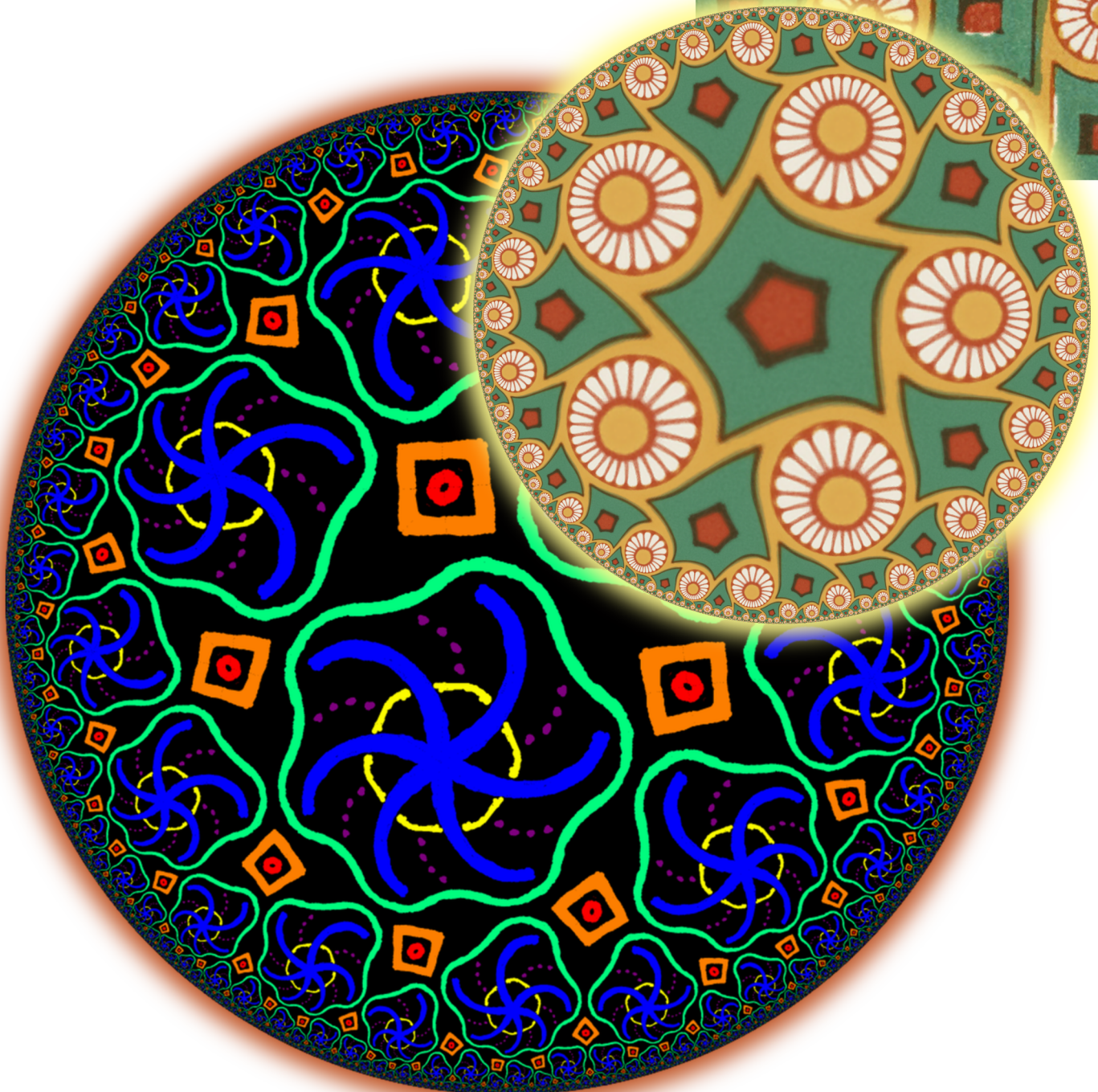


Martin von Gagern

# Creation of Hyperbolic Ornaments

Algorithmic  
and  
Interactive  
Methods







Fakultät für Mathematik

Lehrstuhl für Geometrie und Visualisierung (M10)

# Creation of Hyperbolic Ornaments

## Algorithmic and Interactive Methods

Martin Freiherr von Gagern

Vollständiger Abdruck der von der Fakultät für Mathematik der Technischen Universität München zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr. Christian Liedtke

Prüfer der Dissertation: 1. Univ.-Prof. Dr. Jürgen Richter-Gebert

2. Prof. Dr. Ileana Streinu  
(Smith College, Northampton / USA)

3. Prof. Kenneth Stephenson  
(University of Tennessee, Knoxville / USA)

Die Dissertation wurde am 12. Mai 2014 bei der Technischen Universität München eingereicht und durch die Fakultät für Mathematik am 10. Juli 2014 angenommen.



# Abstract

Hyperbolic ornaments are pictures which are invariant under a discrete symmetry group of isometric transformations of the hyperbolic plane. They are the hyperbolic analogue of Euclidean ornaments, including but not limited to those Euclidean ornaments which belong to one of the 17 wallpaper groups. The creation of hyperbolic ornaments has a number of applications. They include artistic goals, communication of mathematical structures and techniques, and experimental research in the hyperbolic plane. Manual creation of hyperbolic ornaments is an arduous task. This work describes two ways in which computers may help with this process. On the one hand, a computer may provide a real-time drawing tool, where any stroke entered by the user will be replicated according to the rules of some previously selected symmetry group. Finding a suitable user interface for the intuitive selection of the symmetry group is a particular challenge in this context. On the other hand, existing Euclidean ornaments can be transported to the hyperbolic plane by changing the orders of their centers of rotation. This requires a deformation of the fundamental domains of the ornament, and one particularly well suited approach uses conformal deformations for this step, approximated using discrete conformality concepts from discrete differential geometry. Both tools need a way to produce high quality renderings of the hyperbolic ornament, dealing with the fact that in general an infinite number of fundamental domains will be visible in the finite model of the hyperbolic plane. To deal with this problem, an approach similar to ray tracing can be used, variations of which are discussed as well.

# Zusammenfassung

Hyperbolische Ornamente sind Bilder, die invariant bleiben unter den Operationen einer diskreten Symmetriegruppe, bestehend aus längenerhaltenden Abbildungen der hyperbolischen Ebene. Sie stellen hyperbolische Analoga dar zu euklidischen Ornamenten; insbesondere (aber nicht ausschließlich) solche die einer der 17 kristallographischen Gruppen angehören. Die Erstellung von Ornamenten kann einer Vielzahl von Zwecken dienen. Dazu zählen künstlerische Ziele, die Vermittlung von mathematischen Strukturen und Techniken, sowie die experimentelle Erforschung der hyperbolischen Ebene. Die manuelle Erstellung von hyperbolischen Ornamenten ist jedoch sehr anstrengend. Diese Arbeit beschreibt zwei Verfahren, mit denen der Computer bei diesem Prozess behilflich sein kann. Er kann auf der einen Seite ein Werkzeug zum Zeichnen in Echtzeit zur Verfügung stellen, bei dem jeder vom Benutzer gezeichnete Strich gemäß den Regeln einer zuvor festgelegten Symmetriegruppe repliziert wird. Eine Herausforderung ist hierbei die Wahl einer geeigneten Benutzerschnittstelle, mit der diese Symmetriegruppe intuitiv ausgewählt werden kann. Auf der anderen Seite können auch bestehende euklidische Ornamente in hyperbolische Geometrie übertragen werden, wobei die Zähligkeiten ihrer Drehzentren angepasst werden müssen. Dies erfordert eine Deformation der Fundamentalzellen des Ornaments, und ein besonders geeignetes Verfahren ist hier die konforme Deformation. Diese lässt sich annähern durch diskret konforme Abbildungen, wie sie in der diskreten Differentialgeometrie formuliert und berechnet werden. Beide Verfahren müssen am Ende Darstellungen des hyperbolischen Ornaments in hoher Qualität erzeugen. Das wird insbesondere dadurch erschwert, dass im Allgemeinen eine unendlich große Zahl von Fundamentalzellen in einem endlichen Modell der hyperbolischen Ebene sichtbar ist. Um mit diesem Problem umzugehen kann ein Verfahren verwendet werden, das dem Raytracing ähnlich ist, und von dem verschiedene Varianten diskutiert werden.

# Preface

This work is the result of several years of research, going back at least to 2006, while I was still a diploma student. At times, it had been planned that this research would become my diploma thesis, but this research project grew beyond the scope of a diploma thesis. So I wrote that about something simpler, namely interactive drawing and automatic recognition of Euclidean ornaments. This was done in order to keep the larger project about hyperbolic ornaments available as subject for further research and my dissertation. For a long time I intended to first research the best approaches to several problems related to the creation of hyperbolic ornaments, then implement them all in the perfect ornament application, and then write my dissertation about these. But research kept turning up new ideas faster than I could integrate them. As it currently stands, I have a number of distinct proof-of-concept implementations, but no single integrated application. We eventually decided that I should write about all of these results before writing the one application to unite them all. So in this sense, the text you're about to read is a report on the current state of affairs, for a project which is still very much in progress. This text will also be my road map for the implementation, which is the next major project on my agenda.

PDF versions of this work will be published in several places. On the one hand, the Technische Universität München will provide an official electronic publication. On the other hand, I will also make copies available on my own web page. Most likely in different qualities, for reading online or for print. I intend to also maintain a list of errata. All of this will become accessible at the following address:

<http://martin.von-gagern.net/publications/2014-phd/>

For those reading the digital PDF version, I'd like to point out some features. As may be expected, there are hyperlinks within the document, so clicking on a cross reference will take you to the referenced object. A feature which is less common is my use of links for the references section. For most of the works I cite, clicking on the reference will take you to a suitable page describing the document in question. In the presence of sufficient subscriptions, most of these will give you instant access to the referenced papers. So even if you have a printed copy, and

prefer reading that, I very much suggest using the digital version when working with the references.

I would like to thank many people who helped me make this work possible. On the personal level, I'd like to thank my family. My parents and in particular my wife helped me a lot, taking good care of our two daughters while I worked late hours. I'd also like to thank my late grandmother who sponsored printing this work but did not live to see it completed.

Scientifically, I thank first and foremost Jürgen Richter-Gebert. It was his lectures, during my first semester as a student of computer sciences, which got me interested enough in mathematics that these days I am unsure whether I ought to call myself a computer scientist or a mathematician. He has been a constant source of support and inspiration since, motivating me to investigate interesting avenues. Both the idea to work on hyperbolic ornaments at all, as well as the idea to turn Euclidean ornaments into hyperbolic ones, were very much his ideas, and payed off since they opened up a whole forest of interesting mathematical questions and concepts as well as challenging tasks in computer sciences, from algorithm design up to utilization of modern GPU hardware. So thank you, Jürgen!

Various other people who had a notable influence on this work. Boris Springborn had published his 2008 paper on discrete conformality just in time for us to make use of it. He was very helpful in explaining some details and in providing the hyperbolic functional which was essential for my application. He was also the one who pointed out Troyanov's theorem, on which I'll base several proofs about hyperbolization. Christian Stussak held a talk in 2008 which made me aware of the possible applications modern 3D graphics equipment had for the generation of planar images.

Martin von Gagern  
May 2014



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation . . . . .	2
1.2	Basic concepts . . . . .	5
<b>2</b>	<b>Modeling the hyperbolic plane</b>	<b>13</b>
2.1	Models of the hyperbolic plane . . . . .	13
2.1.1	Poincaré disk . . . . .	14
2.1.2	Poincaré half-plane . . . . .	15
2.1.3	Klein-Beltrami . . . . .	16
2.1.4	Hemisphere . . . . .	17
2.1.5	Hyperboloid . . . . .	19
2.1.6	Pseudosphere . . . . .	21
2.1.7	Reasons to choose the Poincaré disk . . . . .	23
2.2	Hyperbolic isometric transformations . . . . .	23
2.2.1	Representing isometries . . . . .	23
2.2.2	The group of isometries . . . . .	32
2.2.3	A catalog of hyperbolic isometries . . . . .	33
2.2.4	Classification of isometries . . . . .	34
2.2.5	Modeling objects . . . . .	39
<b>3</b>	<b>Group navigation</b>	<b>41</b>
3.1	Triangle reflection groups . . . . .	41
3.1.1	Mapping words to transformations . . . . .	42
3.2	Combinatoric group calculations . . . . .	44
3.2.1	Triangle rewriting systems . . . . .	44
3.2.2	Orbifold rewriting systems . . . . .	47
3.2.3	Completion of rewriting systems . . . . .	48
3.2.4	Geometric interpretation . . . . .	53
3.3	Expressiveness . . . . .	59

<b>4</b>	<b>Drawing hyperbolic ornaments</b>	<b>63</b>
4.1	Reverse pixel lookup . . . . .	63
4.1.1	Problematic number of copies . . . . .	63
4.1.2	General solution . . . . .	63
4.1.3	Adjacent pixels . . . . .	65
4.1.4	Supersampling . . . . .	66
4.1.5	Preprocessing . . . . .	69
4.2	OpenGL GPU implementation . . . . .	70
4.2.1	Anatomy of a 2D shader transformation . . . . .	70
4.2.2	Adjacency revisited . . . . .	71
4.2.3	Preprocessing revisited . . . . .	72
4.3	Reified and virtual triangles . . . . .	72
4.3.1	Reified triangles . . . . .	72
4.3.2	Automatic enumeration . . . . .	73
4.3.3	Virtual triangles . . . . .	75
4.4	Grid lines in hyperbolic views . . . . .	76
<b>5</b>	<b>Hyperbolization of ornaments</b>	<b>81</b>
5.1	The big picture . . . . .	81
5.2	Pattern recognition . . . . .	83
5.3	Choice of group . . . . .	84
5.4	Conformal deformation . . . . .	87
5.4.1	Uniqueness . . . . .	88
5.4.2	High symmetry . . . . .	92
5.4.3	Low symmetry with rotations . . . . .	96
5.4.4	Absence of rotations . . . . .	101
5.5	Discrete conformal maps . . . . .	103
5.5.1	Equivalence of triangle meshes . . . . .	103
5.5.2	Circle packings . . . . .	106
<b>6</b>	<b>Outlook</b>	<b>109</b>
6.1	Future development . . . . .	109
6.2	Open questions . . . . .	110
6.3	Things to try . . . . .	111

# Chapter 1

## Introduction



Figure 1.1: An ornament, drawn interactively by the author.

## 1.1 Motivation

This work is about hyperbolic ornaments, which are images that arise when the ideas of symmetric ornaments are applied to hyperbolic geometry. In the context of this work, an ornament is a picture which is symmetric in the sense that some symmetry operation will map that picture onto itself. To set the stage for hyperbolic ornaments, some background information on both ornaments and hyperbolic geometry shall be provided, regarding their history and applications.

Ornaments as a form of art are both ancient and universal. Different cultures all over the world have used very different shapes as elements of their ornaments, and very different techniques to depict those shapes. Entire books have been written about the ornaments in different cultures[27], but this work will instead concentrate on the common mathematical structures behind all of these.

Many laypeople will at first associate the science of mathematics with numbers, with manipulating numbers and performing calculations. Those actually working in this field usually tend to describe their subject a bit differently. To many mathematicians, mathematics is primarily about finding and describing structure. One of the most accessible ways to explain this distinction to the man on the street is by using ornaments. There certainly is a lot of structure underlying such repetitive images, and mathematicians can describe it in terms of symmetry groups and related concepts. So there is a kind of structure which falls into the scope of mathematics but at first glance has little to do with the typical stereotype of computations performed on numbers. It turns out that there still is a connection, that on the one hand the concept of a mathematical *group* describes both the symmetry groups of ornaments and the additive or multiplicative groups of various number systems, while on the other hand the elements of the symmetry groups, the isometric transformations, can be described as matrices whose elements are numbers. Nevertheless, ornaments are a very useful tool in explaining that there is more to mathematics than doing computations on numbers. The visual and artistic approach helps transporting this message on intuitive levels.

The method of choice for precisely describing any mathematical structure is by stating a collection of defining axioms. Geometry is a very useful example for this, since Euclid's attempt at formalizing geometry is one of the first axiomatic approaches in the history of mathematics (in his work "Elements", c. 300 BC). Of special interest is the last of his five axioms, commonly called the axiom of parallels. In one of several equivalent forms, it states that given a line and a point not on that line, there exists exactly one other line through that given point which is parallel to the first line in the sense that the two lines do not intersect. For hundreds of years mathematicians have been looking for ways to avoid including this fact as an axiom, by trying to prove it from the other axioms instead.

It came as something of a surprise when scientists finally discovered that the

fifth axiom can never be proven from the other four axioms. They did so by demonstrating the existence of a geometry where the first four axioms held but the fifth did not. This was the birth of the so-called non-Euclidean geometries, and more precisely of hyperbolic geometry. It happened around 1830, although there was both prior work in preparation for this, as well as later work which offered simpler proofs.[35] By 1868, several models existed which demonstrated that hyperbolic geometry can be built on Euclidean axioms.[6, 47, 3] These models will be discussed in Section 2.1.

This work is about the connection between the two concepts introduced so far: between ornaments on the one hand and hyperbolic geometry on the other hand. This combination is of interest for several reasons. One reason is artistic in nature: by offering a new concept of what constitutes a symmetric ornament, of how it might look and what rules govern its appearance, artists gained freedom to express new ideas, find new looks and new techniques. For the mathematical educator, presenting hyperbolic ornaments offers an accessible way to explain some aspects of the underlying structure of the hyperbolic plane. People are used to ornaments in the Euclidean plane, and will therefore recognize the same structures in a hyperbolic ornament, thus gaining an intuitive grasp of the relation between different geometries. Last but not least, the fusion of ornaments and hyperbolic geometry is of use to mathematical research. It offers new areas of research, new kinds of objects among which structure might be found and described. On the other hand, it also offers ways to visualize structures which might have arisen from different fields of research, like for example finitely represented groups, and by visualizing them gain a deeper understanding in other fields as well.[36, 47]

However, the first steps towards hyperbolic ornaments entailed a lot of painful work. A leading figure in this area was the Dutch artist Escher. Inspired by an illustration of a hyperbolic tiling in a work by Coxeter from 1957[15], he produced a series of four prints of ornaments in hyperbolic geometry. These have been termed “Circle Limit I” through “IV”. Coxeter in turn wrote articles about these prints by Escher, explaining the mathematical details behind them.[16, 17] In a letter to his son Arthur, Escher wrote about the effort required to create these prints:

I worked terribly hard to finally finish that litho, and then with gritted teeth, spent another four days making beautiful prints of that extremely complex circle limit in colors. Each print is a series of twenty printings: five pieces, and each piece four times. All this with the remarkable feeling that this work is a milestone in my development, and that nobody, except myself, will ever realize this. [31, 20, Letter from 20 Mar. 1960]

The combination of skill and effort required for these ornaments severely limited their application. However, things changed drastically with the advent of computer-

generated graphics. With the aid of computers, it is possible these days to create images of hyperbolic ornaments with less artistic skill being required, and more importantly, far less work required from the person creating said images. This text will demonstrate not only one but two ways in which computers may aid the creation of hyperbolic images.

One way is an interactive drawing tool, where any pen stroke entered by the user will immediately be replicated according to the rules of a previously chosen symmetry group. The central ideas required to draw such ornaments will be described in Chapter 4, after Chapter 3 investigated feasible ways of defining suitable symmetry groups. This tool is very useful to communicate certain mathematical concepts. For example, a rough sketch like the one in Figure 1.2 on Page 10 will quickly illustrate a certain situation. It is this kind of interactive experimentation which makes the implementation not only an exercise in applied mathematics, but also a useful tool for basic research in mathematics.

The other way starts from a Euclidean ornament and transforms it in such a way that the result is a related ornament in hyperbolic geometry. Chapter 5 will precisely define this relation between ornaments, and will describe the process which does the required transformations. Rendering the result will again make use of ideas from Chapter 4. The key benefit in this tool is leveraging both the wealth of existing artwork by talented artists and the collection of tools which help in the creation of Euclidean ornaments. With this method, all those ornaments can be hyperbolized very easily.

Each of the next four chapters will be guided by a central question:

**Chapter 2:** What is the hyperbolic plane, how can it be depicted and how can one represent geometric objects and transformations?

This chapter is mainly intended as a kind of dictionary, to help people who are used to a different model to understand the model used in this work.

**Chapter 3:** How can hyperbolic symmetry groups be described in an easily accessible way?

This work introduces a novel idea of using triangle reflection groups as a basis from which users can enter subgroup generators using a graphical user interface.

**Chapter 4:** How can the hyperbolic plane be rendered on screen in a way which conveys the infinity of the depicted objects, in particular the infinite number of copies of the fundamental domain of an ornament?

The key concept here is a process we call reverse pixel lookup, which is very similar to the ray tracing approach used in the generation of photorealistic three-dimensional computer graphics.

**Chapter 5:** How can a Euclidean ornament be turned into a hyperbolic one?

A precise definition of this transformation is followed by various uniqueness and existence proofs, as well as a practical method to compute them using techniques from discrete differential geometry.

## 1.2 Basic concepts

This section will introduce some vocabulary used throughout the remainder of this work. Readers already familiar with these terms can usually rely on the definitions they know.

### Definition 1.1: Isometry

Let  $M$  be some Riemannian 2-manifold. Let  $d(p, q)$  denote the distance between two points  $p, q \in M$ . Then a transformation function  $g : M \rightarrow M$  which satisfies

$$\forall p, q \in M : d(p, q) = d(g(p), g(q))$$

is called an *isometry* of  $M$ . The set of all such isometries on  $M$  is denoted as  $\text{iso}(M)$ .

In this work, the manifold  $M$  will usually be the Euclidean plane  $\mathbb{E}^2$  or the hyperbolic plane  $\mathbb{H}$ . So readers who are not completely comfortable with differential geometry may safely think “plane” for every occurrence of  $M$  in subsequent definitions.

The parenthesized notation  $g(p)$  used above stresses the fact that this is an application of a function. But since deeply nested sets of parentheses make some formulas hard to read, this work will also often use the shorter multiplicative notation  $gp$  (or sometimes  $g \cdot p$ ). This is similar to the way a linear transformation applied to a vector can be expressed by a matrix-times-vector multiplication, but here it will be used even for those cases where the transformation can not be expressed as a matrix multiplication. See Page 30 for such transformations.

**Lemma 1.1: Inverse isometry**

Every isometry  $g \in \text{iso}(M)$  has an inverse  $g^{-1} \in \text{iso}(M)$ .

*Proof:* Suppose  $g \in \text{iso}(M)$  were non-invertible, i.e. not bijective. Then there must be some points  $p, q \in M$  with  $p \neq q$  but  $g(p) = g(q)$ . In this case,  $d(p, q) \neq 0$  but  $d(g(p), g(q)) = 0$ . This contradicts the assumption  $g \in \text{iso}(M)$ .

Also consider  $g^{-1} \notin \text{iso}(M)$ . This can not be the case because

$$d(g^{-1}(g(p)), g^{-1}(g(q)))d(p, q)d(g(p), g(q))$$

so the distance between  $g(p)$  and  $g(q)$  is maintained, hence  $g^{-1}$  must be an element of  $\text{iso}(M)$  as well.  $\square$

**Lemma 1.2: Group of isometries**

The set  $\text{iso}(M)$  of isometries forms a group.

*Proof:* The existence of inverse elements was shown in lemma 1.1. The closedness  $g_1, g_2 \in \text{iso}(M) \Rightarrow g_1 \circ g_2 \in \text{iso}(M)$  follows from the definition of  $\text{iso}(M)$  as the set of *all* isometries. Likewise does the existence of the identity transformation as the neutral element of the group. With the associativity of function concatenation, all group axioms are satisfied.  $\square$

Of particular interest in the context of this work are subgroups of this group.

**Definition 1.2: Symmetry group**

A group  $G \subseteq \text{iso}(M)$  is called a *symmetry group*.

One possible way to look at a symmetry group is interpreting its members as identification rules: if one point is mapped onto another under any element of a symmetry group, then they are elements of the same equivalence class. The equivalence classes formed in this way are called *orbits*.



**Definition 1.3: Orbit**

Let  $G \subseteq \text{iso}(M)$  be a symmetry group on  $M$ . For any point  $p \in M$ , the set

$$Gp := \{g(p) \mid g \in G\} \subseteq M$$

is called the *orbit* of  $p$  under (the operations of)  $G$ .

In the same spirit, let  $S \subseteq M$  denote an arbitrary set of points from the manifold. Then

$$GS = \{g(s) \mid g \in G, s \in S\} \subseteq M$$

is called the *orbit* of  $S$  under (the operations of)  $G$ .

It is quite possible that all points of  $M$  belong to a single orbit. Such scenarios are of little aesthetic interest. The most appropriate tool to preclude these and similar situations is requiring the group to be *discrete*.<sup>[43]</sup>

**Definition 1.4: Discrete symmetry group**

A symmetry group  $G \subset \text{iso}(M)$  is called *discrete* iff none of its point orbits is dense:

$$\exists r \in \mathbb{R} \forall p \in M \forall g \in G : d(p, g(p)) > r$$

Colloquially speaking, this signifies that the symmetry group will never map a point to a point arbitrarily close to the its own preimage. Around every point  $p$  of the plane there is a neighborhood of radius  $r$  which is devoid of other points from the same orbit  $Gp$ .\*

One important thing to note here is the fact that distances are measured using the intrinsic distance measure of the manifold, without regard how this manifold may be visualized. So there will be situations in this work where points from the same orbit *appear* infinitely close to one another, but this is due to the way the manifold is embedded into the Euclidean plane, and not intrinsic to the manifold itself. Details about such embeddings will be given in Section 2.1.

Talking about visualizations, we also need some concept of pictures or images. This work will use both terms where the meaning is clear. In situations where

---

\*Since isometries are *proper*, a discrete symmetry group will act *properly discontinuous* on the manifold, which is a concept studied in several other works<sup>[12]</sup>. Despite slight differences in definition, being discrete and being properly discontinuous is essentially the same thing for symmetry groups, so the latter term won't be used subsequently.

*image* might refer to the image of some object under some map, the term *picture* is preferred. In its purest form, a *picture* can be defined as follows:

**Definition 1.5: Picture**

Let  $M$  be some Riemannian 2-manifold, and let  $C$  be some arbitrary set. Then a function  $P : M \rightarrow C$  is called a *picture* on  $M$  with color space  $C$ .

The manifold  $M$  here will either again denote the hyperbolic or Euclidean plane, or it will denote a subset thereof. One common color space  $C$  would be the set of red, green and blue intensities, which can be written as the product of three intervals, i.e.  $C = [0, 1]^3$ .

It is worth noting that for most practical applications, pictures will be represented using raster images, which basically define a map from some integer grid, bounded by a rectangle, to the color space  $C$ . The grid points for which these are defined are usually called *pixels*, and between pixel positions, colors are not defined by the image, and in some cases will have to be interpolated in some fashion. In the light of these definitions, it is best to view these raster images as *approximations* of pictures in the above sense. This is similar to the way floating point numbers are often used as approximations for real numbers. Section 4.1 in particular will make use of the discrete nature of a raster image to perform efficient computations, but the result will still remain an approximation of some theoretical ideal.

**Definition 1.6: Ornament**

A picture  $P : M \rightarrow C$  together with a discrete symmetry group  $G \subset \text{iso}(M)$  is called an *ornament*  $(P, G)$  iff the symmetry group maps the picture onto itself, i.e.

$$\forall x \in M \forall g \in G : P(x) = P(g(x)) \quad (1.1)$$

For an ornament in the above sense, one has to explicitly state the symmetry group associated with it. This is a bit contrary to everyday use, where one would consider a picture by itself an ornament, with the symmetry group being “obvious”. This obvious choice would be the maximal group, consisting of all possible symmetries of the ornament. The following definition captures this in more formal terms.

**Definition 1.7: Natural symmetry group**

A symmetry group  $G \subset \text{iso}(M)$  is called the *natural symmetry group* of a picture  $P : M \rightarrow C$  iff  $(P, G)$  is an ornament and there is no supergroup  $G' \supsetneq G, G' \in \text{iso}(M)$  for which  $(P, G')$  forms an ornament.

In many cases, the qualification “natural” can be omitted: wherever the author speaks about the symmetry group of a *picture* (as opposed to the symmetry group of an *ornament*, which can be extracted from its tuple representation), it is in fact a reference to its natural symmetry group. The formulation already suggests that this natural symmetry group will be uniquely defined by the picture, which should be verified now.

**Lemma 1.3: Uniqueness of natural symmetry group**

For a given picture  $P$  there exists exactly one uniquely defined symmetry group  $G$  such that  $G$  is the natural symmetry group of  $P$ .

*Proof:* Suppose that  $G_1$  and  $G_2$  were two incomparable maximal groups satisfying Equation (1.1). Then the group generated by their union will satisfy the equation as well, which is a contradiction to the assumed maximality.  $\square$

Note that *every* picture will have a natural symmetry group, since the trivial symmetry group, consisting only of the identity transformation, will always satisfy Equation (1.1). It makes sense to include these “unsymmetric ornaments” in the scope of these general definitions, but they represent only a degenerate corner case of little aesthetic relevance, at least in the context of ornaments. The focus of this work is on ornaments with non-trivial and discrete symmetry groups. Pictures of “real” ornaments allow for several associated symmetry groups, including the trivial and the natural symmetry group. There are applications where it is useful to consider an ornament formed by a picture and only a subgroup of its natural symmetry group. This allows a derived work (e.g. a *hyperbolization*, as will be introduced in Section 5.1) to break some of the symmetries, and in doing so gain artistic freedom.

In the light of definition 1.6, an ornament can be seen as a coloring not of the points of a manifold but instead of the point orbits of some symmetry group of that manifold. The whole ornament can be described by giving its symmetry group and the color of at least one point from every orbit. This concept of taking one point from every orbit is the basis of the following definition.

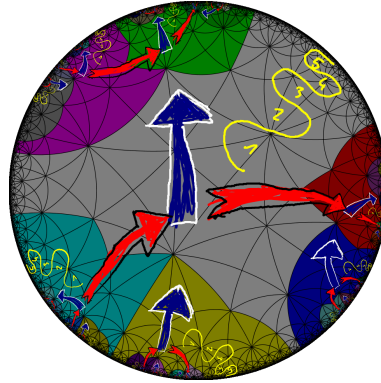


Figure 1.2: Two independent translations and still not cocompact

### Definition 1.8: Fundamental domain

A simply connected and closed sub-manifold  $F \subseteq M$  is called a *fundamental domain* of some symmetry group  $G \subset \text{iso}(M)$  iff

- (i)  $GF = M$
- (ii)  $\forall g \in G : gF \cap F \subseteq \partial F$

In other words, copies of  $F$  will cover the whole plane, and two copies will have no inner points in common. One important aspect used to describe a fundamental domain is its compactness.

### Definition 1.9: Compact

A manifold  $F$  is called *compact* iff every infinite sequence of points  $(p_i \in F)_{i=0}^{\infty}$  is dense around some point  $q \in F$ :

$$\forall p : \mathbb{N} \rightarrow F \exists q \in F \forall r \in \mathbb{R} \exists n \in \mathbb{N} : d(p_n, q) < r$$

There are two subtle aspects to note about this definition. One is the fact that distances are again measured using the intrinsic distance measure of the manifold. The other is the choice of  $q$  as a point of  $F$ . So if  $F$  is again a sub-manifold of  $M$ , then it is not sufficient for a sequence in  $F$  to have a limit point in  $M$ , but that limit point has to lie in  $F$  itself as well. As a consequence, open sets will never be compact.

As stated above, the compactness of fundamental domains is an important quality. Some symmetry groups will allow for a compact fundamental domain,

whereas for others every fundamental domain will be non-compact.

**Definition 1.10: Cocompact**

A symmetry group  $G \in \text{iso}(M)$  is called *cocompact* iff there exists a fundamental domain of  $G$  which is compact.

