}$, every $m \in \Omega$, and every $u \in \{0, 1, 2, 3\}$, we perform steps 2 and 4 with $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}) = Q_k$, n_p interior vertices, and $t_1 \ldots t_m = u^m$. Then we run the Delaunay flip algorithm for each of the six strategies. Here, the minimum and maximum number of flips are respectively denoted by $\alpha_{k,m,u}$ and $\beta_{k,m,u}$. Figure 8 shows a stronger impact of the strategy on the number of flips than experiments A, B, C, D.

To compare the six strategies, we count for each experiment and each strategy the number of times (i.e., the number of pairs $(\alpha_{k,s}, \beta_{k,s})$ for experiments A, ..., D or pairs $(\alpha_{k,m,u}, \beta_{k,m,u})$ for experiments E, ..., H) when the strategy induced the minimum/maximum number of flips among the other strategies. Figure 9 summarizes the results. Overall the minratio and the maxratio strategies seem to regularly achieve the maximum and the minimum (respectively). Remark in particular that in experiments D and H the minratio and the maxratio strategies always induced more and fewer flips, respectively, than any other strategy.

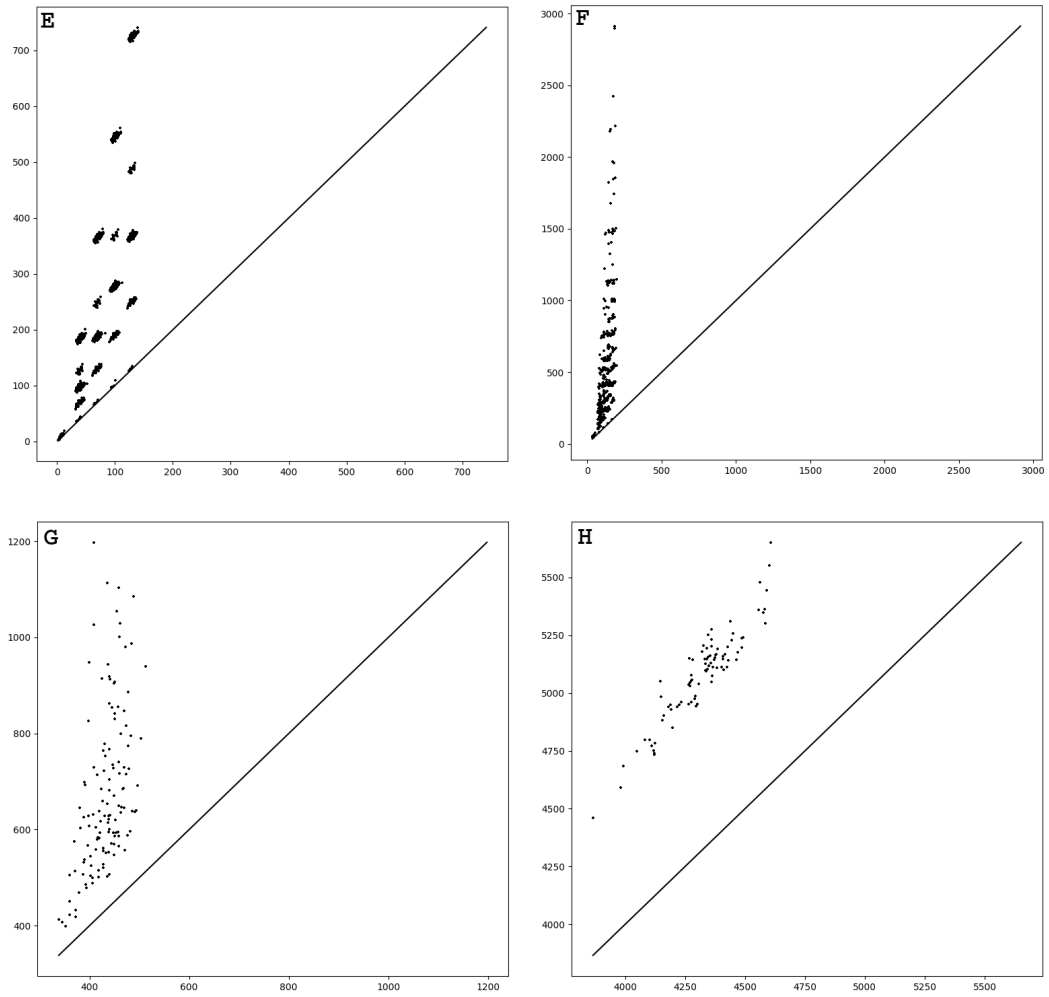

Figure 8 Experiments E, F, G, and H: points $(\alpha_{k,m,u}, \beta_{k,m,u})$, $k \in \{1, \ldots, n_q\}, m \in \Omega, u \in \{0, 1, 2, 3\}$.

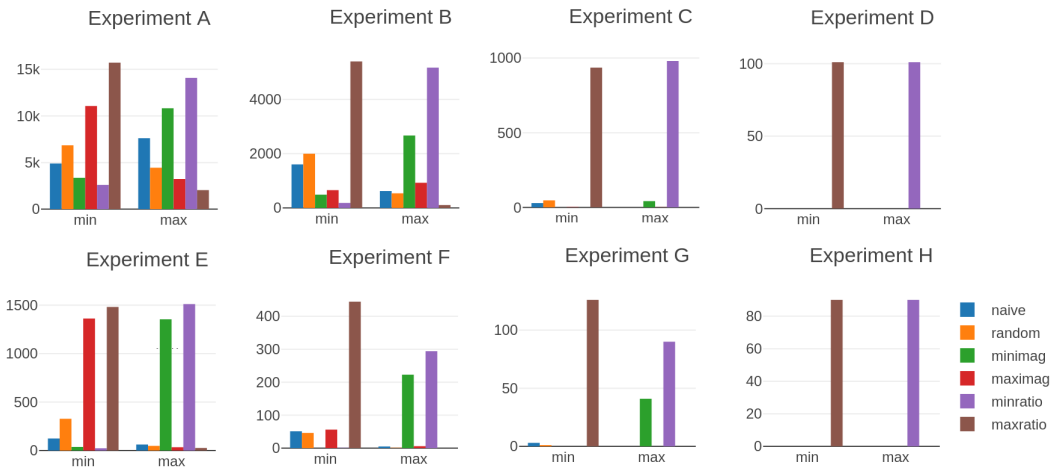

Figure 9 Number of times when each strategy induced the minimum/maximum number of flips

The naive strategy seems to rarely achieve the minimum or the maximum number of flips among the six strategies. In Figures 10 and 11, the ordinate is the number of flips induced by the naive strategy (instead of the maximum among the six strategies); the abscissa is still the minimum number of flips among the six strategies. The figures show that the number of flips required by the naive strategy is close to the minimum. As it runs much faster than all other strategies, we stick to the naive strategy for the experiments of Section 7.

