# Computing a Dirichlet domain for a hyperbolic surface

Vincent Despré, Benedikt Kolbe, Hugo Parlier, Monique Teillaud

# Computing a Dirichlet domain for a hyperbolic surface*

Vincent Despré[1], Benedikt Kolbe[2], Hugo Parlier[3], and Monique Teillaud[4]

[1]Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France, http://vdespre.free.fr/

[2]Hausdorff Center for Mathematics, University of Bonn, Germany , https://hyperbolictilings.wordpress.com/

[3]Department of Mathematics, University of Luxembourg , https://math.uni.lu/parlier/

[4]Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France , https://members.loria.fr/monique.teillaud/

December 1, 2022

**Abstract**

The goal of this paper is to exhibit and analyze an algorithm that takes a given closed orientable hyperbolic surface and outputs an explicit Dirichlet domain. The input is a fundamental polygon with side pairings. While grounded in topological considerations, the algorithm makes key use of the geometry of the surface. We introduce data structures that reflect this interplay between geometry and topology and show that the algorithm finishes in polynomial time, in terms of the initial perimeter length and the genus of the surface.

## 1 Introduction and motivation

Hyperbolic surfaces and their moduli spaces play an ubiquitous role in mathematics, namely, through relationships with other areas including Riemannian geometry, number theory, geometric group theory and mathematical physics. Algorithms for surface groups, as combinatorial or topological objects, have a rich history dating back to Dehn. Recently, in part motivated by applications in other sciences [1, 16], there has been a push to understand hyperbolic structures on surfaces from the point of view of computational geometry.

*This work was partially supported by grant ANR-17-CE40-0033 of the French National Research Agency ANR (project SoS) and INTER/ANR/16/11554412/SoS of the Luxembourg National Research fund FNR https://SoS.loria.fr/.

Dealing with hyperbolic surfaces necessarily involves describing them — or even visualizing them — meaningfully. A fundamental domain (in the hyperbolic plane) with a side pairing is one way to determine a hyperbolic metric on the surface. Lengths of curves in a pants decomposition and their associated pasting parameters (so-called Fenchel-Nielsen coordinates) are another. No matter which construction or parameter set used, it is always interesting to know to which extent two different constructions output the "same" surface, where "same" can take different meanings. However, these representations, either by a fundamental domain or a set of Fenchel-Nielsen coordinates, are not unique, and determining a canonical representation is challenging for either option. In this paper, we tackle this question for fundamental domains, by computing a so-called Dirichlet domain.

Roughly speaking, a Dirichlet domain of a hyperbolic surface is a fundamental polygon in the hyperbolic plane, with a special point where distances to that point in the polygon correspond to distances on the surface. Another way of thinking of them is that it is a Voronoi cell associated to a lift of a single point of the surface to its universal cover $\mathbb{H}^2$. A more precise definition is given in the next section. Note that for hyperbolic surfaces any given surface has infinitely many Dirichlet domains up to isometry. This is in strong contrast to, for example, flat tori. Nonetheless, when describing a surface via fundamental domains, the prize for the most relevant geometric domain undoubtedly goes to Dirichlet domains because they visualize the distance function for a given point. As far as we know, there is only one algorithm in the literature that computes a Dirichlet domain for a hyperbolic surface and a given point [17]. Unfortunately, the run-time of the algorithm is not studied and an analysis seems complicated.

The contribution of this paper is an algorithm that computes a Dirichlet domain efficiently, and its analysis. The point defining the domain is not given as input, but it is part of the output. The Dirichlet domain of a given input point can then be computed with a complexity that only depends of the genus of the surface [10]. Our main result is the following:

**Theorem 1.** *Let $S$ be a closed orientable hyperbolic surface of genus $g$ given by a fundamental polygon of perimeter $L$ and side pairings. A Dirichlet domain for $S$ can be computed in $O((g^2L)^{6g-4})$ time.*

A key ingredient is the use of *Delaunay triangulations* on hyperbolic surfaces, an area of research that has recently gained traction, both from an experimental and a theoretical perspective [3, 7, 4, 14, 8, 12]. Recently, it has been shown that the well-known flip algorithm that computes the Delaunay triangulation of a set of points in the Euclidean plane $\mathbb{E}^2$ also works on a hyperbolic surface; the complexity result announced in Theorem 1 crucially depends on the only known upper bound on the complexity of this Delaunay flip algorithm [11]. The algorithm subsumes the real RAM model. Studying the algebraic numbers involved in the computations goes beyond the scope of this paper.

The paper is structured as follows: In Section 3, we give an overview of the algorithm and we present the data structure. Sections 4 and 5 explain in detail the main two steps of the algorithm, which output a geometric triangulation of the surface having only one vertex. Finally, Section 6 builds on the literature and concludes the proof of Theorem 1 with the last step of the algorithm.

# 2 Preliminaries

We begin by recalling a collection of facts and setting notations, and we refer to [2, 5, 13] for details. The surfaces studied in this paper are assumed to be closed, orientable, and of genus $g \geq 2$. We begin with a topological surface and endow it with a hyperbolic metric to obtain a hyperbolic surface, generally denoted by $S$. A hyperbolic surface is locally isometric to its universal covering space, the hyperbolic plane $\mathbb{H}^2$. Such surfaces can always be obtained by considering the quotient of $\mathbb{H}^2$ under the action of $\Gamma$, a discrete subgroup of isometries of $\mathbb{H}^2$ isomorphic to the fundamental group $\pi_1(S)$.

Let $S := \mathbb{H}^2/\Gamma$ be a hyperbolic surface of genus $g$ and fundamental group $\Gamma$. The projection map is denoted as $\rho : \mathbb{H}^2 \to S = \mathbb{H}^2/\Gamma$. We denote by $\widetilde{x} \in \rho^{-1}(x)$ one of the lifts, to $\mathbb{H}^2$, of an object $x$ on $S$. More generally, objects in $\mathbb{H}^2$ are denoted with $\widetilde{\ }$.

A *fundamental domain* $\mathcal{F}$ for the action of $\Gamma$ is defined as a closed domain, i.e., $\overline{\text{int}(\mathcal{F})} = \mathcal{F}$, such that $\Gamma\mathcal{F} = \mathbb{H}^2$ and the interiors of different copies of $\mathcal{F}$ under $\Gamma$ are disjoint.

For a point $\widetilde{x} \in \mathbb{H}^2$, the *Dirichlet domain* $\mathcal{D}_{\widetilde{x}}$ is defined as the Voronoi cell containing $\widetilde{x}$, of the Voronoi diagram associated to the point set $\Gamma\widetilde{x}$. In other words,

$$\mathcal{D}_{\widetilde{x}} = \{\, \widetilde{y} \in \mathbb{H}^2 \mid d_{\mathbb{H}^2}(\widetilde{x}, \widetilde{y}) \leq d_{\mathbb{H}^2}(\widetilde{x}, \Gamma\widetilde{y}) \,\} = \{\, \widetilde{y} \in \mathbb{H}^2 \mid d_{\mathbb{H}^2}(\widetilde{x}, \widetilde{y}) \leq d_{\mathbb{H}^2}(\Gamma\widetilde{x}, \widetilde{y}) \,\},$$

where the equality is true since $\Gamma$ acts as isometries w.r.t. $d_{\mathbb{H}^2}$. The Dirichlet domain is a compact convex fundamental domain for $\Gamma$ with finitely many geodesic sides [2, §9.4] and is generally considered a canonical choice of fundamental domain. A property of Dirichlet domains, of interest for the conception of algorithms, is that, by the triangle inequality,

$$\operatorname{diam}(\mathcal{D}_{\widetilde{x}}) \leq 2 \operatorname{diam}(S) \leq 2 \operatorname{diam}(\mathcal{D}_{\widetilde{x}}),$$

where $\operatorname{diam}(\cdot)$ denotes the diameter.

