Karlsruhe Institute of Technology

Complexity of SimpleStretchability and Related ER-Complete Problems in Hyperbolic Geometry

Master Thesis of

Nicholas Bieker

At the Department of Informatics
Institute of Theoretical Informatics

Reviewers:	Thomas Bläsius
	Torsten Ueckerdt
Advisors:	Paul Jungeblut
	Thomas Bläsius

Time Period: 29th April 2022 – 28th October 2022

Statement of Authorship

Ich versichere wahrheitsgemäß, die Arbeit selbstständig verfasst, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde sowie die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet zu haben.

Karlsruhe, October 28, 2022

Abstract

There are many problems of geometric nature that are complete in the complexity class $\exists\mathbb{R}$. Their hardness is often shown by reduction from one of the first problems shown to be complete in $\exists\mathbb{R}$, SimpleStretchability. Until now, the underlying geometry was almost always assumed to be Euclidean. The purpose of this thesis is to examine what happens to the complexity of these problems when considering hyperbolic geometry instead. For that, we start with SimpleStretchability, define the hyperbolic problem version HyperbolicSimpleStretchability and show that the hyperbolic and Euclidean versions are equivalent.

One problem where this change of geometry was already considered is the problem of recognizing unit disk graphs. In the Euclidean plane, the problem is well-researched and proven to be $\exists\mathbb{R}$-complete. In the hyperbolic plane, the problem was established to be in $\exists\mathbb{R}$ as well as NP-hard. In this thesis, we add that recognizing hyperbolic unit disk graphs is $\exists\mathbb{R}$-hard as well.

Additionally, we consider other Euclidean geometric problems that are proven to be $\exists\mathbb{R}$-complete via reduction from SimpleStretchability. We want to encourage further research about their hyperbolic variants motivated by the results shown for hyperbolic unit disk graphs. For that, we provide a proof framework for showing $\exists\mathbb{R}$-hardness for their hyperbolic problem versions and apply that framework for the recognition problem of segment graphs. We also present additional problems that could have interesting hyperbolic counterparts.

Deutsche Zusammenfassung

In dieser Arbeit betrachten wir geometrische Probleme, die für die Komplexitätsklasse $\exists\mathbb{R}$ vollständig sind. Die beiden Probleme, mit denen wir uns dabei am meisten befassen, sind SimpleStretchability und die Erkennung von Unit-Disk-Graphen. Für diese Probleme ist bereits bekannt, dass sie $\exists\mathbb{R}$-vollständig sind, wenn man Einbettungen in die euklidische Ebene betrachtet. Wir ändern diese Einbettungen so, dass wir die hyperbolische Ebene statt der Euklidischen dafür nutzen.

Wir zeigen, dass die so entstehenden Problemvarianten HyperbolicSimpleStretchability und Recog(HUDG) ebenfalls $\exists\mathbb{R}$-vollständig sind. Für SimpleStretchability erlangen wir ein noch stärkeres Ergebnis: Die euklidischen und hyperbolischen Problemversionen sind nicht nur gleich schwer, sondern sogar äquivalent.

Dieses Resultat nutzen wir, um die $\exists\mathbb{R}$-Schwere von Recog(HUDG) zu zeigen, wobei vorher bereits bekannt war, dass es NP-schwer und in $\exists\mathbb{R}$ ist. Darüber hinaus bauen wir eine Beweisidee, die nicht nur für Unit-Disk-Graphen, sondern auch für andere geometrische $\exists\mathbb{R}$-vollständige Probleme nutzbar ist. Mit dieser Beweisidee kann für viele andere hyperbolische Problemvarianten ebenfalls $\exists\mathbb{R}$-Schwere gezeigt werden, was wir am Beispiel von Segmentgraphen durchführen. Damit und mit der Einführung von weiteren Kandidaten für das Framework wollen wir die Forschung an weiteren hyperbolischen Problemvarianten ermutigen.

Contents

1. Introduction

In recent years, many problems in different contexts were found to be complete for the complexity class $\exists\mathbb{R}$. The class was introduced by Schaefer [Sch09] and is defined via the decision problem ETR. For a given formula of the Existential Theory of the Reals, consisting of real variables in equations and inequalities that are connected with logical operators, a set of real variables fulfilling this formula has to be found. In a way, ETR can be seen as a real number variant of SAT and $\exists\mathbb{R}$ as a real number extension of NP. An overview of the class and some of its complete problems is given in [Bie20].

One of the contexts that includes many $\exists\mathbb{R}$-complete problems is geometry, in particular recognition problems of intersection graphs. Here, the vertices of a graph $G = (V, E)$ are represented by geometric objects that intersect if and only if the corresponding vertices are adjacent. Two of the first problems of this type that were considered in the context of $\exists\mathbb{R}$ are the recognition problems of unit disk and segment graphs. For unit disk graphs, Kang and Müller [KM12] give a NP-hardness proof that also shows $\exists\mathbb{R}$-hardness. For segment graphs, Schaefer himself [Sch09] and Matoušek [Mat14] show $\exists\mathbb{R}$-hardness via reduction from the same problem Kang and Müller used, SimpleStretchability where we are given a pseudoline arrangement and have to decide if there is a homeomorphic arrangement of lines.

Until now, in the context of $\exists\mathbb{R}$, the underlying geometry of those problems was almost always considered to be Euclidean. We ask the question what happens to the problems and their complexity when we consider hyperbolic geometry instead. This is the purpose of this thesis: To examine $\exists\mathbb{R}$ and its problems of geometric nature in hyperbolic geometry instead of Euclidean geometry. One of the properties of hyperbolic geometry is that in small areas, its behaviour is similar to Euclidean geometry. This indicates that the hyperbolic versions are at least as hard as the Euclidean ones, as Euclidean structures are easily scalable to small areas where they can be transformed to hyperbolic geometry. We exploit this observation at multiple points in the thesis.

We start by defining a hyperbolic variant of the SimpleStretchability problem whose Euclidean version not only serves as the starting point for the hardness proofs of recognizing unit disk and segment graphs, but for many more geometric problems in $\exists\mathbb{R}$. We present an operation that manages to scale down simple hyperbolic line arrangements to arbitrarily small areas as well. Using that operation, we conclude one of the main results of our thesis: The hyperbolic and Euclidean SimpleStretchability problems are equivalent, meaning that a combinatorial description of a pseudoline arrangement is realizable in the Euclidean plane if and only if it is realizable in the hyperbolic plane. This is rather surprising as in general, the hyperbolic plane allows far more possibilities than the Euclidean one.

As a natural next step, we consider the problem of recognizing hyperbolic unit disk graphs. Hyperbolic unit disk graphs were first established by Papadopoulos et al. [PKBV10] in a probabilistic setting. Random hyperbolic unit disk graphs, also known as hyperbolic random graphs, have a hierarchical structure that is very convenient for testing algorithms. In recent years, there have been results for the graph class of intersection graphs of hyperbolic unit as well, by Dohse [Doh22], Bläsius et. al [BFKS21] and Kisfaldu-Bak [KB20]. Additionally, Dohse also considers the complexity of the problem of recognizing hyperbolic unit disk graphs and shows that they are in $\exists\mathbb{R}$ as well as NP-hard, but leaves the question open if they are also $\exists\mathbb{R}$-hard. We give an answer by showing $\exists\mathbb{R}$-hardness, using the equivalence of simple hyperbolic and Euclidean line arrangements as well as the $\exists\mathbb{R}$-hardness proof for recognizing Euclidean unit disk graphs.

Motivated by the results for hyperbolic unit disk graphs, we then generalize our hardness proof to be applicable for more geometric $\exists\mathbb{R}$-complete problems. As an example, we define the hyperbolic version of recognizing segment graphs and show its $\exists\mathbb{R}$-hardness with our proof framework. This hopefully encourages a discussion about considering hyperbolic variants for more geometric problems. We present a few additional candidates that could also be considered in a hyperbolic context.

The thesis is structured in the following way: In Chapter 2, we explain the basics of hyperbolic geometry, complexity and graph theory that are used throughout this thesis. Chapter 3 then deals with line arrangements and the Stretchability problem. Here we introduce the hyperbolic scaling operation and use it to show that the hyperbolic version of SimpleStretchability is equivalent to its Euclidean variant and thus $\exists\mathbb{R}$-complete. We use that in Chapter 4 to show that, while not being equivalent to its Euclidean variant, the problem of recognizing hyperbolic unit disk graphs is also $\exists\mathbb{R}$-complete. Finally, we formalize the ideas of that proof in Chapter 5 to introduce a proof framework for showing $\exists\mathbb{R}$-hardness for hyperbolic variants of other geometric $\exists\mathbb{R}$-complete problems and apply it on the problem of recognizing segment graphs. Additionally, we present a few other problems that are likely to have hard hyperbolic variants as well.

2. Preliminaries

This thesis mainly touches three different areas of theoretical computer science and mathematics. The first one is complexity theory and the complexity class $\exists\mathbb{R}$. We also need geometry and the differences between Euclidean and hyperbolic geometry, and graph theory and certain graph problems like recognizing unit disk graphs. In this chapter, we introduce these areas and the basic concepts needed throughout this thesis.