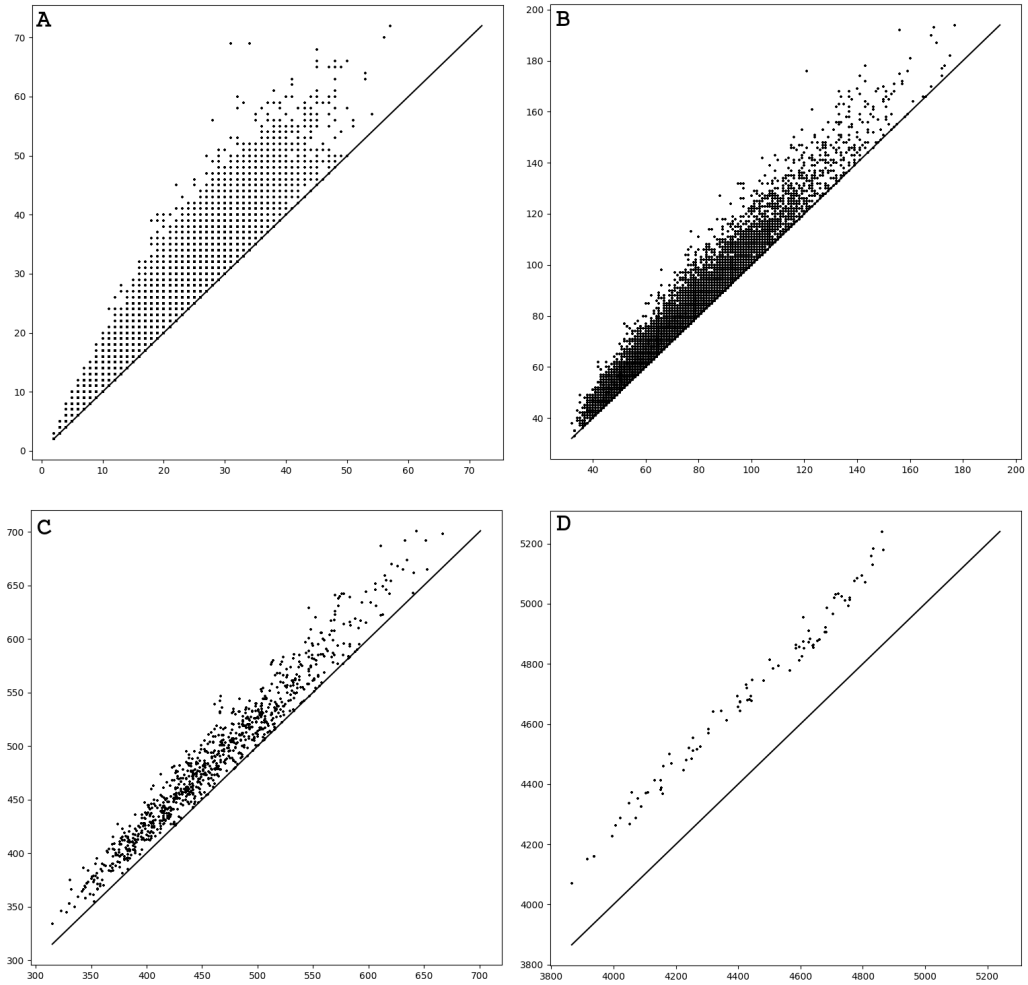

Figure 10 Number of flips induced by the naive strategy with respect to the minimum among the six strategies in experiments A, B, C, and D

Figure 12 illustrates a run of the program. The diameter of the initial domain is about 139 and the diameter of the final domain is smaller than 5.

7 Exploring the relationship between number of flips and diameter

7.1 Rationale for the experiments

Mark Bell [2] showed that the structure of the mapping class group has a very interesting effect on the flip graph of topological triangulations. In this topological setting, the objective is to reduce the number of intersections k of the input triangulation with a fixed curve. The

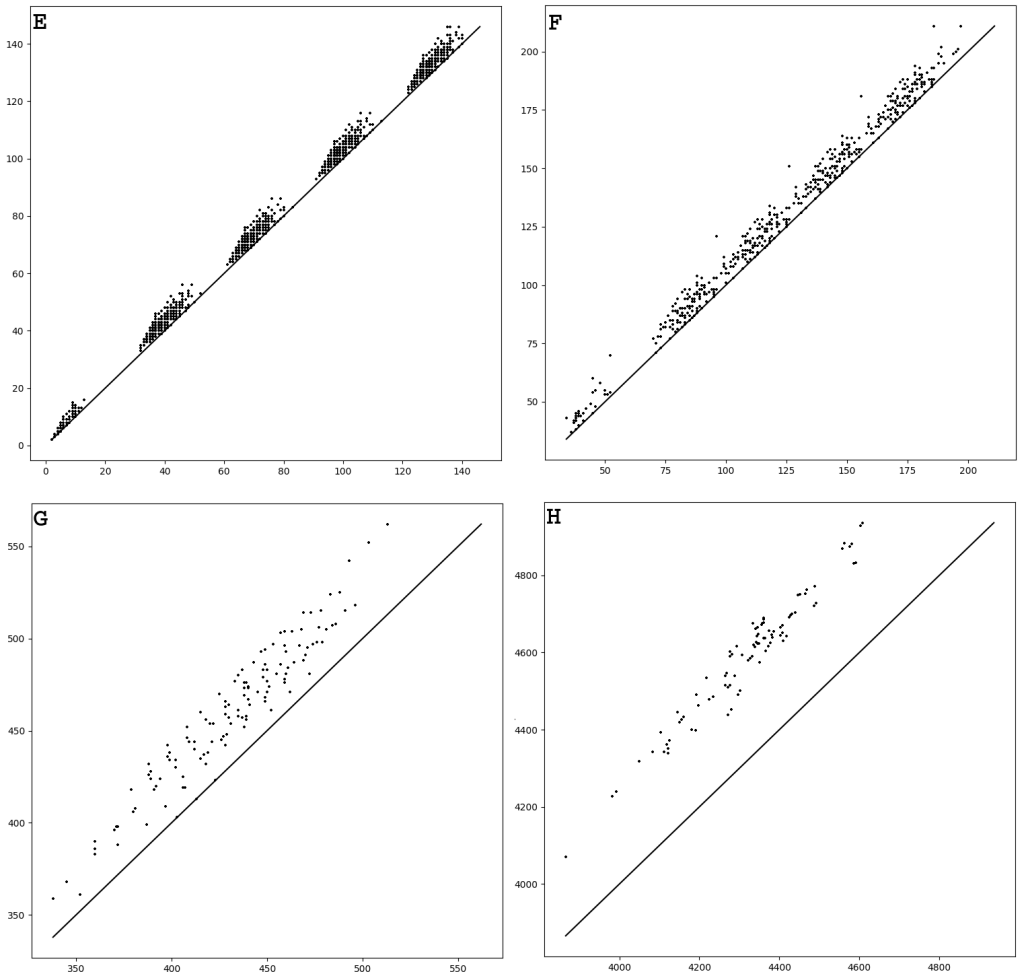

Figure 11 Same as Figure 10, for experiments E, F, G, and H

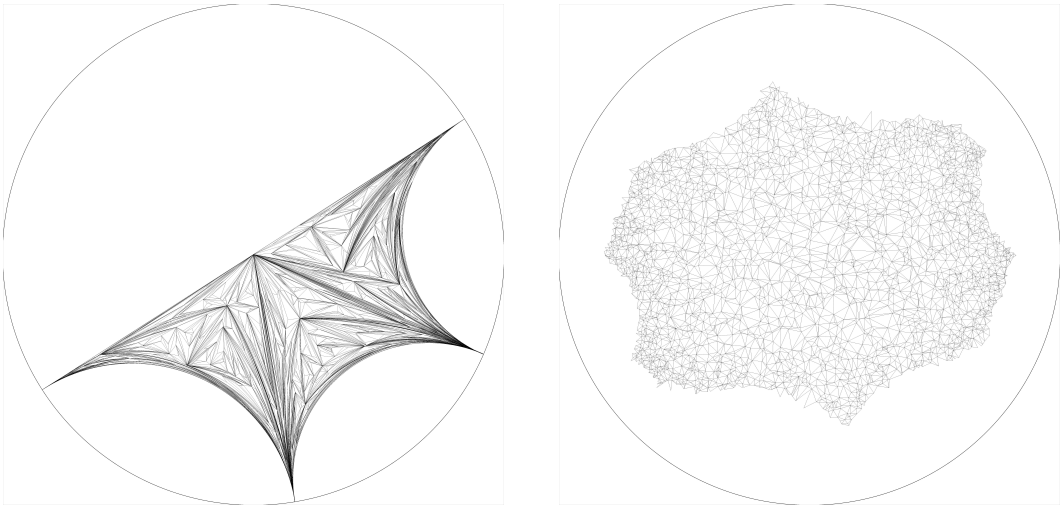

Figure 12 Triangulation with 3001 vertices before (left) and after (right) the flips

main theorem of the paper states that one can always find a flip or a power of a Dehn twist that reduces the number of crossings by a fixed percentage. This result can be seen as follows: either a pseudo-Anosov transformation allows to decrease the number of crossings in a single application, or there exists a power of a Dehn twist that reduces the number of crossings. This gives an algorithm to compute the optimal triangulation using $O(\log(k))$ operations.