In the familiar setup of Euclidean ornaments, an ornament with a discrete symmetry group is cocompact iff it provides translational symmetries in at least two different directions. The 17 wallpaper groups satisfy this criterion, whereas frieze and rosette groups do not.[14] In hyperbolic geometry, the existence of two independent translations is not sufficient for cocompactness. This is illustrated in Figure 1.2. Since it is this cocompactness which is important for most practical distinctions, it might be good to consider groups with compact fundamental domains as the closest analogue to Euclidean wallpaper groups.

An alternate but equivalent definition generalizes compactness from fundamental domains to so called *orbifolds*, as they will be defined below, and then requires that orbifold to be compact.[12]

**Definition 1.11: Orbifold**

In the context of this work\*, the *orbifold*  $O = M/G$  associated with a symmetry group is the quotient space of the underlying manifold  $M$  under the actions of a discrete symmetry group  $G$ .

The most intuitive way to imagine an *orbifold* is by taking a single fundamental domain and gluing its boundary according to the actions of the symmetry group. For a Euclidean ornament this can be actually be done with paper, scissors and glue, at least for some symmetry groups. The orbifold will have a boundary iff the symmetry group contains axes of reflections. Corners on that boundary with interior angle  $\frac{\pi}{m}$  correspond to centers of  $m$ -fold rotation coinciding with these reflections. Centers of  $m$ -fold rotation which do not lie on an axis of reflection will correspond to so called *cone points*, around which the angle sum will only be  $\frac{2\pi}{m}$ .

---

\*This definition of an orbifold is somewhat less general than some established ones[48]. While an orbifold in the above sense will also be an orbifold according to the more general definition, the converse is not necessarily true: there can be orbifolds according to the general definition which can not be obtained as a quotient space. In general, an orbifold has to only *locally* look like such a quotient space. The more general definition is far more complicated, and not required for the tasks at hand, so it has been omitted.

**Definition 1.12: Singular Point**

A point on an orbifold is called singular iff it is either a cone point or a point on the boundary (including corner points). A point of an ornament is called singular iff the corresponding point on the orbifold of that ornament is singular.

Apart from these *singular points*, the metric of the orbifold equals that of the underlying manifold.

**Definition 1.13: Colored orbifold**

An orbifold  $O = M/G$  together with a function  $\tilde{P} : M \rightarrow C$  is called a *colored orbifold*  $(\tilde{P}, O)$  with color space  $C$ .

This generalizes the definition of a *picture* to the case of orbifolds instead of manifolds as the colored objects. There is a canonical correspondence between ornaments in the plane and colored orbifolds associated with their symmetry groups.

## Chapter 2

# Modeling the hyperbolic plane

### 2.1 Models of the hyperbolic plane

There are several possible ways how a portion of some Euclidean space may be used as a model of the hyperbolic plane. All of these models fulfill the same set of axioms, and express the same abstract hyperbolic plane. Therefore the choice of model makes no difference for purely hyperbolic theorems. It does make a difference when visualizing hyperbolic geometry, though.

Each model has its distinct advantages and disadvantages. Some of those will be listed, along with the defining ingredients of each model. These defining ingredients include the following:

**What is a hyperbolic point?** As stated above, only a portion of the enclosing Euclidean space is used to model the hyperbolic plane. Therefore some Euclidean points will correspond to hyperbolic ones, whereas others will not.

**What is a hyperbolic line?** A hyperbolic line is a straight line, i.e. a line of zero curvature with respect to the hyperbolic metric, in the hyperbolic plane. Although the term “geodesics” is well suited to describe this concept of locally shortest connecting curves on a curved manifold, this work will instead use the term “hyperbolic line”, as it stresses the correspondence between Euclidean geometry with its (Euclidean straight) lines and hyperbolic geometry with its hyperbolic lines.

**How are distances measured?** Given two hyperbolic points, which in formulas can be represented as vectors of Cartesian coordinates in the Euclidean space, a formula or procedure to measure their distance with respect to the hyperbolic metric will be stated.

**How are angles measured?** Like for distances, a formula or procedure will describe how hyperbolic angles can be measured in each model.

This work will almost exclusively use the first of the models described below, the Poincaré disk model. The other models are described to give an overview, and to help translating between the various models. Many publications present one of these models as a foundation, while the others are either derived from it or not mentioned at all. As a consequence, some readers of this work here might have a strong feeling about how they visualize the hyperbolic plane, which differs from that of the author. This section here will offer a dictionary to translate mental images between models, in the hope that doing so will help making subsequent sections intuitively understandable to a broader audience of readers. Those readers who feel sufficiently at home in the Poincaré disk model may skip this Section 2.1 if they wish.

### 2.1.1 Poincaré disk

The Poincaré disk model uses the inside of any circular disk as a model of the hyperbolic plane. The most obvious choice for a disk is the unit disk, which will also be the focus of the following description.

**Hyperbolic points** are points inside the unit disk.

**Hyperbolic lines** are circle arcs perpendicular to the unit circle. Hyperbolic lines passing through the origin degenerate to diameters, which can be thought of as arcs of circles with infinite radius.

**Distances** between hyperbolic points  $a$  and  $b$  can be measured based on the Euclidean norm as

$$d(a, b) = \operatorname{arcosh} \left( 1 + \frac{2 \|a - b\|^2}{(1 - \|a\|^2)(1 - \|b\|^2)} \right) \quad (2.1)$$

If the model were not dealing with the *unit* disk, this formula would require a reference to the radius. In differential geometry, the metric is usually defined not in terms of finite distances between distinct points, but instead as infinitesimal line element. Since such a differential approach won't be needed in this work, no such definition is given for this or any of the other models.

**Angles** are measured as the Euclidean angle between the tangents at the point of intersection.



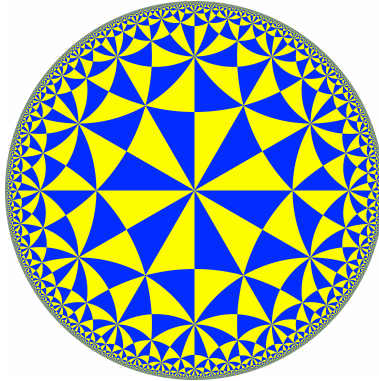


Figure 2.1: A pattern in the Poincaré disk model

**Advantages:** The model is conformal, i.e. it preserves angles. This means that hyperbolic angles between curves equal Euclidean angles at the point of intersection, as described above. This is a very valuable property for aesthetic reasons.

**Disadvantages:** As a hyperbolic line is modeled by a Euclidean circle arc, straight lines appear curved. This can

At times, it makes sense to not only consider the interior of the circle, but also its exterior, as a kind of mirror image of the interior. In that view, every hyperbolic point would be modeled by a *pair* of points, which in turn are related to one another by inversion in the unit circle. The center of the unit circle corresponds to the point at infinity. The main benefit of this approach is the fact that orientation-reversing transformations are sometimes easier to describe if they exchange the two copies of the hyperbolic plane.

### 2.1.2 Poincaré half-plane

The half-plane model can be obtained from the disk model using a Möbius transformation. Another way to visualize the translation between these models is this: if one were to zoom in on the rim of the disk model, particularly towards the bottom-most point of it, then the obtained image would become an increasingly close approximation of the half-plane model.

**Hyperbolic points** are points in the upper half-plane  $\{(x, y)^T \in \mathbb{R}^2 \mid y > 0\}$ .

**Hyperbolic lines** are semicircles and half-lines perpendicular to the horizontal axis.

**Distances** between hyperbolic points  $a$  and  $b$  can be measured using the formula

$$d\left(\begin{pmatrix} a_x \\ a_y \end{pmatrix}, \begin{pmatrix} b_x \\ b_y \end{pmatrix}\right) = \operatorname{arcosh}\left(1 + \frac{(a_x - b_x)^2 + (a_y - b_y)^2}{2a_y b_y}\right) \quad (2.2)$$

**Angles** are measured as the Euclidean angle between the tangents at the point of intersection.

**Advantages:** Isometries can be expressed through the projective general linear group  $\operatorname{PGL}(2, \mathbb{R})$  acting on  $\mathbb{C}$ . There have been extensive studies in that field, so existing results can be used and interpreted geometrically.

**Disadvantages:** less symmetric than the disk; doesn't fit the whole hyperbolic plane into a bounded Euclidean area.

Like in the case of the Poincaré disk, one can also view the whole plane (with the exception of the horizontal axis itself) as two copies of the hyperbolic plane. A simple reflection in the horizontal axis relates Euclidean points which model the same hyperbolic point. When isometries are modeled using  $\operatorname{PGL}(2, \mathbb{R})$  as stated above, then orientation-reversing transformations will exchange the two copies.

### 2.1.3 Klein-Beltrami

The model used by Beltrami and Klein can be inscribed in any (real and non-degenerate) conic section, although it most easily is modeled inside the unit disk as well. In contrast to the Poincaré disk model, this model uses straight line segments to model hyperbolic lines.

**Hyperbolic points** are points inside the fundamental conic (e.g. unit disk).

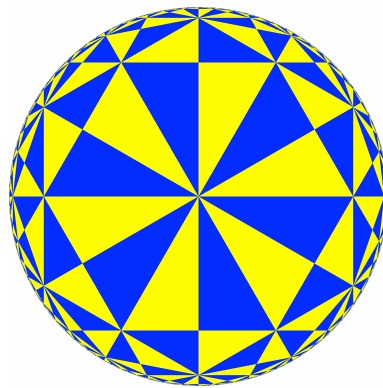


Figure 2.2: A pattern in the Klein-Beltrami model

**Hyperbolic lines** are the intersections of lines with the interior of the fundamental conic.

**Distances** are measured using the cross-ratio. If the line connecting  $a$  and  $b$  intersects the fundamental conic in points  $p$  and  $q$ , then the hyperbolic distance between  $a$  and  $b$  is

$$d(a, b) = \frac{1}{2} |\log(a, b; p, q)| = \frac{1}{2} \left| \log \frac{\|a - p\| \cdot \|b - q\|}{\|a - q\| \cdot \|b - p\|} \right|$$

**Angles** measured in the fashion of a Cayley-Klein geometry involve cross-ratios of four lines, two of which are imaginary tangents to the fundamental conic.

**Advantages:** Lines remain straight, and the whole model can be easily embedded into a framework of real projective geometry.

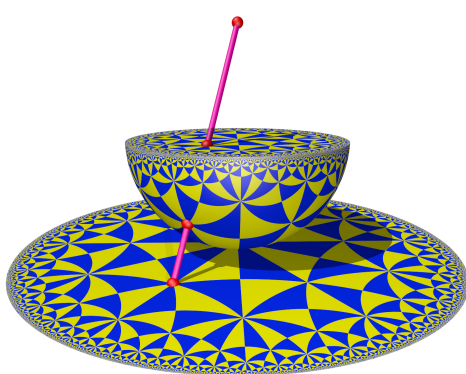
**Disadvantages:** The angle measurement becomes quite involved and differs greatly from the Euclidean angle between the modeling line segments.

### 2.1.4 Hemisphere

The hemisphere is not often used as a model of the hyperbolic plane in its own respect. It does however prove very useful in linking various other models using different projections, as depicted in Figure 2.3.

**Hyperbolic points** are points on the southern hemisphere.

**Hyperbolic lines** are the half-circles resulting from intersecting the southern hemisphere with planes perpendicular to the equator.



(a) Projection to the Poincaré disk



(b) Projection to the Klein-Beltrami model

Figure 2.3: A pattern in the hemisphere model

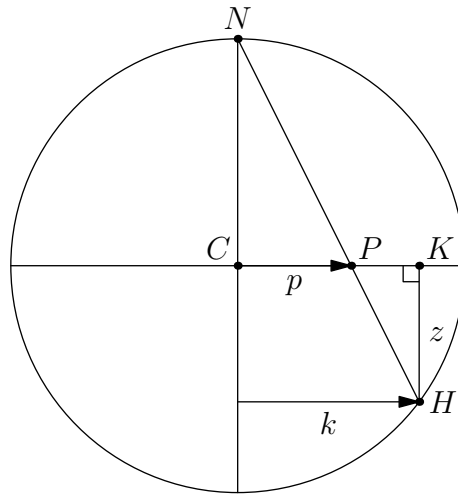


Figure 2.4: Converting between Klein-Beltrami and Poincaré disk model

**Advantages:** useful to link various other models.

**Disadvantages:** requires three dimensions.

One can obtain the Klein-Beltrami model from the hemisphere model via orthogonal projection onto a plane parallel to the equator of the sphere. This can be thought of as a projection from a point infinitely high above the north pole of the sphere. If the projection center is at the north pole itself, then the resulting picture is that of the Poincaré disk, and if the center of projection is on the equator, one obtains the Poincaré half-plane. So there is a way to continuously morph between the Klein-Beltrami model and either of the Poincaré models by moving that point.

One can take the upper hemisphere into account as well. It will be a mirror image of the lower hemisphere. This upper hemisphere corresponds to the exterior of the unit circle in the Poincaré disk model, or the lower half-plane in the half-plane model, if these models make use of the mirror-images discussed above. For the Klein-Beltrami model, however, both points on the full sphere correspond to the same point in the plane. Conversely, points outside the unit circle in the Klein-Beltrami model will have no correspondence on the (real) sphere. There is no mirror image outside the Klein-Beltrami model. In the broader setup of Cayley-Klein geometries, points outside the fundamental conic can be handled as well, but resulting distance or angle measures may become complex. This subject is beyond the scope of this work.

Observing the way how the hemisphere model links the Poincaré disk and the Klein-Beltrami models, one can obtain formulas to convert between these. Consider  $p \in \mathbb{R}^2$  with  $\|p\| < 1$  to be a point in the Poincaré disk, and likewise  $k \in \mathbb{R}^2$  with  $\|k\| < 1$  a point in the Beltrami-Klein model with the unit circle as its fundamental

conic. Both disks are embedded in the equatorial plane of the hemisphere. Then there is a point on the lower hemisphere linking these two. It has coordinates  $(x_k, y_k, z)^T$  where  $z = \sqrt{1 - \|k\|^2}$  is the lower projection of that point onto the sphere. So the  $z$  axis shall be oriented downward. The vertical component of the distance to the north pole is  $1 + z$ . To perform stereographic projection from that north pole onto the equatorial plane, one divides by that distance and obtains

$$p = \frac{1}{1+z}k = \frac{1}{1 + \sqrt{1 - \|k\|^2}}k \quad (2.3)$$

For the converse, the stereographic projection can be characterized by the following condition:

$$\begin{aligned} \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} + \lambda \begin{pmatrix} p_x \\ p_y \\ 1 \end{pmatrix} &\in S^2 = \{v \in \mathbb{R}^3 \mid \|v\| = 1\} \\ \lambda^2 p_x^2 + \lambda^2 p_y^2 + (\lambda - 1)^2 &= 1^2 \\ \lambda^2(p_x^2 + p_y^2 + 1) - 2\lambda &= 0 \\ \lambda_1 = 0 \quad \lambda_2 &= \frac{2}{1 + p_x^2 + p_y^2} \end{aligned}$$

$\lambda_1$  describes the north pole itself, whereas  $\lambda_2$  corresponds to the point on the lower hemisphere. Discarding the third coordinate, one obtains the Cayley-Klein position

$$k = \frac{2}{1 + \|p\|^2}p \quad (2.4)$$

### 2.1.5 Hyperboloid

The hyperboloid model is a three-dimensional model of the hyperbolic plane. If expressed in a Minkowski space with its special bilinear form, many formulas look much like they do for a normal unit sphere in Euclidean space. Its radius, however, is imaginary. Table 2.1 compares formulas for sphere and hyperboloid, exhibiting both the similarities as well as the differences resulting from the different bilinear form and the imaginary radius.[38]

To measure angles, we once again take tangents at the point of intersection. This time the tangent vectors  $v$  and  $w$  don't lie on the surface used to model the hyperbolic plane, but are only tangents to that surface at the point of intersection.

Note that the hyperboloid has two symmetric sheets, one with  $x_0 \geq 1$  and one with  $x_0 \leq -1$ . One way to do hyperbolic geometry on the hyperboloid is to identify

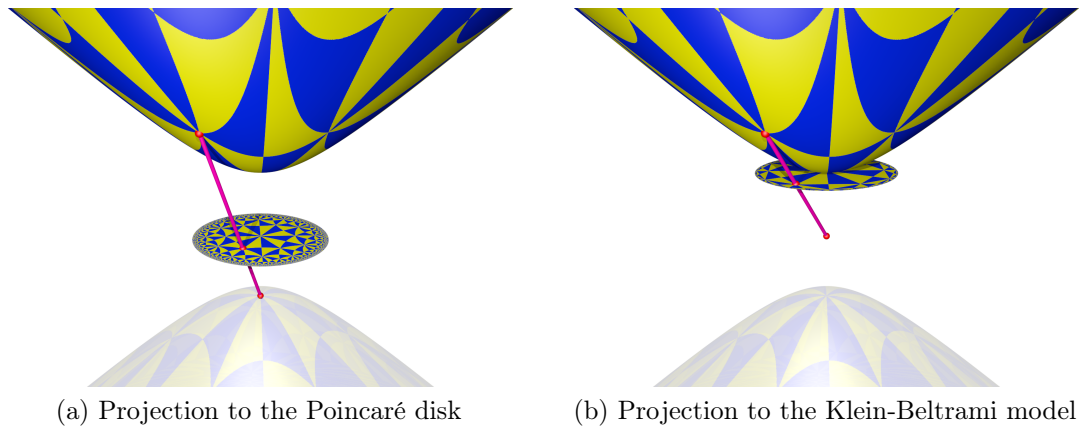


Figure 2.5: A pattern in the hyperboloid model

	Common	Sphere	Hyperboloid
Bilinear form	$p(x, y)$	$x_0y_0 + x_1y_1 + x_2y_2$	$-x_0y_0 + x_1y_1 + x_2y_2$
Quadratic form	$q(x) = p(x, x)$	$x_0^2 + x_1^2 + x_2^2$	$-x_0^2 + x_1^2 + x_2^2$
Radius	$r$	1	$i$
Surface	$\{x \mid q(x) = r^2\}$	$\{x \mid q(x) = 1\}$	$\{x \mid q(x) = -1\}$
Curvature	$\frac{1}{r^2}$	1	-1
Distance	$\ x - y\ $	$\arccos(p(x, y))$	$\operatorname{arccosh}(-p(x, y))$
Angle	$\angle vw$	$\arccos \frac{p(v, w)}{\sqrt{q(v) \cdot q(w)}}$	$\arccos \frac{p(v, w)}{\sqrt{q(v) \cdot q(w)}}$

Table 2.1: Comparing sphere and hyperboloid

antipodal points on these two sheets, just as one identifies antipodal points on the sphere when using it as a model for elliptic geometry. The more common approach, however, is restricting the model to a single sheet.

There is a strong connection between common spherical (or radial) trigonometric functions like  $\cos$  and their hyperbolic counterparts like  $\cosh$ . This connection can be seen most easily when considering areas instead of angles. Consider two points in a unit circle which are seen from the center of the circle under a given angle. That angle, expressed in radians, equals the length of the arc between the points. But it also equals the area of the circle segment they bound. In the same way, the hyperbolic sine and cosine will give coordinates of the hyperbola corresponding to an *area* given as argument to these functions.

**Hyperbolic points** are points on the  $x_0 > 0$  sheet of the hyperboloid  $-x_0^2 + y_0^2 +$

$$z_0^2 = -1.$$

**Hyperbolic lines** are intersections of the hyperboloid sheet with planes through the origin.

**Distances** are measured as  $\operatorname{arcosh}(-p(x, y))$

**Angles** are measured as  $\operatorname{arccos} \frac{p(v, w)}{\sqrt{q(v) \cdot q(w)}}$

**Advantages:** very similar to the sphere.

**Disadvantages:** difficult to visualize in two dimensions, measurements not intuitive.

### 2.1.6 Pseudosphere

The term *pseudosphere* is often used in connection with hyperbolic geometry. It is somewhat ambiguous, though, as different people associate different meanings with it. Beltrami at first used the term “pseudospherical surface” to abstractly describe a surface of constant negative curvature, without restriction to any specific model at all.[7, 47] Some other people use the term *pseudosphere* to refer to the hyperboloid model described above, e.g. [26].

The third possible meaning of the term *pseudosphere* is to describe a surface also known as a *tractricoid*. This is a surface of constant negative curvature that can actually be embedded in three-dimensional Euclidean space. It is obtained by revolving a tractrix about its asymptote. It was again Beltrami who associated this surface with his idea of a pseudosphere.[8]

As the surface is singular around its equator and has poles in the direction of the axis of revolution, it is not a model of the complete hyperbolic plane. Any

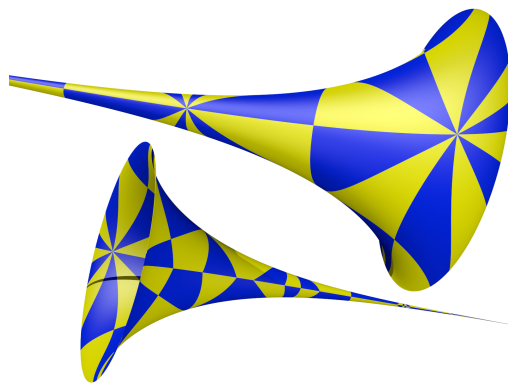


Figure 2.6: A partial pattern on the tractricoid

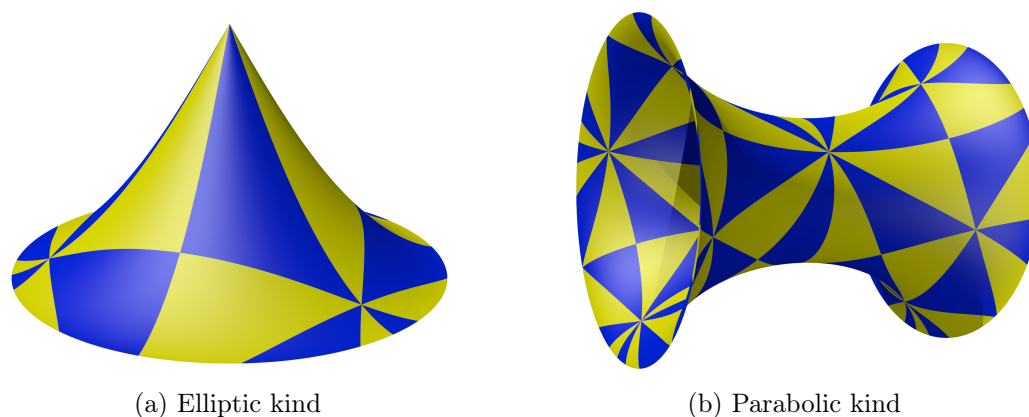


Figure 2.7: Specimens of the two other (non-tractricoid) surfaces of revolution with constant negative curvature[9]

construction that crosses the equator or circles the axis of revolution is not truly planar in a hyperbolic sense. As a consequence, hyperbolic ornaments won't line up, as can be seen in the lower part of Figure 2.6.

**Hyperbolic points** are points on the tractricoid.

**Hyperbolic lines** are geodesics on the tractricoid.

**Distances** are measured as path lengths of geodesics.

**Angles** are angles between the tangents to the geodesics at the point of intersection.

**Advantages:** actually visible surface with intuitive distance and angle measurements.

**Disadvantages:** models hyperbolic geometry only locally; not a model of the whole hyperbolic plane due to singularities.

Note that the tractricoid is not the only surface of constant negative curvature which can be embedded into three dimensions. There are more, and even when restricting oneself to surfaces of rotation, two more families of constant negative curvature surfaces exist, which are depicted in Figure 2.7.[29, 37] The tractricoid is special in that it allows for rays, i.e. segments may be extended infinitely in at least one direction without encountering any singularity. In this sense, the tractricoid is the “most infinite” of all the surfaces of rotation, and those surfaces of rotation are probably the most symmetrical among all the surfaces of constant negative curvature.



There can be no isometric embedding of the whole hyperbolic plane into three-dimensional space.[24] And while surfaces of rotation are among the easiest to describe, they represent only a tiny fraction out of a zoo of surfaces with constant negative curvature and singularities.[34]

### 2.1.7 Reasons to choose the Poincaré disk

As stated at the beginning of this section, the subsequent chapters as well as the described implementations will almost exclusively use the Poincaré disk model. When rendering ornaments in hyperbolic geometry, this seems to be the right choice, considering the following arguments.

- All 3D models are infeasible to represent on a flat medium.
- The Klein-Beltrami model tends to squeeze the image content to the rim of the disk. Furthermore, the fact that its angle measurement does not agree with the Euclidean one makes ornaments appear more irregular than necessary.
- The Poincaré half-plane has less aesthetic value. For example, on the disk a hyperbolic rotation can be modeled using a Euclidean rotation as long as the center of the rotation coincides with the center of the disk. This allows for an additional Euclidean rotational symmetry in addition to the hyperbolic ones. This coincidence of rotations makes hyperbolic rotations in general a bit easier to understand in the disk model, as any rotation can be composed using translations and one around the origin.

## 2.2 Hyperbolic isometric transformations

### 2.2.1 Representing isometries

As stated in the previous section, this work uses the Poincaré disk model of the hyperbolic plane. One therefore wants to describe the interior of the unit circle in such a way that dealing with circles and lines (both perpendicular to the unit circle) comes easy. A natural setup for this is  $\mathbb{C}^1$ , the “complex line”\*. In that environment, the Möbius transformations form a class of transformations which

---

\*The term “line” and the exponent 1 in  $\mathbb{C}^1$  both refer to the fact that we are dealing with a *single* complex number. Nevertheless, even a single complex number is usually visualized by embedding into the real plane  $\mathbb{R}^2$ , turning real and imaginary part of the number into separate real coordinates. This visualization is often called the “complex plane”, but should not be confused with  $\mathbb{C}^2$ , which would be all pairs of complex numbers.

map circles and lines to circles and lines, thereby treating lines as special kinds of circles (with infinite radius).

At first, a line appears to have a topology different from that of a circle. For example, on a line it makes sense to say that one point lies between two other points, whereas on the (unoriented) circle this will always be the case, unless some of these points coincide. In order to obtain a consistent framework in which lines can be treated as special cases of circles, one additional point has to be added to the plane. This “point at infinity” is usually denoted as  $\infty$ . It compactifies the plane into a topological sphere, which is usually called the Riemann sphere. Any Möbius transformation which turns lines into circles and some circles into lines will take the original point at infinity to a finite position, and move a different point to the position at infinity.

We will use a projective approach to provide a unified treatment of both finite and infinite points. Consider a finite point in  $\mathbb{C}^1$ , denoted by a single complex number  $z$ . In the projective complex line  $\mathbb{C}\mathbb{P}^1$ , this coordinate will be *homogenized* to the two-coordinate vector  $(z, 1)^T$ . Any vector which is a non-zero multiple of this one can be used to denote the same value. So in order to obtain the original complex number  $z$ , one has to divide the first coordinate by the second. The point at infinity is denoted using the vector  $(1, 0)^T$ , or in fact any vector with a second component of zero, as scalar multiples are again identified. The zero vector denotes no point at all.\*

$$\text{Homogenization:} \\ \mathbb{C}^1 \ni z \rightsquigarrow \begin{pmatrix} z \\ 1 \end{pmatrix} \in \mathbb{C}\mathbb{P}^1$$

$$\text{Dehomogenization:} \\ \mathbb{C}\mathbb{P}^1 \ni \begin{pmatrix} x \\ y \end{pmatrix} \rightsquigarrow \frac{x}{y} \in \mathbb{C}^1 \cup \{\infty\}$$

### Definition 2.1: Möbius transformation

A Möbius transformation is an automorphism of the Riemann sphere ( $\mathbb{C} \cup \{\infty\}$  or equivalently  $\mathbb{C}\mathbb{P}^1$ ) of the form

$$z \mapsto \frac{az + b}{cz + d} \quad \text{resp.} \quad \begin{pmatrix} z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} z \\ 1 \end{pmatrix} \quad a, b, c, d \in \mathbb{C}; \quad ad - bc \neq 0$$

\*The result of some algebraic operation could still be a zero vector, if the corresponding geometric situation is ill-defined in some way. For example, one can use the adjugate of a transformation matrix to describe the inverse transformation. But if the original transformation is not injective, its matrix is singular and therefore not invertible. Then the adjugate matrix would be the null matrix, and applying it to a point would result in a null vector. This corresponds to the fact that the preimage of a point under such a transformation is no longer uniquely defined.

Besides eliminating the need for special treatment of  $\infty$ , the second notation using homogeneous coordinates has another advantage: as a Möbius transformation in that setup is performed using a matrix-vector multiplication, concatenation of transformations can be easily formulated as a multiplication of matrices.

Multiplying the whole matrix by a complex scalar will multiply the result of the transformation by that same factor. As scalar multiples of homogeneous coordinates denote the same point, scalar multiples of matrices denote the same transformation. One can always choose a factor such that the determinant of the matrix becomes 1, so that every Möbius transformation can be written as an element of  $\text{SL}(2, \mathbb{C})$ . Even then there will still be *two* such represents for every transformation, as multiplying a matrix with  $-1$  will not change its determinant.

Of all these matrices we are interested in those that also map the unit circle onto itself.

### Lemma 2.1

The Möbius transformations which preserve the unit circle are exactly those which can be represented as a matrix of the form

$$M = \begin{pmatrix} \bar{w} & v \\ \bar{v} & w \end{pmatrix} \quad v, w \in \mathbb{C}; \quad \det(M) \neq 0 \quad (2.5)$$

The arrangement of variables in this matrix might appear slightly strange. The reasons for this choice will be explained on Page 39.

*Proof:* The first step is to show that transformations of the given form do in fact preserve the unit circle. For an arbitrary point  $z$  anywhere in the plane, the absolute value of the transformed point is calculated as follows.

$$\begin{aligned} \lambda \begin{pmatrix} z' \\ 1 \end{pmatrix} &= \begin{pmatrix} \bar{w} & v \\ \bar{v} & w \end{pmatrix} \cdot \begin{pmatrix} z \\ 1 \end{pmatrix} \\ z' &= \frac{\bar{w}z + v}{\bar{v}z + w} \\ |z'|^2 &= z' \cdot \overline{z'} = \frac{\bar{w}z + v}{\bar{v}z + w} \cdot \overline{\left( \frac{\bar{w}z + v}{\bar{v}z + w} \right)} = \frac{\bar{w}z + v}{\bar{v}z + w} \cdot \frac{w\bar{z} + \bar{v}}{v\bar{z} + \bar{w}} \\ &= \frac{\bar{w}w\bar{z}z + \bar{w}\bar{v}z + v w\bar{z} + v\bar{v}}{\bar{v}\bar{v}\bar{z}z + \bar{v}\bar{w}z + wv\bar{z} + w\bar{w}} = \frac{|w|^2 \cdot |z|^2 + 2 \operatorname{Re}(\bar{v}wz) + |v|^2}{|v|^2 \cdot |z|^2 + 2 \operatorname{Re}(\bar{v}wz) + |w|^2} \end{aligned} \quad (2.6)$$

If  $z$  lies on the unit circle, then we have

$$|z|^2 = z\bar{z} = 1$$

$$|z'|^2 = \frac{|w|^2 + 2\operatorname{Re}(\overline{v}wz) + |v|^2}{|v|^2 + 2\operatorname{Re}(\overline{v}wz) + |w|^2} = 1$$

Therefore the image of a point on the unit circle will lie on the unit circle as well.