2.1. Complexity Theory

In complexity theory, the goal is to unite different problems into classes based on a defining property, mostly time or space complexity. A *complexity class* thus is a set of problems that have the same defining quality. The most commonly known complexity classes are P and NP. Both classes contain problems based on their time complexity: P consists of problems that are deterministically solvable in polynomial time, while NP allows non-deterministic approaches. In this thesis, we work with a different class: The complexity class $\exists\mathbb{R}$, introduced by Schaefer [Sch09].

Complexity Class $\exists\mathbb{R}$

In order to define $\exists\mathbb{R}$, we first need to introduce the Existential Theory of the Reals. As the name suggests, we consider first order sentences that are existentially quantified. The variables represent real numbers that we combine into formulas via equations, inequalities and logical connectors. Each equation and inequality has to be a polynomial over the existentially qualified real variables. These equations and inequalities are then used in logical formulas and connected with $\wedge$, $\vee$ and $\neg$ to be evaluted. The Existential Theory of the Reals is the set of all true sentences of type $\exists x_1, \ldots, x_n : p(x_1, \ldots, x_n)$ where p is a quantifier-free sentence over the signature $\{0, 1, +, \cdot, <, \leq, =\}$. The corresponding decision problem ETR is defined in the following way:

ETR:
Input: Formula $\exists x_1, \ldots, x_n : p(x_1, \ldots, x_n)$ where p is a quantifier-free sentence over the signature $\{0, 1, +, \cdot, <, \leq, =\}$ with connectives $\{\vee, \wedge, \neg\}$.
Problem: Are there real numbers $x_1, \ldots, x_n$ for which the sentence p is true?

An example of a sentence that belongs in the Existential Theory of the Reals is the following:

$$\phi \equiv \exists x_1, x_2, x_3 : (x_1 + x_2)x_3 = 1 \wedge x_1 < 0$$

As $(x_1 + x_2)x_3 = 1 \wedge x_1 < 0$ is true for $x_1 = -1, x_2 = 2, x_3 = 1$, among other solutions, the sentence indeed is an ETR yes-instance. We can now define $\exists\mathbb{R}$ using ETR:

Definition 2.1 ($\exists\mathbb{R}$)**.** *$\exists\mathbb{R}$ is the set of all problems that are polynomially reducable to ETR.*

In the context of the different complexity classes, $\exists\mathbb{R}$ lies between NP [Sch09] and PSPACE[Tar98] where both subset relations are believed to be proper. $\exists\mathbb{R}$ consists of many different problems from multiple contexts, including graph recognition problems like recognizing unit disk graphs [KM12] or segment graphs [Mat14], [Sch09]. These problems and many others use underlying geometric concepts that are mostly assumed to be of Euclidean nature. In this thesis, we consider hyperbolic geometry instead, which we introduce now.

2.2. Hyperbolic Geometry

When we think of geometry and the plane without further distinctions, we usually think about Euclidean geometry. This is the intuitive model of geometry for humans and lines up with our natural observations. However, it is not the only way to define consistent geometric models. Another model, hyperbolic geometry, plays a large part in this thesis.

Euclidean geometry is formally defined using axioms to correctly describe human observations. One of these axioms, the parallel axiom, is the notion that for each line l and point p with $p \notin l$, there is exactly one line l' parallel to l with $p \in l'$. Hyperbolic geometry is defined in the same way using identical axioms, with the exception of the parallel axiom. In hyperbolic geometry, it is negated: There are infinitely many lines l' through p that are parallel to l.

This results in a different kind of space where, for example, the area of a circle grows exponentially in its radius, as opposed to quadratic growth in Euclidean space. Intuitively, the hyperbolic plane includes exponentially more space than the Euclidean one. However, when we only consider areas where points have small distances to each other, the differences between the hyperbolic and the Euclidean plane become negligible. This observation plays a key part in this thesis.

As we cannot accurately represent hyperbolic space, we rely on models that express the hyperbolic plane in Euclidean space to observe the effects of hyperbolic geometry. Two of the most commonly used models are the Poincaré Disk and the hyperboloid model.

Poincaré Disk Model

In this model, we first need to choose any point O of the hyperbolic plane, called the origin. The hyperbolic plane is represented in Euclidean space as the interior of a unit disk D around the point $(0, 0)$ which represents the origin O. The boundary of the circle is not part of the model and represents infinity. Points on the boundary are called *ideal points*. Hyperbolic lines are either lines through the origin or the segment inside of D of a circle c. This circle c has to be orthogonal to the bounding unit disk, meaning that the tangents of c and D in their intersection points build a right angle. Hyperbolic circles always look like Euclidean circles, although the center of the circle is not the same as the center of the Euclidean circle if it is not the origin. In order to compute the distance of two points P, Q, we define ideal points A, B that are the intersection points of the hyperbolic line with D through P and Q The distance of P and Q now is given by the following formula:

$$d_h(P, Q) = \ln\left(\frac{d_e(A, Q), d_e(P, B))}{d_e(A, P), d_e(Q, B))}\right.$$

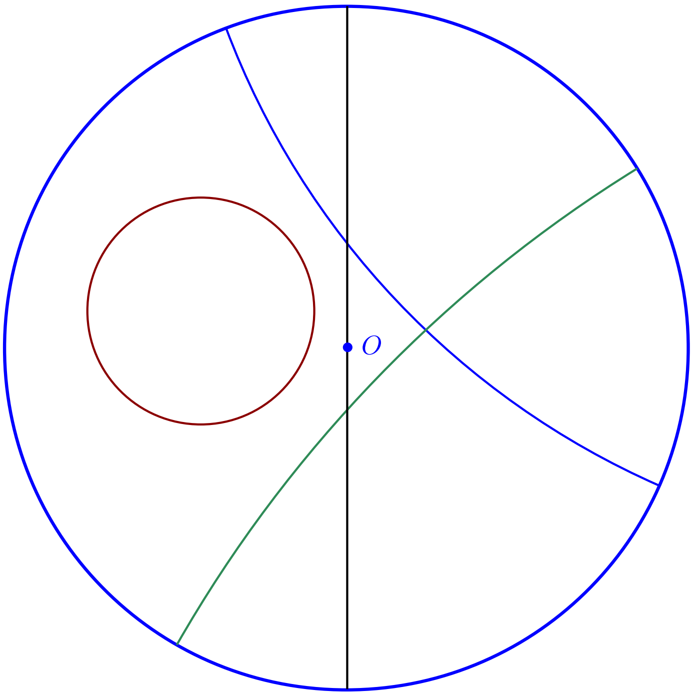

Figure 2.1.: An exemplary arrangement of lines (green and black) and circles (red) in a Poincaré disk (blue).

An arrangement of hyperbolic lines and circles in the Poincaré disk is given in Figure 2.1.

This is the model we mainly use in this thesis. It is intuitive for higher-level ideas as the concepts of lines and circles are simple Euclidean structures that are easily usable. However, we need an additional model for distance calculations as we need to represent hyperbolic distances in ETR-formulas. This distance formula is too complex for that as ETR formulas only allow polynomials. The additional model is the hyperboloid model:

Hyperboloid Model

In order to understand the hyperboloid model, we first introduce the concept of hyperboloids. A *hyperboloid* h is a surface in 3-dimensional Euclidean space. It can either be *one-sheeted* with characteristic formula $x^2/a^2 + y^2/b^2 - z^2/c^2 = 1$ describing the points (x, y, z) that are on h for constants a, b and c. Or it can be *two-sheeted* with characterization $z^2/c^2 - x^2/a^2 - y^2/b^2 = 1$. Geometrically, one-sheeted hyperboloids are the result of rotating a hyperbola around one of its main axes while two-sheeted hyperboloids emerge when rotating around the third axis. One-sheeted hyperboloids are a continouos surface while two-sheeted hyperboloids consist of two separate surfaces.

In the hyperboloid model, the hyperbolic plane is represented by the surface of a hyperboloid embedded in 3-dimensional Euclidean space (shown in Figure 2.2). A point (x, y, z) is on the hyperboloid if $z^2 - x^2 - y^2 = 1$ (also known as the Minkowski quadratic form). This forms a two-sheeted hyperboloid, but we only use one sheet for our model. Thus, we restrict the points to be on the upper sheet by having $z > 0$ as an additional restriction. In this model, hyperbolic lines are obtained by intersecting planes that contain the origin of the 3-dimensional space with the hyperboloid. As two points on the hyperboloid, together with the origin, fully define such a plane and thus a line in our model, we can describe the distance of these two points along the line. The distance formula is given by $d_h(u, v) = \operatorname{arcosh}(B(u, v))$ where B is the Minkowski bilinear form:

$$((u_x, u_y, u_z), (v_x, v_y, v_z)) = u_z v_z - u_x v_x - u_y v_y$$

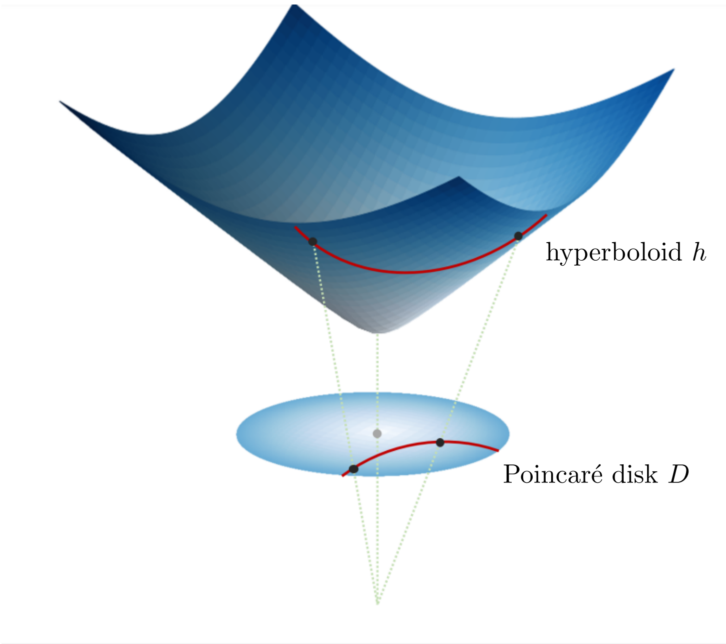

Figure 2.2.: Visualization of the hyperboloid model in relation to a Poincaré disk (Source: [CYRL19]).