## 2.1 Curves, paths, and loops

Recall that a closed curve is the image of $\mathbb{S}^1$ under a continuous map; a curve is non-trivial (or essential) if it is not freely homotopic to a point. Similarly, a path is a continuous image of the interval $[0, 1]$, and the images of 0 and 1 are referred to as its endpoints. A loop is a path whose endpoints are equal; this endpoint is referred to as its basepoint.

For a closed curve or loop $c$, we will denote by $[c]$ its free homotopy class, and, if $c$ is based in a point $p$, by $[c]_p$ its homotopy class of loops based in $p$. For a path $c$ between points $p$ and $q$, we denote by $[c]_{p,q}$ the homotopy class of the path with fixed endpoints. We will readily make use of the fact that if $c$ is closed non-trivial curve on a hyperbolic surface, then in $[c]$ there is a unique closed geodesic. Similarly, if $c$ is a loop based in $p$, in $[c]_p$ there is a unique closed geodesic loop, and if $c$ is a path between $p$ and $q$, in $[c]_{p,q}$ there is a unique geodesic path. If $c$ is a simple closed curve then the closed geodesic in $[c]$ is also simple, but this is no longer necessarily the case for loops or paths with basepoints.

The intersection number $i(c, c')$ between homotopy classes of curves $c$ and $c'$ is defined as the minimal intersection among its representatives. Note that closed geodesics on a hyperbolic surface always intersect minimally. The situation for paths is slightly different. The unique geodesic representatives of paths (with fixed end points) might not intersect minimally. This subtlety plays a key technical role in our story.

### 2.2 Fundamental polygon

Let $S$ be a (closed) hyperbolic surface of genus $g$ and fundamental group $\Gamma$. A polygon $P \subset \mathbb{H}^2$ (i.e., a circular sequence of geodesic edges) bounding a fundamental domain for $\Gamma$ (as defined in the introduction) is called a fundamental polygon. Poincaré's theorem implies that $\Gamma$ is generated by the side pairings on $P$ [2, §9.8]. The edges and vertices of $P$ project to a graph $G_P$ on $S$; the region enclosed by $P$ projects to the unique face of $G_P$.

The numbers $n_G$ of vertices and $m_G$ of edges of $G_P$ satisfy Euler's relation $n_G - m_G + 1 = 2 - 2g$, as there is only one face. It follows that if $G_P$ only has one vertex, then that vertex is incident to the $m_G = 2g$ edges, which are actually all loops. The number of vertices is maximal when they all have degree 3 (then there are no loops); in this case $3n_G = 2m_G$, so, $m_G = 6g - 3$ and $n_G = 4g - 2$. More generally, the number $2m_G$ of edges and vertices of $P$ lies between the two extreme cases: $4g \leq 2m_G \leq 12g - 6$. When $2m_G < 12g - 6$, some vertices of $P$ project to the same vertex of $G_P$, i.e., they belong to the same orbit under $\Gamma$. $G_P$ has a loop for each edge whose vertices are in the same orbit; then the projected point on $S$ is incident to that loop twice.

## 3 Algorithm overview

Let $S$ be a (closed) hyperbolic surface of genus $g$ and fundamental group $\Gamma$.

We propose the algorithm sketched below to compute a Dirichlet fundamental domain of $S$. The output of Step 1 will be denoted with primes; it will be used as input for Step 2, whose output will be denoted with double primes.

1. Construct a system $\beta'_0, \ldots, \beta'_{2g-1}$ of simple topological loops based at the same point $b'$ that cuts $S$ into a disk (Section 4).

2. Find a point $b''$ so that the system of geodesic loops based at $b''$, conjugate to the ones computed in Step (1), is embedded (Section 5).

3. Construct the Dirichlet domain of a lift $\widetilde{b''}$ of $b''$ (Section 6).

Obviously, the complexity of the algorithm heavily depends on the data structure used to store the objects involved in the constructions. As the algorithm actually operates in the universal covering space $\mathbb{H}^2$ of $S$, it is natural to present the data structure in $\mathbb{H}^2$. We assume that, as input, we are given a fundamental polygon $\Pi \subset \mathbb{H}^2$ for $\Gamma$, together with side pairings, as in Section 2.2. The data structure described below is actually equivalent to a combinatorial map [15, Section 3.3] on $S$, enriched with geometric information. In particular, for each vertex $x$ of $G_\Pi$ (the projection of $\Pi$ onto $S$, as in Section 2.2), the sequence of edges around $x$ is ordered (edges that correspond to a loop appear twice).

**Description of the input.** Let a representative $\widetilde{e_i}, i = 0, \dots, m-1$ be chosen for each couple of paired edges of $\Pi$ and denote as $\gamma_0, \dots, \gamma_{m-1}$ the corresponding side pairings in $\Pi$: the other edge of the couple is ${\gamma_i}^{-1}\widetilde{e_i}$, where ${\gamma_i}^{-1}$ is the inverse of $\gamma_i$. We denote the set of the $2m$ edges of $\Pi$ as $E_\Pi$ and the set of its $2m$ vertices as $V_\Pi$. We choose a representative $\widetilde{v_j}, j = 0, \dots, n-1$ for each orbit of vertices of $\Pi$; $n$ is the number of vertices of $G_\Pi$.

Each element of $\Gamma$ can be represented as a word on the alphabet $\mathcal{A}_\Gamma = \{\mathbb{1}, \gamma_0, \dots, \gamma_{m-1},$
${\gamma_0}^{-1}, \dots, {\gamma_{m-1}}^{-1}\}$, where $\mathbb{1}$ denotes the identity in $\Gamma$. Here, letters of $\mathcal{A}_\Gamma$ and the corresponding generators in $\Gamma$ are denoted by the same symbol; this should not cause any confusion.

The data structure is roughly a doubly linked list of edges of $\Pi$, which stores the combinatorial information. Additional information is necessary to store the geometry (i.e., the positions of the vertices of $\Pi$ in $\mathbb{H}^2$) and the side pairings. The data stored for each edge and vertex is constant, so the size of the data structure is $O(g)$ (we do not try to shave constants in the $O()$).

Concretely, for each edge $\widetilde{x} \in E_\Pi$, the data structure stores:

- two pointers $\mathsf{prev}(\widetilde{x})$ and $\mathsf{next}(\widetilde{x})$ that give access to the previous and next edges in $\Pi$, respectively (in counterclockwise order);
- two pointers $\mathsf{source}(\widetilde{x})$ and $\mathsf{target}(\widetilde{x})$ that give access to the source and target of $\widetilde{x}$ in $\Pi$, respectively (in counterclockwise order); when $\rho\widetilde{x}$ is a loop in $G_\Pi$, $\mathsf{source}(\widetilde{x})$ and $\mathsf{target}(\widetilde{x})$ lie in the same orbit under $\Gamma$;
- a pointer to the paired edge $\mathsf{pair}(\widetilde{x})$ in $\Pi$;
- a letter $\mathsf{w}(\widetilde{x}) \in \mathcal{A}_\Gamma$ that encodes the relation between $\widetilde{x}$ and $\mathsf{pair}(\widetilde{x})$:
  $$\mathsf{w}(\widetilde{x}) = \begin{cases} \mathbb{1} & \text{if } \widetilde{x} = \widetilde{e_i} \\ \gamma_i & \text{if } \widetilde{x} = {\gamma_i}^{-1}\widetilde{e_i} \end{cases} \text{ for some } i \in \{0, \dots, m-1\}.$$
  By definition, $\mathsf{pair}(\widetilde{x}) = \begin{cases} {\gamma_i}^{-1}\widetilde{x} & \text{when } \mathsf{w}(\widetilde{x}) = \mathbb{1} \ (\widetilde{x} = \widetilde{e_i}) \\ \gamma_i\widetilde{x} & \text{when } \mathsf{w}(\widetilde{x}) = \gamma_i \end{cases}$.