For the proof of the converse, that all Möbius transformations which preserve the unit circle can be written in the above form, a simple counting argument shall suffice. A generic Möbius transformation has 4 complex matrix entries, but multiplying the matrix by any complex number doesn't change the transformation it describes. For this reason, there are effectively three complex degrees of freedom, and a mapping from any three distinct points in  $\mathbb{CP}^1$  to their images uniquely defines a Möbius transformation. If the unit circle is to be preserved, then the pre-image points can be chosen on the unit circle, which forces the image points to lie on that circle as well. So instead of three points in the plane, one can now choose only three points on the circle, which corresponds to three real degrees of freedom. The above matrix is defined by two complex numbers  $v$  and  $w$ , but multiplication by a real number yields the same transformation. So there are effectively three real degrees of freedom in the above matrix notation. Therefore, the notation can express any Möbius transformation which preserves the unit circle. Note that multiplying the matrix with a non-real complex number will change its form to disagree with the above notation, so there are representatives which express the same transformation but don't have the stated form.  $\square$

Möbius transformations are invertible and act on  $\mathbb{CP}^1$  in a continuous way. Those of the given form preserve the unit circle. Because the same is true for their inverse, they cannot map any points onto the unit circle except those which already are on the unit circle. Because of the continuity, they cannot map part of the interior onto the the interior and part onto the exterior without mapping some part onto the circle itself. But they could still map all of the interior to the exterior, which would not correspond to a hyperbolic transformation. We therefore want to not only preserve the unit circle, but also its interior. This can be enforced by a stronger condition on the determinant.

**Lemma 2.2**

The Möbius transformations which preserve the unit disk (i.e. the interior of the unit circle) are exactly those which can be represented as a matrix of the form

$$M = \begin{pmatrix} \bar{w} & v \\ \bar{v} & w \end{pmatrix} \quad v, w \in \mathbb{C}; \quad \det(M) = 1 \quad (2.7)$$

*Proof:* Assuming  $|z| < 1$ , we want  $|z'| < 1$  in Equation (2.6).

$$\begin{aligned} \frac{|w|^2 \cdot |z|^2 + 2 \operatorname{Re}(\bar{v}wz) + |v|^2}{|v|^2 \cdot |z|^2 + 2 \operatorname{Re}(\bar{v}wz) + |w|^2} &< 1 \\ |w|^2 \cdot |z|^2 + 2 \operatorname{Re}(\bar{v}wz) + |v|^2 &< |v|^2 \cdot |z|^2 + 2 \operatorname{Re}(\bar{v}wz) + |w|^2 \\ |v|^2 (1 - |z|^2) &< |w|^2 (1 - |z|^2) \\ |v|^2 &< |w|^2 \\ \det(M) = \bar{w}w - \bar{v}v = |w|^2 - |v|^2 &> 0 \end{aligned}$$

So the determinant of a matrix with the given form will always be real. If it is positive, then it will map the interior of the unit disk onto itself. Scaling the whole matrix  $M$  with a real coefficient  $\lambda$  will preserve both its form and the transformation it describes, but will scale its determinant by  $\lambda^2$ . Therefore, for  $\lambda = \frac{1}{\sqrt{\det(M)}}$ , the resulting matrix  $\lambda M$  has determinant 1 and will serve as a representative of the transformation satisfying Equation (2.7).  $\square$

Always scaling transformation matrices in such a way that their determinant equals 1 ensures that their product will again have determinant 1. In situations where many such matrices are multiplied, this avoids floating point number overflow in some cases where additional normalization steps would be required otherwise.

Note that the stated requirements on the form of the matrix and its determinant still don't give a unique representative for every transformation: multiplying the matrix by  $-1$  will preserve both the mapping and the determinant. So there are two possible representatives for every transformation.

**Lemma 2.3**

The inverse of a transformation of the form given in Equation (2.7) will again be of that same form.

*Proof:*

$$M^{-1} = \begin{pmatrix} \bar{w} & v \\ \bar{v} & w \end{pmatrix}^{-1} = \frac{1}{\det(M)} \begin{pmatrix} w & -v \\ -\bar{v} & \bar{w} \end{pmatrix} = \begin{pmatrix} w & -v \\ -\bar{v} & \bar{w} \end{pmatrix} \quad (2.8)$$

□

If the original matrix  $M$  did not satisfy  $\det(M) = 1$ , then the rightmost matrix in the above equation would be a real multiple of the actual inverse matrix and therefore still be a representative of the inverse transformation. So the formula above can be used even without strict adherence to the determinant normalization.

If one were to represent the parameters of a transformation using four real numbers, then the above formula would mean changing three signs. However, by switching the representative, one can make do with a single sign switch only, namely that of the real part of  $w$ :

$$M^{-1} = \begin{pmatrix} \bar{w} & v \\ \bar{v} & w \end{pmatrix}^{-1} \sim \begin{pmatrix} -w & v \\ \bar{v} & -\bar{w} \end{pmatrix}$$

**Lemma 2.4**

A Möbius transformation which preserves the unit disk (i.e. one of the form given in Equation (2.7)) is an isometry of the hyperbolic plane.

*Proof:* If the unit disk is preserved under the transformation, then points of the hyperbolic plane are again mapped onto points of the hyperbolic plane. To show that the transformation is an isometry, one can first reformulate the distance function from Equation (2.1) in terms of  $\mathbb{C}^1$ :

$$\begin{aligned} d(a, b) &= \operatorname{arcosh} \left( 1 + \frac{2|a-b|^2}{(1-|a|^2)(1-|b|^2)} \right) \\ &= \operatorname{arcosh} \left( 1 + 2 \frac{(a-b)(\overline{a-b})}{(1-a\bar{a})(1-b\bar{b})} \right) \end{aligned} \quad (2.9)$$

Using this formulation, a straight-forward calculation will show that the fraction in the formula is preserved.

$$\begin{aligned}
& \frac{(a' - b')\overline{(a' - b')}}{(1 - a'\overline{a'})(1 - b'\overline{b'})} \\
&= \frac{\left(\frac{\overline{w}a + v}{\overline{v}a + w} - \frac{\overline{w}b + v}{\overline{v}b + w}\right) \left(\frac{w\overline{a} + \overline{v}}{v\overline{a} + \overline{w}} - \frac{w\overline{b} + \overline{v}}{v\overline{b} + \overline{w}}\right)}{\left(1 - \frac{\overline{w}a + v}{\overline{v}a + w} \frac{w\overline{a} + \overline{v}}{v\overline{a} + \overline{w}}\right) \left(1 - \frac{\overline{w}b + v}{\overline{v}b + w} \frac{w\overline{b} + \overline{v}}{v\overline{b} + \overline{w}}\right)} \\
&= \frac{((\overline{w}a + v)(\overline{v}b + w) - (\overline{w}b + v)(\overline{v}a + w))((w\overline{a} + \overline{v})(v\overline{b} + \overline{w}) - (w\overline{b} + \overline{v})(v\overline{a} + \overline{w}))}{((\overline{v}a + w)(v\overline{a} + \overline{w}) - (\overline{w}a + v)(w\overline{a} + \overline{v}))((\overline{v}b + w)(v\overline{b} + \overline{w}) - (\overline{w}b + v)(w\overline{b} + \overline{v}))} \\
&= \frac{(a(w\overline{w} - v\overline{v}) - b(w\overline{w} - v\overline{v}))(\overline{a}(w\overline{w} - v\overline{v}) - \overline{b}(w\overline{w} - v\overline{v}))}{((w\overline{w} - v\overline{v}) - a\overline{a}(w\overline{w} - v\overline{v}))((w\overline{w} - v\overline{v}) - b\overline{b}(w\overline{w} - v\overline{v}))} \\
&= \frac{\det(M)^2 \cdot (a - b)\overline{(a - b)}}{\det(M)^2 \cdot (1 - a\overline{a})(1 - b\overline{b})} \\
&= \frac{(a - b)\overline{(a - b)}}{(1 - a\overline{a})(1 - b\overline{b})}
\end{aligned}$$

As this fraction is preserved under the transformation, so is the whole value of the inverse hyperbolic cosine, and therefore the distance.  $\square$

Note that there is an alternate and equivalent way to define distances in the hyperbolic plane: not via Equation (2.1) but instead by computing a cross ratio. The four points entering that cross ratio are the two points whose distance shall be computed and the two ideal points on the hyperbolic line connecting them. The resulting cross ratio has to be post-processed (taking the logarithm and multiplying that by a constant) to obtain a length, but these details are unimportant for the point at hand. Using such a definition of distance measurements, one can see that the transformations described by Equation (2.7) will be isometries, even without the calculation above. The reason is that Möbius transformations are projective transformations, and projective transformations preserve the cross ratio. So as long as ideal points get mapped to ideal points by a transformation which preserves cross ratios, hyperbolic lengths will be preserved as well.[39]

As a Möbius transformation which preserves the unit circle is defined by the images of three distinct points on the circle, there are two cases to distinguish: either the transformation preserves the order of points along the unit circle, or it reverses that order. Reversing the order will cause the exterior and the interior of the unit disk to be interchanged, which would not represent a mapping of the hyperbolic plane and is thus forbidden in Equation (2.7). Therefore, Möbius

transformations can only be used to express *orientation-preserving* isometries. In order to represent transformations which reverse orientation, one has to use *anti-Möbius transformations*.

**Definition 2.2: Anti-Möbius transformation**

An *anti-Möbius transformation* is a combination of a Möbius transformation and a conjugation.

The symbol “conj” will be used to denote the conjugation operation of such a transformation.

$$\text{conj} \begin{pmatrix} x \\ y \end{pmatrix} := \begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix}$$

Note that both components of a vector are conjugated, which ensures that equivalence classes in  $\mathbb{CP}^1$  are preserved. Also note that one cannot express this conjugation as a matrix multiplication. A notation like multiplication will nevertheless be used for the concatenation of operators.

**Lemma 2.5**

An anti-Möbius transformation of the form

$$\begin{pmatrix} \bar{w} & v \\ \bar{v} & w \end{pmatrix} \cdot \text{conj} \quad v, w \in \mathbb{C}; \quad \left| \begin{pmatrix} \bar{w} & v \\ \bar{v} & w \end{pmatrix} \right| = 1 \quad (2.10)$$

is an isometry of the hyperbolic plane.

*Proof:* This is obviously true for the case of the conjugation alone, when the matrix represents the identity. In all other cases, one can think of it as two operations, a conjugation followed by a Möbius transformation, each of which is an isometry of the hyperbolic plane.  $\square$

**Definition 2.3: Generalized Möbius transformation**

Möbius transformations and anti-Möbius transformations together form a group, which is called the group of *generalized Möbius transformations*.



**Theorem 2.6: Hyperbolic isometries**

The isometries of the hyperbolic plane are exactly those generalized Möbius transformations which can be written as

$$\begin{pmatrix} \bar{w} & v \\ \bar{v} & w \end{pmatrix} \cdot \text{conj}^c \quad v, w \in \mathbb{C}; \quad c \in \{0, 1\}; \quad \left| \begin{pmatrix} \bar{w} & v \\ \bar{v} & w \end{pmatrix} \right| = 1 \quad (2.11)$$

The exponent  $c$  simply denotes whether or not a conjugation should be performed.  $\text{conj}^1$  is a conjugation whereas  $\text{conj}^0$  denotes the identical transformation.

*Proof:* As we have already shown that both Möbius and anti-Möbius transformations are isometries of the hyperbolic plane, we only have to prove that there are no other isometries besides these. Again we examine the degrees of freedom to show that all possible transformations are already covered. We rely on the fact that an isometry not only preserves lengths, but also the absolute value of angles, as angles can be calculated from length measurements.

A polar coordinate system can be established in the hyperbolic plane by fixing a point as its origin and a half-line originating at this point. Once these two are chosen, every point in the plane can be uniquely identified by its (hyperbolic) distance from the origin and its angle ( $\text{mod } 2\pi$ ) measured from the fixed half-line. A transformation which preserves lengths and angles is already fully specified by its effect on this reference system, as all other points have to remain in fixed relation to it. As hyperbolic isometries only preserve the *absolute value* of the angle, there remains a choice between two possible directions in which angles are measured, in addition to the action on the reference system.

Choosing a point on the unit disk as the image of the origin accounts for two real degrees of freedom. Choosing the direction of zero angle adds a third real degree of freedom. This exactly equals the number of degrees of freedom in the Möbius transformations, as used in our proof of lemma 2.1. The remaining choice of direction reflects whether to use a Möbius or an anti-Möbius transformation.

To obtain a more thorough proof that goes beyond simple counting of degrees of freedom, one can use rotations in the form of Equation (2.14) in combination with translations in the form of Equation (2.13) to transform the reference system. This combination can map the reference system to any location and orientation desired.  $\square$

## 2.2.2 The group of isometries

In lemma 1.2 it has been shown that the isometries of any manifold form a group. In this section here, we will show the same in an algebraic way, and for the specific representation of hyperbolic isometries.

### Theorem 2.7: Group of hyperbolic isometries

The set of all hyperbolic isometries, represented in the fashion of Equation (2.11), forms a group with the concatenation of transformations as the group operation.

*Proof:* The identity transformation is obviously the identity Möbius transformation,

$$\text{id} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \text{conj}^0$$

Writing concatenation of transformations like multiplication, the following rules apply when dealing with conjugations:

$$\text{conj} \cdot \text{conj} = \text{id} \qquad \text{conj} \cdot M = \overline{M} \cdot \text{conj} \qquad (2.12)$$

Using these rules, it is always possible to move all conjugations to the very right of the notation, and to cancel an even number of conjugations, resulting in zero or one conjugations remaining.

The product of two orientation-preserving transformations is computed using matrix multiplication. Its result is again of the required form.

$$\begin{pmatrix} \overline{b} & a \\ \overline{a} & b \end{pmatrix} \cdot \begin{pmatrix} \overline{d} & c \\ \overline{c} & d \end{pmatrix} = \begin{pmatrix} \overline{ac + bd} & \overline{bc + ad} \\ \overline{bc + ad} & \overline{ac + bd} \end{pmatrix}$$

For orientation-preserving transformations, the existence of an *inverse* element of the required form has already been proven in lemma 2.3. Conjugation is its own inverse. When combined, we can invert the matrix and the conjugation independently, exchanging their order, and then re-order the terms as described above.

The associativity of the concatenation of transformations is well-known for orientation-preserving matrix transformations, and obviously still true when conjugations are involved due to the rules on conjugation handling from Equation (2.12).

So we have an associative operation which combines two group elements to form another group element. We also have an identity element and for each element of the group there exists an inverse element inside the group. Therefore all group axioms are satisfied.  $\square$

**Lemma 2.8**

The group of all hyperbolic isometries is isomorphic to  $\text{PGL}(2, \mathbb{R})$

*Proof:* The orientation-preserving transformations can be described by their action on the unit circle. The unit circle is isomorphic to the projective real line, and the action of Möbius transformations which preserve the interior of the unit disk acts as a projective special linear group  $\text{PSL}(2, \mathbb{R})$  on this projective real line, and therefore preserves orientation.

The anti-Möbius transformations of the plane operate on the circle in an orientation-reversing way. So both Möbius and anti-Möbius transformations together correspond to an arbitrary projective transformation of the real line, expressed by the projective general linear group  $\text{PGL}(2, \mathbb{R})$ . [5]  $\square$

This idea has already been foreshadowed in Section 2.1.2. In the half-plane model, the orientation-preserving isometries are exactly the transformations of  $\text{PSL}(2, \mathbb{R})$  acting on the whole complex plane. The orientation-reversing members from  $\text{PGL}(2, \mathbb{R})$  will exchange upper and lower half-plane, but if one identifies them by reflection in the real axis, then the isometries are exactly the real Möbius transformations.

**2.2.3 A catalog of hyperbolic isometries**

The following list gives explicit transformation matrices for a number of basic transformations. They can be used as building blocks for arbitrary transformations. In general the given matrices are not normalized to  $\det(M) = 1$ , so a subsequent normalization step is usually required to achieve the form from Equation (2.11).

**Shifting the origin**  $(0, 0)^T$  to the point  $(v, w)^T$  can be accomplished by the transformation

$$\begin{pmatrix} \bar{w} & v \\ \bar{v} & w \end{pmatrix} \quad (2.13)$$

Depending on the actual representation of the point  $(v, w)^T$ , this can be an arbitrary orientation-preserving transformation which takes the origin to the specified destination. In order to achieve a particular orientation as well, a subsequent rotation might be needed.

**A Rotation** by an angle  $\varphi$  around the origin is given by

$$R_\varphi := \begin{pmatrix} \exp\left(\frac{i}{2}\varphi\right) & 0 \\ 0 & \exp\left(-\frac{i}{2}\varphi\right) \end{pmatrix} \quad (2.14)$$

If the center of rotation is to be any other point, it can be shifted to the origin first, and back after the transformation has been executed.

**A Translation** by a given distance  $x$  along the real axis is given by

$$T_x := \begin{pmatrix} e^x + 1 & e^x - 1 \\ e^x - 1 & e^x + 1 \end{pmatrix} \quad (2.15)$$

In order to achieve a translation by a fixed distance along any other axis, that axis can be transformed to the real axis using shifts and rotations, and back again after the operation.

Note that a translation in the hyperbolic plane differs in one central aspect from a translation in the Euclidean plane. A Euclidean translation will move all points in the plane along *parallel* lines, so it is often represented by a vector of translation. In the hyperbolic case, there is only a single line which, when taken as a whole, will remain fixed under that operation. We call this the *axis of translation*. Any other point in the plane will be moved along a curve. Those curves are lines of fixed distance to the axis of translation, but are not hyperbolic lines. For these reasons, a hyperbolic translation is best described by denoting its axis (with orientation) and stating the distance by which points on that axis will be moved.

## 2.2.4 Classification of isometries

When given a description of an isometry, it is possible to classify it into one of several categories which closely resemble the well-known types of Euclidean isometries. Note that it is conventional to classify Möbius transformations in a similar fashion, but using different terms. To highlight similarities with and distinctions to Euclidean geometry, I'll primarily use names like those used for Euclidean geometry, but mention the names from the Möbius group classifications as well. The key to either classification is an analysis of the fixed points of the transformation.

The first distinction is that between Möbius transformations and anti-Möbius transformations. Let us first concentrate on the former, i.e. the transformations which can be expressed as a simple matrix multiplication without any additional complex conjugation. Those are the transformations which preserve orientation. Let  $z = x + yi$  be the position of a point. In order for this point to be a fixed point, its homogeneous coordinates must be an eigenvector of the transformation matrix.

$$\begin{aligned} \begin{pmatrix} a - bi & c + di \\ c - di & a + bi \end{pmatrix} \cdot \begin{pmatrix} z \\ 1 \end{pmatrix} &= \lambda \begin{pmatrix} z \\ 1 \end{pmatrix} \quad a, b, c, d \in \mathbb{R} \\ \lambda &= (c - di)z + (a + bi) \\ (a - bi)z + (c + di) &= \lambda z = ((c - di)z + (a + bi))z \\ (c - di)z^2 + (2bi)z - (c + di) &= 0 \end{aligned}$$

If  $c = d = 0$ , then the matrix will represent a multiplication by a fixed complex root of unity  $\frac{a-bi}{a+bi}$ , which corresponds to a rotation around the origin, as described above. In this case, the second fixed point would be the point at infinity, which isn't described by the equation as its coordinates  $(1, 0)^T$  don't match the prescribed form. In the classification of Möbius transformations, such a transformation would sometimes be called *circular*, which is a special case of the *elliptic* transformations.

If we even have  $b = c = d = 0$ , then the equation will hold for any  $z$ , thus representing the identity transformation. The identity transformation can be seen as a special case of most of the other classes, so in a complete classification it makes sense to consider it as a distinct class by itself.

If  $c + di \neq 0$ , then there will in general be two fixed points.

$$z_{1,2} = \frac{-2bi \pm \sqrt{(2bi)^2 - 4(c - di)(c + di)}}{2(c - di)} = \frac{-bi \pm \sqrt{c^2 + d^2 - b^2}}{c - di}$$

In case the discriminant  $c^2 + d^2 - b^2$  is positive, both results will be located on the unit circle, as the following computation verifies.

$$|z_1|^2 = z_1 \cdot \bar{z}_1 = \frac{\sqrt{c^2 + d^2 - b^2} - bi}{c - di} \cdot \frac{\sqrt{c^2 + d^2 - b^2} + bi}{c + di} = \frac{c^2 + d^2 - b^2 + b^2}{c^2 + d^2} = 1$$

A similar equation holds for  $z_2$ . Those two points on the unit disk can be considered the ideal “endpoints” of a hyperbolic line, which is uniquely defined by those two points. The corresponding transformation is a hyperbolic translation, moving all points away from one of the fixed points and towards the other, keeping their connecting line as a whole invariant.

In terms of the usual classification of Möbius transformations, such a group would be called *hyperbolic*, although the use of this term here has only a very remote connection to its use in *hyperbolic* geometry. These two uses should not be confused.

If the discriminant is negative, the square root will result in a purely imaginary number. Conjugating that number will change its sign. For this reason, the computation above now expresses a slightly different product, namely

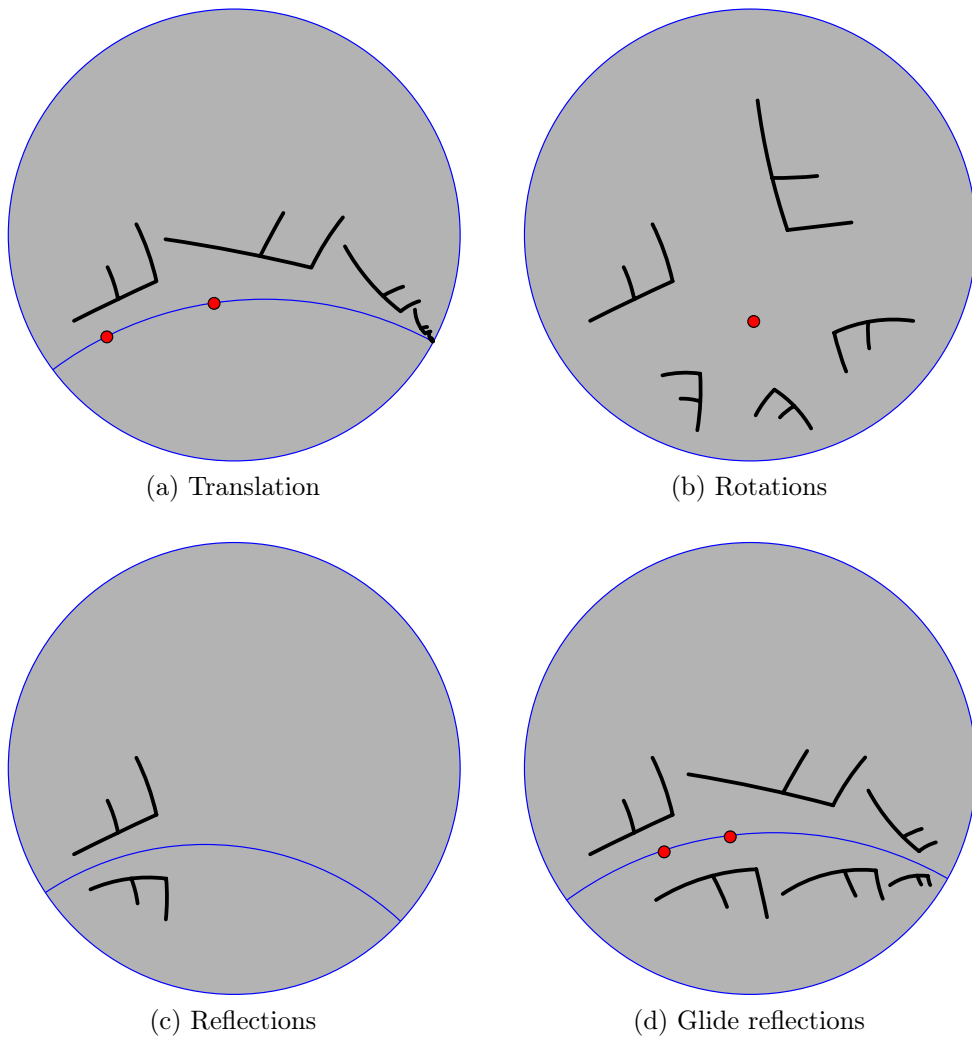


Figure 2.8: Hyperbolic isometries

$$z_1 \cdot \bar{z}_2 = \frac{\sqrt{c^2 + d^2 - b^2} - bi}{c - di} \cdot \frac{\sqrt{c^2 + d^2 - b^2} + bi}{c + di} = \frac{c^2 + d^2 - b^2 + b^2}{c^2 + d^2} = 1$$

This means that  $z_1$  and  $z_2$  are related to one another via an inversion in the unit circle. Taking the double cover of the Poincaré disk (as introduced in Section 2.1.1) into account, those two points in the model are in fact different representatives of the same point in the hyperbolic plane. That single fixed point is the center of a hyperbolic rotation. In the common classification of Möbius transformations, this would be called an *elliptic* transformation.

As a limiting case between these two, a discriminant of zero indicates that the two fixed points coincide in a single point on the unit circle. This denotes a so-called *limit rotation*. Like a rotation, it has no finite fixed lines, but like a translation, it has no finite fixed points either. Instead, all points will be moved along horocycles which pass through the single ideal fixed point. There is no obvious counterpart to this in Euclidean geometry, although depending on the way one translates concepts, one can think of this as a special case of either a translation or a rotation. In the nomenclature of *Möbius* transformations, this would be called a *parabolic* transformation.

So what is usually called an *elliptic* or perhaps even *circular* transformation shall be called a hyperbolic rotation in this work. What is usually called a *parabolic* transformation shall be called a limit rotation, and *hyperbolic* transformations shall be called translations in hyperbolic geometry. There is one more class of Möbius transformations called *loxodromic* transformations. They can be described by matrices  $M$  satisfying

$$\frac{\text{Tr}(M)^2}{\det(M)} \notin [0, 4]$$

If that fraction is greater than 4, this corresponds to the elliptic transformations described above. The loxodromic transformations are a superset of the elliptic ones. In the situation at hand, with the unit disk mapped onto itself, the fraction in the above expression can never become negative. This is because the trace of the matrix  $(a - bi) + (a + bi) \in \mathbb{R}$  is always real, so its square is always positive. And lemma 2.2 showed that the determinant of the transformation matrix will always be positive and can therefore be chosen equal to one. So there are no non-elliptic but loxodromic transformations which model isometries of the hyperbolic plane.

Now for the anti-Möbius transformations. Although the classification of Möbius transformations doesn't usually cover these, the comparisons to Euclidean transformations still yields intuitive names. Due to the conjugation involved here, combining the  $x$  and  $y$  coordinates to a single complex number is less helpful for these computations. Therefore the fixed point condition is more easily written as

$$\begin{aligned} \begin{pmatrix} a - bi & c + di \\ c - di & a + bi \end{pmatrix} \cdot \begin{pmatrix} x - yi \\ 1 \end{pmatrix} &= \lambda \begin{pmatrix} x + yi \\ 1 \end{pmatrix} \\ \lambda &= (c - di)(x - yi) + (a + bi) \\ (a - bi)(x - yi) + (c + di) &= ((c - di)(x - yi) + (a + bi))(x + yi) \\ (ax - by + c) + (-bx - ay + d)i &= ((cx - dy + a) + (-dx - cy + b)i)(x + yi) \\ (ax - by + c) + (-bx - ay + d)i &= ((cx - dy + a)x + (dx + cy - b)y) \\ &\quad + ((cx - dy + a)y - (dx + cy - b)x)i \end{aligned}$$

Component-wise comparison of this equation yields two real equations:

$$\begin{aligned} ax - by + c &= (cx - dy + a)x + (dx + cy - b)y \quad \Rightarrow \quad c = cx^2 + cy^2 \\ d - bx - ay &= (cx - dy + a)y - (dx + cy - b)x \quad \Rightarrow \quad 2(bx + ay) = d(x^2 + y^2 + 1) \end{aligned}$$

If  $c = 0$ , the first of these equations will always be satisfied. In that case, the second equation describes a circle which corresponds to a hyperbolic line in the Poincaré model. If in addition  $d = 0$ , then instead of a regular circle the second equation will describe a line through the origin, which also is a hyperbolic line. Both of these cases represent hyperbolic mirror reflections, which keep all points on the axis of reflection fixed.

If  $c \neq 0$ , then the first equation expresses the requirement that any fixed point must be a point on the unit circle. This corresponds to a glide reflection, with the second equation defining its axis.

To sum up the different classes and the conditions which identify them:

**Without conjugation:** preserves orientation

$$\begin{aligned} \mathbf{b} = \mathbf{c} = \mathbf{d} = \mathbf{0}: & \text{identity} \\ \mathbf{c}^2 + \mathbf{d}^2 > \mathbf{b}^2: & \text{translation} \\ \mathbf{c}^2 + \mathbf{d}^2 < \mathbf{b}^2: & \text{rotation} \\ \mathbf{c}^2 + \mathbf{d}^2 = \mathbf{b}^2: & \text{limit rotation} \end{aligned}$$

**With conjugation:** reverses orientation

$$\begin{aligned} \mathbf{c} = \mathbf{0}: & \text{reflection} \\ \mathbf{c} \neq \mathbf{0}: & \text{glide reflection} \end{aligned}$$



### 2.2.5 Modeling objects

One can use isometries in combination with distinguished objects in order to describe various geometric objects in the hyperbolic plane. One benefit is that it suffices to implement concatenation of isometries, since all other transformations then follow from this elementary operation. This way of describing objects tends to carry more information than many other possible representations, which usually doesn't hurt and can be useful at times.

**Points:** Any point in the hyperbolic plane can be expressed as the image of the origin under some isometry. Additional information are a direction (e.g. the image of the real axis) and a sign (from the conjugation). By the way, this is the reason for the “strange” form of the matrix so far: the origin has homogeneous coordinates  $(0, 1)^T$ , which means that the right column of the matrix,  $(v, w)^T$ , is its image. To keep this simple, the added complexity of the conjugation has been moved to the left column.

**Lines:** Any line in the hyperbolic plane can be expressed as the image of the real axis under some isometry. Additional information includes an orientation, a distinguished point (image of the origin) and a sign (from the conjugation).

**Half-planes** bounded by a given line can be described as the upper half disk (i.e. the set  $\{z \mid \text{Im}(z) > 0, |z| < 1\}$ ) under the same transformation that describes the bounding. Changing the conjugation gives the other half space.

**Convex polygons** can be described as the intersection of a finite number of half-planes.

There exist alternatives to these. One of the most interesting would model hyperbolic lines as elements of some Lie geometry. Disregarding their orientation (and thus switching to Möbius geometry), one could represent them as Hermitian matrices, upon which a Möbius transformation acts by conjugation.[2] The main benefit would be direct access to the center of the Euclidean circle. This approach hasn't been implemented so far, though, and it is not clear whether the gains would outweigh the cost of more multiplications and perhaps also more difficult handling of line orientations.



# Chapter 3

## Group navigation

### 3.1 Triangle reflection groups

There are ways to describe symmetry groups in terms of some algebraic expression like their orbifold symbol.[25] These kinds of descriptions tend to be very symbolic and rather unintuitive. One main goal of this work is to demonstrate a method where one can define specific hyperbolic symmetry groups using visual tools which are easy to understand and use.

The primary idea is to restrict the available groups to those which are subgroups of a given triangle reflection group. This concept will be used a lot in subsequent sections, so an acronym will help to keep the texts shorter and by this hopefully easier to read as well.

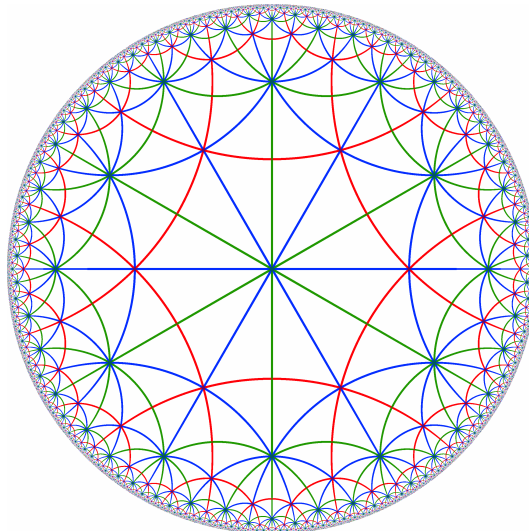


Figure 3.1: The  $(2, 4, 6)$  tiling, with edges of orbit  $a$  in red,  $b$  green and  $c$  blue.

**Definition 3.1: TRSG**

Let  $\Delta$  denote a triangle, and let  $G$  be the group generated by reflections in the edges of the triangle. If that group is discrete and has the triangle  $\Delta$  as a fundamental domain, then that group shall be called a *triangle reflection group*. Any symmetry group which is a subgroup of such a triangle reflection group will be abbreviated as a *TRSG*, short for *triangle reflection subgroup*.