.

As described above, we use this model whenever we need to describe distance in the hyperbolic plane by polynomials, as the distance formula is mostly a polynomial with an additional arcosh as a last step. With a trick we describe later, this can be used in ETR formulas to describe the hyperbolic distance between points.

2.3. Graph Theory

As $\exists\mathbb{R}$ includes many graph-related problems that are interesting for us, we also introduce basics of graph theory. For us, a *graph* $G = (V, E)$ is a pair of the *vertex set* V and the *edge set* $E \subseteq V \times V$, consisting of undirected edges. We denote the number of vertices with $n := |V|$ and the number of edges with $m := |E|$. As we work with geometric graph recognition problems, we often need to embed graphs into either the Euclidean or the hyperbolic plane. Formally, an embedding is a function $f : V \to \mathcal{P}$ that maps each vertex onto a point of the plane $\mathcal{P}$ where $\mathcal{P}$ can either be the Euclidean plane $\mathbb{R}^2$ or the hyperbolic plane $\mathbb{H}^2$.

The main graph problem we consider in this thesis is the problem of recognizing unit disk graphs:

Unit Disk Graphs

The problem of recognizing unit disk graphs, or more formally, intersection graphs of unit disks, asks if a given graph $G = (V, E)$ can be represented by unit disks in a given plane. The disks each represent a vertex, and the arrangement of disks is a valid representation of G if two hyperbolic lines intersect if and only if the corresponding vertices are adjacent. This is equivalent to another characterization of the problem using the concept of embeddings we introduced earlier: A graph $G = (V, E)$ is a unit disk graph if there is an embedding $f : V \to \mathcal{P}$ into a plane $\mathcal{P}$ that fulfills $\{u, v\} \in E \Leftrightarrow d(f(u), f(v)) \leq 1$. Depending on which plane we choose for $\mathcal{P}$, we get different versions of the problem:

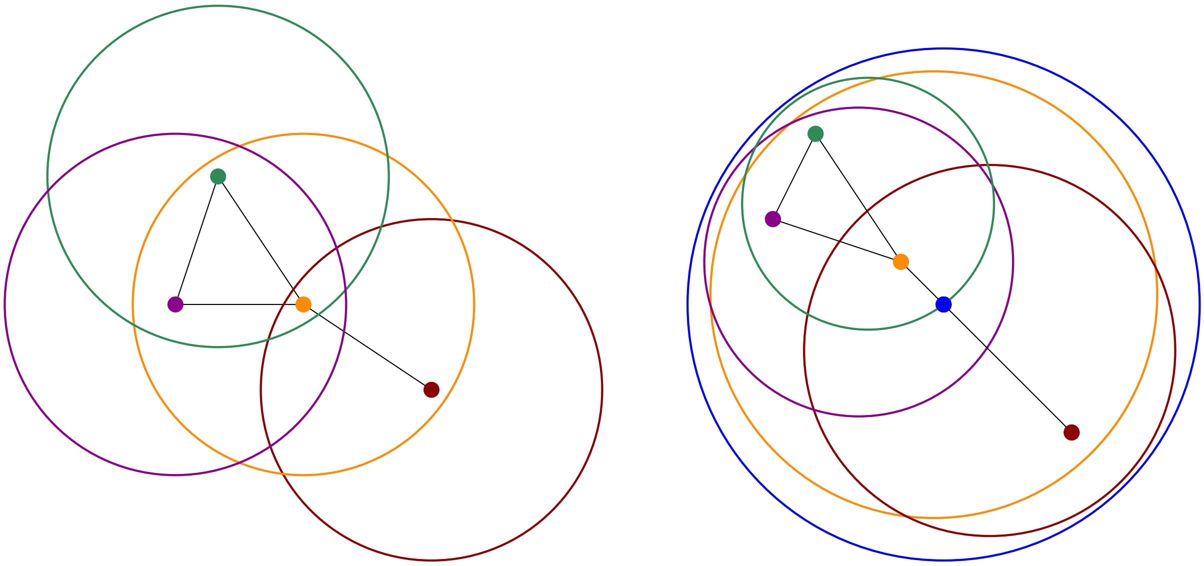

Figure 2.3.: Left: Euclidean embedding. Right: Hyperbolic embedding of the same graph for some threshold distance R in the Poincaré disk model.

For $\mathcal{P} = \mathbb{R}^2$, we get the problem of recognizing Euclidean unit disk graphs, RECOG(EUDG). A *Euclidean unit disk graph* (EUDG) is a graph $G = (V, E)$ that can be embedded into the euclidean plane in a way that the distance between adjacent vertices is at most one and the distance between non-adjacent vertices is more than one. The problem of deciding whether a given graph is an EUDG is defined in the following way:

RECOG(EUDG):
Input: Graph $G = (V, E)$
Problem: Is there an embedding $f_e : V \to \mathbb{R}^2$ into the euclidean plane that fulfills $\{u, v\} \in E \Leftrightarrow d_e(f_e(u), f_e(v)) \leq 1$?

For $\mathcal{P} = \mathbb{H}^2$, we similarly obtain the problem of recognizing hyperbolic unit disk graphs, RECOG(HUDG). However, there is a difference: In the Euclidean case, it does not matter how we choose the threshold distance as we can simply scale the embedding to any value. In the hyperbolic case, choosing different threshold distances can result in different results for a given graph, as there is no scaling operation that preserves the characteristics of the embedding. Because of that, we do not fix a threshold distance R, but rather allow different ones. It is sufficient for G to be considered a HUDG if there is one R for which G can be embedded into $\mathbb{H}^2$. A *hyperbolic unit disk graph*, or HUDG, thus is a graph G that has an embedding into the hyperbolic plane such that a threshold distance R exists where exactly the adjacent vertices have hyperbolic distances smaller than R. The problem of recognition hyperbolic unit disk graphs then is defined in the following way:

RECOG(HUDG):
Input: Graph $G = (V, E)$
Problem: Is there a radius R and an embedding into the hyperbolic plane $f_h : V \to \mathbb{H}^2$ that fulfills $\{u, v\} \in E \Leftrightarrow d_h(f_h(u), f_h(v)) \leq R$?

An embedding of a unit disk graph into Euclidean and hyperbolic space is given in Figure 2.3.

The graph classes of Euclidean and hyperbolic unit disk graphs are well-researched. Relevant results for us include that every EUDG is also a HUDG as shown by Dohse [Doh22], and that there are HUDGs that have no representation via Euclidean unit disks. Dohse also considers the decision problem of recognizing hyperbolic unit disk graphs and shows that it is NP-hard and in $\exists\mathbb{R}$. For the Euclidean counterpart, Kang and Müller show $\exists\mathbb{R}$-hardness [KM12] while $\exists\mathbb{R}$-membership is stated multiple times, but as far as we know not yet explicitly shown. Additionally, Bläsius et al. [BFKS21] introduce a subclass of HUDGs, strongly hyperbolic unit disk graphs, that we consider in Chapter 4.

3. Hyperbolic SimpleStretchability

The goal of this thesis is to consider $\exists\mathbb{R}$-complete problems of geometric nature in the context of hyperbolic geometry and explore their complexity. As a starting point, we take a closer look at one of the most fundamental problems contained in $\exists\mathbb{R}$, SimpleStretchability. Specifically, we discuss what happens to the realizability of pseudoline arrangements if we change the underlying geometry from Euclidean to hyperbolic. SimpleStretchability is special among the $\exists\mathbb{R}$-complete problems because it was proven to be $\exists\mathbb{R}$-complete before $\exists\mathbb{R}$ was established. Mnëv showed that the underlying mathematical concepts of $\exists\mathbb{R}$ and SimpleStretchability are equivalent in his Universality Theorem [Mnë88]. Because of that, there are many geometrical reductions from SimpleStretchability to other problems such as recognizing unit disk graphs [KM12] and segment graphs [Mat14]. For that reason, SimpleStretchability is the obvious problem to start with when shifting the geometry from Euclidean to hyperbolic. We now define the problem in both the Euclidean and hyperbolic version.

3.1. The SimpleStretchability Problem

In order to define the problem, we need the concept of pseudoline arrangements. A *pseudoline* is a curve in Euclidean space that is x-monotone (intersects every vertical line exactly once). An *arrangement* of lines or pseudolines is a concrete drawing of the lines in the according plane. We call a pseudoline arrangement P *stretchable* or *realizable* if a line arrangement L exists that is homeomorphic to P and say that L *realizes* P. A (pseudo)line arrangement is called *simple* if the lines/pseudolines intersect pair-wise exactly once and in each intersection point, only two lines meet. An example of a simple pseudoline arrangement and a corresponding line arrangement is given in Figure 3.1. The corresponding decision problem can be described in the following way:

SimpleStretchability:
Input: Simple arrangement P of pseudolines.
Problem: Does a line arrangement L exist that is homeomorphic to P?