Our problem is different from Mark Bell's: in his study, the number of crossings is an explicit measure of the distance to the goal, while there is no way to know in advance how far the input triangulation is from being Delaunay, and we do not know the homotopy classes of final edges. However, asymptotically, combinatorial intersection metrics are very similar to hyperbolic metrics on surfaces of genus $g \geq 2$. If a triangulation has very long edges (in terms of number of crossings for the topological version or in terms of hyperbolic length in our geometric setting), then in a first stage both strategies will aim at reducing edge lengths. Thus the two problems might have a similar asymptotic efficiency.

This raises two questions:

- Is there any hope to experimentally observe such a similar efficiency? It looks *a priori* unpromising as the above only holds asymptotically.
- Can Mark Bell's result be transposed to the number of flips?

Two sets of experiments will be carried out. The first set will construct the input triangulation by twisting the initial octagon in one direction only; as these twists correspond to reducible elements of $\mathrm{Mod}(S_2)$ (Section 2.3) we expect to observe a linear number of flips. The second set of experiments will twist the octagon in a random way; asymptotically, we should obtain pseudo-Anosov elements of the mapping class group and an asymptotic logarithmic behavior.

We present five experiments named I, J, K, L, and M, all using the naive strategy (see Section 6). We use again the same notation as in Section 5. We follow steps 3 and 4 and keep track of the loosely-symmetric octagon $G[\mathbf{z_0}', \ldots, \mathbf{z_7}']$ obtained in step 4 after the twists; we compute (an approximation represented by a C++ `double` of) the hyperbolic diameter of $G[\mathbf{z_0}', \ldots, \mathbf{z_7}']$ (see Appendix E). As we are only interested in the influence of the diameter, we do not run step 2 (i.e., $n_p = 0$) and the triangulation only has one vertex.

7.2 Exploring with power sequences

Experiments I and J are parameterized by the number n_q of 4-tuples: $n_q = 1$ in I and $n_q = 1,000$ in J. We perform step 4 with $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}) = Q_k$, $n_p = 0$ and $t_1 \ldots t_m = u^{3l}$ for $k \in \{1, \ldots, n_q\}$, $u \in \{0, 1, 2, 3\}$, $l \in \{0, \ldots, 50\}$ and we compute the approximate hyperbolic diameter $\varnothing_{k,l,u}$ of $G[\mathbf{z_0}', \ldots, \mathbf{z_7}']$. We run the Delaunay flip algorithm, counting the number $\alpha_{k,l,u}$ of flips that were needed by the algorithm to terminate. Figure 13 shows the result.

7.3 Exploring with random sequences

Table 3 shows the values of n_q and n_s for experiments K, L, and M. We first construct the set X containing the 11 prefixes of S_k (including the empty sequence) for every $k \in \{1, \ldots, n_s\}$. Then for every $k \in \{1, \ldots, n_q\}$ and every $s \in X$, we perform step 4 with $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}) = Q_k$, $n_p = 0$, and $t_1 \ldots t_m = s$. We compute the approximate diameter $\varnothing_{k,s}$ of $G[\mathbf{z_0}', \ldots, \mathbf{z_7}']$. We run the Delaunay flip algorithm and count the number $\alpha_{k,s}$ of flips that were needed by the algorithm to terminate. Figure 14 shows $\alpha_{k,s}$ as a function of $10 \ln(\varnothing_{k,s})$ for $k \in \{1, \ldots, n_q\}, s \in X$. Here ln denotes the natural logarithm (base e).

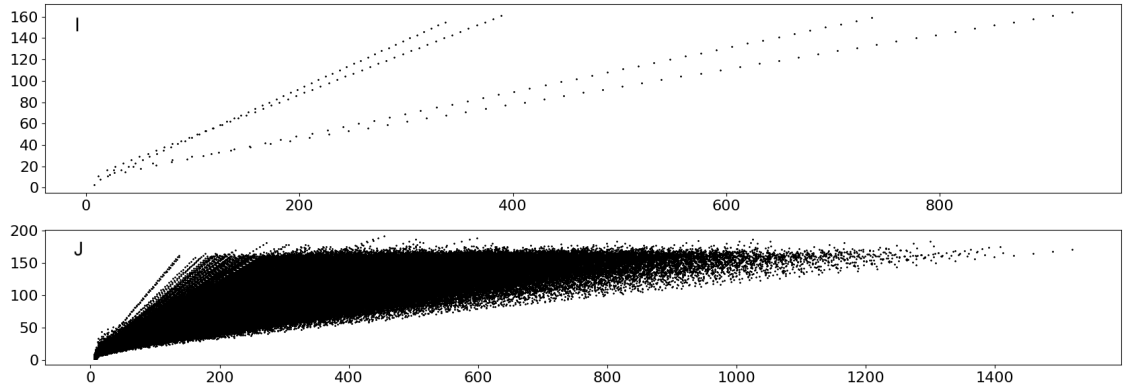

Figure 13 Experiments I and J: number of flips $\alpha_{k,l,u}$ with respect to the (approximate) diameter $\varnothing_{k,l,u}$, $k \in \{1, \ldots, n_q\}, l \in \{0, \ldots, 50\}, u \in \{0, 1, 2, 3\}$

experiment	K	L	M
n_q	1	10	1,000
n_s	10,000	1,000	100

Table 3 Parameters for experiments K, L, and M

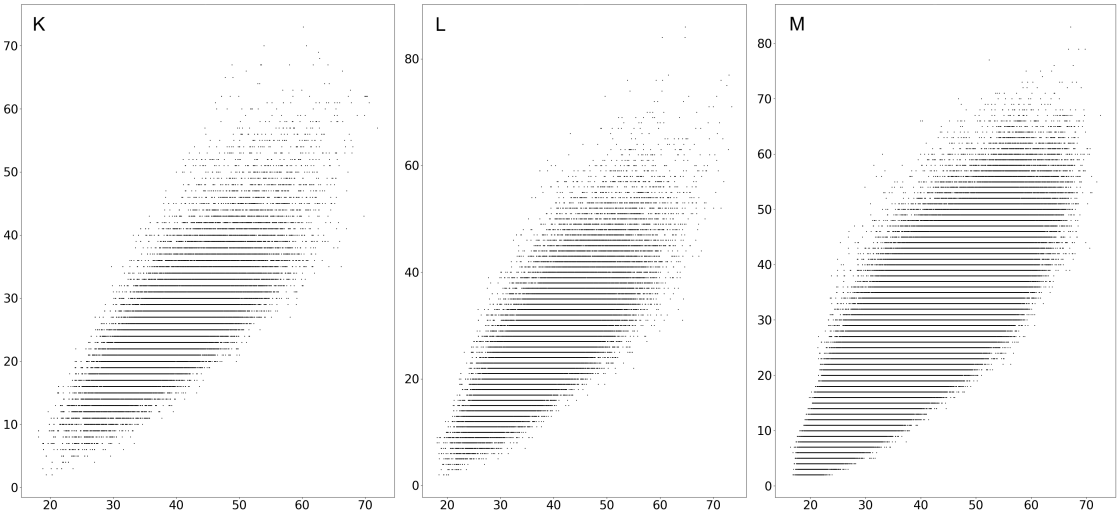

Figure 14 Experiments K, L, and M: number of flips with respect to $10 \ln(\varnothing_{k,s})$, $k \in \{1, \ldots, n_q\}, s \in X$; the maximum diameter is about 1500

7.4 Interpretation of the results

Our experiments show that controlling the elements of the mapping class group $\mathrm{Mod}(S_2)$ used for twisting actually allows us to control the number of flips needed by the flip algorithm. Indeed, in the case of power sequences, we observe that the number of flips is linear in the diameter of the input triangulation: Delaunay flips untwist the triangulation by performing a constant number of flips per iteration of the twist. However, for random sequences, we observe that the number of flips is logarithmic in the diameter of the input triangulation. In practice the Delaunay flip algorithm actually realizes a strategy that is as efficient as Mark Bell's.