The corners of the underlying triangle might be ideal points, i.e. lie on the circle itself, an infinite distance away from any point inside the hyperbolic plane. In one extreme, the whole triangle reflection group itself may be considered a TRSG, whereas in the other extreme the trivial group consisting only of the identity transformation is a TRSG as well. Between those two extrema, many interesting hyperbolic symmetry groups can be found. Section 3.3 will investigate the question of which symmetry groups can be expressed as TRSGs and which cannot.

One obvious thing that *can* be done is defining any symmetry group which arises from the identification of edges of a regular polygon: the axes of symmetry will cut that polygon into congruent right triangles, and an edge identification can be expressed as an identification of corresponding triangles, one on the inside of one edge and its image on the outside of the other edge. The particular choice of triangle defines the orientation of the identification as well.

**3.1.1 Mapping words to transformations**

The fundamental building block for most of our hyperbolic groups will be the reflections in the edges of a single hyperbolic triangle.

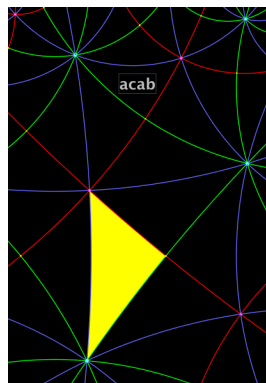


Figure 3.2: From words to triangles.

Let  $\Delta$  be a triangle (preferably near the center of the model) with edges  $a$ ,  $b$  and  $c$ . Let the angle between  $a$  and  $b$  be  $\frac{\pi}{r}$ , that between  $b$  and  $c$  be  $\frac{\pi}{s}$ , and  $\frac{\pi}{t}$  between  $c$  and  $a$ , for a given set of integers  $r$ ,  $s$  and  $t$  fulfilling the hyperbolic angle sum inequality

$$\frac{\pi}{r} + \frac{\pi}{s} + \frac{\pi}{t} < \pi \quad (3.1)$$

We use the same labels  $a$ ,  $b$  and  $c$  to denote reflections in these edges. The group  $G$  generated by these reflections tiles the plane with copies of the original triangle  $\Delta$ . The edge between any two adjacent triangles belongs to exactly one of the orbits  $Ga$ ,  $Gb$  or  $Gc$ . In Figure 3.1 these edge orbits are represented using different colors.

A word  $w \in \{a, b, c\}^*$  (i.e. a finite sequence of arbitrary length consisting of these three letters) denotes a specific element  $\bar{w} \in G$  which can be calculated as a nested function application. For example, the word  $acbc$  describes the transformation one obtains by first performing a reflection in the edge  $c$  of  $\Delta$ , then in  $b$ , then in  $c$  again and finally in  $a$ . Some readers might be more familiar with one of the following equivalent notations:  $(acbc)(x) = (a \circ c \circ b \circ c)(x) = a(c(b(c(x))))$  So while  $w$  is a sequence of letters,  $\bar{w}$  is a transformation in  $\mathbb{CP}^1$ . The map  $w \mapsto \bar{w}$  is not injective: a given transformation can be generated by different words. It is however surjective, since the group is generated by these reflections.

Another way to look at this is by drawing a path from  $\Delta$  to some image  $g\Delta$ , avoiding all triangle corners. Such a path will intersect triangle edges in a certain order. Each such edge is part of one of the edge orbits, i.e.  $Ga$ ,  $Gb$  or  $Gc$ . If one were to write down the corresponding letters  $a$ ,  $b$  or  $c$  from left to right in the order in which they are intersected by the path from  $\Delta$  to  $g\Delta$ , then the resulting word  $w$  will be a representation of  $g$ .

Figure 3.2 illustrates this use of words to denote transformations or triangles. The yellow triangle is the starting point  $\Delta$ . From there, a path to the labeled triangle exists which crosses red, blue, red and green, in this order. So one possible label for that triangle is  $acab$ . The associated transformation is the transformation which takes the yellow triangle to the labeled one.

At first glance, this may seem as if the order was reversed: the path crosses the edges in the order from left to right in the word, whereas the convention established above is that the righternmost transformation is to be applied first. But that reversal is in fact correct, because the reflections are not in the edges of the original triangle, but in some other edge from the same orbit which is incident with the path.

The truth of this method of reading off labels from paths can be visualized as follows: if one were to take the original path, and apply the transformations from left to right, then each of the transformations will fold the path along the edge

where it leaves  $\Delta$ , until at the end of the word the whole path has been folded into  $\Delta$ . So if the operations are performed from left to right, they will take  $g\Delta$  to  $\Delta$ . As performing reflections in reverse order results in the inverse of the combined operation, the original execution order from right to left will take  $\Delta$  to  $g\Delta$ , as claimed.

Another thing worth considering is local reference frames established by conjugation. Every word  $w$  describes a certain operation, originally based on reflections in  $\Delta$ . Instead of using that triangle, one might want to use the edges of a different triangle  $\bar{v}\Delta$  as the basic operations expressed by the word. To reformulate the resulting operation with respect to the original  $\Delta$ , one can first map  $\bar{v}\Delta$  to  $\Delta$  by reverse execution of  $v$ , i.e. by execution of  $v^{-1}$ . Then one performs the operation denoted by  $w$ , and finally one performs  $v$  again to take  $\Delta$  back to  $\bar{v}\Delta$ . Thus, the word  $w$  with respect to  $\bar{v}\Delta$  corresponds to the word  $vwv^{-1}$  with respect to  $\Delta$ .

To cross-check that concept, let  $v$  be the word obtained by reading off the edge labels from  $\Delta$  to  $\bar{v}\Delta$ , and  $w$  the word obtained by reading off the edge labels from  $\bar{v}\Delta$  to some other triangle which we'll show to be  $\bar{v}\bar{w}\Delta$ . Then the transformation taking  $\Delta$  to  $\bar{v}\bar{w}\Delta$  can be split into a part  $v$  taking it to  $\bar{v}\Delta$  and a part  $vwv^{-1}$  taking it from there to  $\bar{v}\bar{w}\Delta$ , to be executed in this order. Hence the total operation is  $(vwv^{-1})v = vw$ . This is exactly the word which can be read off the edge labels along the whole path, without stopping in between.

## 3.2 Combinatoric group calculations

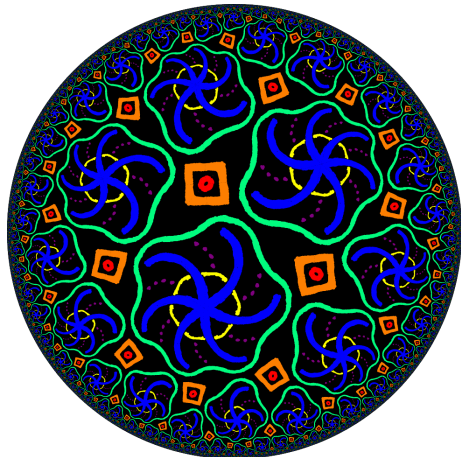
### 3.2.1 Triangle rewriting systems

One powerful tool when working with TRSGs are term rewriting systems.[4]

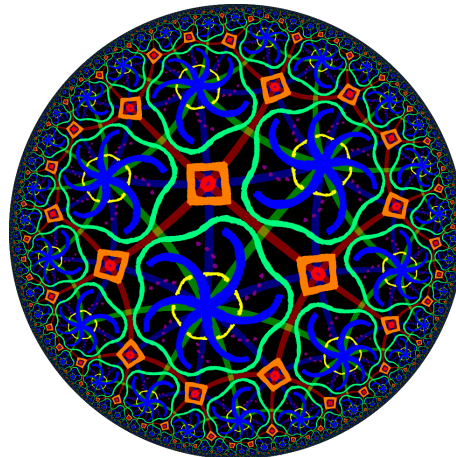
#### Definition 3.2: Term rewriting system

A term rewriting system over some finite alphabet  $A$  is a finite set of rules  $R \subset (A^* \times A^*)$ .

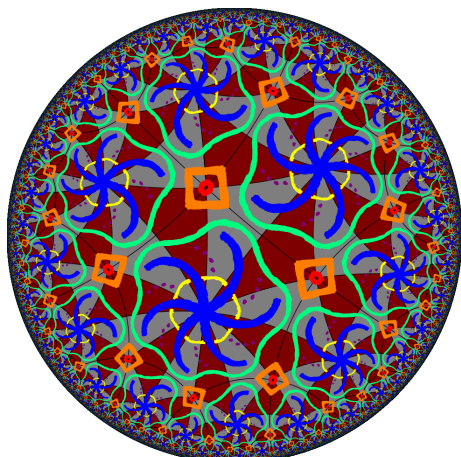
Each rule consists of two words over  $A$ , a left hand side and a right hand side. The idea behind this is two-fold. In an abstract semantic way, both sides of each rule are considered equivalent, and the whole system expresses the equivalence relation generated by these equivalences. In a more technical way, the rules form instructions how a given word should be manipulated: if the word contains the left hand side of any rule, that part of it can be substituted with the right hand side of the same rule.



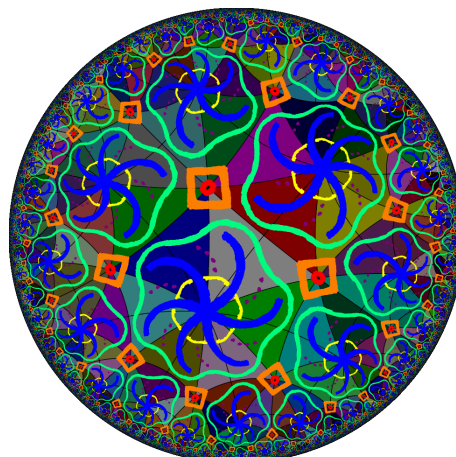
(a) The image from the introduction.



(b) Triangles drawn in, using a hyperbolic brush.



(c) There are two triangle orbits, one white and one red.



(d) Each fundamental domain consists of two triangles.

Figure 3.3: The picture from Page 1 revisited. It is based on triangle reflection group  $(2, 4, 6)$  and has subgroup generators  $a, bcb, bac$ .

The first thing that can be expressed in this way is the triangle group itself. In this context, the operations  $a$ ,  $b$  and  $c$  which we interpreted as isometric transformations so far will now be treated as generators of a group, without any dependency on the geometric interpretation. Subsequent paragraphs will use the labels  $\mathbf{a}$ ,  $\mathbf{b}$  and  $\mathbf{c}$  for these generators. Based on these generators, the triangle reflection group can be expressed as a Coxeter group

$$T = \langle \mathbf{a}, \mathbf{b}, \mathbf{c} \mid 1 = \mathbf{a}^2 = \mathbf{b}^2 = \mathbf{c}^2 = (\mathbf{ab})^r = (\mathbf{bc})^s = (\mathbf{ca})^t \rangle \quad (3.2)$$

We can formulate a term rewriting system expressing these relations, using the empty word  $\varepsilon$  to denote identity.

$$\begin{aligned} \mathbf{a}^2 &\rightarrow \varepsilon \\ \mathbf{b}^2 &\rightarrow \varepsilon \\ \mathbf{c}^2 &\rightarrow \varepsilon \\ (\mathbf{ab})^r &\rightarrow \varepsilon \\ (\mathbf{bc})^s &\rightarrow \varepsilon \\ (\mathbf{ca})^t &\rightarrow \varepsilon \end{aligned} \quad (3.3)$$

This rewriting system expresses the fact that different paths to the same triangle represent the same triangle of the triangular tiling. In other words, every equivalence class of the relation corresponds to exactly one triangle of the triangle reflection group.

Here is an example to illustrate how a rewriting system can be used to compute a normal form for a given triangle. The triangle rewriting system for the  $(2, 4, 6)$  triangle reflection group will eventually (after the completion procedure that will be described in Section 3.2.3) look as follows:

$$\begin{aligned} \text{(i)} \quad & \mathbf{aa} \rightarrow \varepsilon \\ \text{(ii)} \quad & \mathbf{bb} \rightarrow \varepsilon \\ \text{(iii)} \quad & \mathbf{cc} \rightarrow \varepsilon \\ \text{(iv)} \quad & \mathbf{ba} \rightarrow \mathbf{ab} \\ \text{(v)} \quad & \mathbf{caca} \rightarrow \mathbf{acac} \\ \text{(vi)} \quad & \mathbf{cbc bcb} \rightarrow \mathbf{bc bcbc} \\ \text{(vii)} \quad & \mathbf{cbc bcab} \rightarrow \mathbf{bc bcbca} \end{aligned} \quad (3.4)$$

Now this rewriting system can be applied to an arbitrary generator word  $w$  as follows:

$$w = \mathbf{acac babc} \xrightarrow{\text{(iv)}} \mathbf{acac abbc} \xrightarrow{\text{(ii)}} \mathbf{acac ac} \xrightarrow{\text{(v)}} \mathbf{aacacc} \xrightarrow{\text{(i)}} \mathbf{cacc} \xrightarrow{\text{(iii)}} \mathbf{ca}$$



So a transformation or triangle originally described using eight reflections can also be described using only two reflections. The reduction sequence is not necessarily unique; for example the last two steps might have been exchanged. The mentioned completion will ensure that the result does not depend on that order.

### 3.2.2 Orbifold rewriting systems

When interpreted as a symmetry group, the full triangle reflection group has a rather simple structure: every triangle is a fundamental domain of the ornament, and the orbit of a single triangle already covers the whole plane. When dealing with a subgroup of this triangle reflection group, things become more difficult. The most prominent question in this respect is the following: given two triangles in the plane, do they belong to the same orbit, i.e. do they contain the same portion of the artistic content of the ornament? A suitable tool here would be an equivalence relation which has the different orbits as equivalence classes. An alternate view of the same concept is the following: taking exactly one triangle from every orbit, one can glue their edges according not just to adjacency between individual triangles, but according to adjacency between orbits.

If formulated in terms of points of the plane instead of triangles, the resulting object will be the *orbifold* as defined in definition 1.11. In a certain sense, the orbifold is a very minimal description of the combinatorics of the ornament: from it, one can deduce properties like centers of rotation or axes of reflection, while at the same time (and in contrast to the fundamental domain) eliminating arbitrary choices like cuts around a center of rotational symmetry which does not lie on an axis of reflection.

In the world of TRSGs and their group representation using words, an orbit is the set of words describing triangles related to one another by elements of the TRSG. An orbifold is therefore the set of all such orbits, together with information about how these orbits relate to one another, i.e. a mapping from one orbit to another for each of the triangle group generators. Note that in the case of the full triangle reflection group, the orbifold would be only a single triangle orbit which would always map to itself under all triangle reflections.

The rules identifying different paths to the same triangle could be applied to *any* part of the input word. This is because every triangle in the triangulation basically plays the same role, so they can be interchanged without changing anything. The rules identifying different triangles from the same orbit are different, though. They must only be applied at the *beginning* of the input word. This can be achieved by adding a fourth letter **d** which denotes the beginning of the word. So the complete alphabet will now be  $A = \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$ . The new letter **d** will be prepended to every word before processing starts, and every rule must preserve that role of the letter

$d$ , i.e. have both sides starting with it or not contain it at all.

**Definition 3.3: Invalid, anchored and non-anchored words or rules**

A word which contains  $d$  in any position but the first will be called an *invalid* word. The remaining words will be called *valid* words. They fall into two categories: those starting with  $d$  will be called *anchored* words, whereas those not containing any  $d$  will be called *non-anchored* words. Similarly, rules where both sides are anchored words will be called *anchored* rules, and rules where both sides are non-anchored words will be called *non-anchored* rules. All other rules are *invalid*.

Non-anchored rules will identify different paths to the same triangle, while anchored rules identify different triangles of the same orbit. If one of the generators of a given group is an identification of the triangle denoted by word  $v$  with the triangle denoted by  $w$ , then a suitable anchored rewriting rule expressing this relation would be the following:

$$dv \rightarrow dw \quad (3.5)$$

This rule is based on the fact that the triangles  $v$  and  $w$  belong to the same orbit. The resulting rewriting systems, if all subgroup generators have been added in this fashion, will no longer describe equivalence classes for different paths to all the individual triangles, but equivalence classes for all paths ending in any of the triangles belonging to one of the orbits. If the symmetry group is cocompact, then the number of triangle orbits will be finite, and therefore the number of equivalence classes with the subgroup generators added will be finite as well. This is in contrast to the system for the triangle tiling, which has an infinite number of equivalence classes corresponding to the infinite number of triangles in the tiling.

### 3.2.3 Completion of rewriting systems

The rewriting system formulated in Equation (3.3) serves its purpose when read as an equivalence relation, i.e. without regard for the direction of the arrows. If one always applies rules from left to right, there are two important properties the rewriting system should have:[4]

**Termination** (or Noether property) describes the fact that any computation using the rewriting system will always terminate. This is best achieved by defining some kind of well-founded order over the words, and ensuring that every rule will produce a word that is less than the original word according to this order. Every calculation will stop with a minimal element where none of the left hand sides matches.

**Confluence** describes the fact that it doesn't matter in which order rules are applied. Whenever there are multiple locations in the input word where rules could apply, then the resulting words might be different, but there is some common word to which both can be rewritten. The so called Church-Rosser property is equivalent to confluence.[4] It guarantees not only confluence when starting from the same word, but also when starting from different words that are equivalent under the symmetric transitive closure of the rewriting system, i.e. under the system when interpreted as an equivalence relation.

A rewriting system having both of these properties is called *complete*. It is a very powerful tool indeed, because it can decide the word problem: given two words as input, it can calculate whether they are equivalent under the equivalence relation expressed by the rewriting system. It does so by repeatedly applying rewriting rules until none of the rules apply. Because the system is Noetherian, this computation will terminate in finite time. Because of confluence, the result will be independent of the order in which rules were applied, and because of the Church-Rosser property, this final representative will be the same if both words were equivalent to begin with. Obviously, as the rewriting system expresses the equivalence relation, words that can be rewritten to the same final result must have been equivalent all along, so we get this result:

### Corollary 3.1: Deciding word problem

Two words are equivalent with respect to some equivalence relation iff a complete rewriting system expressing that relation computes the same final result for both.

Out of the box, the rewriting system stated above usually isn't complete. One can try to reformulate it in such a way that it still represents the same equivalence relation, but also has the required properties for completeness.

In order to ensure termination, we define an order for the words used. We use the *ShortLex* ordering. It orders words by length first: shorter words are considered less than longer words. Words of equal length are compared lexicographically: the letters at the first position where both words differ decides their order, based on an order on their constituent letters, which we take to be  $\mathbf{a} < \mathbf{b} < \mathbf{c} < \mathbf{d}$ .

Making the original rewriting system Noetherian is quite easy: for every rule, see if its right hand side is less than its left hand side. If not, exchange the sides of the rule. Notice that the ShortLex order is local: if the right hand side of every rule is less than its left hand side, then the rule applied to a word will ensure that the resulting word is less than the original as well.

Ensuring confluence is harder. There is a well-established way to do so: the Knuth-Bendix procedure[28] can be applied to any rewriting system, and if the procedure terminates, then the resulting rewriting system will be confluent. The Knuth-Bendix procedure makes use of a term order, and if we choose that term order to again be the ShortLex order, then the Noetherianness of the system will be preserved. The procedure works by identifying potential overlap between left hand sides, and generating new rules from these so called *critical pairs*. The process has some similarity to Buchberger's algorithm for the construction of Gröbner bases.

For triangle rewriting systems, the Knuth-Bendix procedure will terminate, with one exception: if in the representation of Equation (3.2),  $r = 2$  and of  $s$  and  $t$ , one is equal to 3 and the other is an even number, then the algorithm will not terminate. This problem can be easily avoided by changing the term order, or equivalently by relabeling the edges in such a way that  $r = 3$ .

**Theorem 3.2**

For every triangle reflection group  $(r, s, t)$  there exists a term order on  $A = \{a, b, c\}$  such that the Knuth-Bendix procedure, applied to the rewriting system from Equation (3.2), will terminate.

Recall the relations from the presentation given in Equation (3.2):

$$1 = \mathbf{a}^2 = \mathbf{b}^2 = \mathbf{c}^2 = (\mathbf{ab})^r = (\mathbf{bc})^s = (\mathbf{ca})^t$$

Every combination of two different reflections has an integral count associated with it. A function mapping a pair of generators to that count can help keeping the notation short.

$$\nu(x, y) := \begin{cases} r & \text{if } \{x, y\} = \{\mathbf{a}, \mathbf{b}\} \\ s & \text{if } \{x, y\} = \{\mathbf{b}, \mathbf{c}\} \\ t & \text{if } \{x, y\} = \{\mathbf{c}, \mathbf{a}\} \end{cases}$$

One recurring pattern in many of the final replacement rules are strings consisting of two alternating letters. The relevant number here is not the number of times each character occurs, but instead the total length of the string, which might be odd. The following helper function  $\sigma$  denotes a string of alternating letters  $x$  and  $y$ , starting with  $x$  and with a total length of  $n$ .

$$\sigma(x, y, n) := \begin{cases} (xy)^k & \text{if } 2 \mid n = 2k \\ (xy)^k x & \text{if } 2 \nmid n = 2k + 1 \end{cases}$$

Now the previous two definitions will be combined into a shorthand notation used throughout the following set of rules. The notation again consists of two letters, which will be alternated to form the denoted string. But instead of giving the total length explicitly, the length will be deduced from the count associated with these generators. An integer  $k$  (which may be zero) is subtracted from that count to obtain the length of the resulting string.

$$xy \dot{-} k := \sigma(x, y, \nu(x, y) - k) \tag{3.6}$$

Table 3.1 gives an overview over all the rules for different values of  $r, s, t$ . Single letters in the rewriting rules stand for themselves, while combinations including  $\dot{-}$  use the shorthand notation defined in Equation (3.6). The commas were added for increased readability. Many of the rules are only applicable for some values of  $r, s, t$ . The corresponding conditions are given in the other columns of that table. In cases where no conditions are stated, the rule applies to all possible values.

rewriting rule	$\nu(a, b)$	$\nu(b, c)$	$\nu(c, a)$	$\nu(b, c)$	$\nu(c, a)$
$a, a \rightarrow$					
$b, b \rightarrow$					
$c, c \rightarrow$					
$ba^{-1}0 \rightarrow ab^{-1}0$					
$ca^{-1}0 \rightarrow ac^{-1}0$					
$cb^{-1}0 \rightarrow bc^{-1}0$					
$ca^{-1}1, bc^{-1}0 \rightarrow ac^{-1}0, bc^{-1}1$	$\geq 3$	$\geq 3$	$\geq 3$	even	odd
$cb^{-1}1, ab^{-1}0 \rightarrow bc^{-1}0, ab^{-1}1$	$\geq 3$	$\geq 3$	$\geq 3$	odd	even
$cb^{-1}1, ac^{-1}0 \rightarrow bc^{-1}0, ac^{-1}1$	$\geq 3$	$\geq 3$	$\geq 3$	even	odd
$ca^{-1}1, bc^{-1}1, ac^{-1}0 \rightarrow ac^{-1}0, bc^{-1}1, ac^{-1}1$	$\geq 3$	$\geq 3$	$\geq 3$	odd	even
$cb^{-1}1, ac^{-1}1, bc^{-1}0 \rightarrow bc^{-1}0, ac^{-1}1, bc^{-1}1$	$\geq 3$	$\geq 3$	$\geq 3$	odd	even
$ca^{-1}1, bc^{-1}1, ab^{-1}0 \rightarrow bc^{-1}0, ac^{-1}1, ab^{-1}1$	$\geq 3$	$\geq 3$	$\geq 3$	odd	even
$ca^{-1}1, bc^{-1}1, ab^{-1}0 \rightarrow ac^{-1}0, bc^{-1}1, ab^{-1}1$	$\geq 3$	$\geq 3$	$\geq 3$	odd	odd
$c, ab^{-1}0 \rightarrow bc^{-1}0, ab^{-1}1$					
$ca^{-1}1, bc^{-1}0 \rightarrow ac^{-1}0, b$	$= 2$	$= 2$	$= 2$	odd	odd
$cb^{-1}1, ac^{-1}0 \rightarrow bc^{-1}0, a$	$= 2$	$= 2$	$= 2$	even	even
$ca^{-2}2, bc^{-1}0, ab^{-1}1 \rightarrow ac^{-1}1, bc^{-1}0, ab^{-1}2$	$\geq 3$	$= 2$	$= 2$	even	even
$cb^{-2}2, ac^{-1}0, ba^{-1}1 \rightarrow bc^{-1}1, ac^{-1}0, b$	$\geq 3$	$= 2$	$= 2$	even	even
$ca^{-2}2, ac^{-1}0, ab^{-1}1 \rightarrow ac^{-1}0, ab^{-1}1, ca^{-1}1$	$\geq 3$	$= 2$	$= 2$	even	even
$cb^{-2}2, ac^{-1}0, ba^{-1}1 \rightarrow bc^{-1}1, ac^{-1}0, a$	$\geq 3$	$= 2$	$= 2$	even	even
$ca^{-1}1, b, ac^{-2}2, bc^{-1}0, ab^{-1}1 \rightarrow ac^{-1}0, b, ac^{-2}2, bc^{-1}0, ab^{-1}2$	$\geq 3$	$\geq 3$	$\geq 3$	odd	odd
$cb^{-1}1, a, bc^{-2}2, ac^{-1}0, ba^{-1}1 \rightarrow bc^{-1}0, a, bc^{-2}2, ac^{-1}0, ba^{-1}2$	$\geq 3$	$\geq 3$	$\geq 3$	odd	odd
$ca^{-1}1, bc^{-1}1, ab^{-1}0 \rightarrow ac^{-1}0, bc^{-1}1, a$	$= 2$	$\geq 3$	$\geq 3$	odd	odd
$cb^{-2}2, ab^{-1}0, ca^{-1}1 \rightarrow bc^{-1}1, ab^{-1}0, ca^{-1}2$	$= 2$	$\geq 3$	$\geq 3$	odd	odd
$ca^{-1}1, bc^{-2}2, ab^{-1}0, ca^{-1}1 \rightarrow ac^{-1}0, bc^{-2}2, ab^{-1}0, ca^{-1}2$	$= 2$	$\geq 3$	$\geq 3$	odd	odd
$cb^{-2}2, ab^{-1}0, ca^{-2}2, bc^{-1}0 \rightarrow bc^{-1}1, ab^{-1}0, ca^{-2}2, bc^{-1}1$	$= 2$	$\geq 3$	$\geq 3$	odd	odd
$ca^{-1}1, b, a, bc^{-1}0, ab^{-1}1 \rightarrow ac^{-1}0, b, a, bc^{-1}0, ab^{-1}2$	$= 2$	$\geq 3$	$\geq 3$	odd	even
$cb^{-1}1, a, b, ac^{-1}0, ba^{-1}1 \rightarrow bc^{-1}0, a, b, ac^{-1}0, ba^{-1}2$	$= 2$	$\geq 3$	$\geq 3$	odd	even
$ca^{-2}2, bc^{-1}0, ab^{-1}1 \rightarrow ac^{-1}1, bc^{-1}0, a$	$= 2$	$\geq 3$	$\geq 3$	even	even
$cb^{-2}2, ab^{-1}0, ca^{-1}1 \rightarrow bc^{-1}1, ab^{-1}0, c$	$= 2$	$\geq 3$	$\geq 3$	even	even
$ca^{-3}3, bc^{-1}0, ab^{-1}1, ca^{-1}1 \rightarrow ac^{-2}2, bc^{-1}0, ab^{-1}1, ca^{-1}2$	$\geq 3$	$= 2$	$= 2$	odd	odd
$cb^{-3}3, ac^{-1}0, ba^{-1}1, cb^{-1}1 \rightarrow bc^{-2}2, ac^{-1}0, ba^{-1}1, cb^{-1}2$	$\geq 3$	$= 2$	$= 2$	odd	odd
$ca^{-1}1, bc^{-3}3, ab^{-1}0, ca^{-1}1, bc^{-1}1 \rightarrow ac^{-1}0, bc^{-3}3, ab^{-1}0, ca^{-1}1, bc^{-1}2$	$= 2$	$\geq 3$	$\geq 3$	odd	odd
$cb^{-1}1, bc^{-3}3, ab^{-1}0, ca^{-1}1, bc^{-1}1 \rightarrow bc^{-1}1, ab^{-1}0, ca^{-3}3, bc^{-1}1, ab^{-1}0, ca^{-1}3$	$= 2$	$\geq 3$	$\geq 3$	odd	odd
$ca^{-3}3, bc^{-1}0, ab^{-1}1, ca^{-3}3, bc^{-1}0, ab^{-1}1 \rightarrow ac^{-2}2, bc^{-1}0, ab^{-1}1, ca^{-3}3, bc^{-1}0, a$	$\geq 3$	$= 2$	$= 2$	odd	odd
$cb^{-3}3, ac^{-1}0, ba^{-1}1, cb^{-3}3, ac^{-1}0, ba^{-1}1 \rightarrow bc^{-2}2, ac^{-1}0, ba^{-1}1, cb^{-3}3, ac^{-1}0, b$	$\geq 3$	$= 2$	$= 2$	odd	odd
$ca^{-3}3, ac^{-1}0, ba^{-1}1, cb^{-3}3, ac^{-1}0, ba^{-1}1 \rightarrow bc^{-2}2, ac^{-1}0, ba^{-1}1, cb^{-3}3, ac^{-1}0, a$	$\geq 3$	$= 2$	$= 2$	odd	odd
$c, ab^{-1}0, ca^{-3}3, bc^{-1}1, ab^{-1}0, ca^{-1}1, bc^{-1}1 \rightarrow bc^{-1}1, ab^{-1}0, ca^{-3}3, bc^{-1}1, ab^{-1}0, c$	$= 2$	$\geq 3$	$\geq 3$	odd	odd
$ca^{-1}1, bc^{-3}3, ab^{-1}0, ca^{-1}1, bc^{-3}3, ab^{-1}0, ca^{-1}1 \rightarrow bc^{-1}1, ab^{-1}0, ca^{-3}3, bc^{-1}1, ab^{-1}0, c$	$= 2$	$\geq 3$	$\geq 3$	odd	odd
$c, ab^{-1}0, ca^{-3}3, bc^{-1}1, ab^{-1}0, ca^{-3}3, bc^{-1}1, a$	$= 2$	$\geq 3$	$\geq 3$	odd	odd

Table 3.1: Results of the Knuth-Bendix procedure for triangle reflection groups.

So it is feasible to decide whether or not two words denote the same triangle. However, for the orbifold rewriting systems, there are TRSGs which do not even allow for a complete rewriting system based on the ShortLex order of terms. If no such system can exist, the Knuth-Bendix procedure will never terminate. The following subsection will give a proof for this theorem 3.6 based on geometric arguments.

If the Knuth-Bendix procedure based on the ShortLex order will not result in a finite rewriting system, there remain other avenues to explore in order to decide the word problem in these groups. One possibility is the use of a different term order. A more likely candidate, however, would be the use of automatic groups, which are a more powerful concept which allows computations equivalent to certain infinite rewriting systems[19]. The details of this approach for these particular groups haven't been worked out by the author yet, though, and though cited references sounds promising, we are not certain whether this approach has been taken for the application at hand. At the time of this writing, the word problem on orbifolds is to be considered somewhat problematic at best.

### 3.2.4 Geometric interpretation

In order to show that there can be no finite rewriting system which will always give a ShortLex-minimal representative for each orbit, it is useful to achieve a geometric understanding of the actions of the involved rewriting systems. So for the remainder of this section, we will assume that there were a complete rewriting system for the orbifolds of a given group, and we will study its properties.

Rewriting system (3.3) identifies different paths to the same triangle. One can interpret this as giving each triangle a unique and well-defined canonical name, which is to be the ShortLex-minimal word that denotes this triangle. These are all the words that don't contain any of the left hand sides of the corresponding complete rewriting system.

#### **Definition 3.4: Canonical triangle label**

The canonical label for a triangle is the ShortLex-minimal word denoting this triangle.