3.1.1. Complexity of EuclideanSimpleStretchability

As mentioned above, Mnëv's Universality Theorem shows $\exists\mathbb{R}$-completeness of SimpleStretchability in an indirect way: Mnëv connects an underlying concept of pseudoline arrangements, rank-3 oriented matroids, with the underlying concept of ETR,

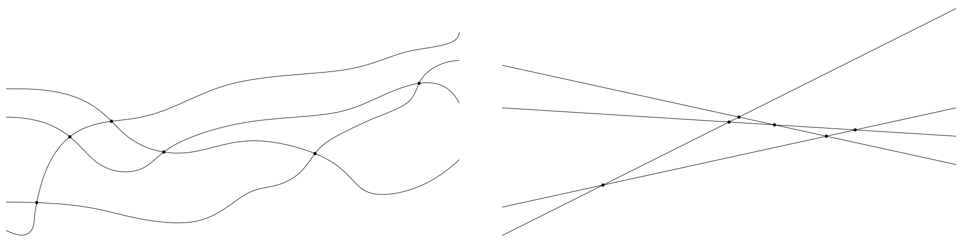

Figure 3.1.: A pseudoline and corresponding line arrangement of four lines.

semialgebraic sets. We do not go into detail here and do not define these concepts as there is an alternative, more intuitive proof and it is not important for the rest of this thesis. The Universality Theorem can be formulated in the following way:

Theorem 3.1 (Universality Theorem ([Mnë88])). *Every semialgebraic set is stably equivalent to the realization space of a rank-3 oriented matroid.*

As the Universality Theorem has been proven before the complexity class $\exists\mathbb{R}$ was established, the reduction from ETR to SimpleStretchability is not easy to grasp. The proof is highly mathematical and uses advanced concepts. In order to establish a more straightforward proof, Matoušek [Mat14] gives a more direct reduction from ETR via an intermediate problem called OrderTypeRealizability which is basically the dual equivalent of SimpleStretchability (lines correspond to points and vice versa). Both the mathematical proof of Mnëv and the geometrical proof of Matoušek show the following theorem:

Theorem 3.2 ([Mat14]). EuclideanSimpleStretchability *is $\exists\mathbb{R}$-complete.*

3.1.2. Combinatorial Description of Pseudoline Arrangements

There are several equivalent ways to encode the pseudoline arrangement, we use two of those in this thesis: We can either specify the intersections of the lines or the regions that the plane is divided into by the lines. Both versions are equivalent to each other as they both uniquely define pseudoline arrangements and can be translated both ways.

We start with presenting the combinatorial description of a pseudoline arrangement $L = \{l_1, \dots, l_n\}$ via intersections, which follows an explanation from Matoušek [Mat14]. This is the model we mainly use in Chapter 3. In the pseudoline arrangement, each of the lines l_i are x-monotone. That means that we can add an additional vertical line l_0 to the left of all intersection points. We call l_0 the border line of the arrangement and demand, in contrast to Matoušek, that l_0 is a line of the arrangement. L thus is an arrangement of $n+1$ lines $l_0, \dots, l_n$. Matoušek only used l_0 implicitly, but it is obvious that we do not change the problem by adding l_0 as we can always add and delete it from a pseudoline arrangement without changing the rest of the arrangement without turning yes-instances into no-instances and vice-versa. For each of the other pseudolines, we then follow them starting at the intersection with l_0 and write down a sequence of the order of intersections with the other lines. The set of these sequences, with the lines being numbered in order of their intersection with l_0 (starting from the top) and the sequences being ordered similarly, forms the *combinatorial description* of the pseudoline arrangement. An example of this process is visualized in Figure 3.2. The problem definition of EuclideanSimpleStretchability via combinatorial definition then is the following:

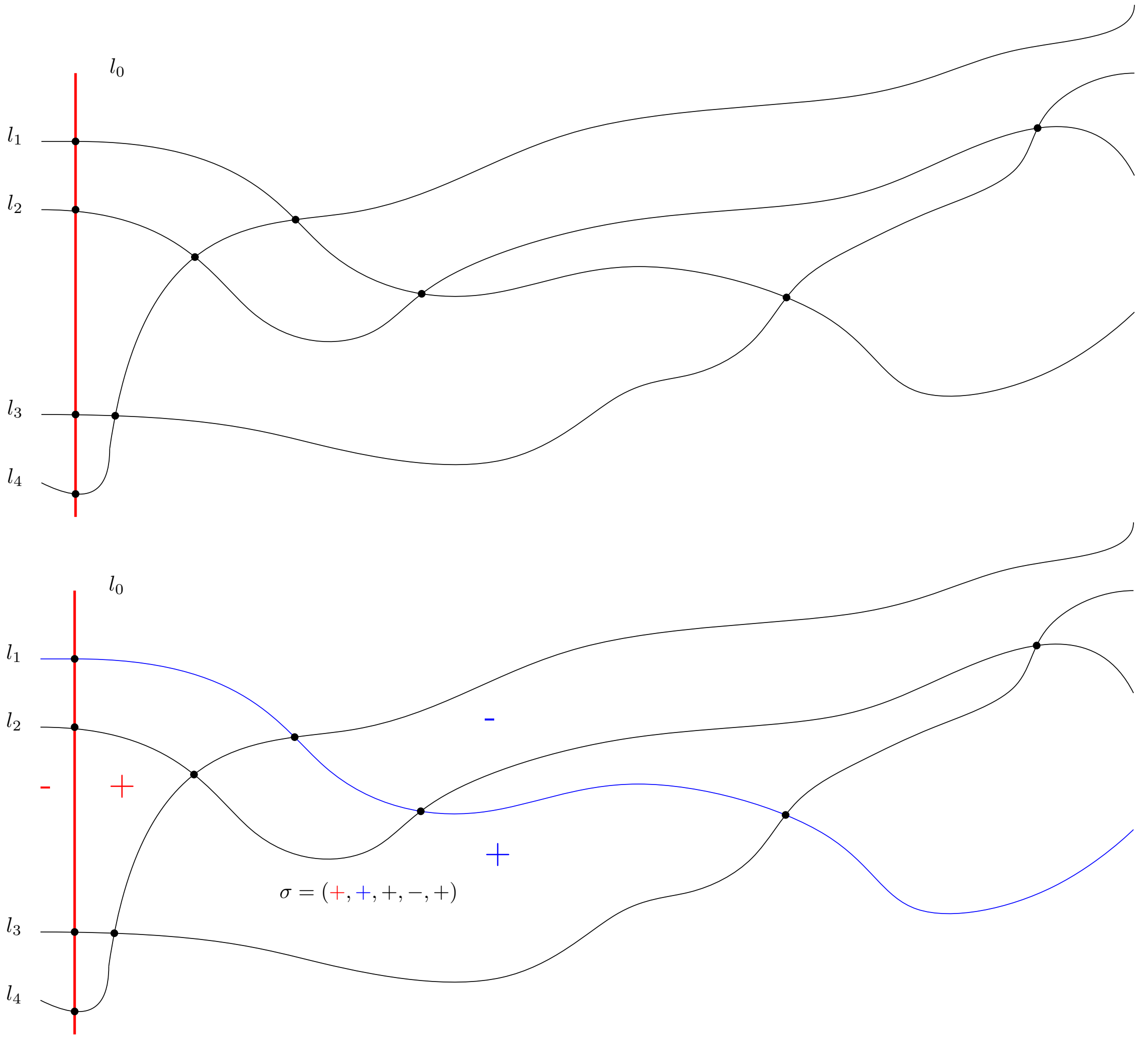

Figure 3.2.: Top: pseudoline arrangement with combinatorial description $(4, 2, 3), (4, 1, 3), (4, 1, 2), (3, 2, 1)$. Below: Orientated line arrangement with one sign vector.

EuclideanSimpleStretchability:
Input: Combinatorial description D of a simple pseudoline arrangement with border line l_0.
Problem: Does a line arrangement L_e in Euclidean space exist that realizes D?

For later proofs in Chapter 4, we need another way of describing a line arrangement combinatorially which follows Kang and Müller [KM12]. This time, we do not specify the order of the intersection points, but we characterize the regions into which the plane is divided. For that, we *orientate* the line arrangement in the following way (we still keep the requirement of the border line l_0): Each line divides the plane into two half planes, which we denote by either $-$ or $+$. For the line l_0, the half plane without intersection points is denoted by $-$, the other one by $+$. For each of the other lines, we can go along them starting at the intersection with l_0 and going towards the other intersection points. The half plane that is to the left in this characterization is denoted by $-$, the other one by $+$. Note that every step we did is applicable in both Euclidean and hyperbolic space. Now, every point p in the plane can be described by a *sign vector* $\sigma(p) \in \{-, 0, +\}^{n+1}$ where $\sigma_i(p)$ describes if p is on the line l_i (0) or in the left ($-$) or right ($+$) half plane.