Surprisingly, the asymptotic behavior of random walks in the mapping class group can be observed in practice with relatively small sequences of twists: even rather short random sequences reach pseudo-Anosov homeomorphisms, yielding the logarithmic behavior.

In light of our experimental results, we conjecture that the complexity of the Delaunay flip algorithm is worst-case linear in the diameter of the triangulation, and logarithmic on average.

References

1 Aline Aigon-Dupuy, Peter Buser, Michel Cibils, Alfred F. Künzle, and Frank Steiner. Hyperbolic octagons and Teichmüller space in genus 2. *Journal of mathematical physics*, 46(3):033513, 2005. doi:10.1063/1.1850177.

2 Mark C. Bell. Simplifying triangulations. *Discrete & Computational Geometry*, 66(1):1–11, 2021. doi:10.1007/s00454-021-00309-0.

3 M. Berger. *Geometry (vols. 1-2)*. Springer-Verlag, 1987.

4 Peter Buser. *Geometry and Spectra of Compact Riemann Surfaces*. Birkhäuser, Boston, 1992. doi:10.1007/978-0-8176-4992-0.

5 Guillaume Damiand. Combinatorial maps. In *CGAL User and Reference Manual*. CGAL Editorial Board, 5.2.1 edition, 2021. URL: https://doc.cgal.org/5.2.1/Manual/packages.html#PkgCombinatorialMaps.

6 Vincent Despré, Benedikt Kolbe, and Monique Teillaud. Representing infinite hyperbolic periodic Delaunay triangulations using finitely many Dirichlet domains (v2). Research report, INRIA, July 2021. URL: https://hal.inria.fr/hal-03045921v2.

7 Vincent Despré, Jean-Marc Schlenker, and Monique Teillaud. Flipping geometric triangulations on hyperbolic surfaces. In *Proceedings of the 36th International Symposium on Computational Geometry (SoCG'20)*, pages 35:1–35:16, 2020. URL: https://hal.inria.fr/hal-02886493, doi:10.4230/LIPIcs.SoCG.2020.35.

8 Matthijs Ebbens, Iordan Iordanov, Monique Teillaud, and Gert Vegter. Delaunay triangulations of generalized Bolza surfaces. Research report, INRIA, 2021. URL: https://hal.inria.fr/hal-03080125v2.

9 Benson Farb and Dan Margalit. *A primer on mapping class groups*, volume 49 of *Princeton Mathematical Series*. Princetown University Press, Princetown, 2012. doi:10.1515/9781400839049.

10 Albert Fathi, François Laudenbach, and Valentin Poénaru. *Thurston's Work on Surfaces (MN-48)*. Princeton University Press, 2012.

11 Michael Hemmer, Susan Hert, Sylvain Pion, and Stefan Schirra. Number types. In *CGAL User and Reference Manual*. CGAL Editorial Board, 5.3 edition, 2021. URL: https://doc.cgal.org/5.3/Manual/packages.html#PkgNumberTypes.

12 Iordan Iordanov and Monique Teillaud. Implementing Delaunay triangulations of the Bolza surface. In *Proceedings of the 33rd International Symposium on Computational Geometry (SoCG'17)*, pages 44:1–44:15, 2017. doi:10.4230/LIPIcs.SoCG.2017.44.

13 Nikolai V. Ivanov. *Subgroups of Teichmüller modular groups*, volume 115 of *Translations of Mathematical Monographs*. American Mathematical Soc., 1992. doi:10.1090/mmono/115.

14 Joseph Maher. Random walks on the mapping class group. *Duke Mathematical Journal*, 156(3):429–468, 2011. doi:10.1215/00127094-2010-216.

15 Bojan Mohar and Carsten Thomassen. *Graphs on Surfaces*. Johns Hopkins University Press, Baltimore, 2001.

16 Georg Osang, Mael Rouxel-Labbé, and Monique Teillaud. Generalizing CGAL periodic Delaunay triangulations. In *Proceedings 28th European Symposium on Algorithms*, pages 75:1–75:17, 2020. Best Paper Award (Track B: Engineering and Applications). URL: https://hal.inria.fr/hal-02923439, doi:10.4230/LIPIcs.ESA.2020.75.

17 Chee Yap *et al.* The CORE library project. URL: http://cs.nyu.edu/exact/core_pages/.

A Cross-ratios and Delaunay flips

We define the map $\phi : \mathbb{C} \times \mathbb{C} \to \mathbb{C}$ by $\phi(x, y) = 1 - (1 - x) \cdot y$ for every $(x, y) \in \mathbb{C}^2$.

To prove Lemma 5, we first give a straightforward lemma. Here, the triangulation $\mathcal{T}$ may be finite or infinite.

▶ **Lemma 4.** *Consider a triangulation $\mathcal{T}$ of $\mathbb{H}$ and an edge e of $\mathcal{T}$. Denote by f, g, h, k the edges, oriented counter-clockwise, of the quadrilateral formed by the two trianles of $\mathcal{T}$ that are incident to e. Assume that e is Delaunay-flippable and let $\mathcal{T}^\star$ be the triangulation obtained from $\mathcal{T}$ when replacing of e by the other diagonal $e^\star$ of the quadrilateral. Then:*

- $\mathcal{R}_{\mathcal{T}^\star}(e^\star) = \mathcal{R}_{\mathcal{T}}(e)/(\mathcal{R}_{\mathcal{T}}(e) - 1)$.
- $\mathcal{R}_{\mathcal{T}^\star}(w) = \phi(\mathcal{R}_{\mathcal{T}}(w), \mathcal{R}_{\mathcal{T}}(e))$ *for* $w \in \{f, h\}$.
- $\mathcal{R}_{\mathcal{T}^\star}(w) = \phi(\mathcal{R}_{\mathcal{T}}(w)), 1/\mathcal{R}_{\mathcal{T}^\star}(e^\star))$ *for* $w \in \{g, k\}$.

It is clear that the cross-ratio of any edge of $\mathcal{T}$ other than $\{e, f, g, h, k\}$ remains unchanged after the flip.

Proof. Consider the notation defined by Figure 15.

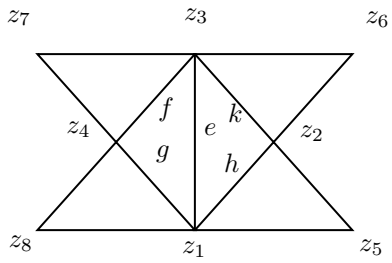

Figure 15 Notation for the proof of Lemma 4 (geodesic edges are represented by straight line segments)

A straightforward computation gives:

$$[z_1, z_2, z_5, z_3] \cdot [z_1, z_2, z_3, z_4] = [z_1, z_2, z_5, z_4]$$
$$[z_2, z_3, z_6, z_4] \cdot [z_2, z_3, z_4, z_1] = [z_2, z_3, z_6, z_1]$$
$$[z_3, z_4, z_7, z_1] \cdot [z_1, z_2, z_3, z_4] = [z_3, z_4, z_7, z_2]$$
$$[z_4, z_1, z_8, z_2] \cdot [z_2, z_3, z_4, z_1] = [z_4, z_1, z_8, z_3].$$

The result follows. ◀

Let us now state the result on $\mathcal{S}$.