For each vertex $\widetilde{y} \in V_\Pi$, the data structure stores:

- $\mathsf{point}(\widetilde{y})$, which is the representative point of its orbit: $\mathsf{point}(\widetilde{y}) = \widetilde{v_j}$ for some $j \in \{0, \dots, n-1\}$;
- a word $\gamma_{\widetilde{y}}$ on $\mathcal{A}_\Gamma$ (equivalently, $\gamma_{\widetilde{y}} \in \Gamma$), which specifies the precise position $\gamma_{\widetilde{y}}\,\mathsf{point}(\widetilde{y})$ of $\widetilde{y}$ in $\mathbb{H}^2$.

The graph $G_\Pi$ lifts in the universal covering space $\mathbb{H}^2$ to the (infinite) graph $\rho^{-1}G_\Pi = \Gamma\Pi$. In particular, the sequence of edges of $\Gamma\Pi$ incident to a given vertex $\widetilde{v} \in \rho^{-1}v$ is a sequence of lifts of the edges incident to $v$ in $G_\Pi$. Each of these lifts is the image by an element of $\Gamma$ of an edge of $\Pi$ (Figure 1). The following result is straightforward from the data structure. We still prove it for completeness.

**Lemma 2.** *Let $e$ be an edge of $G_\Pi$ and $v$ a vertex of $e$. The sequence of edges of $G_\Pi$ incident to $v$ can be found in time $O(g)$.*

*Proof.* Without loss of generality (this can always be achieved by renaming edges and vertices of $\Pi$) $\widetilde{v} = \mathsf{source}(\widetilde{e})$ (as in Figure 1 for $e = e_0$), or $\widetilde{v} = \mathsf{target}(\widetilde{e})$. Consider the first case. After $e$, the next edge incident to $v$ in counterclockwise order in $G_\Pi$ is given in $\Pi$ by $\widetilde{x} = \mathsf{prev}(\widetilde{e})$, whose vertex $\mathsf{target}(\widetilde{x})$ is $\widetilde{v}$. The next edge incident to $v$ in $G_\Pi$ is given by $\mathsf{prev}(\mathsf{pair}(\widetilde{x}))$, whose target vertex lies in the same orbit as $\widetilde{v}$ under $\Gamma$. And so on: a sequence of accesses to $\mathsf{pair}(\cdot)$ and $\mathsf{prev}(\cdot)$ allows us to find the edges of $\Gamma\Pi$ incident to $\widetilde{v}$ in counterclockwise order. The process perfoms a constant number of accesses for each edge incident to $\widetilde{v}$, and the number of such edges is linear in $g$ as recalled above. The case when $\widetilde{v} = \mathsf{target}(\widetilde{e})$ is similar: $\mathsf{next}(\cdot)$ is used instead of $\mathsf{prev}(\cdot)$. □

In addition, the precise positions in $\mathbb{H}^2$ of all vertices of $\Pi$ in the orbit $\rho^{-1}v$ can be computed along the process using the information $\mathsf{point}(\cdot)$ and $\mathsf{w}(\cdot)$ stored in the data structure, without changing the complexity.

Relations in the finitely presented group [6, Chapter 5.5] $\Gamma$ can be deduced by comparing the two sequences of edges —clockwise and counterclockwise— around each vertex.

# 4 Constructing the initial system of simple loops

The combinatorial part of Step 1 of the algorithm is quite common in the topology literature: it consists in computing a spanning tree $\mathcal{T}$ of $G_\Pi$, then the edges of $\mathcal{T}$ are contracted, so that each vertex of $\mathcal{T}$ is merged into the root, and each edge of $G_\Pi$ that is not an edge of $\mathcal{T}$ is transformed into a loop based at the root. This is illustrated in genus 2 by Figure 2(Top). In this example, $\mathcal{T}$ has three edges $e_1, e_5$, and $e_6$. If $v$ is chosen as the root, edge $e_0$ transforms into a loop based at $v$ when $e_1$ and $e_6$ are contracted.

However, topology is not enough in this work. We actually compute the geometry of each loop that is obtained from the contraction of $\mathcal{T}$ by precisely computing a lift.

The main result of this section is as follows:

**Proposition 3.** *Let $S$ be a closed orientable surface of genus $g$ and $\Pi$ a fundamental polygon of $S$ with $2m$ edges and side pairings as described in Section 3. A system of loops based at a common point on $S$, given by a circular list of geodesic segments in $\mathbb{H}^2$ and side pairings, can be constructed in time $O(g^3)$. The total length of this system of loops is $O(gL)$, where $L$ denotes the perimeter of $\Pi$.*

The construction algorithm proceeds in three phases:

*(i).* Compute a spanning tree $\mathcal{T}$ of $G_\Pi$. A root $b$ is chosen for $\mathcal{T}$, together with an edge $e_0$ incident to $b$ in $G_\Pi \setminus \mathcal{T}$ and lifts $\widetilde{b}$ and $\widetilde{e_0}$ in $\Pi$. Up to a renaming of representatives in orbits, we can assume that $\widetilde{b} = \mathsf{source}(\widetilde{e_0})$.

*(ii).* Construct a new fundamental domain $\Pi'$, as a polygon whose edges are paths consisting of $O(g)$ geodesic segments in $\Gamma\Pi$: in each such path, one segment is a lift of an edge of $G_\Pi \setminus \mathcal{T}$, and the other segments are lifts of edges of $\mathcal{T}$; the endpoints of each path lie in the orbit of $\widetilde{v}$. The side pairings in $\Pi'$ are also computed.