The *orientated combinatorial description* of the line arrangement now is the set of all sign vectors. This process is also shown in Figure 3.2.

3.1.3. Hyperbolic SimpleStretchability

We now define SimpleStretchability in the hyperbolic setting. We obviously can consider line arrangements in the hyperbolic plane similarly to the Euclidean case. This has been done by Dress et al. [DKM02], but not in the context of the decision problem SimpleStretchability. The requirements for simplicity of line arrangements, that every pair of lines must intersect and no more than two lines intersect in a point, can be formulated for hyperbolic geometry as well. However, we cannot simply transfer the concept of pseudolines to the hyperbolic space as it is not clear how to transfer the concept of x-monotonocity into the different models of the hyperbolic plane. Because of that, we use the same input as in the Euclidean case: combinatorial descriprions of pseudoline arrangements that can be represented by x-monotone Euclidean curves. For the description of the intersections, we need to again include the border line l_0. This poses the question if we exclude possible simple hyperbolic line arrangements if we force the border line to be included. We answer that question negatively in Lemma 3.26, but find that this does not hold for general hyperbolic line arrangements in Section 3.4. The problem can now be defined in the same way as in the Euclidean case:

HyperbolicSimpleStretchability:
Input: (Orientated) Combinatorial description D of a simple pseudoline arrangement with border line l_0.
Problem: Does a line arrangment L^h in hyperbolic space exist that realizes D?

As already done in the problem definition, we clarify if a line arrangement, or any structure that has to be defined, is Euclidean or hyperbolic by adding e or h to the notation, L^e for Euclidean line arrangements and L^h for hyperbolic line arrangements.

In this chapter, we use the Poincaré disk model as our model for hyperbolic geometry. As a short reminder, in this model the hyperbolic plane is embedded in the interior of a unit circle in the Euclidean plane, which we call D. Lines are either represented by circles c that are orthogonal to D, or lines through the origin. Technically, the lines representation is only the segment of c inside of D, but for simplicity we often identify those segments with the whole circle. Additionally, we always choose the origin O in a way that the border line is the vertical line through O and every intersection point is to the right of l_0. Additionally, we want every line other than l_0 to be represented by a circle, not by other origin lines. An example of such a hyperbolic line arrangement is given in Figure 3.3.

Now that we have defined HyperbolicSimpleStretchability, we can investigate its complexity. The Euclidean version is $\exists\mathbb{R}$-complete as mentioned above. The question now is if the hyperbolic version is equally complex. In Section 3.2, we answer that question affirmatively and obtain the even stronger result that the problem does not really change as a combinatorial description is realizable in Euclidean space if and only if it is realizable in hyperbolic space. This intuitively seems right as the difference between the geometrys is that hyperbolic geometry allows for additional parallel lines, but we do not have any parallel lines in our line arrangement as one requirement is that each pair of lines must intersect. Accordingly, the main result of this chapter is the following:

Theorem 3.3. *Let D be a combinatorial description of a* SimpleStretchability *instance. Then D is realizable by a line arrangement L^e in the Euclidean plane if and only if D is realizable by a line arrangement L^h in the hyperbolic plane.*

The key observation for the proof of the chapter is the following (on an intuitive basis): If we only look locally at sufficiently small parts of the hyperbolic plane, lines and distances

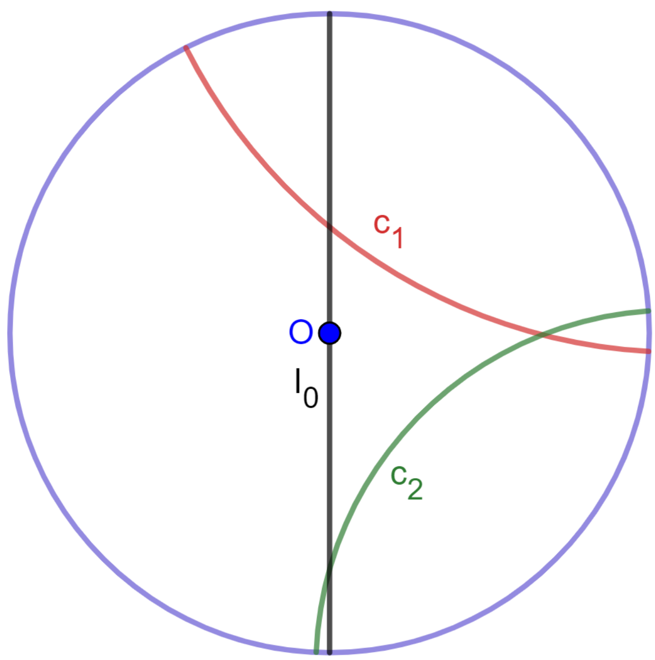

Figure 3.3.: A hyperbolic line arrangement of three lines.

are almost similar to their Euclidean counterparts, and become even more similar the more we "zoom in". This property is what the proof of Theorem 3.3 relies on. We need scaling operations, for hyperbolic and Euclidean line arrangements, that manage to force any line arrangement, more precise its intersection points as lines are infinite, into an arbitrarily small area to use the similarity there. For the Euclidean case, normal Euclidean scaling does the trick. In the hyperbolic case, it is not obvious how such a way of scaling can be obtained.

The plan for the rest of chapter is the following: We first introduce a scaling method for hyperbolic line arrangements and prove its correctness. Then we prove Theorem 3.3. As the instances are realizable in Euclidean space if and only if they are realizable in hyperbolic space and EuclideanSimpleStretchability is $\exists\mathbb{R}$-complete, it follows that HyperbolicSimpleStretchability is also $\exists\mathbb{R}$-complete. This will be the base of the reductions for the rest of the thesis.

3.2. Hyperbolic Scaling of Simple Pseudoline Arrangements

Now that we have motivated why we need a way of scaling down hyperbolic line arrangements, we define the key characteristics such a scaling operation needs to have to be useful for us. For the rest of this chapter, we always use the version of combinatorial descriptions that characterizes pseudoline arrangements by their order of intersections. The first property that the scaling operation obviously needs to preserve is the combinatorial description of the line arrangement. Additionally, as mentioned above, we need to reduce the space in which the intersection points lie by an arbitrary amount. This suffices as the order intersection points uniquely defines the description of the line arrangement. In order to adress this space easily, we use the immediate surrounding of the origin. The other goal for the operation thus is to move every intersection point of lines arbitrarily close to the origin.

At this point, we need to clarify how we use the Poincaré disk model in this chapter. The model displays the hyperbolic plane as part of the Euclidean plane, specifically as the interior of a unit disk. However, we are using the disk as part of the Euclidean plane around it and do not constrain our observations to only the interior of the disk. Subsequently, our argumentations and observations are of Euclidean nature. As lines in the Poincaré disk are represented by circle segments for us (apart from l_0), we are also interested in the centers of those circles which is outside of D. We therefore use Euclidean terminology and distances for our proofs in this chapter.

Before we start to think about candidates of scaling operations, we present the notation we use in this proof and the rest of the thesis if we talk about hyperbolic line arrangements. In this section, we only consider hyperbolic line arrangements and thus drop the additional annotations to distinguish Euclidean and hyperbolic geometry. For any hyperbolic line l that is not an origin line, we need the following characteristics.

- c: the circle that represents line l in the arrangement
- r: the radius of c
- M: the center of c
- q: the ray starting at O and going through M
- α: the angle between q and l_0 in O
- P: the intersection point of c and q inside of D

We often use c directly as the name of a hyperbolic line l^h in a line arrangement L^h and sometimes call α the "angle of c" for simplicity. Doing so also implies that the line is not an origin line, as those are not represented by circles. If we need to differentiate between lines, we add the corresponding indices to our notations. The notations are visualized in Figure 3.4. We are also interested in the Euclidean distance of these points to the origin O as that is what our scaling operation is supposed to minimize. For that reason, we denote the distance of any point A to the origin as $d_A := d(O, A)$.

3.2.1. First Try: Euclidean Scaling

In order to find a scaling method that fulfills those requirements, we first start with a naive approach: We could try to just scale down each of the intersection points S in a Euclidean sense: multiply their distance d_S to the origin with a scaling factor while keeping their angles constant. This way, the hyperbolic lines remain circles c and their order of intersections remains identical while the intersection points move arbitrarily close to the origin, seemingly fulfilling the requirements we have for the scaling operation. However, as the Poincaré disk D remains a unit circle and invariant to the scaling, the circles are not orthogonal to D anymore, which means that scaling does not result in a valid hyperbolic line arrangement.