▶ **Lemma 5.** *Consider a triangulation T of $\mathcal{S}$ and an edge e of T. Let f, g, h, k be the edges of T such that e, f, g and e, h, k (oriented counter-clockwise) bound the triangles incident to e in T. Assume that e is Delaunay-flippable and let $T^\star$ be the triangulation obtained from T after the flip of e and $e^\star$ be the new edge replacing e. Then the following holds:*

- $\mathcal{R}_{T^\star}(e^\star) = \mathcal{R}_T(e)/(\mathcal{R}_T(e) - 1)$.
- *If* $f \neq h$ *then* $\mathcal{R}_{T^\star}(w) = \phi(\mathcal{R}_T(w), \mathcal{R}_T(e))$ *for every* $w \in \{f, h\}$.
- *If* $f = h$ *then* $\mathcal{R}_{T^\star}(f) = \phi(\phi(\mathcal{R}_T(f), \mathcal{R}_T(e)), \mathcal{R}_T(e))$.
- *If* $g \neq k$ *then* $\mathcal{R}_{T^\star}(w) = \phi(\mathcal{R}_T(w), 1/\mathcal{R}_{T^\star}(e^\star))$ *for every* $w \in \{g, k\}$.
- *If* $g = k$ *then* $\mathcal{R}_{T^\star}(g) = \phi(\mathcal{R}_T(g), \phi(\mathcal{R}_T(g), 1/\mathcal{R}_{T^\star}(e^\star)), 1/\mathcal{R}_{T^\star}(e^\star))$.

Before going to the proof, note that some edges in $X = \{e, f, g, h, k\}$ may be equal. However, e, f, g are pairwise-distinct and e, h, k, too, as they bound faces of $\widetilde{T}$. Also, $f \neq k$ and $g \neq h$ because the interior angles of faces of T are all less than π. So the only two possible equalities in X are between f and h and between g and k.

One easily sees that the cross-ratio of any edge $w \notin X$ remains unchanged.

Proof. Consider the lift $\widetilde{T}$ of T. Choose some fixed lift $\widetilde{e}$ of e and let $\widetilde{f}, \widetilde{g}, \widetilde{h}, \widetilde{k}$ be edges of $\widetilde{T}$ such that $\widetilde{e}, \widetilde{f}, \widetilde{g}$ and $\widetilde{e}, \widetilde{h}, \widetilde{k}$ are the two faces incident to $\widetilde{e}$ in $\widetilde{T}$, oriented counter-clockwise. By swapping $\widetilde{f}, \widetilde{g}$ to $\widetilde{h}, \widetilde{k}$ if needed we can also assume that each $w \in X$ is lifted to $\widetilde{w}$. We define $\widetilde{X_1}$ as $\{\widetilde{f}, \widetilde{h}\}$ if $f \neq h$ or as $\{\widetilde{f}\}$ if $f = h$. We define $\widetilde{X_2}$ similarly for g and k. This way $\widetilde{X}$ contains exactly one lift of each element of X. Now we define $\widetilde{E}$ as the set of all lifts of e that are adjacent to one of the faces of $\widetilde{T}$ having an edge in $\widetilde{X}$. The possible configurations are summarized in Figure 16. Now we consider the infinite triangulation $\widetilde{T'}$ of the hyperbolic

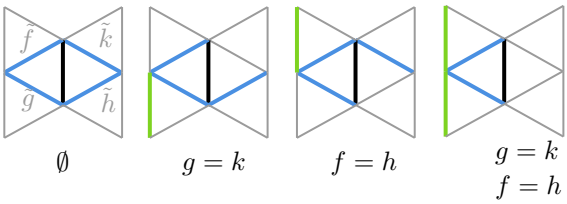

Figure 16 The possible configurations: $\widetilde{e}$ is the black edge, $\widetilde{X_1} \cup \widetilde{X_2}$ is shown in blue, and $\widetilde{E} \setminus \{\widetilde{e}\}$ is shown in green

plane obtained from $\widetilde{T}$ after flipping each element of $\widetilde{E}$. We denote by $\widetilde{e^\star}$ the edge of $\widetilde{T'}$ resulting from the flip of $\widetilde{e}$. Then for every $\widetilde{w} \in \widetilde{X} \setminus \{\widetilde{e}\}$ we have $\mathcal{R}_{T^\star}(w) = \mathcal{R}_{\widetilde{T'}}(\widetilde{w})$ and $\mathcal{R}_T(w) = \mathcal{R}_{\widetilde{T}}(\widetilde{w})$. Also, $\mathcal{R}_{T^\star}(e^\star) = \mathcal{R}_{\widetilde{T'}}(\widetilde{e^\star})$ and $\mathcal{R}_T(e) = \mathcal{R}_{\widetilde{T}}(\widetilde{e})$. The result follows by computing $\mathcal{R}_{\widetilde{T'}}(\widetilde{w})$ from $\mathcal{R}_{\widetilde{T}}(\widetilde{w})$ and $\mathcal{R}_{\widetilde{T'}}(\widetilde{e^\star})$ from $\mathcal{R}_{\widetilde{T}}(\widetilde{e})$, using Lemma 5 for each $\widetilde{w} \in \widetilde{E}$. ◀

B Flipping edges: details and pseudo-code

In this section, we show how an edge e of T is flipped. We use the notation of Section 3.3. F denotes the map that associates the cross-ratio $\mathcal{R}_T(f)$ to each edge f of T.

Algorithm 1 shows the operations performed in the combinatorial map, using the CGAL package [5]. The notation d_f stands for a dart supporting an edge f. Given a dart d we denote by $[d]$ the edge supported by d.

$d_f \leftarrow d_e.next$; $\quad d_g \leftarrow d_f.next$;
$d'_e \leftarrow d_e.opposite$; $\quad d_h \leftarrow d'_e.next$; $\quad d_k \leftarrow d_h.next$;
$d_f.next, d_e.next, d_k.next \leftarrow d_e, d_k, d_f$;
$d_g.next, d_h.next, d'_e.next \leftarrow d_h, d'_e, d_g$;

Algorithm 1 Flipping e in a combinatorial map.

Algorithm 2 computes an anchor for the triangulation $T^\star$ obtained after the flip. We claim that after the execution of the algorithm the new anchor actually represents one of the two faces of $\widetilde{T^\star}$ that are adjacent to $\widetilde{e^\star}$. First remark that such a face, name it $\widetilde{t'}$, shares two vertices with $\widetilde{t}$ and that the vertex of $\widetilde{t'}$ that is not shared with $\widetilde{t}$ can be computed from the three vertices of $\widetilde{t}$ and the cross-ratio of e in T: this computation is done by ϕ. To define ϕ we first define the subset $\Omega \subset \mathbb{C}^4$ as the set of 4-tuples $(x, y, z, r) \in \mathbb{C}^4$ such

that x, y, z are pairwise-distinct and $r(z-y) \neq (z-x)$. Now we can define $\phi : \Omega \to \mathbb{C}$ by $\phi(x, y, z, r) = (xr(z-y) + y(x-z))/(r(z-y) + x - z)$ for every $(x, y, z, r) \in \Omega$. This map ϕ is well-defined. Now we will explain why ϕ computes this third vertex of $\widetilde{t'}$ and why Algorithm 2 always gives to ϕ inputs that are in Ω. Consider some infinite triangulation $\mathcal{T}$ of $\mathbb{H}$ and some edge w of $\mathcal{T}$. Denote by u_1 and u_3 the vertices of w and by u_2 and u_4 the two other vertices of the two faces of $\mathcal{T}$ containing w: assume that u_1, u_2, u_3, u_4 are in counter-clockwise order. A simple computation shows that $(u_1, u_2, u_3, \mathcal{R}_{\mathcal{T}}(w)) \in \Omega$ and $u_4 = \phi(u_1, u_2, u_3, \mathcal{R}_{\mathcal{T}}(w))$. The correctness of Algorithm 2 then follows by case analysis.

switch δ **do**
  case d_e **do**
    $\delta \leftarrow d_h$;
    $a_2 \leftarrow \phi(a_2, a_3, a_1, F([d_e]))$;
  end
  case d'_e **do**
    $\delta \leftarrow d_f$;
    $a_2 \leftarrow \phi(a_2, a_3, a_1, F([d_e]))$;
  end
  case d_f *or* d_h **do**
    $a_3 \leftarrow \phi(a_1, a_2, a_3, F([d_e]))$;
  end
  case d_g *or* d_k **do**
    $a_3 \leftarrow \phi(a_3, a_1, a_2, F([d_e]))$;
  end
end

Algorithm 2 Updating the anchor $A = (\delta, a_1, a_2, a_3)$. The dart δ is modified if $\delta \in \{d_e, d_f, d_g, d'_e, d_h, d_k\}$.