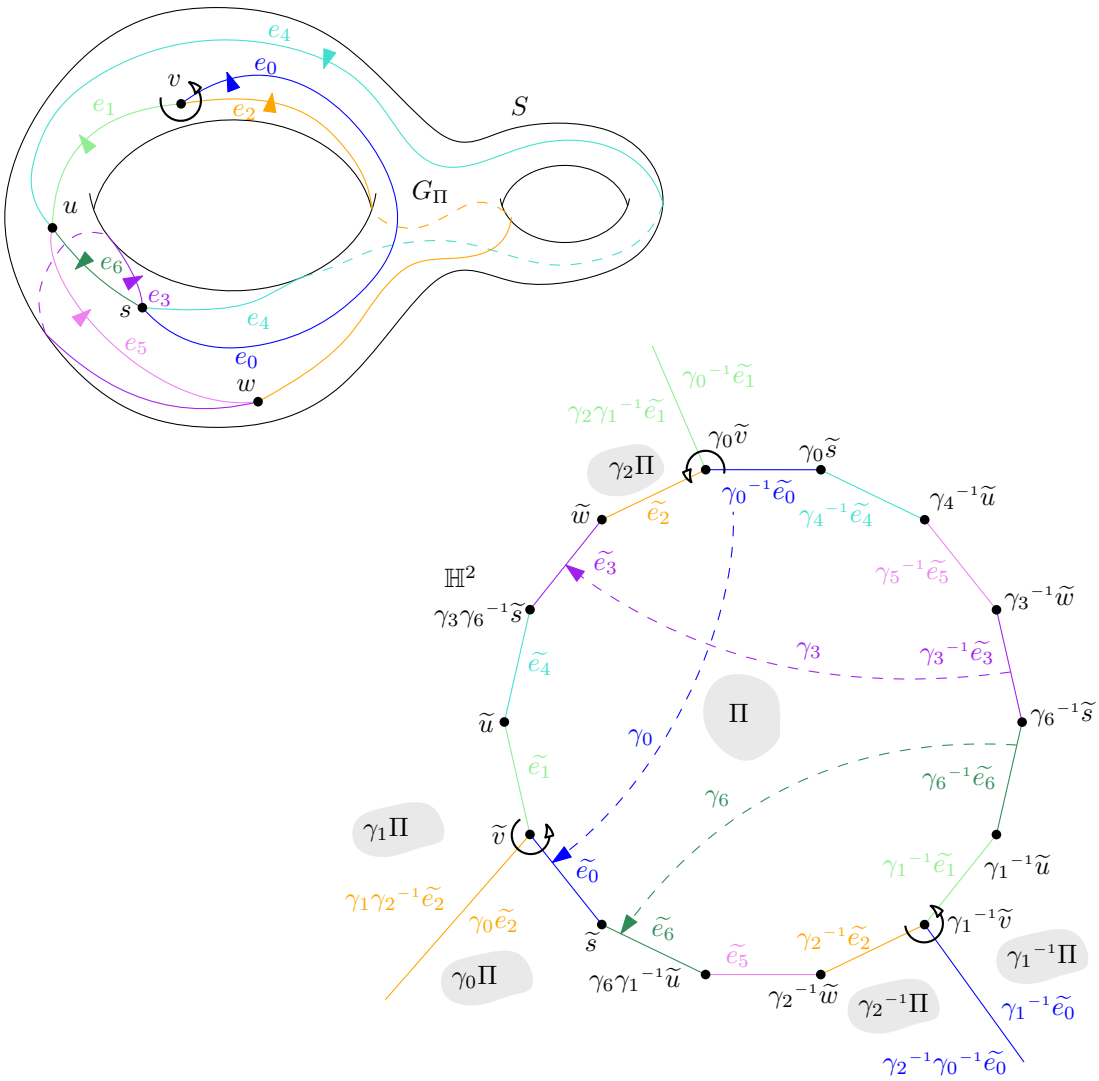


Figure 1: (Top) The graph $G_\Pi$. The arrow around vertex $v$ shows its incident edges. (Bottom) The fundamental polygon $\Pi$. Vertices $\widetilde{s}, \widetilde{u}, \widetilde{v}$, and $\widetilde{w}$ of $\Pi$ are chosen as representatives of the orbits of $s, u, v$, and $w$, respectively. The arrows show the combinatorics of the tiling $\Gamma\Pi$ at the three vertices of $\Pi$ in the orbit $\rho^{-1}v$: $\widetilde{v}, \gamma_0\widetilde{v} = \gamma_2{\gamma_1}^{-1}\widetilde{v}$, and ${\gamma_1}^{-1}\widetilde{v} = {\gamma_2}^{-1}\gamma_0$.

*(iii).* Replace each path computed in the previous stage by the geodesic segment between its vertices and keep the side pairings.

Note that $\Pi'$ is a fundamental domain, but not a fundamental polygon in the sense of Section 2.2: its edges are paths consisting of several geodesic segments; the geodesic segments between its vertices (i.e., endpoints of these paths) will intersect in general, so they do not bound a fundamental domain. We will call such a polygon a *topological polygon*. Section 5 will present the construction of a fundamental polygon from this topological polygon (Step 2 of the algorithm).

The rest of this section is devoted to proving Proposition 3, by detailing the construction.

*Proof.* As in Section 3, $n$ denotes the number of vertices of $G_\Pi$. Phase *(i)* is performed by a standard constuction of a minimum spanning tree $\mathcal{T}$ in $O(m \log n)$ or $O(m + n \log n)$, i.e., $O(g \log g)$. The tree has $n - 1$ edges.

Phase *(ii)* consists of walking along the edges of $\Gamma\Pi$. The walk constructs the new fundamental domain $\Pi'$ in counterclockwise order and stores it in a data structure that is very similar to the data structure defined in Section 3 for $\Pi$. However some of its elements have a different meaning, which will be detailed in the sequel; in particular, the elements $\mathsf{pair}(\cdot)$ are actually not yet side pairings in this phase, but temporary elements of $\Gamma$.

As a preprocessing step, for each edge $x$ of $G_\Pi \setminus \mathcal{T}$, we find the path $\mathsf{p}(x)$ on $S$ whose homotopy class contains the loop that will eventually replace $x$: it is given by the (unique and simple) path in $\mathcal{T}$ from the root to a first vertex of $x$, followed by $x$, and finally by the path in $\mathcal{T}$ from the second vertex of $x$ to the root. In the example of Figure 2(Top), $e_0$ is replaced by a loop based at $b = v$ that is homotopic to the sequence $\mathsf{p}(e_0) = e_0 \cdot e_6 \cdot e_1$, where $\cdot$ denotes concatenation of paths. The path for edge $e_4$ is $\mathsf{p}(e_4) = e_1 \cdot e_4 \cdot e_6 \cdot e_1^{-1}$; here, edge $e_1$ is traversed in both directions.

The walk starts at $\widetilde{b}$ and first considers edge $\widetilde{e_0}$ chosen in stage *(i)*. For each considered edge $\widetilde{x}$ not in $\rho^{-1}\mathcal{T}$, by the pre-processing we have just mentioned, we look for lifts of edges of $\mathsf{p}(x)$ in order in $\Gamma\Pi$. This is easily done by a sequence of operations $\mathsf{next}(\cdot), \mathsf{prev}(\cdot)$, and $\mathsf{pair}(\cdot)$ on edges of $\Pi$, and turning around their vertices $\mathsf{source}(\cdot)$ and $\mathsf{target}(\cdot)$ as in Lemma 2 until a lift of the next element of $\mathsf{p}(x)$ is found. On the way, the elements $\mathsf{w}(\cdot)$ of $\Gamma$ found in the data structure are collected so that the precise lift of each edge or vertex of $\Pi'$ is known.

Each time a lift of a path $\mathsf{p}(x)$, i.e., an edge of $\Pi'$, has been found, the algorithm proceeds to the next one. Note that edges (i.e., paths) appear on $\Pi'$ in the same order as the order in which the corresponding edges appear on $\Pi$: indeed, contracting the edges of $\mathcal{T}$ does not change the order in which edges on $S$ are traversed to describe the boundary of the face of $G_\Pi$.