As a way to use a similar idea of Euclidean scaling, but define valid hyperbolic line arrangements for each scaling factor, we could also try to move each of the points P closer to the origin. Again, we define a scaling factor $\mu \in (0, 1)$ and linearly scale down each P by setting $d_P(\mu) = \mu \cdot d(P)$. We then use $P(\mu)$ to define a new circle c_μ that intersects the ray q of angle α in $P(\mu)$, but is also orthogonal to D. We know that this would not pose any problems when scaling down Euclidean lines, however, it is not immediately clear that this produces unique circles for every μ. We show that this operation is indeed well-defined in the following lemma:

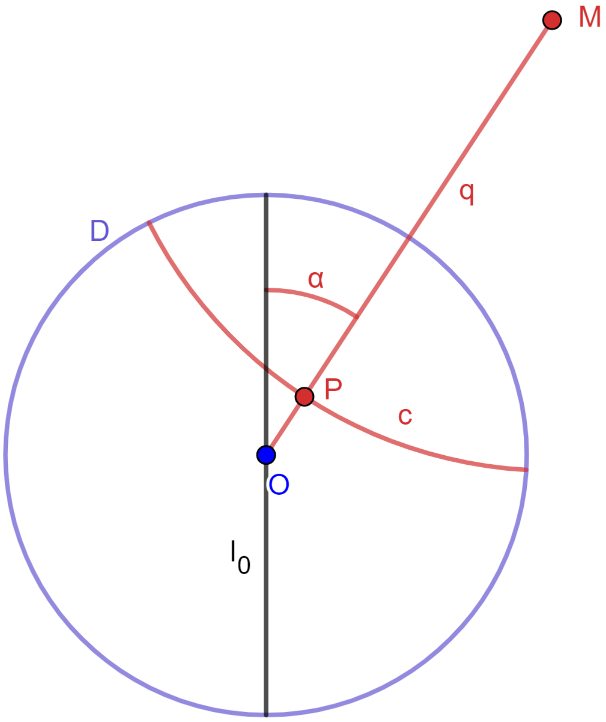

Figure 3.4.: Left: The notations for each line c. Right: The notations for the intersection point of c_1 and c_2.

Lemma 3.4. *For each angle α and distance $d \in (0, 1)$, there is one and only one hyperbolic line c with $d_P = d$ and angle α.*

Proof. Let $d \in (0, 1)$ and α be a fixed angle. We have to prove two statements: There is a circle c with angle α and $d_P = d$, and these requirements uniquely defines the radius of the circle. For that, we think about what defines c: In order to represent a hyperbolic line in the Poincaré disk, c has to be orthogonal to D. This means that there has to be a right triangle Δ with sides of length d_M, r and 1, depicted in Figure 3.5, where $d_M = r + d_P$. If we now fix $d = d_P$, there is only one radius r that fulfills the equation $(r + d)^2 = d_M^2 = r^2 + 1 \Leftrightarrow 2rd + d^2 = 1 \Leftrightarrow r = (1 - d^2)/2d$. M thus is unambigously defined by angle α and $d_M = r + d_P = (1 - d^2)/2d + d = (1 + d^2)/2d$ and there is one and only one circle c with center M and radius r. □

Note that this proof also gave us formulas for the radius and distance to the center of c:

Corollary 3.5. *For each angle α and distance $d = d_P$, the unique radius of c is given by $r = (1 - d^2)/(2d)$ and the distance to the center is given by $d_M = (1 + d^2)/(2d)$.*

This shows us that the scaling is indeed well-defined. It also fulfills the second requirement we had for the scaling operation as it clearly brings the hyperbolic lines and thus the intersection points arbitrarily close to the origin. However, it is not consistent and in many cases changes the combinatorial description of L^h. One of those cases is depicted in Figure 3.6. Nevertheless, we use Euclidean scaling later in Section 3.3 in a different context.

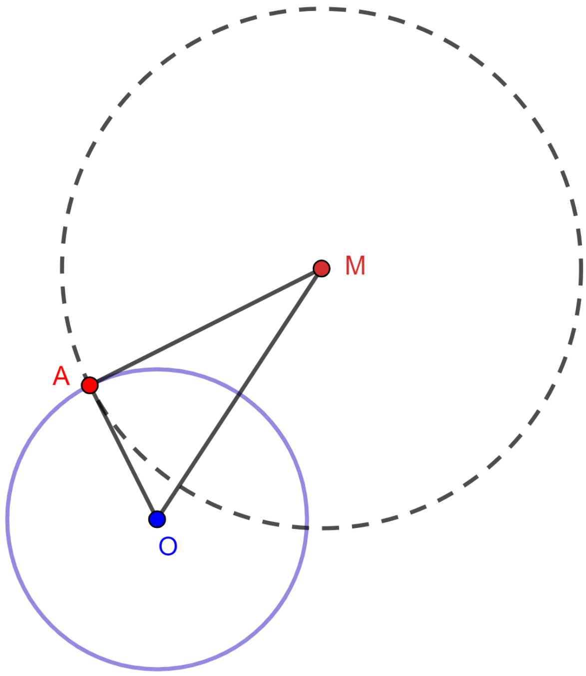

Figure 3.5.: The triangle Δ that uniquely defines the circle c.

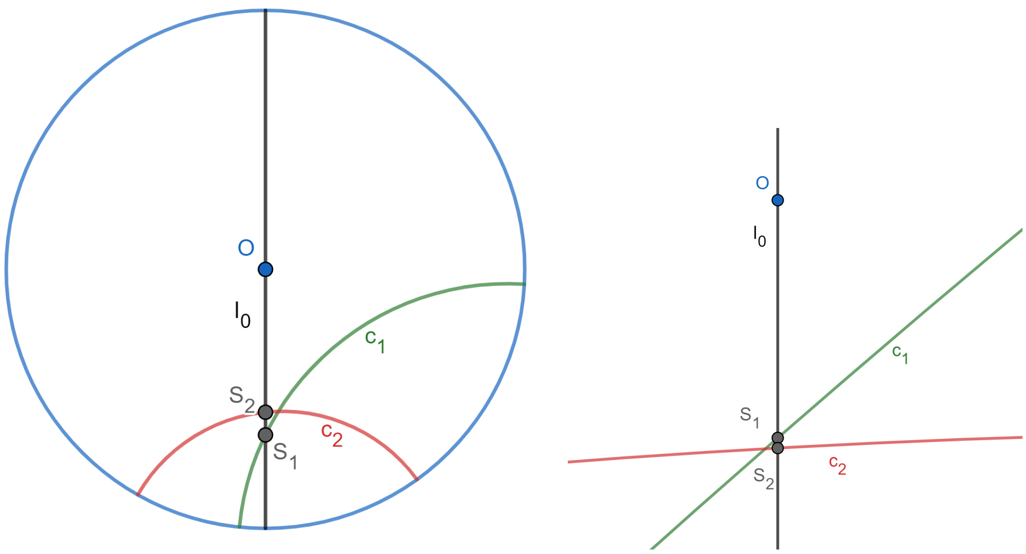

Figure 3.6.: Left: Hyperbolic line arrangement with three lines. Right: Line arrangement for $\mu = 0.2$, the order of intersections on l_0 has changed.

3.2.2. Geometric Observations

The last subsection shows that we need to dive deeper into hyperbolic line arrangements in the Poincaré disk in order to find a scaling method that fits both the requirements. For that, we now study characteristics of the hyperbolic lines and their intersection points. We then use the results to define a scaling method that does not change the combinatorial description of L^h while shrinking the area in which the intersection points are situated.

Our first observation clarifies the relation between d_M and r, which we need for the following proofs:

Lemma 3.6. *For any hyperbolic line c with center M in the Poincaré disk, the radius of c is given by $r = \sqrt{d_M^2 - 1}$.*

Proof. Let c be a hyperbolic line. As in the proof of Lemma 3.4, we again use the right triangle Δ. This time, we simply apply Pythagoras' theorem to obtain: $d_M^2 = r^2 + 1 \Leftrightarrow r = \sqrt{d_M^2 - 1}$. □

We now consider in what cases two hyperbolic lines c_1 and c_2 intersect. For that, we first need to define the point H for two hyperbolic lines $c_1 \neq c_2$, not necessarily intersecting: If O, M_1 and M_2 are not collinear, we define H as the intersection point of the altitude of the triangle ΔOM_1M_2 (the line through O that is orthogonal to M_1M_2) and the line defined by M_1 and M_2. If the points are collinear, this triangle is not defined and we set $H := O$. As we see in the following lemmas, H is closely related to the intersection point of hyperbolic lines c_1 and c_2.

Lemma 3.7. *Two hyperbolic lines $c_1 \neq c_2$ intersect if and only if $d_H > 1$.*

Proof. Let c_1 and c_2 be hyperbolic lines and H defined above. Note that c_1 and c_2 intersect if and only if $r_1 + r_2 > d(M_1, M_2)$ as M_1 and M_2 and thus c_1 and c_2 are closest to each other on the line connecting M_1 and M_2.

Case 1: $H = O$: This case occurs only if O, M_1 and M_2 are collinear. In that case, M_1 and M_2 can either be on the same side or on different sides of O. If they are on different sides, c_1 and c_2 cannot intersect because $r_1 + r_2 < d_{M_1} + d_{M_2} = d(M_1, M_2)$ as M_1 and M_2 are situated on the same origin line. If they are on the same side of O, then $\alpha_1 = \alpha_2$. Assume that c_1 and c_2 intersect. Due to symmetry, for any intersection point S of c_1 and c_2 on the inside of D that is not on $q = q_1 = q_2$, we find a second intersection point, S' also inside of D by mirroring S on q. This contradicts the fact that two hyperbolic lines can at most intersect in one point. Thus, c_1 and c_2 have to intersect in $P = P_1 = P_2$ as it is the only point of c_1 and c_2 on q. However, in that case Lemma 3.4 states that there is only one circle c with angle $\alpha_1 = \alpha_2$ and distance d_P. We conclude that $c_1 = c_2$, contradicting the starting assumptions.