Updating the cross-ratios during a flip is done by Algorithm 3, which is a direct implementation of Lemma 5.

$F([d_f]) \leftarrow 1 - (1 - F([d_f])) \cdot F([d_e])$;
if $d_f.opposite = d_h$ **then**
  $F([d_f]) \leftarrow 1 - (1 - F([d_f])) \cdot F([d_e])$;
else
  $F([d_h]) \leftarrow 1 - (1 - F(d_h])) \cdot F([d_e])$;
end
$F([d_e]) \leftarrow F([d_e])/(F([d_e]) - 1)$;
$F([d_g]) \leftarrow 1 - (1 - F([d_g]))/F([d_e])$;
if $d_g.opposite = d_k$ **then**
  $F([d_g]) \leftarrow 1 - (1 - F([d_g]))/F([d_e])$;
else
  $F([d_k]) \leftarrow 1 - (1 - F([d_k]))/F([d_e])$;
end

Algorithm 3 Updating the cross-ratios.

C Details for Section 4

C.1 Experiment on algebraic numbers (Section 4.1)

Algorithm 4 updates the cross-ratios through the sequence of 5 flips described in Section 4.1. It is is a straightforward implementation of Lemma 5

Input : The cross-ratios $R_0, \dots, R_4$
for $k = 0, \dots, 4$ **do**
  $R_k \leftarrow R_k/(R_k - 1)$;
  if $k \geq 1$ **then**
    $R_{k-1} \leftarrow 1 - (1 - R_{k-1})/R_k$;
  end
  if $k \leq 3$ **then**
    $R_{k+1} \leftarrow 1 - (1 - R_{k+1})/R_k$;
  end
end

Algorithm 4 Updating $R_0, \dots, R_4$ along the sequence of 5 flips.

C.2 Approximation algorithm

This section gives additional details on the construction of the rational admissible 4-tuple shown in the proof of Proposition 2 in Section 4.2.

▶ **Definition 6.** *Let $z_0, z_1, z_2, z_3 \in \mathbb{D} \setminus \{0_{\mathbb{C}}\}$ and $\varepsilon > 0$. We say that (z_0, z_1, z_2, z_3) is* ε-valid *if for any $k \in \{0, 1, 2, 3\}$ and $1 \leq l < m \leq 3$ the following properties are satisfied:*

- $\arg z_0 = 0$
- $0_{\mathbb{C}} \notin B(z_k, \varepsilon)$
- $\forall x \in B(z_l, \varepsilon), \forall y \in B(z_m, \varepsilon), 0 < \arg x < \arg y < \pi$.

Now let $\mu > 0$. If moreover $|\mathcal{A}(G[-z_0, z_0, z_1, z_2, z_3]) - 2\pi| < \mu$ then (z_0, z_1, z_2, z_3) is (ε, μ)-admissible.

In what follows we consider some $\varepsilon, \mu > 0$ and some rational (ε, μ)-admissible 4-tuple $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$. We describe below an algorithm returning a rational admissible 4-tuple $(\mathbf{z'_0}, \mathbf{z'_1}, \mathbf{z'_2}, \mathbf{z'_3})$ such that $\mathbf{z'_k} \in B(\mathbf{z_k}, \varepsilon)$ for every $k \in \{0, 1, 2, 3\}$. We will prove the correctness of the algorithm in Proposition 8 under some assumptions on ε, μ and on the input 4-tuple. Before describing the algorithm we state a preliminary Lemma.

▶ **Lemma 7.** *At least one of the 2 triangles $G[-\mathbf{z_0}, \mathbf{z_2}, \mathbf{z_3}]$ and $G[\mathbf{z_0}, \mathbf{z_2}, \mathbf{z_1}]$ has hyperbolic area bigger than $\frac{\pi}{2} - \frac{\mu}{2}$.*

Proof. This is clear since $G[-\mathbf{z_0}, \mathbf{z_0}, \mathbf{z_2}]$ is a triangle so its hyperbolic area is at most π. ◀

▶ **Proposition 8** (Correctness of the approximation algorithm)**.** *Assume $\varepsilon \in]0, 1[$ and $\mu \in]0, \pi/6[$. We introduce the following parameter:*

$$R = \max_{0 \leq k \leq 3} d(0_{\mathbb{C}}, \mathbf{z_k}).$$

If the following assumption is satisfied:

$$\varepsilon > 12\mu e^{6R} \tag{5}$$

then the approximation algorithm 5 is well-defined and correct.

Input : Reals $\varepsilon, \mu > 0$ and a rational (ε,μ)-admissible 4-tuple $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$
Output : A rational admissible 4-tuple $(\mathbf{z'_0}, \mathbf{z'_1}, \mathbf{z'_2}, \mathbf{z'_3})$ s.t. $\mathbf{z'_k} \in B(\mathbf{z_k}, \varepsilon)$, $\forall k$
if $\mathcal{A}(G[-\mathbf{z_0}, \mathbf{z_2}, \mathbf{z_3}]) > \frac{\pi}{2} - \frac{\mu}{2}$ **then**
  $\mathbf{f} : z \mapsto \frac{z+\mathbf{z_0}}{\mathbf{z_0}z+1}$;
  $P_0 \leftarrow \operatorname{Im}\left[(1 - \mathbf{f(z_0)}\overline{\mathbf{f(z_1)}})(1 - \mathbf{f(z_1)}\overline{\mathbf{f(z_2)}})\right]$;
  $P_1 \leftarrow \operatorname{Im}\left[(1 - \mathbf{f(z_0)}\overline{\mathbf{f(z_1)}})(1 - \mathbf{f(z_1)}\overline{\mathbf{f(z_2)}})(1 - \mathbf{f(z_2)}\overline{\mathbf{f(z_3)}})\right]$;
  $\lambda \leftarrow P_0/(P_0 - P_1)$;
  $V \leftarrow \lambda \mathbf{f(z_3)}$;
  return $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{f}^{-1}(V))$;
else
  $\mathbf{f} : z \mapsto \frac{z-\mathbf{z_0}}{-\mathbf{z_0}z+1}$;
  $P_0 \leftarrow \operatorname{Im}\left[(1 - \mathbf{f(z_3)}\overline{\mathbf{f(-z_0)}})(1 - \mathbf{f(z_2)}\overline{\mathbf{f(z_3)}})\right]$;
  $P_1 \leftarrow \operatorname{Im}\left[(1 - \mathbf{f(z_3)}\overline{\mathbf{f(-z_0)}})(1 - \mathbf{f(z_2)}\overline{\mathbf{f(z_3)}})(1 - \mathbf{f(z_1)}\overline{\mathbf{f(z_2)}})\right]$;
  $\lambda \leftarrow P_0/(P_0 - P_1)$;
  $V \leftarrow \lambda \mathbf{f(z_1)}$;
  return $(\mathbf{z_0}, \mathbf{f}^{-1}(V), \mathbf{z_2}, \mathbf{z_3})$;
end