Stripping the last letter from each of these words results in a word denoting an adjacent triangle. The shortened word obviously still doesn't contain the left hand side of any of the rules, therefore it is the ShortLex label of that adjacent triangle. In this fashion, every triangle has a unique parent triangle with a shorter word, up

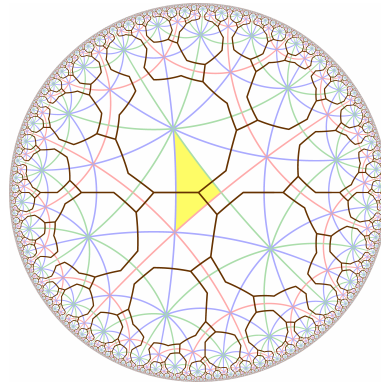


Figure 3.4: ShortLex tree, rooted at the yellow center  $\Delta$

to the central triangle  $\Delta$ . These links connect all the triangles in an infinite tree, with  $\Delta$  at its root. We call this the *ShortLex tree* of the triangle tiling.

The addition of rules like Equation (3.5) to the rewriting system adds relations that identify different triangles of the same orbit. The rewriting result will be a canonical label for the whole orbit. Again the rewriting system will return the minimal word among all the words for a given orbit.

**Definition 3.5: Canonical orbit label**

The canonical label for an orbit is the ShortLex-minimal word denoting any of its triangles.

The words used as canonical orbit labels denote triangles, one for every orbit. Taking one triangle from every orbit gives a fundamental domain. So in this way we have a canonical fundamental domain for the symmetry group.

**Definition 3.6: Canonical fundamental domain**

The canonical fundamental domain of a symmetry group consists of all those triangles denoted by the ShortLex-minimal representatives for each orbit.\*

What the added rules do in the complete rewriting system is this: Whenever the path described by a word leaves the canonical fundamental domain, its initial

---

\*Note that this canonical fundamental domain is mostly relevant for this chapter. It might not always be a convex polygon, which will be required for rendering as described in Section 4.1. Another fundamental domain will be used there. In other words, the canonical fundamental domain is a useful concept for theoretical proofs, but not always the best choice in practice.



portion will be rewritten to the corresponding triangle from the same orbit within the canonical fundamental domain.

**Lemma 3.3: Required orbit identification rules**

Let  $w \in A^*$  be the canonical label of a triangle in the canonical fundamental domain, and let  $g \in A$  denote a single reflection such that  $wg$  is the canonical label of an adjacent triangle outside the fundamental domain. Furthermore let  $v$  be the canonical label of the orbit of  $wg$ . Then every complete rewriting system that computes canonical labels for the orbits of the symmetry group will contain the rule

$$d wg \rightarrow d v$$

or a similar rule with the same left hand side and any right hand side that will eventually (through other rules) be rewritten to  $d v$ .

*Proof:* The rule is permissible for the rewriting system, because it identifies triangles of the same orbit. The rule is required, because  $d wg$  is illegal as the final result of the rewriting system, and because no other rule may apply to  $d wg$ . If any other rule were to apply to  $d wg$ , its left hand side would match a substring of  $d wg$ . So there are two cases to distinguish:

Case 1: a rule matching  $d w$ . This cannot be, because  $w$  is the canonical label of a triangle in the fundamental domain, and hence the canonical label of an orbit. The canonical labels of the orbits correspond to the irreducible words of the rewriting system. Therefore  $d w$  must be irreducible, and no rule can apply.

Case 2: a rule matching  $w g$ . As this part does not contain the letter  $d$ , none of the orbit-identifying rules can apply. The only rules in effect are those which map words to canonical triangle names. As  $w g$  is the canonical label of its triangle, those rules can not apply either. Therefore no rule can apply to  $w g$ .

The right hand side of the rule has to eventually rewrite to  $d v$ , because  $v$  is the canonical label of the orbit that triangle  $w g$  belongs to.  $\square$

**Lemma 3.4: Sufficient orbit identification rules**

If a complete rewriting system contains an anchored rule for every edge of the ShortLex tree that leaves the canonical fundamental domain, mapping it to the triangle of the corresponding orbit within the fundamental domain, and if the rewriting system furthermore has all the unanchored rules required to map unanchored words to canonical triangle names, then the rewriting system will rewrite every anchored word to the canonical name of the corresponding orbit.

*Proof:* Assume that  $dw \in A^*$  is an anchored word denoting a triangle outside the canonical fundamental domain, and assume that none of the rules of the rewriting system applies. As the rewriting system is guaranteed to rewrite  $w$  to the canonical name of the triangle, this means that  $w$  has to be canonical already. Therefore it corresponds to a graph in the ShortLex tree. As the triangle lies outside the canonical fundamental cell, and the root of the tree lies within, this path has to cross the boundary at some point. As such a crossing would correspond to an applicable rule, this is a contradiction to the assumption that no rule applies. As a consequence, as long as  $dw$  denotes a triangle outside the fundamental domain, some rule must apply. Within the fundamental domain, the unanchored part of the rewriting system ensures minimal labels.  $\square$

This latter result can be directly used to show that at least for a certain class of symmetry groups, a suitable complete rewriting system does exist.

**Theorem 3.5: Rewriting systems for finite fundamental domains**

Whenever a TRSG is cocompact, i.e. its fundamental domain consists of only a finite number of triangles, then there exists a complete rewriting system which maps each anchored word to the canonical label of its orbit.

*Proof:* If the canonical fundamental domain is finite, so is its boundary. Therefore there can be only a finite number of edges of the ShortLex tree which do cross that boundary. Per lemma 3.4, one anchored rule for every such edge, in addition to the complete triangle rewriting system, is sufficient to map each anchored word to its canonical orbit label.  $\square$

Note that this proves only the existence of such a rewriting system. The fact that the Knuth-Bendix procedure will always terminate after a finite number of steps with such a rewriting system as its result appears likely, but still remains to be proven. Note that in the Euclidean case, those crystallographic groups which

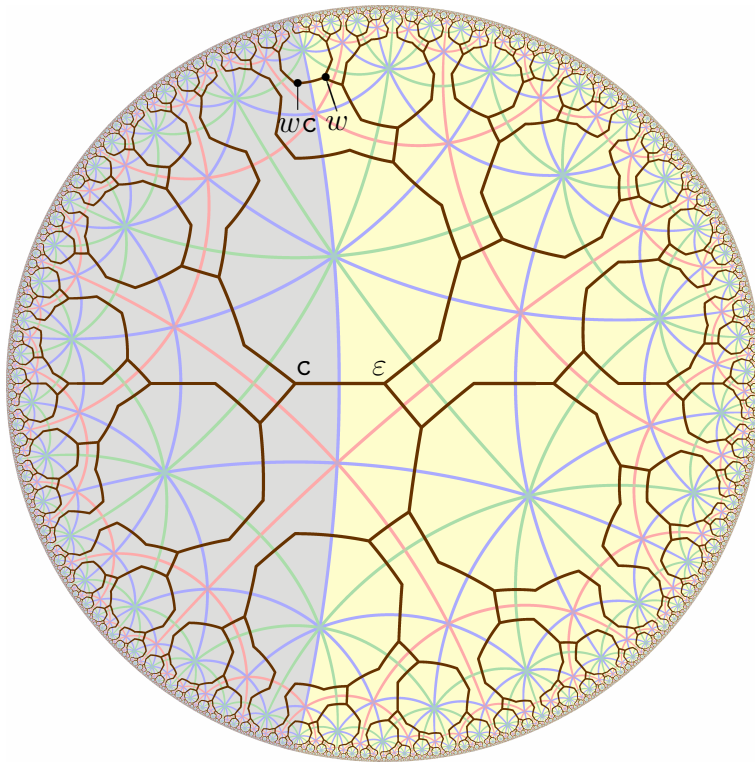


Figure 3.5: ShortLex tree crossing the boundary of the central fundamental domain (yellow)

have a finite fundamental domain are exactly the 17 well-known wallpaper groups (which are also summarized in Table 5.1 on Page 85). So for one possible definition of the term, the “hyperbolic wallpaper groups” can all be described using a complete orbifold rewriting system.

On the other hand, as claimed before, not every TRSG allows for a complete rewriting system for its orbifold. To find a suitable counter-example, one can have a closer look at the image of the ShortLex tree depicted in Figure 3.5. A single reflection in one of the edges of the central triangle  $\Delta$  will result in two fundamental domains, one on either side of the line along the edge. For reflections **a** (red) and **b** (green), the ShortLex tree crosses that boundary line only once. But for edge **c** (blue), *every* edge incident with that line will be crossed by the ShortLex tree. For edges sufficiently close to the center, this fact is readily apparent from the image, but for all the infinitely many triangles, a proof is in order.

**Theorem 3.6: Non-existence of complete orbifold rewriting system**

Not every TRSG allows for a complete rewriting system to calculate the ShortLex-minimal representative for each of its orbits.

*Proof:* This counter-example is based on the triangle reflection group

$$T = \langle a, b, c \mid 1 = a^2 = b^2 = c^2 = (ab)^2 = (ac)^4 = (bc)^6 \rangle$$

The corresponding complete triangle rewriting system can be computed using the Knuth-Bendix procedure and can be written like this:

$$\begin{aligned} aa &\rightarrow \varepsilon \\ bb &\rightarrow \varepsilon \\ cc &\rightarrow \varepsilon \\ ba &\rightarrow ab \\ caca &\rightarrow acac \\ cbcacb &\rightarrow bcbcbc \\ cbcacb &\rightarrow bcbcbca \end{aligned}$$

Adding  $c$  as the only non-trivial element of the symmetry group, one obtains a TRSG with an infinite fundamental domain. This corresponds to the added rule  $dc \rightarrow d$ .

Now take the path to a different triangle incident with the boundary of the fundamental domain. To keep the formulas easier, it makes sense to choose one which has the same orientation as the original fundamental domain, i.e. take not the next edge on the boundary but the one after that. Taking the one depicted above the center, one can read its canonical label from the ShortLex tree as

$$w = bcbcabca$$

$w$  is a canonical triangle label, as no left-hand side rule of the triangle rewriting system applies to it. Applying the rewriting system to its mirror image  $cw$ , one obtains

$$cw = \underline{c}bcbcabca \rightarrow bcbcb\underline{c}aca \rightarrow bcbcb\underline{a}cac \rightarrow bcbcbacac = wc$$

So  $wc$  is the canonical triangle label for this mirror image, which means that here the ShortLex tree crosses the boundary of the fundamental domain, which can be seen in the figure as well. As proven in lemma 3.3, this requires a rule with a

specific left-hand side to be included in the rewriting system. More specifically, one could use the anchored subgroup generator rule to obtain  $\mathbf{d}cw \rightarrow dw$ . To make the system confluent again, a rule like  $\mathbf{d}wc \rightarrow dw$  is required.

As  $w$  denotes a triangle in pretty much the same position as  $\Delta$ , powers of  $w$  will yield further triangles incident with the axis of reflection. The considerations above can easily be extended to those powers, as  $c$  has been shown to commute with  $w$ . So for all of these triangle pairs, there is an edge of the ShortLex tree which crosses the boundary of the fundamental domain. Per lemma 3.3, this would require an infinite number of rewriting rules, based on the pattern

$$\mathbf{d}w^n c \rightarrow dw^n \quad \forall n \in \mathbb{N}_0$$

As rewriting systems are considered to consist of only a finite number of rules, there can be no complete rewriting system containing all of these required rules.  $\square$

### 3.3 Expressiveness

The approach of defining hyperbolic symmetry groups as TRSGs is a very intuitive one, and flexible enough to describe quite a large number of interesting symmetry groups. It is, however, not expressive enough to describe all possible discrete hyperbolic symmetry groups. This section will investigate what additional freedoms a general hyperbolic symmetry group might provide, which of these can be covered by the TRSG approach, and what kinds of additional user interface tools might help to increase the set of obtainable symmetry groups.

In order to talk about possible symmetry groups, it is important to make exceptionally clear when two symmetry groups should be considered the same. Probably the most useful concept here is one which will be called a *geometric symmetry group* in this work.

#### Definition 3.7: Geometric symmetry group

Let  $M$  be any two-dimensional Riemannian manifold (in this work usually the hyperbolic plane), and let  $G \subset \text{iso}(M)$  be a *discrete* symmetry group over  $M$ . Then the *geometric symmetry group*  $[G]$  shall be defined as the equivalence class

$$[G] := \{G' \subset \text{iso}(M) \mid \exists T \in \text{Diff}(M) : G' = TGT^{-1}\}$$

So two symmetry groups will be considered the same up to conjugacy. Intuitively speaking, the group will be considered the same even if its defining geometric operations are all subject to the same diffeomorphism  $T$ . The most important

diffeomorphisms to consider here are isometries, and in the Euclidean case also scalings. In a certain sense, this concept of a geometric symmetry group is somewhere between a purely combinatoric view, where all isomorphic groups would be considered the same, and the original definition of a symmetry group which would consider two groups different even if their ornaments are only shifted or rotated (or perhaps scaled) versions of one another. Since the hyperbolic plane does not have any intrinsic concept of a distinguished point of origin, or an intrinsic reference frame for directions, the choice of coordinate system is pretty much arbitrary. A reasonable user interface should allow its users to perform isometries on the hyperbolic plane as a whole, thus providing access to all the equivalent representatives of a geometric symmetry group. The word “geometric” in the term just defined should express the fact that it captures all the aspects which can be observed intrinsically from within the geometry, like lengths and angles, but without artificial dependencies on an external reference frame.

**Corollary 3.7: Convex polygonal fundamental domain**

For every discrete cocompact symmetry group in the hyperbolic plane there exists a fundamental domain which is a convex polygon.

The Voronoi cells of the orbit of any non-singular point of the plane will provide a tiling of the plane using convex polygons which are fundamental domains. The converse is not true: not every convex fundamental domain is a Voronoi cell for some non-singular point. Many interesting fundamental domains can be obtained as the limit case where the defining point approaches some singularity. The shape of these will usually depend on the direction from which the point moves towards the singularity. But for now it is enough to know that at least one convex polygonal fundamental domain does exist, we don’t need to describe every possible one.

**Lemma 3.8: Parametric description of a fundamental domain**

Any geometric symmetry group of the hyperbolic plane with a compact fundamental domain can be uniquely defined using the following parameters describing a convex polygonal fundamental domain of that group:

- (i) The number  $n$  of corners of the fundamental polygon, with  $n \geq 3$ . From this count one can define  $V := \{v \in \mathbb{N} \mid 1 \leq v \leq n\}$  as an index set identifying the corners, and  $E := \{(v, (v \bmod n) + 1) \in V \times V \mid 1 \leq v \leq n\}$  as the index set identifying the edges.
- (ii) The interior angles for these corners, denoted as a map  $\alpha : V \rightarrow (0, \pi]$ . These angles have to satisfy the hyperbolic angle inequality  $\sum_{k=1}^n \alpha(k) <$

$(n - 2)\pi$ .

- (iii) The lengths of the edges, denoted as a map  $l : E \rightarrow \mathbb{R}^+$ . The interdependencies between these lengths and the angles are discussed below.
- (iv) A set of identification rules, expressing which edges are identified with one another. This is a map  $g : E \rightarrow E$  which needs to be an involution, i.e.  $\forall e \in E : g(g(e)) = e$ , in order to make the identifications relation symmetric.
- (v) A direction flag indicating the orientation of these identifications, denoted as a map  $d : E \rightarrow \{\text{same, opposite}\}$ . Here, “same” direction means that the arrows usually used to denote edge identifications will both be clockwise, or both be counter-clockwise, necessarily resulting in a non-orientable orbifold. “opposite” direction means that one arrow will be clockwise and the other counter-clockwise, so that crossing that boundary will maintain orientation. For reasons of symmetry,  $d(e) = d(g(e))$  must hold for all  $e \in E$ .

*Proof:* As stated in corollary 3.7, every discrete symmetry group with compact fundamental domain will have a convex polygonal fundamental domain. So the description of the group via its fundamental polygon will always be possible. The shape of a fundamental polygon is uniquely defined via its interior angles and edge lengths. This definition is up to isometries, which matches the fact that we are dealing with *geometric* symmetry groups here.

The parametrization above assumed that every edge is identified with exactly one edge. In arbitrarily chosen fundamental domains there might be cases where part of one edge gets identified with (part of) another edge, while the other part of it is identified with (part of) a third edge. In this case there will be a point along the edge where the identification scheme changes. That point can be considered a corner of the polygon, with  $\pi$  as the interior angle. Using this kind of reinterpretation, all edges can be split until they all are identified with just a single edge.

When all edge associations are fixed, the direction information  $d$  can be used to specify which point along the edge gets mapped onto which one. This scheme assumes that the endpoints of  $e$  will be associated with the endpoints of  $g(e)$ . There are symmetry groups where it would be possible to identify the endpoints of one edge with interior points of another edge. Sliding identified edges along one another in this fashion is usually called a *fractional Dehn twist*. But again these situations can be broken up by introducing corners with an interior angle of  $\pi$  on all affected edges at the locations where their matched edges have an endpoint. After this step, the edge identifications fit in the framework outlined above.  $\square$

So how much of this can be accomplished using TRSGs? The information outlined in (i), (iv) and (v) which describes the topology of the orbifold is rather

easy to achieve:

**Theorem 3.9: Realizable topologies**

For any possible discrete cocompact hyperbolic symmetry group, there exists a subgroup of a triangle reflection group with the same topology.

*Proof:* Choosing a polygon with a given number of corners is easy: For every  $n \geq 3$ , a regular  $n$ -gon made up from  $2n$  congruent triangles can be obtained by setting the order of one triangle corner to 2, that of a second corner to  $n$  and that of the third corner in such a way that the hyperbolic angle inequality is satisfied. One possible way would be always setting that third parameter to 7, since this count is large enough to allow all possible edge counts including  $n = 3$ .

Edge identifications can be expressed for this polygon by identifying triangles which are incident to the pair of edges in question. One triangle must be outside and the other inside the polygon. The choice of which triangle is chosen, i.e. to which half of the edge it is incident, governs the orientation of the identification, namely “same” or “opposite”.  $\square$

There are three aspects for which a TRSG doesn’t offer sufficient control. One is the order of the centers of rotation. The edge identifications indicate which corners belong to the same orbit, and the sum of all angles incident with that orbit must be an integral fraction of  $2\pi$  and will define the order of that center of rotation. So one useful tool would be something where a user can select one orbit of corners and change the order of rotation for it.

The other two aspects are metric in nature: The ratios between edge lengths as well as the distribution of the angle sum over the angles incident with a given corner orbit are both subject to modifications. They exhibit strong inter-dependencies. Nevertheless, a usable tool to control these parameters should allow modifying a single angle or edge, and adjust all the others in a suitable fashion. What exactly “suitable” means in this context still remains to be investigated.

The tools described above, namely changing orders of rotation and redistributing lengths and angles, should suffice to represent any possible cocompact hyperbolic symmetry group. Nevertheless, it might make sense to introduce additional tools, e.g. to slide identified edges against one another or to perform other high-level operations.



## Chapter 4

# Drawing hyperbolic ornaments

This section will focus on various tricks used when drawing an ornament to a screen or some other pixel-based output medium.

### 4.1 Reverse pixel lookup

#### 4.1.1 Problematic number of copies

The most obvious problem arises from the fact that the whole hyperbolic plane in all its infinity can and usually will be displayed embedded into a circle which is completely visible on the screen. A naive algorithm would start with one fundamental domain and then create copies of it one after the other. As there usually is an infinite number of such copies required to tile the visible portion of the plane, it will never actually terminate with a mathematically correct result. Of course, with displays being composed of pixels, one might terminate calculation when the copies become smaller than one pixel in size, or when they are less than one pixel from the boundary of the circle, or both. But for reasonable display sizes this still leaves far more copies than can be computed in a real time scenario. Most of the time will be spent computing transformations for copies which are too small for their detailed appearance to matter in any case, thus wasting a lot of computation time.

#### 4.1.2 General solution

The solution to this problem is reversing the direction of the control flow: instead of starting with a central domain and calculating which pixels it affects, one can start with each individual pixel and apply group elements until the location is within the central domain. The color at that location in the central domain can then be used to color the pixel chosen at the start. As this approach makes the pixels first class

**Algorithm 4.1: Reverse Pixel Lookup**

$C$  = the central fundamental domain, a colored convex polygon

- 1: **for** every pixel  $p$  inside the unit circle **do**
- 2:      $z \leftarrow$  the position of  $p$
- 3:     **while**  $z \notin C$  **do**
- 4:          $e \leftarrow$  an edge of  $C$  for which  $z$  lies on the outside
- 5:          $t_e \leftarrow$  the transformation associated with  $e$
- 6:          $z \leftarrow t_e \cdot z$
- 7:     **end while**
- 8:      $c \leftarrow$  the color of  $C$  at position  $z$
- 9:     apply color  $c$  to pixel  $p$
- 10: **end for**

citizens, instead of the fundamental domains, it will give a mathematically correct result (as much as the pixel grid approximation allows) in finite time: there is only a finite number of pixels visible on the screen, and for each pixel inside the unit circle, a finite number of group generators will relate it to the central fundamental domain.

The general idea of this approach has been outlined in algorithm 4.1. It requires the central fundamental domain to be a convex polygon, so that each edge divides the hyperbolic plane into an *inside* and an *outside* half-plane, and the fundamental domain itself is the intersection of all the *inside* half-planes. For each edge, there is an associated transformation describing the edge identification of the symmetry group. To be more precise, for an edge  $e$  of the central fundamental domain the associated transformation  $t_e$  will map the adjacent fundamental domain on the other side of  $e$  onto the central fundamental domain. The associated transformations of all edges together form a set of semigroup generators for the underlying symmetry group.

Each application of one such generator should bring  $z$  one fundamental domain step closer to the central fundamental domain, although no guarantees are made that the shortest possible path will be taken. Line 4 does not specify how the next step is chosen, but it appears best to choose an edge from which the given point has maximum distance, emphasizing the outsideness of the position with respect to that edge. Instead of exact hyperbolic distance, some value which can be calculated more easily might be chosen instead. For example, if the edge is modeled as the image of the real axis under some transformation, the way Section 2.2.5 describes, then insiderness is decided by looking at the sign of the imaginary part

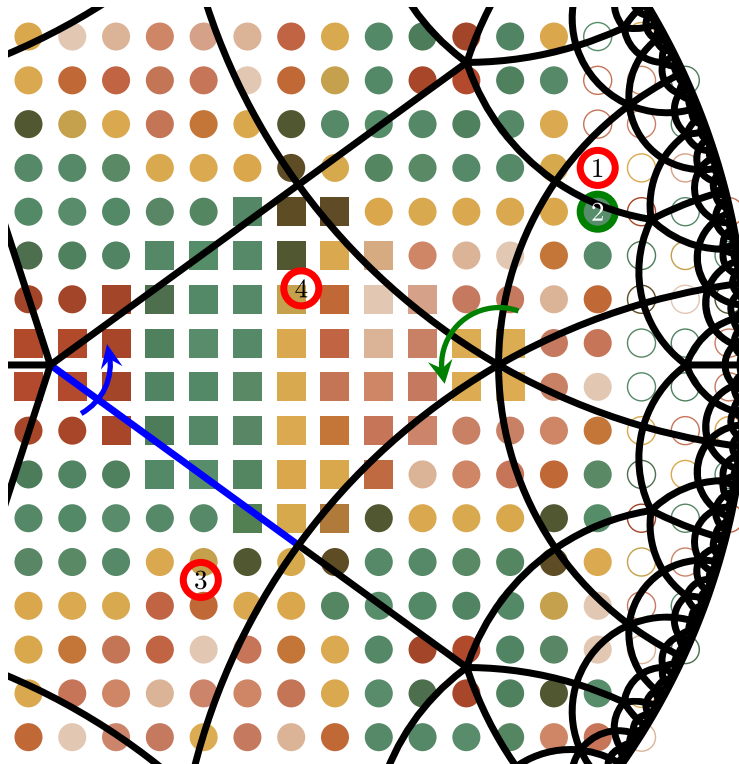


Figure 4.1: Steps of the reverse pixel lookup: (1) original position, (2) using adjacency information, (3) applying transform from adjacent pixel, (4) applying group generator associated with the blue edge.

after applying the inverse transformation to the point in question. The value of that imaginary part will give a reasonably good approximation of the actual distance, and can be obtained very cheaply. In that setup, testing insiderness and determining the edge with maximum outside distance can both be performed in a single loop over all edges.

### 4.1.3 Adjacent pixels

In the form of algorithm 4.1, there is a lot of duplicate work: if two adjacent pixels are processed one after the other, then in most cases they will belong to the same fundamental domain. For this reason, the edges chosen in line 4 will usually be the same. And even if the two pixels come from different domains, those domains will usually be close to one another, and the loop will choose different edges only at the end.

Algorithm 4.2 is an improved computation which exploits information from adjacent pixels. To speed things up, one can assume that for every pixel, the

transformation taking its predecessor to the central fundamental domain will be a good approximation of the one required for the current pixel. So one would apply that transformation to the current pixel position in line 3, right before entering the loop. To obtain a suitable transformation for the next pixel, the loop would not only have to track the current position  $z$ , but also the current transformation  $t$ , as is done in line 8.

Making use of the previously processed pixel assumes that pixels are traversed in such a way that subsequent iterations usually process adjacent pixels. But simply processing the output picture one row of pixels after the other will likely yield bad results: the closer a pixel lies to the rim of the unit disk, the more the numeric precision of the transformation will be lost due to rounding errors. As a result, it is a bad idea to start with pixels close to the rim and use them as a basis to speed up computation of pixels closer to the center. Instead, one should start reasonably close to the center and work outwards.

One way to do this is to render each quadrant of the image separately. Both rows and columns should be iterated away from the center. Taking rows as the outer loop, these should be processed upwards in the upper half, but downwards in the lower half. In the inner loop, pixels should be processed from right to left in the left half but from left to right in the right half of the resulting image. After each iteration of the inner loop, the transformation  $t$  should be reset, as the first pixel of the next row will be closer to the center than the last pixel of the previous one.

For most pixels, there will be two adjacent pixels which have already been processed, one from the previous row and one from the previous column. One might consider leveraging this information as well, by applying both these transformations in turn and using the result which lies closer to the central fundamental domain. This would require storing  $r$  transformations for the pixels of the previous row. In practice, however, the cost of always applying two transformations, together with the added code complexity, outweigh any expected performance gain, so using a single adjacent pixel is enough.

Note that algorithm 4.2 will draw the central row and column of the image twice. Shifting the coordinate system by half a pixel could easily avoid this, as could a case distinction in the quadrant rendering code. Both of these modifications would make the code much harder to read, though, so they have been omitted in this exposition here.

#### 4.1.4 Supersampling

For presentation on a computer display, pixel resolution is not enough. Just choosing one color from the central fundamental domain leads to aliasing effects which make curved lines appear rugged and generally directs attention away from

**Algorithm 4.2: Reverse Pixel Lookup using adjacent pixels**

$r =$  radius of the unit disk in pixels

```

1: function LOOKUPPIXEL( $x, y, t$ )
2:    $z \leftarrow \frac{x}{r} + \frac{y}{r}i$ 
3:    $z \leftarrow t \cdot z$ 
4:   while  $z \notin C$  do
5:      $e \leftarrow$  an edge of  $C$  for which  $z$  lies on the outside
6:      $t_e \leftarrow$  the transformation associated with  $e$ 
7:      $z \leftarrow t_e \cdot z$ 
8:      $t \leftarrow t_e \cdot t$ 
9:   end while
10:   $c \leftarrow$  the color of  $C$  at position  $z$ 
11:  apply color  $c$  to pixel  $(x, y)$ 
12:  return  $t$ 
13: end function
14:
15: function RENDERQUADRANT( $s_x, s_y$ )
16:    $t_1 \leftarrow$  identity
17:   for  $y \leftarrow 0$  to  $r - 1$  do
18:      $t_2 \leftarrow$  LOOKUPPIXEL( $0, s_y \cdot y, t_1$ )
19:      $x \leftarrow 1$ 
20:     while  $x^2 + y^2 < r^2$  do
21:        $t_2 \leftarrow$  LOOKUPPIXEL( $s_x \cdot x, s_y \cdot y, t_2$ )
22:        $x \leftarrow x + 1$ 
23:     end while
24:   end for
25: end function
26:
27: RENDERQUADRANT(+1, +1)
28: RENDERQUADRANT(-1, +1)
29: RENDERQUADRANT(-1, -1)
30: RENDERQUADRANT(+1, -1)

```

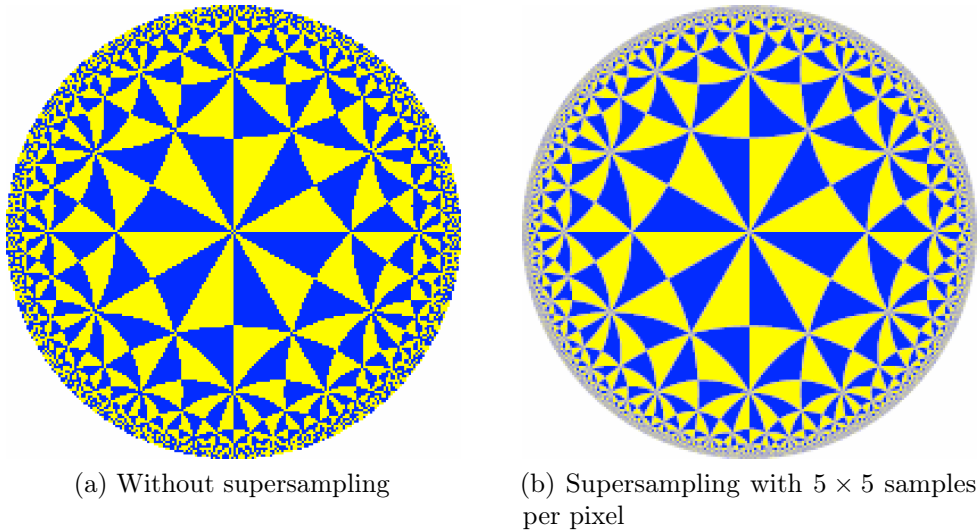


Figure 4.2: Two renderings at a low resolution of  $200 \times 200$  pixels

the displayed image and towards the medium used to display it. Users today rightly expect their graphics to be anti-aliased and appear smooth.

There are basically two techniques for anti-aliasing. One computes the area of a pixel which is covered by a given geometric primitive. This works well for cases with few simple geometric primitives, but as hyperbolic transformations will turn even simple primitives into rather complex objects, this approach is not feasible for the application at hand.

The other approach, which works much better for our purpose, is called *supersampling*. The idea is to compute color values for multiple positions inside a given pixel, and take their average to color the pixel in the final result.

This will make the image appear smoother and less noisy, particularly towards the rim. It will suggest a level of detail even past what is actually visible, thus conveying the idea of an infinite hyperbolic plane far better than an aliased image ever could.

The concept of supersampling is well established in areas like ray tracing. There, elaborate analyses have been performed in order to determine suitable positions of the individual samples within the pixel. The main reason why simple regular patterns are not desirable in that context is because they are likely to cause moiré pattern artifacts when aligned with regular elements of the rendered scene. Such regular elements are highly unlikely in a hyperbolic ornament, as the hyperbolic lines and the pixel grid lines don't run parallel over long distances. Therefore, a regular pattern works well enough for most hyperbolic ornaments.

### 4.1.5 Preprocessing

The map from result pixel locations to locations in the fundamental domain is independent from the image contained in the fundamental domain. So in scenarios where this image is expected to change over time, some performance can be gained by computing this map only once, and using it repeatedly to copy pixel colors. Updating the target image then only requires a fixed and rather small amount of time. Prominent use cases for this kind of preprocessing include real-time drawing of ornaments with live updates to the resulting image, but also hyperbolizations (in the sense of Chapter 5) of live video data, e.g. from a webcam or similar.

Of course, this gain in performance comes at a cost: for every pixel of the resulting image, an index into the data structure representing the source fundamental domain has to be stored. If supersampling is used, as described in Section 4.1.4, then the number of indices to be stored increases accordingly. For large destination images and high degrees of supersampling, the memory demands can become prohibitive.