Case 2: $H \neq O$: We consider the right triangles ΔOHM_1 and ΔOHM_2 as depicted in Figure 3.7 and compute $d(M_i, H)$ for $i = 1, 2$ via Pythagoras' theorem:

$$d(M_i, H)^2 + d_H^2 = d_{M_i}^2 \Leftrightarrow d(M_i, H) = \sqrt{d_{M_i}^2 - d_H^2}$$

Note that $d(M_1, M_2) = d(M_1, H) + d(H, M_2) = \sqrt{d_{M_1}^2 - d_H^2} + \sqrt{d_{M_2}^2 - d_H^2}$ and that c_1 and c_2 intersect if and only if $r_1 + r_2 > d(M_1, M_2)$. In Lemma 3.6, we have shown that $r_i = \sqrt{d_{M_i}^2 - 1}$. Together, this yields that c_1 and c_2 intersect if and only if

$$\sqrt{d_{M_1}^2 - 1} + \sqrt{d_{M_2}^2 - 1} > \sqrt{d_{M_1}^2 - d_H^2} + \sqrt{d_{M_2}^2 - d_H^2}.$$

We conclude that c_1 and c_2 intersect if and only if $d_H^2 > 1 \Leftrightarrow d_H > 1$ which completes the proof. $\square$

We see that, if lines c_1 and c_2 intersect, O, M_1 and M_2 cannot be collinear. We need H again later, so we fix the following notations for intersecting hyperbolic lines $c_1 \neq c_2$ (depicted in Figure 3.7):

- S: the intersection point of c_1 and c_2
- γ: the angle of S
- H: the point where the altitude of ΔOM_1M_2 intersects the line defined by M_1 and M_2

With that in mind, we now look at characteristics of the intersection points in the Poincaré disk, as we need to keep them in the same order while scaling:

Lemma 3.8. *Let c_1, c_2 be two hyperbolic lines that intersect in point S. Then S lies on the origin line defined by H and $d_S = d_H - \sqrt{d_H^2 - 1}$.*

Proof. Let c_1, c_2 be two intersecting hyperbolic lines. In order to prove that S is on the same origin line as H, we consider the triangle ΔOM_1M_2. The points A that are on the same origin line as H have fulfill the equation $d(A, M_1)^2 - d(A, M_2)^2 = d(M_1)^2 - d(M_2)^2$ as in that case, the line through A that is orthogonal to M_1M_2 includes H as well. That is the case for S as $d(S, M_i)^2 = r_i^2 = d(M_i)^2 - 1$ for $i \in \{1, 2\}$ where the last part is derived from Lemma 3.6. This delivers $d(S, M_1)^2 - d(S, M_2)^2 = d_{M_1}^2 - 1 - d_{M_2}^2 + 1 = d_{M_1}^2 - d_{M_2}^2$. Thus, S is indeed on the altitude of the triangle ΔOM_1M_2. The situation is depicted in Figure 3.7.

For the distance d_S, we consider the triangle ΔSHM_1. This is obviously a right triangle with right angle in H. We compute the distance $d(S, H)$ between S and H, d_S is then d_H minus that distance. For that, we can use Pythagoras' theorem:

$$r_1^2 = d(S, H)^2 + d(M_1, H)^2 = d(S, H)^2 + d_{M_1}^2 - d_H^2$$
$$\Leftrightarrow d(S, H)^2 = r_1^2 - d_{M_1}^2 + d_H^2 = \sqrt{d_{M_1}^2 - 1}^2 - d_{M_1}^2 + d_H^2 = d_H^2 - 1$$

Here, we computed $d(M_1, H)$ via applying Pythagoras' Theorem again on the right triangle ΔOHM_1. Thus, the distance between S and H is $\sqrt{d_H^2 - 1}$ and $d_S = d_H - \sqrt{d_H^2 - 1}$. $\square$

This clarifies the situation for the intersection points between two hyperbolic lines represented by circles. In the case of intersection points C_0 between lines c and l_0, the intersection points trivially lie on origin line l_0. We define a right triangle ΔOMH_0 with right angle in H_0, the point on l_0 with $d_{H_0} = \cos(\alpha)d_M$ (shown in Figure 3.7). Inserting $d_{H_0} = \cos(\alpha)d_M$ into the second part of the last proof, we obtain the corresponding result to Lemma 3.8 for d_{S_0}:

Corollary 3.9. *Let c be a hyperbolic line intersecting l_0. The intersection point S_0 has distance $d_{S_0} = \cos(\alpha)d_M - \sqrt{\cos(\alpha)^2 d_M^2 - 1}$ to the origin.*

Note that if we keep the lines defined by M_1 and M_2 parallel throughout the scaling, the intersection point stays on the same origin line. We later uphold that when defining the scaling operation and the observation is important for proving its correctness. Additionally, if we increase the d_H, the distance of S and O decreases, which is the desired result. So our scaling operation should, in some way, scale back the centers of the circles on the same ray q starting at the origin. Our last observation clarifies that this is indeed a well-defined process:

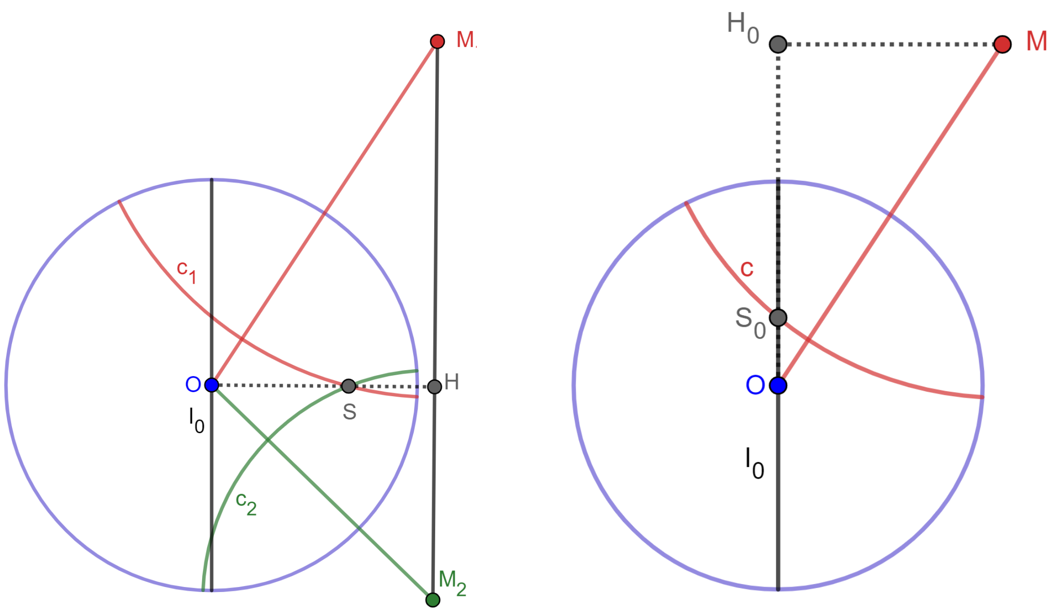

Figure 3.7.: Left: Definition of point H in triangle OM_1M_2. Right: The triangle used to compute d_{S_0}.

Lemma 3.10. *For each angle α and distance $d \geq 1$, there is one and only one hyperbolic line c with center M, $d_M = d$ and α as the angle of c.*

Proof. Let $d \geq 1$ and α be fixed. M is unambigously defined by the requirements. We now need to show that there is only one circle with center M that is orthogonal to D. This translates to there having to be two right triangles with points O, M and a third point on D that completes the right triangle. Now Thales' theorem tells us that the points that complete O and M to a right triangle lie on the circle c_T with diameter OM. There are exactly two of those points with identical distances to O and M. Due to symmetry, we pick one of them and call it A. Now we have a center M and an additional point on the circle A, which defines the circle c unambiguously. We depict the proof in Figure 3.8. $\square$

3.2.3. Hyperbolic Scaling

These observations now allow us to define our hyperbolic scaling operation. As observed, we need to linearly scale back the centers of the circles to bring the intersection points closer to the origin while leaving them on straight lines. In order to do that, we use a scaling factor $\lambda \geq 1$, where $\lambda = 1$ describes the initial line arrangement (Note that we use different names of scaling factors in order to differentiate between Euclidean (μ) and hyperbolic (λ) scaling). We scale back the centers M on ray q by multiplying d_M with λ to achieve that. We denote points, structures and distances, for example a hyperbolic line c, that are scaled with factor λ by $c(\lambda)$. If we do not give a specific scaling factor, the initial line arrangement is described. The geometric observations we made in the last section ensure the correctness of this approach.

Formally, we start with a hyperbolic line arrangement L^h, consisting of $n+1$ lines in the Poincaré disk. The border line l_0 includes the origin and is not changed by our

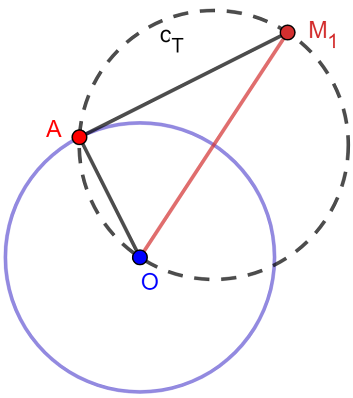

Figure 3.8.: The circle c_T whose intersection point A with the Poincaré disk uniquely defines the circle c.