Algorithm 5 Approximation algorithm

Proof. We will only consider the case $\mathcal{A}(G[-\mathbf{z_0}, \mathbf{z_2}, \mathbf{z_3}]) > \frac{\pi}{2} - \frac{\mu}{2}$ as the same arguments hold for the other case after application of Lemma 7. Using the notations introduced in the algorithm we will bound P_0 and P_1 and then deduce that λ is well-defined. Only then we will prove $V \in B(\mathbf{f(z_3)}, \varepsilon)$. We have $P_0 = \operatorname{Im}[Z_0]$ and $P_1 = \operatorname{Im}[Z_1]$ with

$$\begin{aligned} Z_0 &= (1 - \mathbf{f(z_0)}\overline{\mathbf{f(z_1)}})(1 - \mathbf{f(z_1)}\overline{\mathbf{f(z_2)}}) \\ Z_1 &= (1 - \mathbf{f(z_0)}\overline{\mathbf{f(z_1)}})(1 - \mathbf{f(z_1)}\overline{\mathbf{f(z_2)}})(1 - \mathbf{f(z_2)}\overline{\mathbf{f(z_3)}}). \end{aligned}$$

We already proved $\arg Z_1 = \frac{1}{2}\mathcal{A}(G[-\mathbf{z_0}, \mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}])$ so $|\arg Z_1 - \pi| < \frac{\mu}{2}$ holds since $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$ is (ε,μ)-admissible. Since $\mu < \pi$ then $|P_1| < \sin(\frac{\mu}{2}) < \frac{\mu}{2}$. Also we have

$$\begin{aligned} \arg Z_0 &= \frac{1}{2}\mathcal{A}(G[-\mathbf{z_0}, \mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}]) \\ &= \frac{1}{2}(\mathcal{A}(G[-\mathbf{z_0}, \mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}]) - \mathcal{A}(G[-\mathbf{z_0}, \mathbf{z_2}, \mathbf{z_3}])). \end{aligned}$$

So by Definition 6 and since $\frac{\pi}{2} - \frac{\mu}{2} < \mathcal{A}(G[-\mathbf{z_0}, \mathbf{z_2}, \mathbf{z_3}]) < \pi$ then

$$\frac{\pi}{2} - \frac{\mu}{2} < \arg Z_0 < \frac{3\pi}{4} + \frac{3\mu}{4}.$$

Moreover $\mathbf{f}$ has translation length $d(0_{\mathbb{C}}, \mathbf{z_0}) \leq R$ so for every $k \in \{0,1,2,3\}$ we obtain $d(0_{\mathbb{C}}, \mathbf{f(z_k)}) \leq 2R$ which implies $|\mathbf{f(z_k)}| \leq \tanh(R)$. From that we deduce

$$\begin{aligned} 1 > P_0 &> \left(1 - \tanh(R)^2\right)^2 \cdot \sin\left(\frac{3\pi}{4} + \frac{3\mu}{4}\right) \\ &\geq e^{-4R} \cdot \sin\left(\frac{3\pi}{4} + \frac{3\mu}{4}\right) \\ &> \frac{1}{3}e^{-4R} \end{aligned}$$

since $\mu < \frac{\pi}{6}$ and $\sin(\frac{7\pi}{8}) > \frac{1}{3}$. From the bounds on P_0 and P_1 and using Assumption (5) we deduce $P_0 > P_1$ so λ is well defined. Also we get

$$1 < \lambda < \frac{1}{1 - \frac{3}{2}\mu e^{4R}}.$$

It remains to prove $V \in B(\mathbf{f}(\mathbf{z_3}), \varepsilon)$. For the sake of clarity we denote by D the hyperbolic distance $d(0_{\mathbb{C}}, \mathbf{f}(\mathbf{z_3}))$. It is enough to show the following:

$$\lambda \tanh(\frac{D}{2}) < \tanh(\frac{D+\varepsilon}{2}).$$

To prove that we first remark that $x \mapsto \tanh(x) - x/2$ is increasing on $[0, 1/2[$ and maps 0 to 0 so since $\varepsilon/2 \in]0, 1/2[$ then $\tanh(\varepsilon/2) \geq \varepsilon/4$. From that and by applying Assumption (5) we obtain

$$\mu e^{4R} < \tanh(\varepsilon/2) e^{-2R}$$

and we conclude with the following implications:

$$\begin{aligned}
& 3\mu e^{4R} < \tanh(\varepsilon/2) e^{-2R} \\
\implies \quad & \frac{3}{2}\mu e^{4R} < \frac{1}{2}\tanh(\varepsilon/2)(1 - \tanh(R)^2) \\
\implies \quad & \frac{\tanh(D/2) + \tanh(D/2)^2 \tanh(\varepsilon/2)}{\tanh(D/2) + \tanh(\varepsilon/2)} < 1 - \frac{3}{2}\mu e^{4R} \\
\implies \quad & \frac{\tanh(\frac{D}{2})}{\tanh(\frac{D+\varepsilon}{2})} < 1 - \frac{3}{2}\mu e^{4R} \\
\implies \quad & \lambda \tanh(\frac{D}{2}) < \tanh(\frac{D+\varepsilon}{2}).
\end{aligned}$$

That concludes the proof. ◀

D Details for the generation of input (Section 5)

D.1 Generating an initial rational 4-tuple (step 1)

We follow the construction of 4-tuples [1, Section 3] recalled in Section 3.1 but only compute rational approximations of algebraic numbers.

We first construct for $k \in \{1, 2, 3\}$ the real and imaginary parts x_k and y_k of a complex number z_k; they are represented as `float` numbers in python and constructed in $[-1, 1]$ using the `random` method of the `random` package: x_k `= 2*random.random()-1`. That simulates a uniform distribution. We fail here if one of the points $\{z_1, z_2, z_3\}$ lies outside $\mathbb{D}$, or if Inequality (3) is not satisfied.

Then we construct the `float` numbers x_0 and y_0 representing the real and imaginary parts of z_0 as described by Equality (4). From that we construct for each $k \in \{0, 1, 2, 3\}$ the real and imaginary parts $\mathbf{x_k}$ and $\mathbf{y_k}$ of $\mathbf{z_k}$ as rational approximations of x_k and y_k: $\mathbf{x_k}$ = `int(N *` x_k`) / N`, where the parameter $\mathtt{N} \in \mathbb{N} \setminus \{0_{\mathbb{N}}\}$ determines the quality of the approximation and `int` is native in python. We arbitrarily chose $\mathtt{N} = 100$ in each computation. We fail if $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$ is not valid.

The rational 4-tuple $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$ is not necessarily admissible. However, by the construction method, it can be seen as a rational approximation of some admissible 4-tuple and it satisfies the hypothesis of Proposition 8. So, an admissible 4-tuple can be computed from it as in Section 4.2 (see also Algorithm 5).

To simplify notation, we still denote as $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$ the rational admissible 4-tuple that we obtain.

D.2 Generating points in an admissible symmetric octagon (step 2)

Consider the rational admissible 4-tuple $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$ obtained after step 1. In this section, we describe our method to construct a point $\mathbf{p} \in \mathbb{Q} + i\mathbb{Q}$ in the closure of the admissible symmetric octagon $\mathcal{P}[\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}]$, simulating a uniform distribution with respect to the hyperbolic metric.

The method uses inexact computation so it can fail especially if the Euclidean area of $\mathcal{P}[\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}]$ is close to 0. In particular this is why we do not generate such points in the admissible loosely-symmetric octagon resulting from the twists in step 4.

We start by dividing $\mathcal{P}[\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}]$ into 6 hyperbolic triangles $\Delta_1, \ldots, \Delta_6$. We compute the hyperbolic area of each triangle as a C++ (native) `double` number using Equality (2). Then we choose the triangle Δ_k that will contain $\mathbf{p}$ with probability $\frac{\mathcal{A}(\Delta_k)}{\sum_{l=1}^{6} \mathcal{A}(\Delta_l)}, k \in \{1, \ldots, 6\}$. By a translation we can assume that $0_{\mathbb{C}}$ is a vertex of Δ_k. We construct as `double` numbers the real and imaginary parts of a complex number $p \in \mathbb{D}$, simulating a uniform choice within the closure of Δ_k. To construct $\mathbf{p}$ from p we cast the real and imaginary parts of p into `CGAL::Gmpq` numbers [11]. Then we check using Lemma 9 whether $\mathbf{p}$ actually belongs to the closure of Δ_k; if so then we return $\mathbf{p}$.