When updates to the image are localized (e.g. drawing of individual lines as opposed to a webcam image which will change over its whole area), some performance can be gained by storing the inverse of the mapping as well: for every pixel in the central fundamental domain, store a list of corresponding sample positions in the whole ornament. When a pixel in the central fundamental domain changes its color, iterating over the attached list will identify all the visible pixels which are affected by this and need to be updated.

Applications might want to perform operations on the ornament which change the map between pixels instead of the content of the fundamental domain. The most likely example is an application that allows the user to drag the hyperbolic plane, thus conjugating the symmetry group with an arbitrary translation. In those cases, the preprocessing approach cannot be used to yield exactly the same results as a full Reverse Pixel Lookup on every pixel would compute. So one possible solution is to discard the old preprocessing data, and either compute the new map or fall back to the iterative pixel lookup without preprocessing.

In cases where the time consumed by either of these solutions is unacceptably long, the outdated preprocessing data might instead be used as a temporary approximation, until the updated map has been computed. So in the scenario of the user dragging the view of the hyperbolic plane, the global translation would be applied to every sample location, and the resulting location would then be used to look up an index in the preprocessed map. As the input position to the map has to be rounded to the nearest sample position, that step would introduce some positional error, reducing the quality of the resulting image. But as long as the number of supersamples is sufficiently high, and the global transformation is a translation by a sufficiently small distance, the overall effect of this error will not

become too pronounced.

In order to keep the translation distance low, it is important to leverage the symmetries of the ornament. Applying elements of the symmetry group, it is always possible to choose the global transformation in such a way that the central fundamental domain and its image under that transformation overlap, thus limiting the maximum distance to approximately the diameter of the central fundamental domain.

## 4.2 OpenGL GPU implementation

Modern graphics processing units (GPUs) provide an interesting alternative to implementing the Reverse Pixel Lookup algorithm on the CPU. There are some important differences between computations performed on the CPU and the GPU.

**Parallelism.** While a typical CPU today contains between one and eight cores, a GPU may have over a thousand fragment shaders working in parallel.

**Speed.** While the number of operations per second performed by a CPU core is much higher than the same figure for a GPU shader unit, the sheer number of shader units makes GPU computations potentially a lot faster, for applications that can work well in parallel.

**Data Dependencies.** As the GPU computes colors for multiple pixels in parallel, it is impossible for one such computation to use the result of a preceding one. The algorithm must take this into account to be efficient.

### 4.2.1 Anatomy of a 2D shader transformation

OpenGL[42] can be used to access the GPU using an open standard. As OpenGL is usually associated with rendering scenes in three dimensions, it might not be readily apparent how that can be used for two-dimensional computations. Getting rid of the third dimension is pretty easy, though: simply use a scene description consisting of a single flat surface (usually a quad) to fill the visible area, and have the camera look directly at it using an orthogonal projection. Now the surface of this primitive will directly correspond to the content of the viewport.

In a primitive scene, this surface would simply show some interpolation of the colors associated with its vertices, using the default functionality provided by 3D graphics cards for a long time now. In order to generate the detailed appearance of the primitive on the fly, more recent graphics hardware is required, which provides



a programmable fragment shader.\* The fragment shader is one of the final stages of processing. For every sample (usually pixel) position a geometric primitive (like this quad) touches, the fragment shader computes the associated color as it appears to the user. So in effect, a shader program will be called once for every pixel of the window, and can compute the color of that pixel.

When working with OpenGL, the shader program is written in the OpenGL Shader Language (GLSL), a C-like language designed for this purpose. The source code is sent to the graphics driver which then compiles it for execution on the available hardware. The facilities available to a shader program differ very much from those of most programs written in general purpose programming languages. There usually is no way to e.g. access files on disk or print custom error messages. All input variables to the fragment shader must explicitly be set by the application.

In the case of a hyperbolic ornament, input usually consists of a few simple data structures describing the group generators, and an image providing the content of the central fundamental domain. That image is made available to the shader program as a texture. The program may retrieve the color for any location within this texture, so once the correct position has been determined, copying the color is straightforward.

In addition to the change of language, there are some other modifications to be made in order to adjust the Reverse Pixel Lookup algorithm, as described in previous sections, to execution on the GPU. The following sections will investigate how the techniques introduced in Section 4.1 can be applied to GPU computations.

## 4.2.2 Adjacency revisited

First of all, the fragment shader runs the same program independently for every pixel. So there must be no top level loop iterating over all the pixels; that part is taken care of by OpenGL.

While Section 4.1.3 argued that a lot of work could be saved by reusing data from adjacent pixels, this benefit cannot be exploited on the GPU. As it processes pixels in parallel, the computations for them are executed independently, so there is no way to pass information from the computation of one pixel to that of another one.

On the other hand, when using supersampling as described in Section 4.1.4, the information from adjacent samples may be re-used. For this reason, it is better to compute all the samples of a pixel in a single run of the custom shader program, exploiting that adjacency information. OpenGL has a mode to perform

---

\*The use of custom shader programs is specified in version 2.0 of the OpenGL specification, dating from 2004. Before that version, extensions had to be used for this purpose.

supersampling itself, but that would cause independent runs of the whole shader program, with no adjacency information carried over.

### 4.2.3 Preprocessing revisited

It is possible to access preprocessed data from a shader program as well. For every sample position, the corresponding location in the central fundamental domain can be stored in a second texture. The texture should have a suitable internal format, e.g. two 16-bit integer components, so that it is possible to store an arbitrary input pixel address for each output sample position.

It is important that this lookup texture is accessed using nearest-neighbor-interpolation. At the boundary between two fundamental domains, adjacent sample positions for the output image might correspond to completely different (e.g. opposite) positions in the input image of the fundamental domain. Interpolating between these positions would lead to some pixel in between being chosen, where either boundary choice would have been a much better match as it lies closer to the intended orbit.

The generation of the lookup table itself can be done on the GPU as well. To do so, the fragment shader will be used to render not directly to the screen but instead to the image of the lookup texture, which for this purpose would be attached to an off-screen framebuffer. The changes to the shader code required to output a position instead of the texture color at said position would be minimal.

## 4.3 Reified and virtual triangles

There are several occasions when the user interface needs to deal with the triangles of the underlying triangle reflection group. The most obvious case is during group definition, when the user has to enter triangles which are to be identified. But even if the group definition is complete, filling the triangles in different colors to denote domains or orbits will show details about the structure of the symmetry group.

The following text will sketch several approaches to this problem. Even though new implementations should probably rely on the last approach, using virtual triangles only, the intermediate approaches were historically used by the author in various implementations. Learning from their drawbacks might be instructive.

### 4.3.1 Reified triangles

The most straight-forward idea is having all those triangles reified, i.e. represented as actual objects in memory. Of course, “all” is too strong a term in this context, as an infinite number of hyperbolic triangles is embedded in the finite unit disk

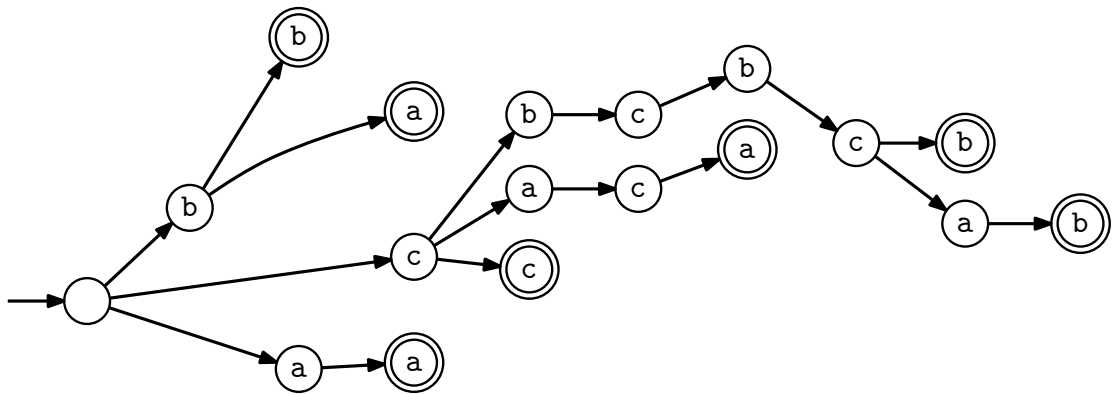
which represents the hyperbolic plane. So one has to somehow limit the depth of the enumeration, in the hope that the missing triangles will be sufficiently small so they won't be noticed. The number of triangles required for this might well require significant amounts of memory.

But even enumerating a finite subset of all the triangles can be a difficult task. Starting from one central triangle, each triangle of the reflection group has an associated transformation which maps the central triangle to that given triangle. But due to rounding errors, different paths which should lead to the same triangle will lead to slightly different transformation matrices. Therefore, using these transformations as keys in some kind of equality-based dictionary like a hash table will result in triangles being enumerated repeatedly. So instead of the equality-based comparison, a similarity-based one is required. One might use some kind of multi-dimensional binary tree to accomplish this, e.g. a so-called QuadTree.

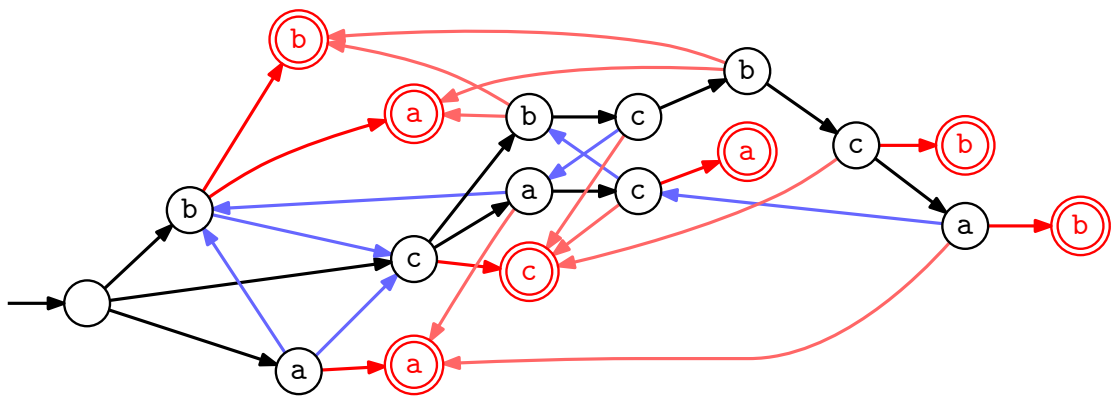
Nevertheless, the deeper the enumeration proceeds, the harder it becomes to distinguish numerical rounding errors from actual differences between triangles. This can lead to strange artifacts. One way to avoid most of these is to keep track of the adjacency between triangles, and use that information to decide equivalence on a combinatoric level. In such an approach, each triangle has pointers to the three neighboring triangles. For every new triangle found by the enumeration, those pointers should be set if the corresponding neighbors already exist. For the link to the direct parent triangle in the enumeration tree, this is easy. But if the corner angles and the orders of rotation they imply are known during enumeration, then it becomes possible to detect situations where the newly found triangle is the last one incident with a given corner. In that case, adjacency links to triangles other than the enumeration parent can be filled in as well. Only directions for which no adjacent triangle could be found in this way should be considered for further enumeration.

### 4.3.2 Automatic enumeration

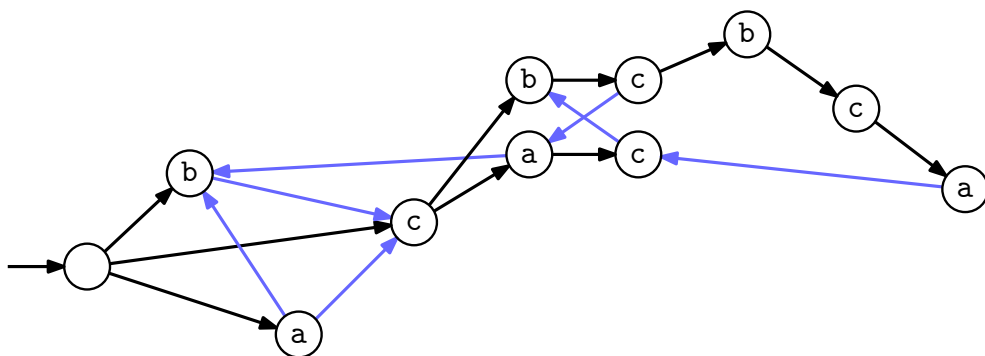
There is a faster way to enumerate all the triangles of a triangle reflection group. Based on the ideas of Section 3.2.1, one can formulate a triangle rewriting system for the reflection group, and complete it using the Knuth-Bendix procedure described in Section 3.2.3. This system will compute a canonical label for every triangle, so in order to obtain an enumeration process which traverses each triangle exactly once, one could enumerate those canonical labels. A name is canonical if and only if it doesn't contain any left-hand side word of the complete triangle rewriting system as a substring. Using parallel matching methods described by Aho and Corasick in [1], one can construct a deterministic finite automaton which will detect those left-hand sides. So this automaton will enter an accept state if and only if its input is non-canonical. Inverting the acceptance pattern, one obtains an automaton



(a) Matching all left sides



(b) Back edges inserted



(c) Accepting states removed

Figure 4.3: Construction of an enumeration automaton

which accepts only canonical words. One can even replace the non-accepting states with a single fail state, or remove them altogether. Run in generating mode, this automaton will emit all canonical triangle labels. A breadth-first search will enumerate them ordered by word length, so that triangles closer to the central triangle will be found before those closer to the rim, and an interrupted search of this otherwise infinite traversal will yield a reasonable subset of the infinite set of all triangles.

Here is an example to illustrate this. It is for the triangle group  $(2, 4, 6)$ , which can also be described as a Coxeter group like this:

$$T = \langle a, b, c \mid 1 = a^2 = b^2 = c^2 = (ab)^2 = (ac)^4 = (bc)^6 \rangle$$

The complete rewriting system has already been given in Equation (3.4) and is repeated below:

- (i)  $aa \rightarrow \varepsilon$
- (ii)  $bb \rightarrow \varepsilon$
- (iii)  $cc \rightarrow \varepsilon$
- (iv)  $ba \rightarrow ab$
- (v)  $caca \rightarrow acac$
- (vi)  $cbcbcb \rightarrow bcbcbc$
- (vii)  $cbcbcab \rightarrow bcbcbca$

To only enumerate ShortLex labels of triangles, none of the left sides must occur in the labels of these triangles. So all the left side words are combined into a single automaton, like Figure 4.3 (a) depicts. In the next step, backward-pointing edges are introduced, searching for a word which started later. After all accept states are removed, the resulting automaton as depicted in (c) will only enumerate ShortLex-minimal labels, and therefore touch each triangle exactly once.

### 4.3.3 Virtual triangles

If one wants to avoid both the memory consumption caused by reified triangles as well as the numeric problems, one can instead use an approach similar to the reverse pixel lookup used for the artistic content of the ornament. For each pixel on the screen, the reverse pixel lookup will compute a sequence of transformations which will map it to a position inside the central domain. If one considers the full triangle reflection group, then that central domain will be the central triangle. The sequence of transformations applied will identify the triangle in that triangle reflection group. It can be considered a word in the corresponding finitely represented group.

The word derived in this fashion will not be uniquely defined. Different pixels of the same triangle might be transformed in different order, and might therefore be identified by different but equivalent words in the group. In cases where a unique identifier is required, a term rewriting system can be used to compute it.

The triangles in this approach are “virtual” as they are identified only during the computation of a single pixel. There is no persistent object to represent a triangle after that computation is complete. Because of this, it might be necessary to repeat computations for each pixel in a triangle, but unless those computations are really expensive, this will still be faster than handling the large amounts of memory required for reified triangles.

## 4.4 Grid lines in hyperbolic views

As outlined above, the user should be able to choose a triangle from a hyperbolic triangle tiling. The most suitable way to present this triangle tiling is using a grid, outlining the boundaries between triangles. The question what line width to use for these edges is quite a subtle one.

**A hyperbolic brush** would become thinner towards the rim, in correspondence with the hyperbolic distance metric. Practical observations show that such lines are unsuitable over the whole area of the image. Choosing a thick brush, the lines will be too thick near the center of the disk, where they appear bulky and are perceived more as filled areas than as lines. Choosing a thinner brush will result in the pattern becoming indistinguishable too soon, at positions still quite far away from the rim of the disk.

**An Euclidean brush** would usually be chosen such that its lines are as thin as possible while remaining clearly distinguishable. This gives good results in the inner area of the image, but at the very rim, the lines are thicker than the tiles they delineate, giving a cluttered appearance.

The two methods can be combined to profit from their individual strengths and mitigate their weaknesses. Towards the rim, a (rather thick) hyperbolic brush would be the way to go, giving a proper thinning of the lines and suggesting the infinity of tiles at the boundary. Towards the center, an Euclidean pen would be preferable, in order to limit the maximum line thickness and maintain the perception of the lines as lines, not filled areas.

Sudden switches from one pure mode to the other at a fixed distance from the center of the image might cause inconsistencies in appearance near the switch-over. So instead of using one or the other, a weighted sum can be used. To formulate this concept, let us consider the general setup of any simple line width, at first without

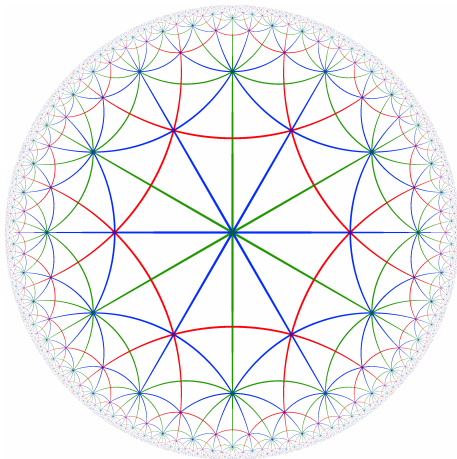
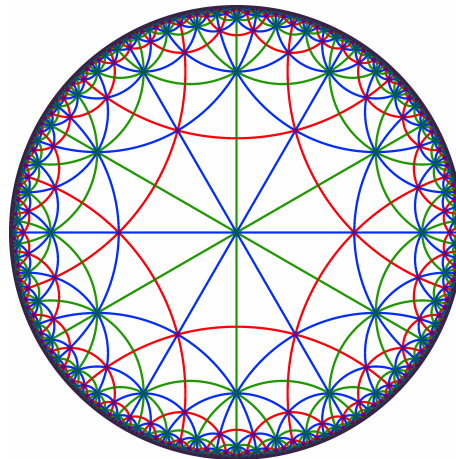
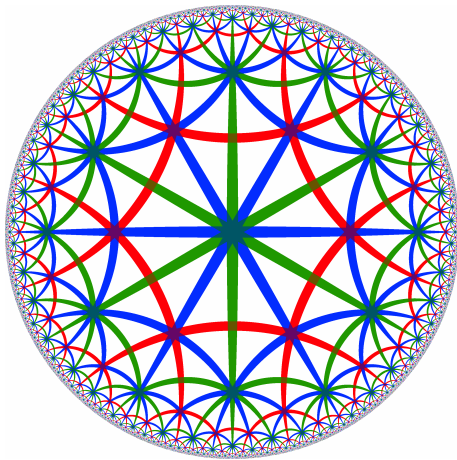
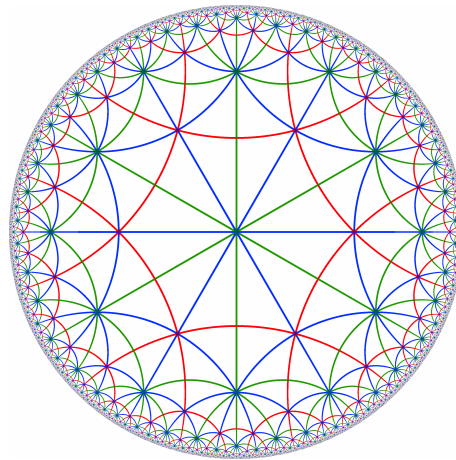
(a) A thin hyperbolic brush;  $w_h = 0.02$ (b) An Euclidean brush;  $w_E = 0.01$ (c) A thick hyperbolic brush;  $w_h = 0.1$ (d) A mixed brush;  $w_E = 0.01, w_h = 0.1$ 

Figure 4.4: Grids drawn with different brushes

reference to any specific geometry. One way to think about brushes with a given line width is in terms of coloring all the points which are no more than a certain distance away from a given mathematical line. The distance between point and line is usually derived from the distance between a pair of points. One common choice uses the orthogonal projection, but expressing that in terms of hyperbolic geometry is rather complicated\*. What we already have is a concept of reflections along a line, leading to the following alternate definition:

**Definition 4.1: Distance between point and line**

The distance between a point and a line is half the distance between the point and its reflection in the line.

This definition can be applied directly to hyperbolic geometry. To compute an Euclidean distance to a hyperbolic line, one needs to mix concepts, doing a hyperbolic reflection but an Euclidean distance measurement. As long as the radius of the circle used to model the hyperbolic line is much larger than the distance to be measured, this is a fair approximation of the distance between a point and a circle. The factor two in the definition is in fact a good thing, as the width of a drawn line is measured from one outline boundary to the other, not to the central line. So if  $d$  is the distance between a point and its reflection, than the point lies inside the brushed line, and will be colored accordingly, if that distance is less than the line width.

$$d < w \implies \frac{1}{w}d < 1$$

In the case of a single mode brush, the distance  $d$  would be either the Euclidean or the hyperbolic distance between the two points. In a combined brush, we can use the weighted sum of both:

$$\frac{1}{w_E}d_E + \frac{1}{w_h}d_h < 1$$

Towards the rim, where  $d_h \gg d_E$ , the hyperbolic width  $w_h$  imposes an upper limit on the line width and causes the lines to appear infinitesimally thin in the limit case, as one would expect in a hyperbolic image. Near the center (and if the Euclidean coordinate system is scaled in such a way that the circle is actually the unit circle), hyperbolic distance metric using Equation (2.1) almost matches the Euclidean one up to a constant factor, and we have  $d_h \approx 2d_E$ . In that case, the combined line

---

\*Probably the easiest way would be switching to the Klein-Beltrami model using Equation (2.4), where an orthogonal projection onto one axis can be done by simply setting one coordinate to zero.



width is limited above by  $w_E$ . This prevents lines from becoming excessively thick in the central area of the image.



## Chapter 5

# Hyperbolization of ornaments

This section makes heavy use of several terms which have been introduced in Section 1.2. Knowledge of these terms is assumed, so if the precise meaning of some term seems unclear, the reader should revisit that section for reference.

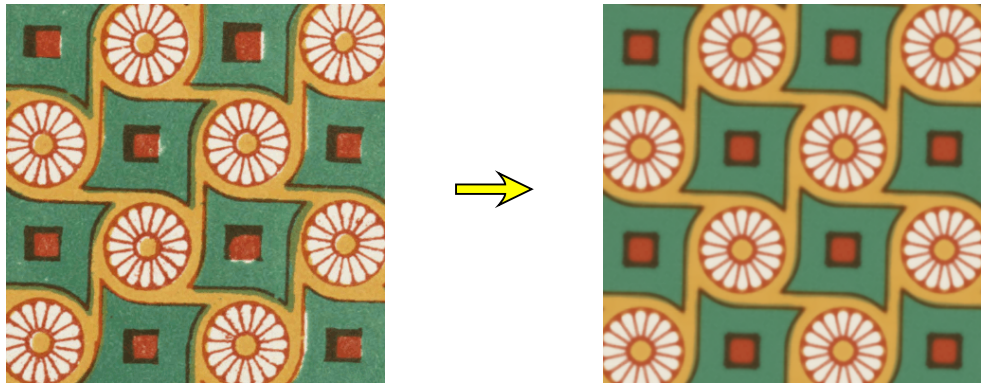
### 5.1 The big picture

One very interesting method to obtain a hyperbolic ornament is the conversion of an Euclidean ornament, which has one of the 17 wallpaper groups as its symmetry group, into a hyperbolic ornament. This process has been published in some detail in [22], so only selected aspects will be outlined below.

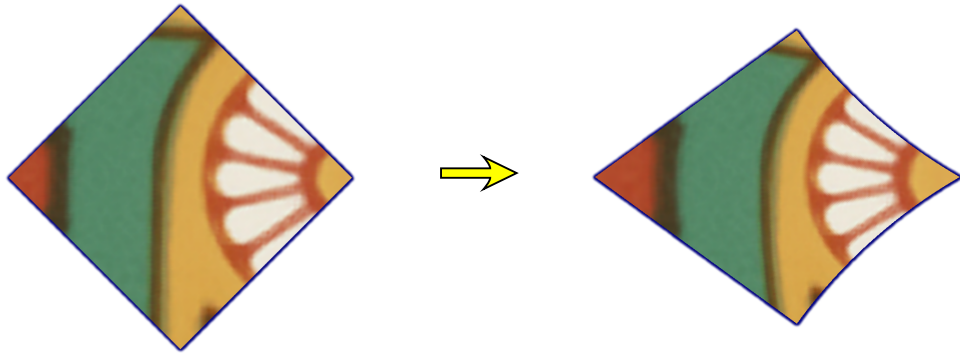
#### Definition 5.1: Hyperbolization

Let  $(P, G)$  be an Euclidean ornament with picture  $P : \mathbb{E}^2 \rightarrow C$  and a discrete symmetry group  $G \subset \text{iso}(\mathbb{E}^2)$ . Likewise let  $(P', G')$  be a hyperbolic ornament with picture  $P' : \mathbb{H} \rightarrow C$  and a discrete symmetry group  $G' \subset \text{iso}(\mathbb{H})$ . Let  $(\tilde{P}, O)$  be the colored orbifold corresponding to  $(P, G)$  and likewise  $(\tilde{P}', O')$  for  $(P', G')$ .  $(P', G')$  is called a *hyperbolization* of  $(P, G)$  iff there exists a homeomorphism  $f$  between  $O$  and  $O'$  such that  $\forall x \in O : \tilde{P}(x) = \tilde{P}'(f(x))$ . The function  $f$  will be called the homeomorphism of the hyperbolization.

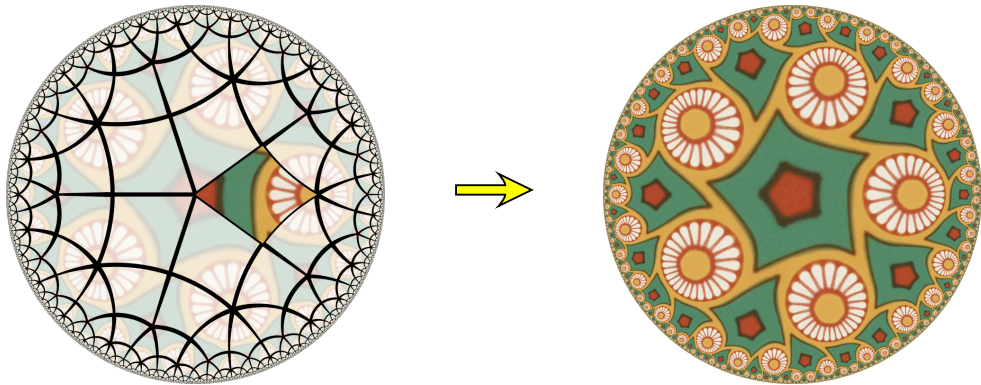
The above definition is pretty broad, since the requirements for  $f$  are rather weak. Due to this, the definition covers various possible approaches, but leaves a lot of room for arbitrary decisions. It turns out that a conformal deformation of the orbifold results in a hyperbolic ornament which most subjects will perceive as being closely related to its Euclidean original. At the same time, requiring conformality provides a nice mathematical framework, getting rid of a number of arbitrary



(a) Pattern recognition



(b) Conformal deformation



(c) Output rendering

Figure 5.1: Steps of the hyperbolization process

decisions and ensuring uniqueness of the solutions to several sub-problems which will be described later on. In a certain sense, conformal maps provide a natural glue between the Euclidean plane and the Poincaré models of hyperbolic geometry: angles are measured the same way in both, so they form a kind of common ground while lengths have to change. Therefore, this work will be dealing with *conformal hyperbolizations*.

### Definition 5.2: Conformal hyperbolization

A hyperbolization is called a *conformal hyperbolization* iff its homeomorphism  $f$  is conformal everywhere, with the possible exceptions of those points which are centers of rotation in either ornament.

The process of creating a conformal hyperbolization with the aid of a computer program as implemented by the author can be summarized as follows. Some of the crucial steps have been illustrated in Figure 5.1.

**Input image:** Starting point is a raster image of the original Euclidean ornament, possibly with imperfect symmetry.

**Pattern recognition:** The symmetry group of the ornament is determined using autocorrelation. Using that symmetry group, a fully symmetric fundamental domain can be computed.

**Choice of group:** The user has some choice about the way in which the hyperbolization should be performed. In the most common cases, this is done by selecting new orders for the centers of rotation present in the ornament. In more complicated cases, some point has to be designated as a new center of rotation.

**Conformal deformation:** The shape and coloring of a single fundamental domain of the resulting ornament is computed, using a discrete conformal map.

**Output image:** The raster image for the output is generated using the reverse pixel lookup described in Section 4.1.

## 5.2 Pattern recognition

The pattern recognition is based on autocorrelation. This is first used to identify possible displacement vectors and thus identify the translational subgroup of the symmetry group. Once that is done, a single translational cell, averaged from

the whole ornament, can be glued to torus (which agrees nicely with the periodic nature of FFT) and then analyzed for further features like centers of rotation and the likes. The method is based on [32]. Details about this process were described in the diploma thesis of the author and shall not be reproduced here in full.

The main goal of this pattern recognition is obtaining a really symmetric ornament, even if the input is not fully symmetric like the one shown in Figure 5.1 (a).

### 5.3 Choice of group

The most obvious method of finding a hyperbolic symmetry group suitable for a hyperbolization of a given Euclidean ornament is adjusting the orders of its centers of rotation. Since this is not the only possible form of a conformal hyperbolization, it is useful to introduce a term for this kind of hyperbolization.

#### **Definition 5.3: Faithful hyperbolization**

Let  $f$  denote the homeomorphism of a conformal hyperbolization. If  $f$  maps corners of the Euclidean orbifold  $O$  to corners of the hyperbolic orbifold  $O'$ , and likewise cone points of  $O$  to cone points of  $O'$ , then this hyperbolization will in the context of this work be called a *faithful hyperbolization*.

Expressed in terms of the ornament, this has the consequence that centers of rotation are mapped to centers of rotation, albeit the order of the rotation may change. One obvious precondition necessary for such a hyperbolization is the existence of centers of rotation in the Euclidean ornament. This means that only 13 of the 17 Euclidean wallpaper groups can be hyperbolized in this fashion.

A conformal hyperbolization which is not faithful will introduce new centers of rotation (i.e. transforming a “one-fold rotation” into a proper rotation), remove such centers (by changing their order to one), or both. Such modifications can cause a quite drastic departure from the structure of the original ornament, and as such should be used sparingly. The subsequent paragraphs will restrict discussion of non-faithful hyperbolizations to those cases where adding a single center of rotation will enable the hyperbolization of an ornament which did not contain any centers of rotation to start with.