This is illustrated in Figure 2(Bottom). The walk starts from $\widetilde{b}$ and follows $\widetilde{e_0}$. The next edge in $\Pi$ is $\mathsf{next}(\widetilde{e_0}) = \widetilde{e_6}$, which projects onto the next edge in $\mathsf{p}(e_0)$. Then we must look for a lift of $e_1$ incident to $\mathsf{target}(\widetilde{e_6})$. This is done by going to $\mathsf{pair}(\widetilde{e_6}) = {\gamma_6}^{-1}\widetilde{e_6}$ and turning around its source vertex. The first edge in counterclockwise order is ${\gamma_1}^{-1}\widetilde{e_1}$; the source vertex of its image $\gamma_6{\gamma_1}^{-1}\widetilde{e_1}$ is the target vertex of $\widetilde{e_6}$ and the walk traverses it. Its target vertex is $\gamma_6{\gamma_1}^{-1}\widetilde{b}$, which is in the orbit of $\widetilde{b}$. We have now found the lift of $\mathsf{p}(e_0)$ in $\Gamma\Pi$ that forms the first edge of $\Pi'$: it is the sequence $\widetilde{e_0} \cdot \widetilde{e_6} \cdot \gamma_6{\gamma_1}^{-1}\widetilde{e_1}$. From vertex $\gamma_6{\gamma_1}^{-1}\widetilde{b}$ we will now construct the edge of $\Pi'$ corresponding to $\widetilde{e_2}$, as $\widetilde{e_2}$ is the edge following

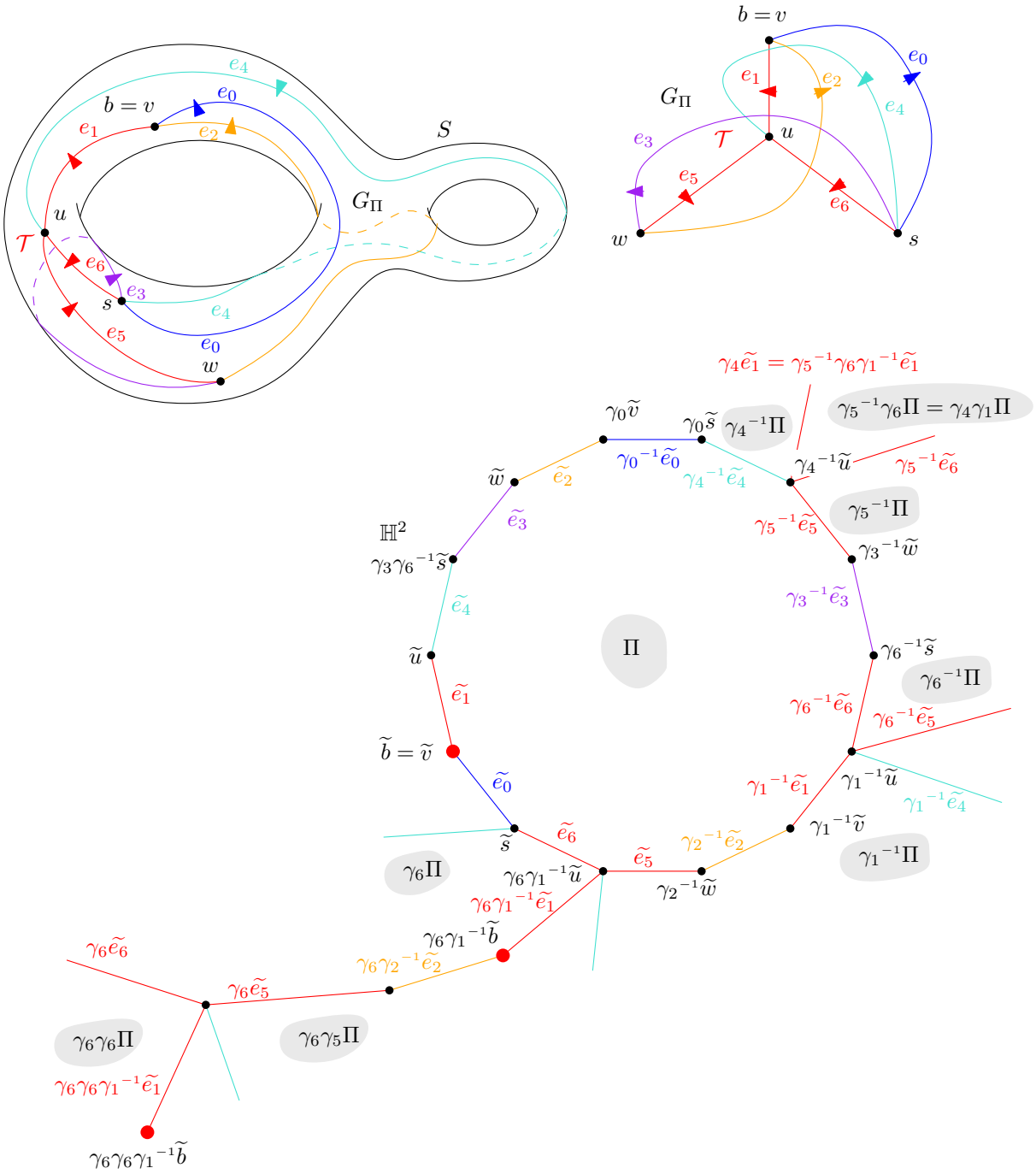


Figure 2: (Top) The spanning tree $\mathcal{T}$ of $G_\Pi$ has edges $e_1, e_5, e_6$ and is rooted at $v = b$. Each edge of $G_\Pi \setminus \mathcal{T}$ is replaced by a loop based at $b$ when contracting $\mathcal{T}$. (Bottom) The path in $\mathbb{H}^2$ corresponding to the loop replacing $e_0$ starts at $\widetilde{b}$ and ends at $\gamma_6\gamma_1{}^{-1}\widetilde{b}$.

$\gamma_0{}^{-1}\widetilde{e_0}$ in $\Pi \setminus \rho^{-1}\mathcal{T}$. We know that $\mathtt{p}(e_2) = e_2 \cdot e_5 \cdot e_1$. Here, turning around the source vertex of $\gamma_1{}^{-1}\widetilde{e_1}$ gives $\gamma_2{}^{-1}\widetilde{e_2}$, and the walk continues with $\gamma_6\gamma_2{}^{-1}\widetilde{e_2}$, then turning around the source of $\gamma_2{}^{-1}\widetilde{e_2}$ we find $\widetilde{e_5}$, and the walk follows $\gamma_6\widetilde{e_5}$. So far we have only followed edges of $\gamma_6\Pi$, as the edge that we were looking for when turning around vertices was always the first one. However, this is not the case after $\gamma_6\widetilde{e_5}$. The target of the representative $\gamma_5{}^{-1}\widetilde{e_5}$ is $\gamma_4{}^{-1}\widetilde{u}$, around which we must turn until we find a lift of $e_1$; the next edge of $\Gamma\Pi$ that we follow is thus $\gamma_6\gamma_5\gamma_5{}^{-1}\gamma_6\gamma_1{}^{-1}\widetilde{e_1} = \gamma_6\gamma_6\gamma_1{}^{-1}\widetilde{e_1}$, which finishes the edge of $\Pi'$ corresponding to $\widetilde{e_2}$. Next, we would continue with $\widetilde{e_3}$ in the same vein. And so on.

Note that, as we are constructing the fundamental domain $\Pi'$, following the order of the edges of $\Pi$, each edge $\widetilde{e'_i}$ of $\Pi'$ defines a topological loop $\beta'_i$ based at $b$ on $S$, which represents the homotopy class $[e'_i]$. Such an edge $\widetilde{e'_i}$ is formed by a sequence of edges of $\Gamma\Pi$ that corresponds to the path $\mathtt{p}(e_i)$, for $e_i \in G_\Pi \setminus \mathcal{T}$, and will naturally be paired with another sequence for the same $\mathtt{p}(e_i)$ (traversed in the opposite direction around $\Pi'$). The words associated with the edges in the two sequences differ by an element of $\Gamma$, which gives the side pairing $\gamma'_i \in \Gamma$ for $\Pi'$.

Phase *(iii)* is easy. It consists in replacing each edge of $\Pi'$ by the geodesic segment between its two vertices, and keeping the associated side pairings. As mentioned above, the corresponding geodesic loops may intersect on $S$, though the topological loops that we choose to represent their homotopy classes only intersect at their common basepoint.