▶ **Lemma 9.** *Consider pairwise-distinct points $z_1, z_2, z_3 \in \mathbb{D}$ and the oriented geodesic l containing z_1 and z_2 and oriented from z_1 to z_2. The oriented geodesic l disconnects $\mathbb{D}$ into 2 open regions and we consider the region R being on the left of l. We define $\tau : \mathbb{D} \to \mathbb{D}$ by*

$$\tau(z) = \frac{z - z_1}{1 - \overline{z_1} z}$$

for every $z \in \mathbb{D}$. Then $z_3 \in R$ if and only if

$$Im\left[\frac{\tau(z_3)}{\tau(z_2)}\right] > 0$$

and there is equality if and only if $z_3 \in l$.

Proof. The result follows from remarking that τ is an orientation preserving isometry of $\mathbb{D}$ sending z_1 to $0_{\mathbb{C}}$. ◀

D.3 Constructing the data structure (step 4)

After step 1, step 2, and step 3 we are given a rational admissible 4-tuple $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$, points $(p_1, \ldots, p_{n_p}) \in (\mathbb{Q} + i\mathbb{Q})^{n_p}$ lying in the closure of $\mathcal{P}[\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}]$ and a sequence $t_1, \ldots, t_m$ of twists. The rational admissible 4-tuple $(\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3})$ defines the closed hyperbolic surface $\mathcal{S}$.

We construct the vertices $\mathbf{z_0}', \ldots, \mathbf{z_7}'$ of the rational admissible loosely-symmetric octagon resulting from twisting $\mathcal{P}[\mathbf{z_0}, \mathbf{z_1}, \mathbf{z_2}, \mathbf{z_3}]$ according to the sequence $t_1, \ldots, t_m$ (see Section 3.2). We also construct from $\mathbf{p_1}, \ldots, \mathbf{p_{n_p}}$ new points $(\mathbf{p_1}', \ldots, \mathbf{p_{n_p}}') \in (\mathbb{Q} + i\mathbb{Q})^{n_p}$ lying in the closure of $G[\mathbf{z_0}', \ldots, \mathbf{z_7}']$. This is done recursively as for every twist and every point p lying in the closure of the octagon to be twisted we have an orientation preserving isometry τ such that either p of $\tau(p)$ lies in the closure of the octagon resulting from the twist (see

Section 3.2). We also compute the orientation preserving isometries $(\tau'_k)_{0\leq k\leq 7}$ mapping the opposite sides of $G[\mathbf{z_0}', \dots, \mathbf{z_7}']$.

Now we construct recursively a sequence $T_0, \dots, T_{n_p}$ of triangulations of $G[\mathbf{z_0}', \dots, \mathbf{z_7}']$. We start with the triangulation T_0 whose edges are the eihgt sides of $G[\mathbf{z_0}', \dots, \mathbf{z_7}']$ and the five geodesic segments between $\mathbf{z_0}'$ and $\mathbf{z_2}', \mathbf{z_3}', \mathbf{z_4}', \mathbf{z_5}', \mathbf{z_6}'$. The triangulation T_0 is represented by a combinatorial map M_0 and a map P_0 associating to each vertex v of M_0 its position $P_0(v)$ in $\mathbb{D}$. Then for $k \in \{1, \dots, n_p\}$ the triangulation T_k is obtained from T_{k-1} by splitting the triangle containing $\mathbf{p_k}'$ into three triangles. In the end we get a triangulation T_{n_p} together with its combinatorial map M_{n_p} and the map P_{n_p} giving the position of each vertex in $\mathbb{D}$. By identifying the edges of T_{n_p} that are opposite sides of $G[z'_0, \dots, z'_7]$ we obtain a triangulation T of $\mathcal{S}$.

We can finally construct from M_{n_p} and P_{n_p} the triple (M, F, A) representing T (see Section 3.3). The combinatorial map M is easily obtained from M_{n_p} by setting $\beta_2(d) = d'$ and $\beta_2(d') = d$ (see Figure 2) for any 2 distinct darts d and d' of M_{n_p} supporting 2 edges corresponding to opposite sides of $G[z'_0, \dots, z'_7]$. The anchor $A = (\delta, a_1, a_2, a_3)$ is defined by choosing δ in M_{n_p}: the dart δ belongs to a face (v_1, v_2, v_3) of M_{n_p} and is based at v_1; we set $a_k = P_{n_p}(v_k)$ for every $k \in \{1, 2, 3\}$. Now consider some edge e of M. There are 2 cases. If e results from an edge of M_{n_p} that was not a side of $G[\mathbf{z_0}', \dots, \mathbf{z_7}']$ then computing its cross-ratio in T_{n_p} or equivalently in T is straightforward. If e results from the identification of 2 edges e_1 and e_2 of M_{n_p} then we compute $F(e)$ as follows. We denote by a, b the vertices in M_{n_p} of e_1 and by c, d the vertices of e_2 such that $P_{n_p}(a), P_{n_p}(b), P_{n_p}(c), P_{n_p}(d)$ appear in counter-clockwise order on the boundary of $G[\mathbf{z_0}', \dots, \mathbf{z_7}']$: when identifying e_1 and e_2 to construct M from M_{n_p} the vertex a is identified with d and b is identified with c. We consider $k \in \{0, \dots, 7\}$ such that orientation preserving isometry τ'_k maps $P_{n_p}(d)$ to $P_{n_p}(a)$ and maps $P_{n_p}(c)$ to $P_{n_p}(b)$. The edge e_1 belongs to a unique face f_1 of M_{n_p} and we denote by u_1 the vertex of f_1 that is neither a nor b. Similarly the edge e_2 belongs to a unique face f_2 of M_{n_p} and we denote by u_2 the vertex of f_2 that is neither c nor d. Then $F(e) = [P_{n_p}(a), \tau_k(P_{n_p}(u_2)), P_{n_p}(b), P_{n_p}(u_1)]$.

E Computation of the approximation of the diameter (Section 7)

Consider a rational admissible loosely-symmetric octagon O given by the 16 rational numbers representing the real and imaginary parts of its 8 vertices. The interior angles of O are not greater than π (since they sum up to 2π and the interior angles of opposite vertices are equal) so the hyperbolic diameter of O is the maximum of the hyperbolic distances between any two of its vertices. For every pair z_1, z_2 of two such distinct vertices we compute an approximation represented by a C++ `double` D of the hyperbolic distance between z_1 and z_2. The maximum (obtained using `std::max`) of these $\binom{8}{2}$ values is an approximation of the hyperbolic diameter of O.

We compute every such D as follows. The isometry $f : z \mapsto (z - z_1)/(1 - z_1 z)$ maps $\mathbb{Q} \cap \mathbb{D}$ to a subset of $\mathbb{Q}$ and maps z_1 to 0. We compute the exact rational value r_2 of the square of the modulus of $f(z_2)$. Then we convert r_2 to a `CORE::Expr` r'_2 and set $x = (1 + \texttt{CGAL::sqrt}(r'_2))/(1 - \texttt{CGAL::sqrt}(r'_2))$. The number D is an approximation of the natural logarithm $\ln(x)$ of x obtained by first casting x to a string s. This string s contains the string representation s_1 of the lower integer rounding k of $\log_{10}(x)$. Also s contains the string representation s_2 of an approximation of $x \cdot 10^{-k}$. The value of D is obtained as $\texttt{std::stoi}(s_1) * \texttt{std::log}(10) + \texttt{std::log}(\texttt{std::stod}(s_2))$.