In the faithful cases, the choices made by the user are purely combinatoric: for every orbit of rotation centers in the existing ornament, the new order has to be selected. This is either an integer  $n$ , corresponding to a  $n$ -fold rotation, or the

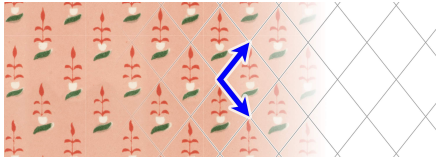
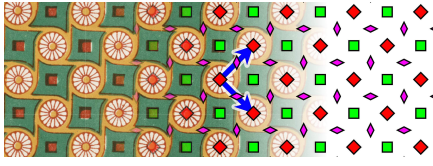
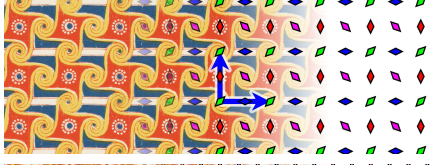
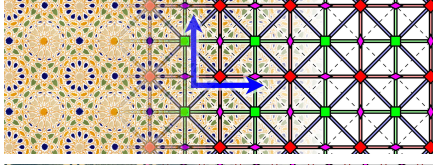
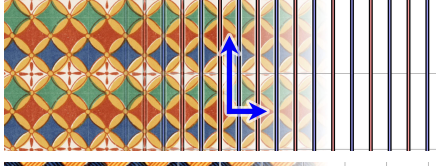
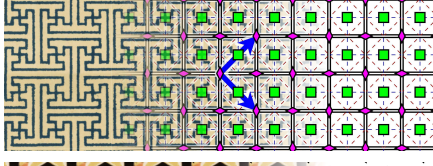
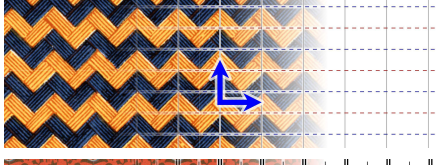
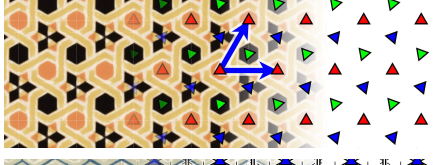
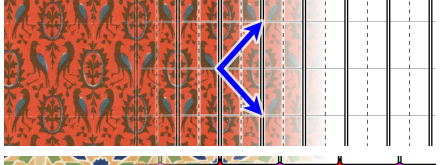
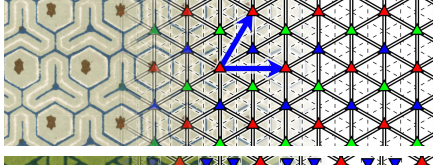
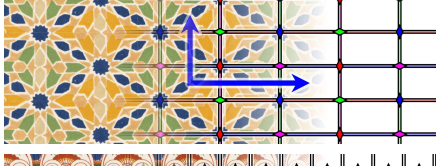
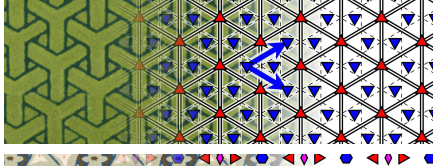
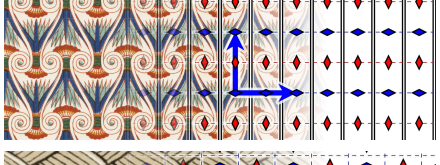
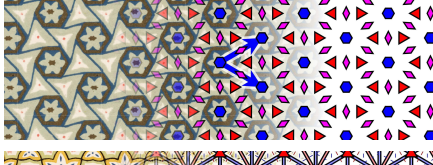
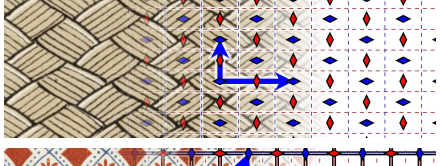
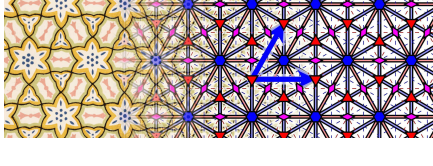
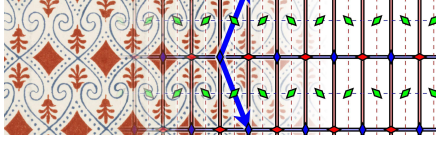
o		442	
2222		*442	
**		4*2	
x x		333	
*x		*333	
*2222		3*3	
22*		632	
22x		*632	
2*22			

Table 5.1: Euclidean ornaments, their wallpaper groups and orbifold symbols

character	geometric	topological	cost
*	reflection	boundary comp.	1
$\times$	glide reflection	crosscap	1
$\circ$	translation	handle	2
$n$ (after *)	rotation on axis	corner	$\frac{n-1}{2n}$
$\infty$ (after *)	of reflection		$\frac{1}{2}$
$n$ (before or without *)	rotation not incident	cone point	$\frac{n-1}{n}$
$\infty$ (before or without *)	with a reflection		1

Table 5.2: Characters of an orbifold symbols

special value  $\infty$ , corresponding to a limit rotation where the center of rotation lies on the boundary of the unit disk and the angle of rotation is zero.

The chosen orders have to satisfy some inequality in order to ensure compatibility with hyperbolic geometry. One way to formulate this condition is using the *orbifold symbol*. [14] This is a sequence of characters which encode the structure of the orbifold, and hence of the symmetry group. Here is a short guide to how such a symbol is formed. A positive natural number or the symbol  $\infty$  is used to denote a center of rotation. A  $*$  represents a boundary component of the orbifold, which corresponds to a reflection in the ornament. If a center of rotation (i.e. a number or  $\infty$ ) is listed after a  $*$ , then it lies on that boundary and hence corresponds to a corner of the orbifold. Otherwise it is in the interior and therefore a cone point. The symbol  $\times$  is used to denote a cross cap, corresponding to glide reflections in the ornament, whereas the symbol  $\circ$  expresses a handle of the orbifold, roughly corresponding to purely translational symmetry. Each of these parts of the orbifold symbol comes with a certain “cost”, as summarized in Table 5.2.



**Theorem 5.1: Orbifold curvature**

If each part of an orbifold symbol is associated with a cost according to Table 5.2, then the sum of these costs will be equal to 2 for Euclidean ornaments, but will be greater than 2 for hyperbolic ones and smaller for spherical ones.

Conway, Burgiel and Goodman-Strauss give a very accessible proof for this statement.[14] The essence of it is that this sum of costs corresponds to the Euler characteristic of the orbifold, which in turn indicates the sign of its curvature.

So the new orders for the centers of rotation have to be chosen in such a way that the total cost of the resulting orbifold symbol is greater than 2. This will automatically result in a fundamental domain which satisfies the hyperbolic angle sum inequality. In general, the orders will increase with respect to their original Euclidean values, corresponding to a decrease in angles of rotation. As long as the order of *some* rotation centers increases sufficiently, the order of others may remain the same, or even decrease.

In order to allow a hyperbolization of a Euclidean ornament which does not exhibit any rotational symmetry, it is necessary to use a non-faithful hyperbolization, and introduce at least one center of rotation. The user is allowed to choose a point of the ornament for this, subject to certain constraints which depend on the Euclidean symmetry group.

## 5.4 Conformal deformation

The choices made by the user are sufficient to uniquely determine the group of the hyperbolic ornament, up to a global isometry of the whole ornament. However, they do not directly give that group, in terms of e.g. locations for the singular points within a single fundamental domain. For some groups, determining such coordinates is pretty easy, but for others, the final shape of the fundamental domain will only be found during the conformal deformation process.

After discussing uniqueness of possible hyperbolizations, their existence will be investigated in this section. This is mostly done assuming perfect mathematical operations, with little regard to actual numerical implementations. Section 5.5 will then give details on how these hyperbolizations can actually be computed numerically.

### 5.4.1 Uniqueness

#### Theorem 5.2: Unique hyperbolization

Let  $(P, G)$  denote a Euclidean ornament with a cocompact symmetry group, and  $G' \subset \text{iso}(\mathbb{H})$  be a hyperbolic symmetry group. The corresponding orbifolds shall be denoted as  $O$  resp.  $O'$ . Let  $r \subset O \times O'$  describe those points which are centers of rotation in either orbifold, and how these should correspond to one another.

Then there can be at most one conformal hyperbolization  $(P', G')$  with an associated homeomorphism  $f : O \rightarrow O'$  such that  $\forall (x, x') \in r : f(x) = x'$ .

In other words, once you have fixed the hyperbolic symmetry group, and stated which centers of rotation will get mapped to which (with one of them possibly a center of one-fold rotation), then a resulting conformal hyperbolization is unique if it exists at all. Since in many cases the hyperbolic symmetry group is fixed by the user's choices, at least up to isometry, this means that in those cases the user's choices uniquely determine the hyperbolization, again up to isometry.

*Proof:* Suppose there were two different hyperbolizations matching the stated requirements. This would mean two different conformal maps from the Euclidean orbifold  $O$  to the same hyperbolic orbifold  $O'$ . The hyperbolic orbifold is the same since it is completely determined by  $G'$ , which is part of the preconditions of this theorem. Denote these two conformal maps as  $f_1$  and  $f_2$ . Then the homeomorphism

$$\varphi : O \rightarrow O \qquad \varphi := f_2^{-1} \circ f_1$$

maps the Euclidean orbifold onto itself. The remainder of this proof will have to demonstrate that  $\varphi$  is necessarily the identity map.

Since both  $f_1$  and  $f_2$  are conformal except at the centers of rotation in either ornament,  $\varphi$  will be conformal as well, with the possible exception of the denoted centers of rotation. This set of possibly non-conformal points is simply the first half of the rotation correspondence  $r$ . It will be written as

$$s \subset O \qquad s := \{a \mid \exists b : (a, b) \in r\}$$

One can cut open the orbifold to obtain a polygonal fundamental domain  $F$  in the Euclidean plane, which will be treated as the complex number plane here:  $F \subset \mathbb{C}$ .

$$\omega : (F \setminus \partial F) \rightarrow O \qquad \omega(z) := Gz$$

shall denote the map from the interior of the fundamental domain to the orbit, and hence to the orbifold. It is conformal and has an inverse  $\omega^{-1}$ . It can be extended to cover the whole complex plane:\*

$$\tilde{\omega} : \mathbb{C} \rightarrow O \qquad \tilde{\omega}(z) = Gz$$

This function  $\tilde{\omega}$  can also be obtained from  $\omega$  via analytic continuation along any path which avoids centers of rotation on the Euclidean orbifold. For normal points in the inside of the orbifold, this follows from the fact that a neighborhood around them is conformally equivalent to a region of the complex plane. For points on the boundary, the Schwartz reflection principle ensures that the analytic continuation in the plane matches the reflection denoted by the group. So with the exception of the denoted centers of rotation,  $\tilde{\omega}$  is conformal.

There exists an open subset  $U \subseteq O$  of the orbifold which does not contain any center of rotation, which itself does not intersect any of the cuts that turned  $O$  into  $F$ , and whose image  $\varphi U$  does not intersect any cut either. This gives rise to a conformal function

$$\Phi : \omega^{-1}U \rightarrow F \qquad \Phi := \omega^{-1} \circ \varphi \circ \omega$$

So  $\Phi$  takes a point from the fundamental domain, maps it to the orbifold, applies  $\varphi$  to it and maps it back into the plane. Those points in the Euclidean plane which correspond to centers of rotation in either ornament shall be denoted as  $S$ :

$$S \subset \mathbb{C} \qquad S := \{z \in \mathbb{C} \mid \omega(z) \in s\}$$

Using analytic continuation along paths in  $\mathbb{C} \setminus S$ ,  $\Phi$  can be extended to a function  $\tilde{\Phi}$  on  $\mathbb{C} \setminus S$ . As a result,  $\tilde{\omega} \circ \tilde{\Phi}$  is the analytic continuation of  $\omega \circ \Phi = \varphi \circ \omega$ . But the analytic continuation of  $\varphi$  along such paths is  $\varphi$  itself, since the path never leaves its domain. And the analytic continuation of  $\omega$  is  $\tilde{\omega}$ . This yields

$$\tilde{\omega} \circ \tilde{\Phi} = \varphi \circ \tilde{\omega}$$

---

\*A note about notation: tildes will be used to denote analytic continuations, while lower case and upper case letters are often orbifold and plane representations of the same idea.

This guarantees that the following diagram commutes:

$$\begin{array}{ccc} \mathbb{C} \setminus S & \xrightarrow{\tilde{\omega}} & O \setminus s \\ \downarrow \tilde{\Phi} & & \downarrow \varphi \\ \mathbb{C} \setminus S & \xrightarrow{\tilde{\omega}} & O \setminus s \end{array}$$

To phrase this verbally: it makes no difference whether we switch from the plane to the orbifold and then execute transformation  $\varphi$  on the orbifold, or whether we execute its counterpart  $\tilde{\Phi}$  in the plane and then switch to the orbifold.

In terms of the orbit of the result, it makes no matter which element of a given orbit is used as the input to  $\tilde{\Phi}$ :

$$\forall z \in \mathbb{C} \setminus S \forall g \in G: \quad \tilde{\omega}(\tilde{\Phi}(g(z))) = \varphi(\tilde{\omega}(g(z))) = \varphi(\tilde{\omega}(z)) = \tilde{\omega}(\tilde{\Phi}(z))$$

But we will need an even stronger condition: we want to show that  $\tilde{\Phi}$  commutes with elements of  $G$ , or in other words, that different elements of the same orbit get mapped in a parallel fashion.

To obtain that, consider a closed analytic path  $p: [0, 1] \rightarrow O \setminus \partial O \setminus s$  which lies fully within the non-singular points of the orbifold  $O$ . That path will correspond to multiple analytic paths in the plane, two of which shall be denoted as  $P, P': [0, 1] \rightarrow \mathbb{C}$ . These two paths satisfy the equation

$$\forall t \in [0, 1]: \tilde{\omega}(P(t)) = \tilde{\omega}(P'(t)) = p(t) \quad (5.1)$$

so for every parameter  $t$ , the corresponding points  $P(t)$  and  $P'(t)$  will belong to the same orbit.

For small steps  $\Delta t \ll 1$ , there is a disk of sufficiently small radius  $\varepsilon > 0$  around  $P(t)$  such that exactly one point inside that disk belongs to the orbit  $p(t + \Delta t)$ . That point will be defined as the position of  $P(t + \Delta t)$ . An exception to this rule would be a point  $P(t)$  located on a line of reflection, but since  $p$  does not encounter the boundary of  $O$ ,  $P$  will not cross any lines of reflection. So knowing  $p$  and knowing  $P(t)$  for a single value of  $t$ , one already has full knowledge of  $P$ .

The two paths  $P$  and  $P'$  start in points from the same orbit, so  $P'(0) = g(P(0))$  for some  $g \in G$ . If  $P'$  were equal to  $g \circ P$ , then Equation (5.1) would be satisfied. And since we just argued that the path  $p$  on the orbifold and the starting point  $P'(0)$  already fully defines that path  $P'$ , that path has to be equal to  $g \circ P$ . So any two analytic paths in the plane which do not encounter any singular points and which map to the same path on the orbifold will necessarily differ by the same symmetry group element  $g$ .

Now take a concrete path  $P$  with  $P(1) = \tilde{\Phi}(P(0))$ . This path is a continuous description of the effect of  $\tilde{\Phi}$ . Since  $\tilde{\Phi}$  was defined to avoid the boundary of the

orbifold, so in the domain of  $\Phi$  such a path  $P$  could be chosen in such a way that it avoids lines of reflection. By continuation, the same is true for  $\tilde{\Phi}$  except it  $P(0)$  and  $P(1)$  themselves lie on lines of reflection. Using  $\tilde{\omega}$ , this path  $P$  can be mapped to the orbifold, resulting in a path  $p$ . That path  $p$  can be used to apply  $\tilde{\Phi}$  to any other preimage  $P'(0) = g(P(0))$  from the same orbit. Since  $p$  encodes the operation of  $\varphi$  and therefore  $P'$  encodes the operation of  $\tilde{\Phi}$ , we have  $P'(1) = \tilde{\Phi}(P'(0))$ . But since the two paths will stay related via  $g$  for their whole length, we also have  $P'(1) = g(P(1))$ . Taken together we obtain

$$\tilde{\Phi}\left(g(P(0))\right) = \tilde{\Phi}\left(P'(0)\right) = P'(1) = g\left(P(1)\right) = g\left(\tilde{\Phi}(P(0))\right)$$

If  $P(0)$  were to lie on a line of reflection, then  $p(0)$  would lie on the boundary of  $O$ . In that case, the image point  $p(1) = \varphi(p(0))$  would lie on the boundary as well, since both  $f_1$  and  $f_2$  will map the boundary of  $O$  to the boundary of  $O'$ . So  $P(1)$  would again lie on a line of reflection, and the above considerations could be carried out the same way, since no ambiguities arise about which side of the line has to be chosen. Therefore  $\tilde{\Phi}$  is compatible with the symmetry group  $G$  of the Euclidean ornament in the following sense:

$$\forall z \in \mathbb{C} \setminus S \quad \forall g \in G : \tilde{\Phi}(g(z)) = g(\tilde{\Phi}(z))$$

Now one can consider a function which describes the change in position effected by the above.

$$\Psi : \mathbb{C} \setminus S \rightarrow \mathbb{C} \qquad \Psi(z) := \tilde{\Phi}(z) - z$$

This function is compatible with respect to the symmetry group in a slightly different sense:

$$\forall z \in \mathbb{C} \setminus S \quad \forall g \in G : \\ |\Psi(g(z))| = \left| \tilde{\Phi}(g(z)) - g(z) \right| = \left| g(\tilde{\Phi}(z)) - g(z) \right| = |\Phi(z) - z| = |\Psi(z)| \quad (5.2)$$

Recall that  $F$  is a fundamental domain of the Euclidean ornament. Its image  $\tilde{\Phi}(F)$  is a fundamental domain as well, since  $\varphi$  is a homeomorphism. Every fundamental domain of a wallpaper group is bounded. For this reason, there exists a bound on the distance between any point in  $F$  and any point in its image.

$$b := \sup_{z_1, z_2 \in F} \left| z_1 - \tilde{\Phi}(z_2) \right| \in \mathbb{R}$$

Due to the symmetry expressed in Equation (5.2), this bound extends to the whole plane, with the exception of the centers of rotation.

$$\forall z \in \mathbb{C} \setminus S : |\Psi(z)| = \left| \tilde{\Phi}(z) - z \right| \leq b$$

Since  $\Psi$  is bounded on its whole domain, it is in particular bounded in a region around every center of rotation. According to Riemann's theorem on removable singularities [21, 4.2], this means that the function  $\Psi$  can be extended analytically to a function  $\tilde{\Psi} : \mathbb{C} \rightarrow \mathbb{C}$  which includes the centers of rotation, so it will be defined on the whole complex plane.

A bounded conformal map on the complex plane has to be constant, due to Liouville's theorem.[13, IV.3.4] Therefore,  $\Psi$  is constant:

$$\exists c \in \mathbb{C} \forall z \in \mathbb{C} : \Psi(z) = c$$

As a consequence of this,  $\tilde{\Phi}$  has to be a translation:

$$\tilde{\Phi}(z) = z + c$$

Since both  $f_1$  and  $f_2$  associated centers of rotation according to  $r$ ,  $\varphi$  fixes these. Therefore  $\tilde{\Phi}$  must leave the orbits of centers of rotation fixed. So the translation  $\tilde{\Phi}$  must be an element of  $G$ . And if  $\tilde{\Phi} \in G$  then  $\varphi$  must be the identity transformation on the orbifold. So we can conclude

$$\varphi = f_2^{-1} \circ f_1 = \text{id} \quad \Rightarrow \quad f_1 = f_2$$

There can be at most one conformal map satisfying the stated requirements.  $\square$

### 5.4.2 High symmetry

The simplest case for which the existence of a conformal hyperbolization can be shown is the case of a faithful hyperbolization applied to a Euclidean ornament which contains at least one center of at least three-fold rotation. This case can also be described via the topology expressed by the orbifold symbol.

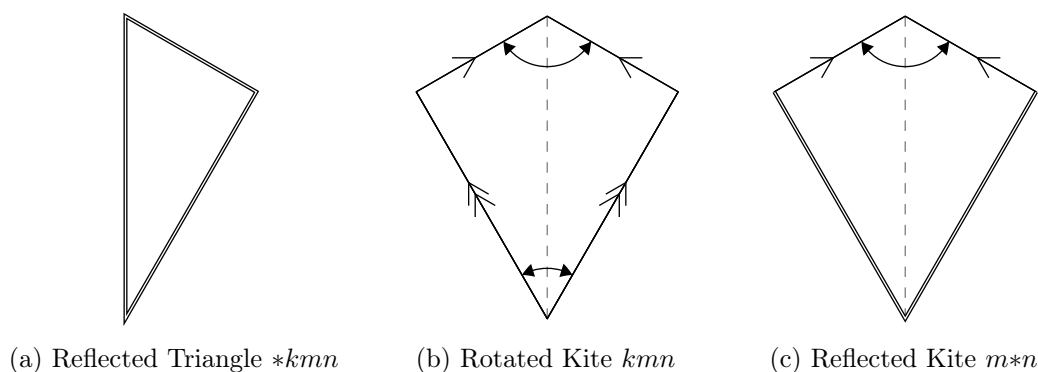


Figure 5.2: Families of triangle-based orbifolds

**Definition 5.4: High symmetry**

A symmetry group is called *highly symmetric* (or said to be of *high symmetry*) iff its orbifold symbol has one of the following three forms:

- (i)  $*kmn$
- (ii)  $kmn$
- (iii)  $m*n$

with  $k, m, n \in \mathbb{N}_{>1} \cup \{\infty\}$ . Otherwise the group will be said to be of *low symmetry*.

If the Euclidean orbifold is highly symmetric, then so is its faithful hyperbolization, according to the above definition. In that case, the orbifold – described as a fundamental domain with edge identification rules – will have one of the three topologies depicted in Figure 5.2.

**Reflected Triangle**, orbifold symbol  $*kmn$ , Euclidean groups  $*632, *442, *333$ :

The fundamental cell of the ornament is a triangle which is bounded by reflections on all three sides.

**Rotated Kite**, orbifold symbol  $kmn$ , Euclidean groups  $632, 442, 333$ :

The fundamental cell of the ornament can be chosen as a kite, with the axis of reflection along a line between an  $m$ -fold and an  $n$ -fold rotation, while the two symmetric corners belong to the same orbit, namely that of the  $k$ -fold rotation. One of the angles on the axis of rotation might even be  $\pi$ . The choice is usually non-unique, as a different assignment of actual centers of rotation to the letters  $k, m, n$  in the above description yields a different (but equivalent) fundamental domain.

**Reflected Kite**, orbifold symbol  $m*n$ , Euclidean groups  $4*2, 3*3$ :

The fundamental cell of the ornament can be chosen as a kite, with the axis of reflection joining suitable representatives for the two centers of rotation. The two symmetric angles off that axis will be right angles.

In each of these three cases, the fundamental domain is either a single triangle, or a kite composed from two symmetric triangles. All three interior angles of these triangles are already specified by the new orders for the centers of rotation. In hyperbolic geometry, the three interior angles already fully specify a triangle up to isometry. Phrased in different words, all similar triangles are congruent. So in the highly symmetric case, the shape of the hyperbolic orbifold is already fully determined by the combinatorics of the group.

### Lemma 5.3: Unique triangle map

Let  $A$  and  $B$  be triangles on two surfaces of constant (and perhaps different) curvature (i.e. sphere, Euclidean or hyperbolic plane). If the curvature for either triangle is negative, then ideal points as corners are permissible. Let their corners be labeled  $A_1, A_2, A_3$  resp.  $B_1, B_2, B_3$  in counterclockwise order.

Then there exists a unique homeomorphism  $f$  which

- conformally maps the interior of  $A$  to the interior of  $B$ ,
- maps the boundary of  $A$  to the boundary of  $B$  and
- maps the corner  $A_i$  to the corner  $B_i$  for  $i \in \{1, 2, 3\}$ .

*Proof:* The core of the proof behind this is the Riemann mapping theorem.[11] It states that any non-empty simply connected open subset of the complex plane can be mapped to the open unit disk using a conformal map. For every surface of constant curvature, a sufficient subset of that surface can be mapped onto the complex plane in a conformal way. For the Euclidean plane this is trivial, and treating the hyperbolic plane as the unit disk via the Poincaré model is not much harder. The sphere can be mapped to the complex plane via stereographic projection, where the center of projection has to be chosen outside the triangle since it will be the only point that does not map to the complex plane. In each of these cases, the interior of each triangle will be such an open set, so it will map to the unit disk. These maps can be extended to include the boundary. Let  $f_A$  be such a map from  $A$  to the unit disk, and  $f_B$  be the corresponding map from  $B$  to the unit disk.

According to Poincaré, the mapping is unique up to a Möbius transformation of the unit disk.[11] A Möbius transformation which preserves the unit circle is uniquely determined by the images of three points on that circle. To preserve the



interior, the orientation of these three points must be maintained. One can choose three arbitrary points on the unit circle, call them  $C_1, C_2, C_3$  in counterclockwise order. Then  $f_A$  can be chosen such that it maps  $A_i$  to  $C_i$ , and likewise  $f_B$  such that it takes  $B_i$  to  $C_i$ . Then  $f_B^{-1} \circ f_A$  will map from  $A$  via the unit disk to  $B$ . Since it maps from  $A_i$  via  $C_i$  to  $B_i$  it will preserve the identity of the corners. The boundary of  $A$  gets mapped to that of  $B$  via the boundary of the unit disk.

The choice of  $C_1, C_2, C_3$  does not affect the result. A different choice would result in a different map  $f_A$ , but that difference would be exactly canceled by the difference in  $f_B$ . Therefore, the map is already uniquely determined by the two triangles and the way their corners are labeled.  $\square$

It is possible to write down an exact formula for the map  $f_A$  or  $f_B$ . It is a Schwarz-Christoffel mapping that can be expressed in closed form with the use of the hypergeometric function  ${}_2F_1$ . [18, 23] But that formulation is numerically unstable, particularly in the vicinity of the points  $C_i$ , where large parts of both triangles get squashed to a tiny area near the rim of the unit disk. Computing an inverse of that function is even harder, so numerical interpolation methods are the most useful tool to approximate such an inverse function. On the whole, the transformation via the unit disk is a useful theoretical tool, but should be avoided in practical applications like this.

Instead, a concept called *discrete conformal maps* will be used. Details on their computation will be given in Section 5.5, but at this point it makes sense to treat that implementation as a black box, and only consider the input and output of that black box. Input can consist of a triangulated polygon (in Euclidean or hyperbolic geometry), together with a desired interior angle for every corner of the polygon. The algorithm will (barring numerical problems) find a new triangulated polygon with the desired angles. It has been conjectured that for a sufficiently fine mesh, the maps obtained from this will converge towards smooth conformal maps. Whenever a smooth conformal map could satisfy the angle requirements stated above, it is assumed that the discrete algorithm will be able to find a reasonable approximation. This might depend on various aspects, the combinatorics of the triangulation among them. Therefore, a possible failure to achieve such a map should be considered a bug with the implementation, but not a fundamental problem with the theory behind all this.

**Lemma 5.4: Existence of highly symmetric hyperbolization**

Given a highly symmetric Euclidean ornament, and a map assigning new orders to the orbits of the centers of rotation in such a way that the resulting orbifold will be hyperbolic, then there exists a faithful hyperbolization of the ornament with the requested orders.

*Proof:* For the  $*kmn$  case, this is trivial, since the fundamental domain is a single triangle. For the other two cases, the fundamental domain, if chosen as described above, will decompose into two symmetric triangles. According to lemma 5.3, a conformal map can be used to transform one half of the Euclidean fundamental domain to the corresponding half of the hyperbolic one. This map is an analytic function, so it has an analytic continuation into the other half. According to the Schwarz reflection principle [41], this continuation will simply be the mirror image of the first half, so it will map the second half of the Euclidean fundamental domain to the second half of the hyperbolic one, while remaining conformal even along the “seam”. So the whole fundamental domain can be mapped conformally. The identifications of the boundary of the fundamental domain fit in well with analytic continuation, too. The reason is again the Schwarz reflection principle, this time applied twice to describe a rotation. It ensures that the depicted fundamental domain and its rotated copy glued to one of the identification edges will both be subject to a single analytic map.  $\square$

**5.4.3 Low symmetry with rotations**

A faithful transformation of a symmetry group with low symmetry is more difficult than the high symmetry case discussed in Section 5.4.2. The problem here is the fact that the orbifold symbol does not uniquely define the orbifold in metric terms. There are real parameters which control its shape. But not every Euclidean shape will match every hyperbolic one with the same orbifold symbol. Instead, certain compatibility conditions have to be met.

Any triangle can be conformally mapped onto any other triangle, with corners being mapped to corners, as shown in lemma 5.3. However, for quadrilaterals this is no longer the case. While every quadrilateral can still be mapped to every other, and three corners can be mapped to corners again, the fourth corner may end up somewhere on the edge of the destination quadrilateral, unless the shapes of the two quadrilaterals are compatible. One tool to measure this compatibility is the conformal modulus.[30] A conformal map between two quadrilaterals which maps all four corners to corners does exist iff the conformal modulus of the two quadrilaterals is the same. But one can already see this from Riemann’s mapping

theorem: That map is unique up to Möbius transformation, which can in this situation be defined by three points on the unit circle which are the images of triangle corners. The fourth corner cannot be chosen arbitrarily, and for two quadrilaterals cannot be made to match if they don't do so automatically (as they do for compatible quadrilaterals).

Consider a possible hyperbolization of the symmetry group  $*2222$  as an example. The fundamental domain in the Euclidean case has to be a rectangle. Choosing new orders for the centers of rotation at its corners defines the hyperbolization. In order to construct the hyperbolic fundamental domain, one could start at an arbitrary point and use that as the starting point of two line segments which enclose the first hyperbolic angle. The length of these line segments isn't known at this point, so one may for the moment assume these lengths as free parameters. At the other ends of line segments with these chosen lengths, two more hyperbolic angles could be drawn, leading to two rays which would intersect in the fourth corner.

Now the two lengths of the initial line segments control that fourth corner. If both lengths are very short, the resulting quadrilateral is small in terms of its hyperbolic area, and since area corresponds to angle deficit, it is almost Euclidean in terms of its angle sum. On the other hand, if both lengths were made sufficiently large, they would intersect at infinity, causing an ideal fourth corner with an angle of zero. The actual angle as defined by the chosen hyperbolic orders has to lie somewhere between those two extremes, with zero included as a possibility. So by scaling both lengths with the same factor, the angle can be adjusted.

But this still leaves a one parameter family of quadrilaterals, described e.g. by the ratio between the two initial lengths. From this whole family, only a single element will be compatible with the original rectangular fundamental domain of the Euclidean ornament. Within this family of hyperbolic fundamental domains with the desired interior angles, there exists one element for which the conformal modulus equals the conformal modulus of the Euclidean ornament.

The most general wallpaper group with two-fold symmetries is the group  $2222$ . The shape of its orbifold is governed by multiple degrees of freedom. If one considers two vectors which together generate the translational subgroup, then scaling both corresponds to a scaling of the ornament, which does not affect possible hyperbolizations. But the ratio between the lengths of these two translation can be seen as one parameter which does affect the shape of the hyperbolic orbifold, and the angle between these vectors would represent a second parameter. So there is a two parameter family of Euclidean ornaments, which lead to different hyperbolizations even for the same choices of hyperbolic orbifold symbol.

Finding a suitable hyperbolic fundamental domain which is compatible with the Euclidean one is particularly difficult due to the fact that in general a straight-lined boundary of the Euclidean fundamental domain will not be mapped to a straight-

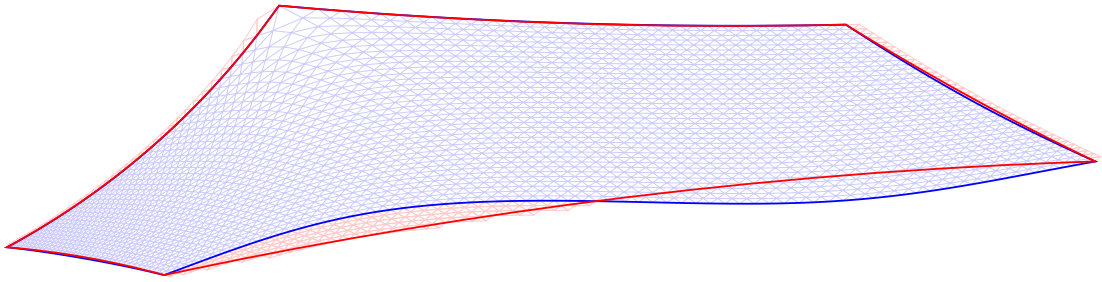


Figure 5.3: Fundamental domains connecting centers of rotation using hyperbolic straight edges (red) or the images of Euclidean straight edges (blue).

lined hyperbolic fundamental domain. In other words, a hyperbolization will map geodesics on the Euclidean orbifold to some curves on the hyperbolic orbifold which are not geodesics. This is true for most geodesics of hyperbolizations in general, but so far the boundary of the fundamental domain has been an exception to the rule, now it is affected as well. Figure 5.3 illustrates this effect.