As the edges of $\Pi'$ project by construction to loops, all based at the same point, there are $2g$ such loops on $S$ and $\Pi'$ has $4g$ edges, each consisting of $O(g)$ edges (and vertices) of $\Gamma\Pi$. By Lemma 2, $O(g)$ operations are performed at each vertex. This shows the complexity annonced in Proposition 3. The bound on the sum of the lengths of the geodesic loops also follows directly. □

Note that during the traversal detailed in the proof, we have computed for each vertex $\widetilde{x}$ of $\Pi'$ the element $\gamma \in \Gamma$ such that $\widetilde{x} = \gamma\widetilde{b}$. We store these elements of $\Gamma$ in a table $\mathtt{t}$, which will be used in the sequel, in addition to the data main data structure.

We denote the output of this step 2 as follows: we re-index the sides of the topological polygon $\Pi'$ (which has $4g \leq 2m$ edges) so that the side pairings are denoted as $\gamma'_0, \ldots, \gamma'_{2g-1}$ and the corresponding $2g$ topological loops on $S$ are $\beta'_0, \ldots, \beta'_{2g-1}$; these loops on $S$ do not intersect except at their common basepoint $b$, which is now renamed to $b'$ for global consistency of notation, as announced at the beginning of Section 3.

# 5 Finding an embedded system of loops

We want to find a collection of geodesic loops on a hyperbolic surface $S$, all based in a single point and disjoint otherwise, and such that the complementary region of the loops is a convex hyperbolic polygon. What we show is that in fact we can retain the choice of topological loops $\beta'_0, \ldots, \beta'_{2g-1}$ made in Section 4 by moving the basepoint appropriately to ensure that their geodesic realizations satisfy the desired properties.

Consider the set of topological loops $\beta'_0, \ldots, \beta'_{2g-1}$ all based at point $b'$ constructed in the previous section. We choose a pair that intersects minimally

exactly once which, up to reordering, we can suppose are $\beta_0'$ and $\beta_1'$. For future reference we set $L_0 := \max\{\ell(\beta_0'), \ell(\beta_1')\}$, where $\ell$ denotes the length.

**Remark 4.** *We can fix any loop to be $\beta_0'$ and find a loop $\beta_1'$ intersecting it exactly once. Indeed, the set $\beta_0', \ldots, \beta_{2g-1}'$ contains curves that pairwise intersect at most once, and are all non-separating and thus homologically non-trivial. As it generates homotopy, it also generates homology and in particular every curve must be intersected by at least one other curve. As they can intersect at most once, they intersect exactly once.*

We begin by taking the unique geodesic loops, based in $b'$, in the free homotopy classes of $\beta_0'$ and $\beta_1'$, and we replace the curves with these geodesic representatives (we keep the same notation for convenience). Now we further consider the unique simple closed geodesic representatives in the free homotopy class of $\beta_0'$ and $\beta_1'$, which we denote $\beta_0''$ and $\beta_1''$, respectively. By hypothesis, they intersect in a single point $b''$, which will be our new basepoint.

We now define a path between $b''$ and $b'$ as follows. We consider a single lift $\widetilde{\beta_0'}$ of $\beta_0'$. Its endpoints both correspond to distinct lifts of $b'$ which are related by a unique translation $g_0$ in $\Gamma$. The copies of $\widetilde{\beta_0'}$ by iterates of $g_0$ form a broken geodesic line with the same end points at infinity as the geodesic axis of $g_0$. This singular geodesic, which we denote $\hat{\beta}_0'$, separates $\mathbb{H}^2$ into two half-spaces, only one of which is convex. We now choose an endpoint of $\widetilde{\beta_0'}$ and consider a lift of $\beta_1''$ that lies in the convex half-space. This lift we denote by $\widetilde{\beta_1'}$ and, as before, we consider the corresponding translation $g_1$ in $\Gamma$ and its geodesic axis and its corresponding singular geodesic $\hat{\beta}_1'$,. Now, we obtain $\widetilde{b''}$ as the intersection of the axes of $g_0$ and $g_1$. We consider the unique geodesic path $\widetilde{c}$ between $\widetilde{b'}$ and $\widetilde{b''}$ and its projection $c$ on $S$. We first observe that we can control the length of this path $c$:

**Lemma 5.** $\ell(c) < 2L_0$.

*Proof.* We observe that the axis of $g_0$ must lie in an $R$ neighborhood of $\hat{\beta}_0'$ where $R < \ell(\beta_1')$. In particular, the axis of $g_1$ intersects $\widetilde{\beta_0'}$. Similarly, the axis of $g_0$ intersects $\widetilde{\beta_1'}$. Now the proof essentially follows from drawing a picture of the above situation in $\mathbb{H}^2$ (see Figure 3).

By following an arc of $\widetilde{\beta_0'}$ from $\widetilde{b'}$ and then a segment of length $\ell(\widetilde{\beta_1''})$ on the axis $g_1$, we obtain a path between $\widetilde{b'}$ to $\widetilde{b''}$. As such, we have

$$\ell(\widetilde{c}) < \ell(\widetilde{\beta_0''}) + \ell(\widetilde{\beta_1''})$$

and so by passing to the surface

$$\ell(c) < 2L_0.$$

□

Observe that for $i = 0, 1$, $\beta_i''$, based in $b''$, is freely homotopic to $c^{-1} \cdot \beta_i' \cdot c$ and that there is a homeomorphism of $S$, isotopic to the identity, that takes $b'$ to $b''$ and that sends (the homotopy class of) $\beta_i'$ to $\beta_i''$. This homeomorphism is often referred to as the point pushing map (see for instance [13, Section 4.2] for details).

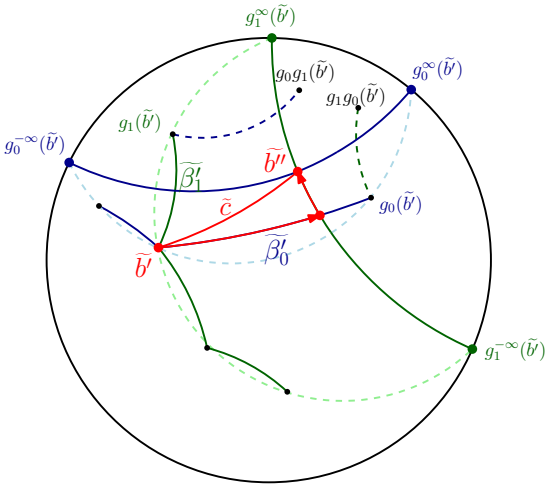


Figure 3: $c$ is homotopic on $S$ to the projection of the concatenation of the red arc of $\widetilde{\beta_0'}$ from $\widetilde{b'}$ and the red segment of the axis of $g_1$.

We can apply this same homeomorphism to the remaining curves. For $i = 0, \ldots, 2g-1$ we set the homotopy class of loop $\beta_i''$ to be:

$$[\beta_i'']_{b''} = [c^{-1} \cdot \beta_i' \cdot c]_{b''}. \tag{5.1}$$

As we have just moved the basepoint by a homeomorphism, the homotopy classes $[\beta_i'']_{b''}$ all have simple representatives and can be realized disjointly outside of $b''$. The following lemma implies that their unique geodesic representatives enjoy this same property. It is well known to specialists, but we include a proof sketch for completeness.

**Lemma 6.** *Let $\Sigma$ be a hyperbolic surface with piecewise-geodesic boundary such that the interior angles on the singular points $s_0, \ldots, s_{k-1}$ of the boundary are cone points of angle $\leq \pi$. If $[\alpha]_{p_i,q_i}, [\alpha']_{p_j,q_j}$ are simple homotopy classes of paths (with endpoints $p_i, p_j, q_i, q_j$ in the set $s_0, \ldots, s_{k-1}$), and disjoint except for possibly in their endpoints, then the unique geodesic representatives are also simple and disjoint.*

*Sketch of proof.* We consider $\widetilde{\Sigma}$, the universal cover of $\Sigma$, which we view as a (geodesically convex) subset of $\mathbb{H}^2$. We lift $\partial\Sigma$ to $\widetilde{\Sigma}$ and representatives of $[\alpha]_{p_i,q_i}$ and $[\alpha']_{p_j,q_j}$, which are simple and disjoint, to the universal cover. Observe that being simple and disjoint is equivalent to *all* individual lifts in $\mathbb{H}^2$ being simple and pairwise disjoint. Now take two individual lifts of either $\alpha$ or $\alpha'$, and their unique geodesic representatives. We will see that they are also disjoint. Note that in general, given two simple disjoint paths in the hyperbolic (or Euclidean) plane, the unique geodesics between their endpoints might intersect (as already mentioned in Section 2.1). However:

Observation: *Let $C \subset \mathbb{H}^2$ be a convex with non-empty boundary, and $p_0, q_0, p_1, q_1 \in \partial C$. Let $\alpha_1 : [0,1] \to C$ and $\alpha_2 : [0,1] \to C$ be simple paths, disjoint in their interior, with $\alpha_0(0) = p_0, \alpha_0(1) = q_0$ and $\alpha_1(0) = p_1, \alpha_1(1) = q_1$. Then the unique geodesic between $p_0$ and $q_0$ and the unique geodesic between $p_1$ and $q_1$ are disjoint in their interior as well.*

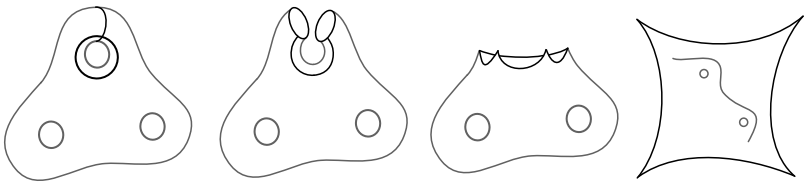

Figure 4: A visualization of the cutting along $\beta_0''$ and $\beta_1''$.

A key point is that, thanks to the angle condition on the cone points, $\widetilde{\Sigma}$ is a convex region of $\mathbb{H}^2$. (This is just a slightly more sophisticated observation than the elementary fact that a polygon with all interior angles less than $\pi$ is convex.) The observation now implies that the lifts of geodesics corresponding to $\alpha$ and $\alpha'$ are disjoint in their interior if and only if there are representatives of $[\alpha]_{p_i,q_i}$ and $[\alpha']_{p_j,q_j}$ that are, too, which, by hypothesis, is the case. □

We can now apply Lemma 6 to the geodesic representatives of $[\beta_i'']_{b''}$. For simplicity we denote by $\beta_i''$ the unique geodesic loop in the corresponding homotopy class.

**Theorem 7.** *Let $\beta_0'', \ldots, \beta_{2g-1}''$ be a set of topological loops based in $b''$ that cuts a surface $S$ into a disk. Assume that $\beta_0''$ and $\beta_1''$ are closed geodesics. Then, the geodesic loops homotopic to $\beta_0'', \ldots, \beta_{2g-1}''$ are simple and pairwise disjoint in their interiors. Furthermore, by cutting $S$ along those geodesics and lifting to $\mathbb{H}^2$, one obtains a convex hyperbolic polygon with $4g$ edges.*

*Proof.* As $\beta_0''$ and $\beta_1''$ are closed geodesics, they form 4 angles in $b''$, and the opposite ones are equal. These angles thus satisfy $2\theta + 2\theta' = 2\pi$ so in particular both $\theta$ and $\theta'$ are strictly less than $\pi$. Thus by cutting along $\beta_0''$ and $\beta_1''$, we obtain a genus $g-1$ surface with a boundary consisting of 4 geodesic segments, and with 4 cone point singularities of angles $< \pi$ (see Figure 4)).

We now proceed inductively for $i \geq 3$ and consider the unique geodesic path $\beta_i''$, which by virtue of Lemma 6, has disjoint interior from the previous geodesic segments. Furthermore, as each segment further splits the angles, the angles are all less than $\pi$.

The end result is a polygon with all interior angles less than $\pi$ which, by elementary hyperbolic geometry, is convex. □

**Proposition 8.** *Let $S$ be hyperbolic of genus $g$ and $\Pi'$ a topological fundamental polygon of $S$ with $4g$ edges and side pairings as described at the end of Section 4. A convex fundamental polygon $\Pi''$ with its side pairing and whose vertices project to a single vertex on $S$, can be constructed in $O(g)$ time. The perimeter of $\Pi''$ has length $O(gL')$, where $L'$ denotes the total geodesic length of the sides of $\Pi'$.*

*Proof.* We need to compute the output convex polygon $\Pi''$ i.e., $4g$ lifts of $b''$ and $2g$ side pairings $\gamma_0'', \cdots, \gamma_{2g-1}''$. As homotopy classes of $\beta_i''$ and $\beta_i'$ are conjugates for $i = 0, \ldots, 2g-1$ (Equation 5.1), the side pairing $\gamma_i''$ is equal to $\gamma_i''$ for each $i$.

The key point here is the computation of a lift of $b''$. The first step consists in finding the loops $\beta_0'$ and $\beta_1'$ satisfying $i(\beta_0', \beta_1') = 1$. As shown in Remark 4, we can choose any loop for $\beta_0'$. We also fix $\widetilde{b_0'}$ to be an endpoint of one of the two paired sides of $\Pi'$ that are lifts of $\beta_0'$. We compute the ordered sequence

of loops around $b'$ as in Lemma 2, in $O(g)$ operations; recall that each loop $\beta'_0, \ldots, \beta'_{2g-1}$ appears twice in the sequence (Section 2.2). We take as $\beta'_1$ one of the loops that alternate with $\beta'_0$ in the sequence, and choose for $\widetilde{\beta'_1}$ one of its two lifts that are incident to $\widetilde{b'_0}$.

The second step consists in finding the free geodesics in the homotopy classes of $\beta'_0$ and $\beta'_1$, respectively. In $\mathtt{t}$, we find the word $g_0$ on $\{\gamma'_0, \cdots, \gamma'_{2g-1}\}$ representing the translation that sends $\widetilde{b'_0}$ to the other endpoint of $\widetilde{\beta'_0}$ (see Figure 5). The sequences $g_0^n(\widetilde{b'_0})$ and $g_0^{-n}(\widetilde{b'_0})$ converge in $\mathbb{C}$ to two points on the unit circle: these points are the two (infinite in $\mathbb{H}^2$) fixed points of the translation $g_0$, i.e., the two solutions of equation $g_0(z) = z$ in $\mathbb{C}$. The axis of $g_0$, i.e., the geodesic between these two points, projects onto $S$ to the free geodesic in $[\beta'_0]$.

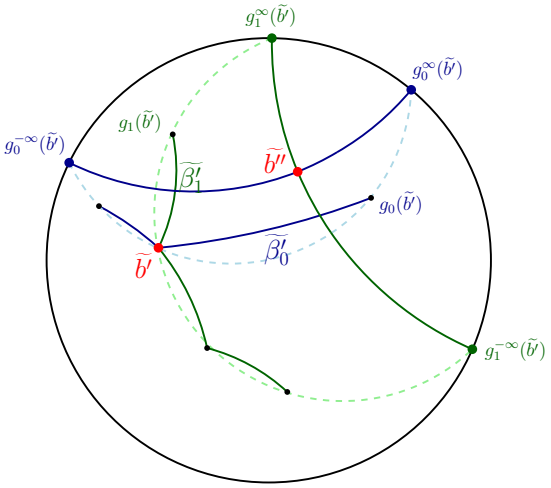


Figure 5: The computation of $b''$ from $b'$.