The practical solution to this problem makes use of one property of the definition of discrete conformal deformations: they operate on triangle edge lengths, without any need for coordinates assigned to the vertices. As such, it is possible to triangulate the Euclidean *orbifold*, not a fundamental domain. As a consequence, the arbitrary choice of seams made when cutting up the orbifold is not represented in the triangulation. Adjusting the angles for the cone points leads to a hyperbolized orbifold, which can be used to obtain the positions of the centers of rotation, possible shapes of fundamental domains, and an approximation of the conformal map between Euclidean and hyperbolic fundamental domain.

On the theoretical side, Riemann's mapping theorem cannot be directly used, since it assumes open subsets of the complex plane, whereas the orbifold of 2222 has spherical topology (and indeed the metric of a tetrahedron) and as such cannot be embedded into the complex plane. But there is another result which can be used. It is due to Troyanov[49], and I will quote it here verbatim.

**Theorem 5.5: Troyanov's Theorem A**

Let  $S$  be a compact Riemann surface. Let  $p_1, p_2, \dots, p_n$  be points of  $S$  and  $\theta_1, \theta_2, \dots, \theta_n$  be positive numbers. Assume

$$2\pi\chi(S) + \sum_{i=1}^n (\theta_i - 2\pi) < 0$$

Then any smooth negative function on  $S$  is the curvature of a unique conformal metric having at  $p_i$  a conical singularity of angle  $\theta_i$ .

The orbifold of the Euclidean ornament has symbol 2222, and can be embedded into  $\mathbb{R}^3$  as a tetrahedron. Any polyhedral surface is what Troyanov calls a *Generalized Riemann surface with divisor*. As such, it can either be directly taken as the surface  $S$  for the above theorem (which the statement of theorem 5.5 as quoted above does not cover, but which in fact works quite nicely with the proof of the theorem), or one could use Troyanov's Theorem B – which basically is a flat version of the one above – to transform the tetrahedron conformally (except at the corners) onto the sphere and use that as  $S$ .

The next step is verifying the above condition. The Euler characteristic  $\chi$  of the sphere is  $\chi(S) = 2$ , so the above equation can be translated to

$$4\pi + \sum_{i=1}^4 (\theta_i - 2\pi) < 0$$

$$\sum_{i=1}^4 \theta_i < 4\pi$$

and using  $r_i$  as the orders, which implies  $\theta_i = \frac{2\pi}{r_i}$ , one obtains

$$\sum_{i=1}^4 \frac{1}{r_i} < 2$$

So the sum of angles must be less than  $4\pi$  which implies that the sum of the inverse orders must be less than 2. Which exactly matches the requirement resulting from theorem 5.1 for hyperbolic ornaments. So the theorem can be used to prove existence of faithful hyperbolizations:

**Corollary 5.6: Existence of 2222 hyperbolizations**

Let  $O$  be the orbifold of an Euclidean ornament with symmetry group 2222.  $A, B, C, D \in O$  shall denote the four cone points. Furthermore, let  $r_A, r_B, r_C, r_D \in \mathbb{N}_{\geq 1} \cup \{\infty\}$  be a choice of hyperbolic orders of rotation which satisfies the requirement for hyperbolic angles, i.e. the cost for the resulting hyperbolic orbifold will be greater than 2. Then there will always exist a unique faithful hyperbolization of that Euclidean ornament which assigns the given orders to the centers of rotation.

The case of ideal points is not explicitly covered by theorem 5.5 since that only covers *positive* angles. However, it still provides a strong intuition that this case can be handled in a similar way, since zero angles at ideal points can be obtained from a limit process. The larger one of the given orders, e.g.  $r_A$ , becomes, the greater the angle deficit of the resulting fundamental domain will be, and therefore its area. This rules out the most likely reasons against the existence of a faithful hyperbolization in these cases, namely the possibility that some corner points might coincide in the limit. A more rigorous proof would require revisiting the arguments used by Troyanov, adapting them to that case. This has not been done here.

We called 2222 the most general wallpaper group with two-fold symmetries because every other group with two-fold symmetries is a supergroup of that. One can use this relation to hyperbolize the groups  $22^*$ ,  $22 \times$  and  $2^*22$ , by treating them as special cases of 2222. Simply removing any orientation-reversing elements from the symmetry group will lead to a subgroup which has a 2222 orbifold symbol. In this subgroup, some distinct orbits of rotational centers correspond a single orbit of the original symmetry group. If these related centers of rotation are assigned the same order during hyperbolization of the 2222 subgroup, then the mirror lines of the original ornament will necessarily map to mirror lines in the hyperbolization.

The argument behind this claim employs the symmetry of the situation. Centers of rotation in the 2222 subgroup which are symmetric with respect to such a line of reflection originate in the same orbit of the original symmetry group. So they will be assigned the same order, and as a consequence, the hyperbolic orbifold, which is unique, will be symmetric as well. And since the map itself is unique as well, as shown in Section 5.4.1, the image of the mirror line has to follow the axis of symmetry of the orbifold. If it were to deviate from that axis in one direction, then one could simply reflect the orbifold as a whole to obtain another hyperbolization, which contradicts the uniqueness so there can be no such deviation.

The axes of glide reflection will not necessarily map to hyperbolic lines. This can be most easily seen in the case of  $22 \times$ , where glide reflections are a defining feature which does not emerge from a combination of other symmetry group generators. But

there still are axes of glide reflection in the resulting image, namely the hyperbolic lines which connect those points where the original axes of glide reflection intersect.

#### 5.4.4 Absence of rotations

If there are no rotations in the original Euclidean ornament, then no faithful hyperbolization can exist. Therefore it becomes necessary to designate an arbitrarily chosen point as a center of one-fold rotation. In the most general case of the symmetry group  $\circ$ , that choice has artistic significance as depicted in Figure 5.4, but from a mathematical point of view, any point may be chosen. A hyperbolization will then increase the order of that point, thus turning it into a real center of rotation.

The problems which arise are similar to those already observed for symmetry group  $2222$ . The orbifold of  $\circ$  is a topological torus and can therefore not be



(a) Euclidean original



(b) New center in the background



(c) New center in the flower

Figure 5.4: Hyperbolizations of  $\circ$

embedded into the complex plane, so again Riemann's mapping theorem does not directly apply. On the other hand, that torus is a Riemann surface therefore theorem 5.5 will again apply. Since the Euler characteristic of the torus is  $\chi = 0$ , the angle condition becomes  $\theta_1 < 2\pi$  or  $r_1 > 1$  respectively. Any increase in the order of the newly designated center of one-fold rotation will result in a possible and uniquely defined hyperbolization.

**Corollary 5.7: Existence of  $\circ$  hyperbolizations**

Let  $O$  be the orbifold of an Euclidean ornament with symmetry group  $\circ$ , and let  $A \in O$  be an arbitrary point on that orbifold. Furthermore, let  $r \in \mathbb{N}_{>1} \cup \{\infty\}$ . Then there will always exist a unique conformal hyperbolization of that Euclidean ornament in which the point  $A$  is turned into a cone point of order  $r$ , and no additional cone points are introduced.

One can again reduce most of the remaining groups to the group  $\circ$  by removing the orientation-reversing elements of the group while keeping the translations. Special care might be required in choosing the point of one-fold rotation in such a way that it is compatible with the symmetry of the ornament. For groups  $**$  and  $*\times$ , it seems best to place the designated point onto the axis of reflection. This way, the two fundamental domains of the original symmetry group which make up a single fundamental domain of the translation subgroup will share that point, so the symmetry of the ornament is maintained by the hyperbolization and the axis of reflection stays a line.

For the symmetry group  $\times\times$ , the Klein bottle, no such symmetric placement is possible. No matter where in the original orbifold the designated center of rotation is chosen, there will always be two copies of it in the orbifold of the  $\circ$  subgroup. So it is not possible to hyperbolize this group by introducing a *single* center of rotation into its  $\circ$  subgroup. It is however possible to introduce *two* centers of rotation, both with the same angle, and both related to one another via the elements of the group obtained by factoring the  $\circ$  subgroup out of the full  $\times\times$  symmetry group. The angle requirement of theorem 5.5 is again satisfied whenever the common order of this pair is greater than one.

Putting all of this together, one can state the following:



**Theorem 5.8: Existence of hyperbolizations**

For every Euclidean wallpaper group, a conformal hyperbolization is possible. If the wallpaper group contains any rotations, then the hyperbolization can be faithful. Otherwise it can be faithful with the exception of a single point which is turned into a new center of rotation.

Possible choices for the introduction of that point were given above, but that does not prove that a hyperbolization will have to choose these points. So a symmetric choice according to the rules above is sufficient for the existence of a hyperbolization, but not necessarily required.

## 5.5 Discrete conformal maps

The central tool to actually compute the conformal maps for the hyperbolizations described above is the concept of discrete conformal maps. Or more precisely, *one* concept for these. There are several distinct discretizations of conformality. The one used for hyperbolizations is based on the conformal equivalence of triangle meshes.[45]

### 5.5.1 Equivalence of triangle meshes

The basic concept is pretty simple. It captures the idea that a continuous map locally acts like an isotropic scaling. So the scale factor will depend on the location (and in the continuous case will vary smoothly over the manifold) but it does not depend on the direction. The discretization of a manifold in this model is a combinatoric triangulation  $M = (V, E, T)$  together with edge lengths  $\ell : E \rightarrow \mathbb{R}_{>0}$ .

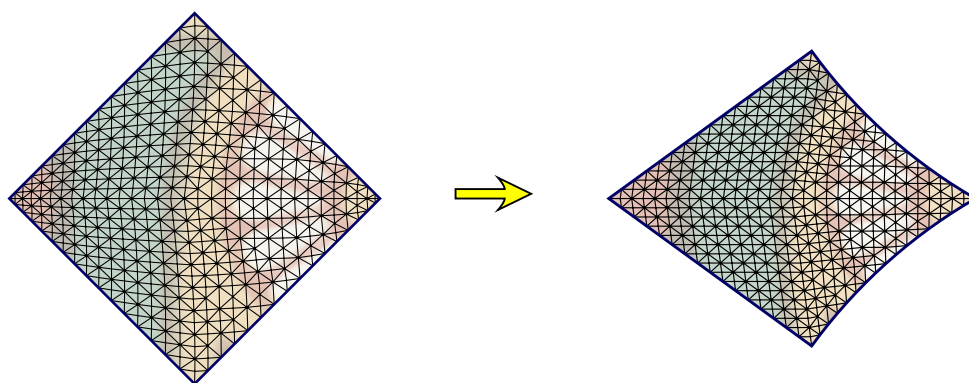


Figure 5.5: A discrete conformal deformation

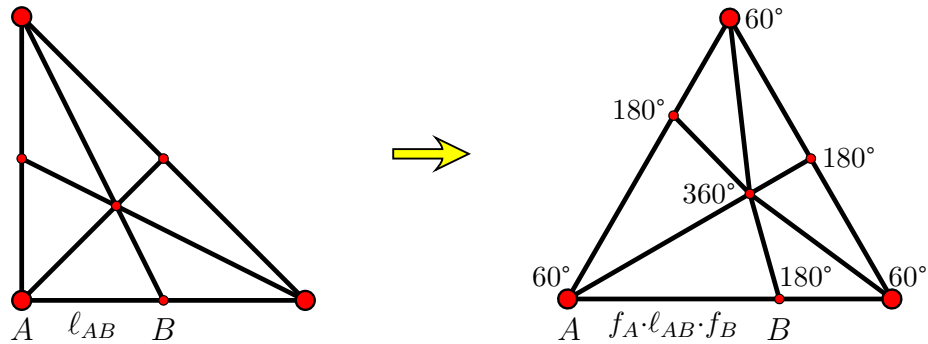


Figure 5.6: Example of conformal equivalence

A discrete conformal transformation of this graph will preserve the combinatorics but change the metric. Instead of the original lengths  $\ell$ , new lengths  $\tilde{\ell}$  will be used for the edges. The change in length should depend on location but not direction, which is modeled by discrete scale factors which are associated with the vertices, not the edges. This is captured by the following definition:

**Definition 5.5: Discrete conformal equivalence**

Two metrics  $\ell$  and  $\tilde{\ell}$  on a given triangulation  $M = (V, E, T)$  are said to be conformally equivalent iff there exists some assignment  $u : V \rightarrow \mathbb{R}$  such that

$$\forall (i, j) \in E : \tilde{\ell}_{ij} = e^{(u_i + u_j)/2} \ell_{ij}$$

This definition apparently was used in [33] in 2004, but [45] also provided a variational method approach to computing the scale factors  $u_i$  required to obtain some prescribed angle sums around the vertices. The application presented originally was conformal flattening of surface textures: by prescribing angle sums of  $2\pi$  for every interior vertex, a contoured surface could be mapped to the flat plane. But the input doesn't have to be 3d coordinates; having the edge lengths is enough.

There even exists a version of the variational principle which operates in the hyperbolic plane, so it will compute hyperbolic triangles instead of Euclidean ones. One thing worth noting in this respect is the fact that the functional used for the variational principle was obtained using hyperbolic geometry.[10] So even for the Euclidean case, ideal hyperbolic polyhedra in three dimensions were essential to obtain the required formulas. Therefore the authors of that approach were well acquainted with hyperbolic geometry, and had already worked out the hyperbolic

version by the time we asked them for it.[44]

There are basically two ways to go about this business of hyperbolizing ornaments using discrete conformal triangle meshes. The mathematically exact approach starts by triangulating the Euclidean fundamental domain. Then the angles at the singularities are conformally adjusted to match the desired hyperbolic orbifold symbol. As already mentioned in Section 5.4.3, one can actually encode the topological structure of the orbifold into the triangulation graph. This avoids the need to make arbitrary cuts and the difficulty of ensuring that these cuts will line up correctly after the transformation.

Once the transformed lengths have been computed using unconstrained convex non-linear optimization, these lengths can be used to compute triangle positions. If the topology of the orbifold is encoded into the triangulation, then a single triangle of that orbifold might get represented by more than one triangle in the plane. This happens because instead of cutting open the orbifold along the images of straight Euclidean lines, it is preferable to have a fundamental domain delimited by straight hyperbolic lines, as the rendering algorithm described in Section 4.1 is based on a convex polygonal fundamental domain. Therefore, some triangles might get cut apart by the boundary of the fundamental domain. And in any case, that boundary is not known until sufficiently many triangles incident to singularities have been placed to define the corners of that fundamental domain. Figure 5.3 also illustrates the situation, since it not only shows straight edges and their images, but also the triangle mesh in relation to these.

One drawback is that the whole computation has to be executed in hyperbolic geometry, which is slower than a comparable computation in Euclidean geometry. This observation gives rise to a second approach. Instead of triangulating the fundamental domain in the Euclidean preimage, the hyperbolic image is triangulated. The resulting triangles are however embedded into the Euclidean plane using the Poincaré disk model, and are subsequently treated as Euclidean triangles. This is a valid approximation in many cases because the smaller a triangle gets, the less angle deficit it has, and the less pronounced is the difference between Euclidean and hyperbolic geometry. Sufficiently small triangles are almost Euclidean. One drawback of this approach is the fact that it requires prior knowledge about the shape of the hyperbolic fundamental domain. This is only the case if the symmetry group is highly symmetric, as defined and used in Section 5.4.2. Otherwise, the shape has to be determined from the result of the transformation, which means that the transformation has to start with the Euclidean ornament. So that is what we implemented: highly symmetrical ornaments get hyperbolized by triangulating their hyperbolic cell and transforming that to Euclidean angles and metric. Ornaments with lower symmetry will have their Euclidean orbifold triangulated and transformed to the hyperbolic scenario using hyperbolic computations during the optimization.

One other relevant property of these conformal equivalences is the fact that it is very well suited to interpolation. For every pair of source and destination triangle there exists a uniquely defined projective transformation which maps not only the corners of one onto the other, but also the circumcircle. Iff the triangles are related to one another by a discrete conformal map, then the projective interpolation will line up along the edges of the triangles. The factors used to scale the homogeneous coordinates of the corners in order to obtain this projective transformation can be directly read off the scale factors used for the lengths in the discrete conformal map of the mesh itself. Namely a vertex  $v_i = (x, y, 1)$  of the original triangle will map to a vertex  $\tilde{v}_i = e^{-u_i}(\tilde{x}, \tilde{y}, 1)$  of the transformed mesh.[45] This interpolation allows for fairly smooth images of the hyperbolized ornament even at moderate resolution of the triangulation.

One thing this approach does not handle yet is changing the order of a singularity to infinity. The problem in that respect is the fact that the non-linear optimization uses edge lengths of triangles to compute angles from them. If one of the angles is to be zero, the two adjacent edges will be infinite. The remaining third edge length is insufficient to determine the other two angles. It might be possible to formulate the relation between these two angles and the edge length, and incorporate that into the variational principle. But that remains a task for a later time. In the meantime, simply increasing the order of a center to large but finite order gives a reasonably good approximation of infinite order, and works well enough if the computation is done from Euclidean to hyperbolic. The reverse direction, used for greater speed, suffers more severely from numeric problems if orders become large.

### 5.5.2 Circle packings

Of the various definitions for discrete conformality, we had considered and abandoned one other before we successfully adapted the equivalence of triangle meshes described in the previous subsection. That other approach was discrete conformal transformations expressed using circle packings. In the most simple scenario (and that was basically all we considered before switching approach), a circle packing whose touch graph is a triangulation of the sphere or equivalently the disk can be transformed to a combinatorially equivalent circle packing of the unit disk.[46]

This is very much a discrete version of Riemann's mapping theorem, which entails some benefits but also some drawbacks. Perhaps the major benefit is that this approach has been proven to converge to continuous conformal maps in the limit. Although the same has been conjectured for the equivalence of triangle meshes, the proof for that is, to the best of our knowledge, still outstanding. On the other hand, the approach as described suffers from some of the same problems that made the use of continuous maps infeasible in the first place. Most notably, since the target of the deformation is the unit circle, circles close to singularities in

the original fundamental domain will get mapped very close to the rim of the unit disk, which entails severe problems with regard to numeric stability. Furthermore, since the transformation most easily starts by intersecting a regular hex grid with the shape that should be mapped onto the unit disk, this would mean that the shape of the hyperbolic fundamental domain would have to be known in advance, which it is not. A different kind of problem comes from the fact that there is no obvious way to interpolate between (the centers of) these circles. Even worse, if indeed centers of circles are used as the basis for interpolation, or perhaps touching points in addition to or instead of these, then at the corners of the triangles, where the deformation is most pronounced, one would be forced to use extrapolation instead of interpolation, at the cost of less control over numeric errors.

While we were still trying to control these different problems, we learned of the freshly published work on equivalent meshes. Even in this first publication it already avoided using the unit disk as an intermediate step, and it already exhibited the high quality projective interpolation scheme associated with it. So at that time, we abandoned our work on circle packings and concentrated on equivalent meshes.



# Chapter 6

## Outlook

### 6.1 Future development

The various approaches described in this work have almost all been realized in some proof of concept implementation. However, these implementations are often distinct projects. In particular, there is one application for real time drawing of hyperbolic ornaments as described in Chapter 4, a second one to do hyperbolization of Euclidean ones as described in Chapter 5 and a third one which was mainly used to create most of the illustrations used throughout this work. While the real time drawing is still based on reified triangles as described in Section 4.3.1, the one used for the illustrations demonstrated the superiority of the approach outlined by Section 4.3.3. But the image generation was designed for high resolutions, flexible code and mathematical precision, with little regard for speed. In terms of speed, the best result is a viewer for the results of hyperbolizations, since that makes use of OpenGL in the way Section 4.2 describes it.

One major goal for this project is the creation of an integrated application which combines all of these aspects. It should have facilities to draw ornaments in Euclidean and hyperbolic geometry, with a similar user interface for both. While the display should most likely be based on OpenGL, it should also be possible to create images at greater resolution than those used for display on screen. Both pattern recognition and the hyperbolization of Euclidean ornaments should be integrated into this application. The user interface should be easily accessible to lay people, but on the other hand offer powerful tools for users with expert knowledge. For example, while novice users might be presented with a fixed set of predefined hyperbolic groups, the expert will have access to the concepts for group definition described in Chapter 3.

The project is currently planned under the name “Morenaments 2”. It is intended as common successor to two prior projects by the author: “Morenaments euc” for

drawing and recognizing Euclidean ornaments and “Morenaments hyp” for drawing hyperbolic ornaments based on triangle reflection groups.

One challenge is current technology for applications on the web. While on the one hand, running Java applications is becoming increasingly hard, with ever more configurations to be made and security warnings scaring away potential users, JavaScript with WebGL and HTML 5 on the other hand have made big steps towards graphics-intensive applications using that platform. Therefore, it seems likely that a new implementation intended to be used mainly from within the web browser should build on these technologies, even if that means rewriting large portions of the current Java code base.

## 6.2 Open questions

Most of the work for this unified project will likely be an exercise in software design, but based on mathematical concepts already understood and demonstrated by the existing proof of concept implementations. However, there are some features which require still more mathematical understanding before they can be implemented.

One open problem is the adaptation of the variational principle for discrete conformal triangle meshes to the case of zero degree angles and ideal points, as discussed at the end of Section 5.5.1.

A second open problem is the search for a suitable method to compute (discrete) conformal transformations on the sphere. The theory of discrete conformal maps can be applied to spherical geometry as well, but it loses one important property: the functional which governs the optimization process is no longer convex, so optimization techniques like gradient descent will diverge. There are currently several people exploring avenues to make this work.

Towards the end of Section 3.2.3, automatic groups have been suggested as a possible alternative to term rewriting systems. There should be some experiments on the feasibility of that approach, and if those look good, a generic proof that they will always work would be nice to have.

In Section 5.4.4 we stated some sufficient conditions for the placement of new centers of rotation, which would ensure the preservation of all symmetries even if the hyperbolization was performed using the underlying subgroup of translations only. It would be good to have a list of required conditions here as well, so that users can have a maximum of flexibility in making that choice.



## 6.3 Things to try

Section 2.1.4 described how one could continuously morph between three different models of hyperbolic geometry. It would be nice to see this realized visually. One option would be a film project to illustrate this connection. Another would be an interactive component within *Morenaments 2*.

Section 2.2.5 suggested alternative ways to represent hyperbolic lines, using ideas from Lie geometry. Some more thought and perhaps a bit of experimentation might show whether they pose a useful representation for the applications at hand.

As mentioned earlier, not every cocompact hyperbolic symmetry group is a triangle reflection subgroup. Section 3.3 suggested approaches to obtain other symmetry groups from such a triangle reflection subgroup by a sequence of subsequent modifications. This has to be tested yet, in particular to investigate how intuitive such a tool would be.

There is an alternative to the GPU-based preprocessing approach described in Section 4.2.3: Instead of only allowing a nearest-neighbor search for the corresponding position in the central fundamental domain, one could also do interpolation. To make this possible, one would have to ensure that points which are close to one another in the orbifold are also close together in the texture image which is the result of the preprocessing step. This can be achieved by replicating the topological structure of the underlying orbifold in color space of the texture.[40] Since that space is four-dimensional (three colors and alpha), even orbifolds which cannot be embedded into three-space without self-intersection might still get embedded into that space. There are however still lots of things to investigate, in particular whether the low resolution of color space can be compensated by making two textures, one with main value and one with round-off errors. Another thing to check is how susceptible this approach is to bias close to the rim of the hyperbolic plane, where interpolation positions will be far apart.



# Bibliography

- [1] AHO, A. V.; CORASICK, M. J. Efficient string matching: an aid to bibliographic search. In: *Communications of the ACM* 18 (1975), June, No. 6, pp. 333–340.
- [2] ANAN'IN, S. *Möbius transformation by 3 points in the Minkowski model*. <http://mathoverflow.net/a/156208>
- [3] ARCOZZI, N. *Beltrami's Models of Non-Euclidean Geometry*. pp. 1–30 in: SALVATORE, C. *Mathematicians in Bologna 1861–1960*. 2012, Springer. ISBN 978-3-0348-0226-0.
- [4] BAADER, F.; NIPKOW, T. *Term rewriting and all that*. 2008, Cambridge University Press. ISBN 978-0-521-77920-3.
- [5] VON BARTH, N. et al. Wikipedia: *Projective linear group*. 28 January 2014.
- [6] BELTRAMI, E. Teoria fondamentale degli spazii di curvatura costante. In: *Annali di Matematica Pura ed Applicata* 2 (1868), August, No. 1, pp. 232–255.
- [7] BELTRAMI, E. *Saggio di interpretazione della geometria Non-Euclidea*. 1868.
- [8] BELTRAMI, E. Sulla superficie di rotazione che serve di tipo alle superficie pseudosferiche. In: *Giornale di matematiche* 10 (1872).
- [9] BIANCHI, L. *Vorlesungen über Differentialgeometrie*. 1910, Teubner.
- [10] BOBENKO, A.; SPRINGBORN, B.; PINKALL, U. Discrete conformal maps and ideal hyperbolic polyhedra. arXiv:1005.2698 [math.GT] (2010), May.
- [11] BOLDT, A. et al. Wikipedia: *Riemann mapping theorem*. 16 January 2014.
- [12] CANNON, J. W. *Geometric Group Theory*. Chapter 6, pp. 261–305 in: DAVERMAN, R. J.; SHER, R. B. *Handbook of Geometric Topology*. 2001, Elsevier. ISBN 978-0-444-82432-5.
- [13] CONWAY, J. B. *Functions of one complex variable*. 1975, Springer. ISBN 978-0-387-90061-2.

- [14] CONWAY, J. H.; BURGIEL, H.; GOODMAN-STRAUSS, C. *The Symmetries of Things*. 2008, A K Peters. ISBN 978-1-56881-220-5.
- [15] COXETER, H. S. M. Crystal symmetry and its generalizations. In: *Transactions and Proceedings of the Royal Society of Canada* 3 (1957), No. 51, pp. 1–11.
- [16] COXETER, H. S. M. The Non-Euclidean Symmetry of Escher’s Picture ‘Circle Limit III’. In: *Leonardo* 12 (1979), No. 1, pp. 19–25.
- [17] COXETER, H. S. M. The trigonometry of Escher’s woodcut “Circle Limit III”. In: *The Mathematical Intelligencer* 18 (1996), No. 4, pp. 42–46.
- [18] DRISCOLL, T. A.; TREFETHEN, L. N. *Schwarz-Christoffel Mapping*. 2002, Cambridge University Press. ISBN 978-0-521-80726-5.
- [19] EPSTEIN, D. B. A.; CANNON, J. W.; HOLT, D. F.; LEVY, S. V. F.; PATERSON, M. S.; THURSTON, W. P. *Word Processing in Groups*. 1992, A. K. Peters, Ltd. ISBN 978-0-86720-244-1.
- [20] ESCHER, M. C. et al. *De magie van M.C. Escher*. 2003, Taschen. ISBN 978-3-8228-2801-4.
- [21] FREITAG, E.; BUSAM, R. *Funktionentheorie*. 1995, Springer. ISBN 978-3-540-58650-0.
- [22] VON GAGERN, M.; RICHTER-GEBERT, J. Hyperbolization of Euclidean Ornaments. In: *The electronic journal of combinatorics* 16 (2009), May, No. 2.
- [23] HARMER, M. Note on the Schwarz Triangle Functions. In: *Bulletin of the Australian Mathematical Society* 72 (2005), No. 3, pp. 385–389.
- [24] HILBERT, D. Ueber Flächen von constanter Gaussscher Krümmung. In: *Transactions of the American Mathematical Society* 2 (1901), pp. 87–99.
- [25] HUSON, D. H. *2DTiler* 2.0, 2005. <http://ab.inf.uni-tuebingen.de/software/2dtiler/welcome.html>
- [26] IVERSEN, B. *Hyperbolic Geometry*. December 1992, Cambridge University Press. ISBN 978-0-521-43528-4.
- [27] JONES, O. *The grammar of ornament*. 1910, Bernard Quaritch.
- [28] KNUTH, D. E.; BENDIX, P. B. *Simple word problems in universal algebras*. pp. 342–376 in: SIEKMANN, J. H.; WRIGHTSON, G. *Automation of Reasoning*. 1983, Springer. ISBN 978-3-642-81957-5.
- [29] KREYSZIG, E. *Differentialgeometrie*. 1968, Akademische Verlagsgesellschaft Geest & Portig K.-G.

- [30] KÜHNAU, R. *The conformal module of quadrilaterals and of rings*. Chapter 3, pp. 99–129 in: KÜHNAU, R. *Geometric Function Theory*, vol. 2. 2005, Elsevier. ISBN 978-0-444-51547-6.
- [31] LEYS, J. *Hyperbolic Escher*. [http://www.josleys.com/show\\_gallery.php?galid=325](http://www.josleys.com/show_gallery.php?galid=325)
- [32] LIU, Y.; COLLINS, R. T.; TSIN, Y. A computational model for periodic pattern perception based on frieze and wallpaper groups. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26 (2004), March, No. 3, pp. 354–371.
- [33] LUO, F. Combinatorial Yamabe flow on surfaces. In: *Communications in Contemporary Mathematics* 6 (2004), No. 5, pp. 765–780.
- [34] MCLACHLAN, R. A gallery of constant-negative-curvature surfaces. In: *The Mathematical Intelligencer* 16 (1994), No. 4, pp. 31–37.
- [35] MILNOR, J. W. Hyperbolic geometry: The first 150 years. In: *Bulletin (New Series) of the American Mathematical Society* 6 (1982), No. 1.
- [36] POINCARÉ, H. Théorie des groupes fuchsien. In: *Acta Mathematica* 1 (1882), No. 1, pp. 1–62.
- [37] POZNYAK, È. G.; SHIKIN, E. V. Surfaces of negative curvature. In: *Journal of Soviet Mathematics* 5 (1976), No. 6, pp. 865–887.
- [38] REYNOLDS, W. F. Hyperbolic Geometry on a Hyperboloid. In: *The American Mathematical Monthly* 100 (1993), May, No. 5, pp. 442–455.
- [39] RICHTER-GEBERT, J. *Perspectives on Projective Geometry*. February 2011, Springer. ISBN 978-3-642-17285-4.
- [40] RICHTER-GEBERT, J. Personal communication, 2014.
- [41] SCHWARZ, H. A. Ueber einige Abbildungsaufgaben. In: *Journal für die reine und angewandte Mathematik* 70 (1869), pp. 105–120.
- [42] SEGAL, M.; AKELEY, K. *The OpenGL<sup>®</sup> Graphics System*, Version 3.0, August 2008. <http://www.opengl.org/registry/doc/glspec30.20080811.pdf>
- [43] SENECHAL, M. *Quasicrystals and geometry*. 1996, Cambridge University Press. ISBN 978-0-521-57541-6.
- [44] SPRINGBORN, B. Personal communication, 2008.
- [45] SPRINGBORN, B.; SCHRÖDER, P.; PINKALL, U. Conformal equivalence of triangle meshes. In: *ACM Transactions on Graphics* 27 (2008), No. 3.

- [46] STEPHENSON, K. *Introduction to Circle Packing*. June 2005, Cambridge University Press. ISBN 978-0-521-82356-2.
- [47] STILLWELL, J. *Sources of Hyperbolic Geometry*, vol. 10. American Mathematical Society. ISBN 978-0-8218-0922-8.
- [48] THURSTON, W.P. *The Geometry and Topology of Three-Manifolds*, 1980. Lecture notes, Princeton University.
- [49] TROYANOV, M. Prescribing curvature on compact surfaces with conical singularities. In: *Transactions of the American Mathematical Society* 324 (1991), No. 2, pp. 793–821.