We repeat the same process with $\widetilde{\beta'_1}$ and find the geodesic in $\mathbb{H}^2$ that projects to the free geodesic in $[\beta'_1]$. The point $\widetilde{b''_0}$ comes as the intersection point of the two geodesics in $\mathbb{H}^2$. We now define $\widetilde{\beta''_0}$ as the geodesic segment between $\widetilde{b''_0}$ and $w_0(\widetilde{b''_0})$, and $\widetilde{\beta''_1}$ in the same way. This step is performed in constant time.

We can now compute the $4g$ lifts of $b''$ that are the vertices of $\Pi''$ by applying the elements of $\mathtt{t}$ to $\widetilde{b''_0}$. This last step has complexity $O(g)$. Additionally, we have

$$\ell(\beta''_i) \leq \ell(\beta'_i) + 2\ell(c) = 5 \cdot (\max_j \ell(\beta'_j))$$

for each $i = 1, \ldots, 2g-1$ (by Lemma 5) and thus the perimeter of $\Pi''$ is $O(g)$ times bigger than the perimeter $\Pi'$. □

# 6 Finding a Dirichlet domain from an embedded system of loops

We first summarize what we have obtained so far. We started with a polygon $\Pi$ of perimeter $L$ and we obtained a convex polygon $\Pi''$ of total length $O(g^2 L)$.

Additionally, all vertices of $\Pi''$ project on a single vertex $b''$ on $S$. This construction has complexity $O(g^3)$ by Propositions 3 and 8. At this point, it is easy to compute a Dirichlet domain. Indeed, we can now triangulate $\Pi''$ easily since it is convex and, thus, we obtain a geometric triangulation $T$, on to which the Delaunay flip algorithm can be applied [11]. The complexity of this algorithm depends on the diameter of $T$, for which the perimeter of $\Pi''$ is an upper bound.

The output of the flip algorithm is a Delaunay triangulation $DT$ of $S$ with the single vertex $b''$ computed in Section 5. To obtain a Dirichlet domain from $DT$, we just have to compute the triangles of $\widetilde{DT}$ incident to a lift $\widetilde{b''}$ of $b''$ and their dual: we compute the circumcenter of each triangle to obtain the vertices of the Dirichlet domain and we put a geodesic between vertices that correspond to adjacent triangles around $\widetilde{b''}$. This step is also clearly done in $O(g)$ operations. Putting all together we obtain the following theorem:

**Theorem 9.** *Let $S$ be a closed orientable hyperbolic surface of genus $g$ given by a fundamental polygon of perimeter $L$ and side pairings. A Dirichlet domain of $S$ can be computed in time $O(f(g^2L) + g^3)$ where $f(\Delta)$ is the complexity of the flip algorithm for a triangulation of diameter $\Delta$ with a single vertex.*

Using the best known bound for the flip algorithm so far, we obtain Theorem 1 stated in the introduction as a corollary. Note that the constant in the $O()$ depends on the metric on $S$. However, there are experimental and theoretical insights suggesting that the actual complexity of the flip algorithm may be much better [9].

# References


[1] N.L. Balazs and A. Voros. Chaos on the pseudosphere. *Physics Reports*, 143(3):109–240, 1986. doi:10.1016/0370-1573(86)90159-6.

[2] Alan F. Beardon. *The Geometry of Discrete Groups*. Springer-Verlag, 1983.

[3] Mikhail Bogdanov, Olivier Devillers, and Monique Teillaud. Hyperbolic Delaunay complexes and Voronoi diagrams made practical. *Journal of Computational Geometry*, 5:56–85, 2014. doi:10.20382/jocg.v5i1a4.

[4] Mikhail Bogdanov, Monique Teillaud, and Gert Vegter. Delaunay triangulations on orientable surfaces of low genus. In Sándor Fekete and Anna Lubiw, editors, *32nd International Symposium on Computational Geometry (SoCG 2016)*, volume 51 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 20:1–20:17, Dagstuhl, Germany, 2016. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik. doi:10.4230/LIPIcs.SoCG.2016.20.

[5] Peter Buser. *Geometry and spectra of compact Riemann surfaces*. Modern Birkhäuser classics. Birkhaäuser, Boston, Mass., 2nd edition, 2010.

[6] H. S. M. Coxeter and W. O. J. Moser. *Generators and Relations for Discrete Groups*. Springer-Verlag, Berlin, Heidelberg, New York, Tokyo, 1957.

[7] Jason DeBlois. The centered dual and the maximal injectivity radius of hyperbolic surfaces. *Geometry and Topology*, 19(2):953–1014, 2015. doi:10.2140/gt.2015.19.953.

[8] Jason DeBlois. The Delaunay tessellation in hyperbolic space. *Mathematical Proceedings of the Cambridge Philosophical Society*, 164(1):15–46, 2018. doi:10.1017/S0305004116000827.

[9] Vincent Despré, Loïc Dubois, Benedikt Kolbe, and Monique Teillaud. Experimental analysis of Delaunay flip algorithms on genus two hyperbolic surfaces. Preprint, INRIA, May 2021. URL: https://hal.inria.fr/hal-03462834/.

[10] Vincent Despré, Benedikt Kolbe, and Monique Teillaud. Representing infinite hyperbolic periodic Delaunay triangulations using finitely many Dirichlet domains. Preprint, INRIA, July 2021. URL: https://hal.archives-ouvertes.fr/hal-03045921.

[11] Vincent Despré, Jean-Marc Schlenker, and Monique Teillaud. Flipping geometric triangulations on hyperbolic surfaces. In Sergio Cabello and Danny Z. Chen, editors, *36th International Symposium on Computational Geometry (SoCG 2020)*, volume 164 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 35:1–35:16, Dagstuhl, Germany, 2020. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.SoCG.2020.35.

[12] Matthijs Ebbens, Hugo Parlier, and Gert Vegter. Minimal Delaunay triangulations of hyperbolic surfaces. In Kevin Buchin and Éric Colin de Verdière, editors, *37th International Symposium on Computational Geometry (SoCG 2021)*, volume 189 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 31:1–31:16, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.SoCG.2021.31.

[13] Benson Farb and Dan Margalit. *A Primer on Mapping Class Groups (PMS-49)*. Princeton University Press, 2012. URL: http://www.jstor.org/stable/j.ctt7rkjw.

[14] Iordan Iordanov and Monique Teillaud. Implementing Delaunay triangulations of the Bolza surface. In *33rd International Symposium on Computational Geometry (SoCG 2017)*, pages 44:1–44:15, Brisbane, Australia, July 2017. doi:10.4230/LIPIcs.SoCG.2017.44.

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

[16] Nikolai C Passler, Xiang Ni, Guangwei Hu, Joseph R Matson, Giulia Carini, Martin Wolf, Mathias Schubert, Andrea Alù, Joshua D Caldwell, Thomas G Folland, et al. Hyperbolic shear polaritons in low-symmetry crystals. *Nature*, 602(7898):595–600, 2022. doi:10.1038/s41586-021-04328-y.

[17] John Voight. Computing fundamental domains for Fuchsian groups. *Journal de Théorie des Nombres de Bordeaux*, 21(2):467–489, 2009. doi:10.5802/jtnb.683.