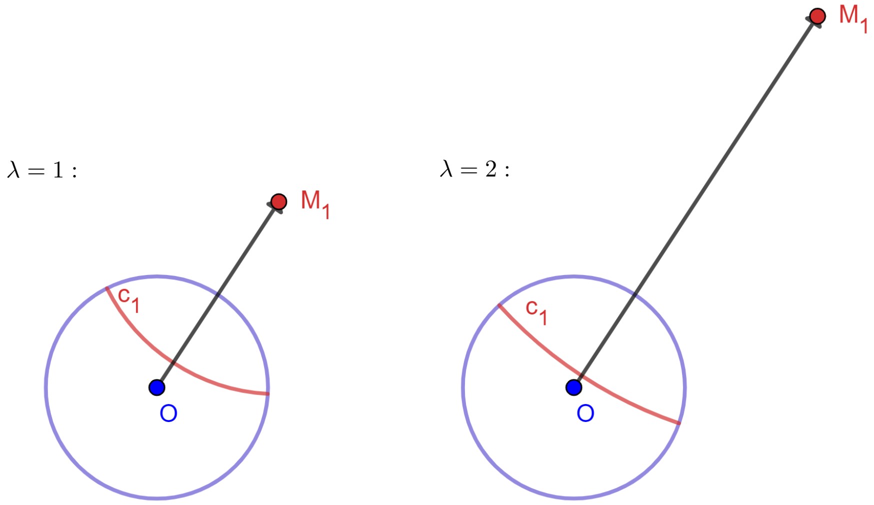

Figure 3.9.: Left: A hyperbolic line with center M_1. Right: The scaled line for $\lambda = 2$.

scaling operation. For the other lines l represented by circles c, we consider their center $M = (\alpha, d_M)$ where x and y are the Euclidean coordinates of M. For each scaling factor $\lambda \in \mathbb{R}, \lambda \geq 1$, we define a new line arrangement $L(\lambda)$ by placing each center $M(\lambda)$ at $(\alpha, \lambda \cdot d_M)$. The lines $c(\lambda)$ of the new arrangement are defined by the unique circle from Lemma 3.10 with center $M(\lambda)$ that is orthogonal to the Poincaré disk. This process is shown in Figure 3.9.

Before we prove that this operation indeed produces a valid simple line arrangement with identical combinatorial description for each scaling factor λ, we first show the effects of the scaling. The result we need to generate is that the intersection points of the lines get closer to the origin with increasing scaling factor and converge towards the origin. As the distances between the intersection points get smaller, the line arrangement becomes more and more similar to a Euclidean line arrangement, which is the goal of the operation. In Figure 3.10, we depict the line arrangement from Figure 3.3 for scaling factors $\lambda = 1, 2$ and 4. Indeed, the intersection points are moved to the center on a straight line and the combinatorial description is not changed, which we now formally prove.

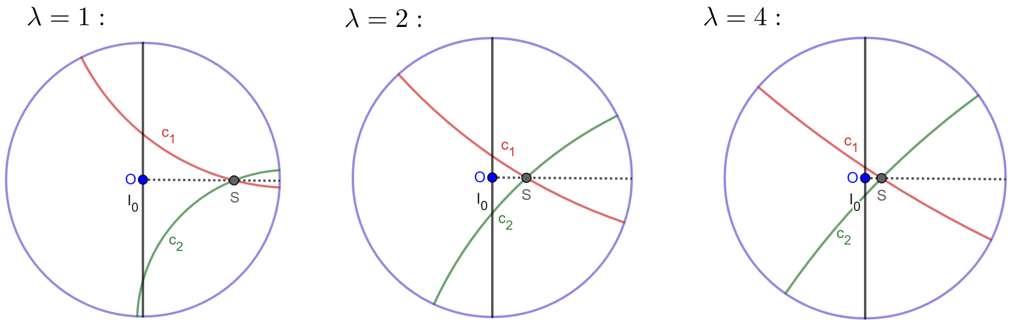

Figure 3.10.: The line arrangement for scaling factor 1, 2 and 4.

We use the geometric observations to prove the correctness of the scaling operation. In this context, correctness means that, for any given hyperbolic line arrangement L, the scaling operation indeed produces a valid line arrangement $L(\lambda)$ for every scaling factor $\lambda \geq 1$. Additionally, the combinatorial description of each of the $L(\lambda)$ has to be identical.

In order to prove those statements, we first use another lemma to formally state the impacts of the scaling on the lines c and intersection points S:

Lemma 3.11. *Let L be a simple hyperbolic line arrangement, c_1 and c_2 two hyperbolic lines in L intersecting in S and $\lambda \geq 1$ a scaling factor. Then:*

1. *$c_1(\lambda)$ and $c_2(\lambda)$ intersect in $S(\lambda)$.*
2. *$S(\lambda)$ is part of the origin line defined by S.*
3. *$d_S(\lambda) = \lambda \cdot d_H - \sqrt{(\lambda^2 \cdot d_H^2 - 1)}$.*

Proof. Let L be a hyperbolic line arrangement and $\lambda \geq 1$. We show the statements for two lines c_1, c_2 represented by circles, the statements for intersections with l_0 can be shown similarly. Our main tool in this proof is an application of the intersect theorem: We compute the relative distances of $M_1(\lambda)$ and $M_2(\lambda)$:

$$\frac{M_1(\lambda)}{M_2(\lambda)} = \frac{\lambda \cdot M_1}{\lambda \cdot M_2} = \frac{M_1}{M_2}.$$

From that, we can conclude multiple results. First, the line through $M_1(\lambda)$ and $M_2(\lambda)$ is parallel to the one through M_1 and M_2 due to the intercept theorem and thus $d_H(\lambda) = \lambda \cdot d_H$. We know that $d_H > 1$ from Lemma 3.7 as c_1 and c_2 intersect. As $\lambda \geq 1$ is also true, $d_H(\lambda)$ is strictly greater than 1 and thus $c_1(\lambda)$ and $c_2(\lambda)$ intersect due to Lemma 3.7.

In order to prove the second statement, we again use that the line through $M_1(\lambda)$ and $M_2(\lambda)$ is parallel to the one through M_1 and M_2. Because of that, the altitudes in the triangles ΔOM_1M_2 and $\Delta OM_1(\lambda)M_2(\lambda)$ are identical. Due to Lemma 3.8, both S and $S(\lambda)$ are on that altitude and thus on the same origin line.

The third statement follows from inserting $d_H(\lambda) = \lambda \cdot d_H$ into Lemma 3.8. □

Using that, we now prove the correctness of the hyperbolic scaling operation:

Theorem 3.12. *For every scaling factor $\lambda \geq 1$ and given hyperbolic line arrangement L, $L(\lambda)$ is a valid simple hyperbolic line arrangement and the combinatorial description of L and $L(\lambda)$ is identical.*

We have already proven the geometrical backgrounds of the observations we need for this proof. We need to show three statements: The scaling operation produces a valid and unique hyperbolic line arrangement for each $\lambda \geq 1$, the line arrangement $L(\lambda)$ is simple and the combinatorial description of $L(\lambda)$ is identical to the one of L_1.

The proof of the first statement follows directly from Lemma 3.10. For the second and third statement, our main tool is Lemma 3.11.

Lemma 3.13. *If L is a simple line arrangement, then the line arrangement $L(\lambda)$ also is a simple line arrangement for every $\lambda \geq 1$.*

Proof. Simple line arrangements have two defining characteristics: Each pair of lines intersect, and no more than two lines intersect in the same point. Let L be such a simple hyperbolic line arrangement. The first property directly follows from the first statement of Lemma 3.11. For the second characteristic, we use the third statement of Lemma 3.11: Assume that there is an intersection point S where at least three lines and corresponding circles intersect. The distance formula $d_S(\lambda) = \lambda \cdot d_H - \sqrt{(\lambda^2 \cdot d_H^2 - 1)}$ is injective for $\lambda \geq 1$ and only depends on the initial d_H. It follows that three or more lines can only intersect in the same point if they have the same initial d_H, so they have intersected in L already, but L is simple per definition. This concludes the proof. □

Now that we have shown that our line arrangement $L(\lambda)$ stays a simple line arrangement for every $\lambda \geq 1$, all that is left to show is that the combinatorial description also remains identical:

Lemma 3.14. *Let L be a hyperbolic line arrangement. For each scaling factor $\lambda \geq 1$, the line arrangement $L(\lambda)$ has the same combinatorial description as the line arrangement L.*

Proof. Let L be a hyperbolic line arrangement and $\lambda \geq 1$. In order to prove the statement, we need to show the two defininf properties of the line arrangement: The order of the intersection points on the border line l_0 stays the same, and for each line c, the order of the intersecting lines on $c(\lambda)$ is the same as on c. As those two properties define a combinatorial description, if both are met we have the same combinatorial description.

Both properties can be shown in the same way: Assume that the combinatorial description is not identical. We consider the line c where the combinatorial description has first changed, meaning that two intersection points S, S' changed their order. As our operation is continuous, this means that there is a λ^* for that $S(\lambda^*)$ and $S'(\lambda^*)$ fall on the same point. For that λ^*, at least three lines intersect in a point, thus the line arrangement $L(\lambda^*)$ is not simple, contradicting Lemma 3.13. This concludes the proof. □

Now that we have proven the necessary components, we can conclude Theorem 3.12:

Proof of Theorem 3.12. Let L be a hyperbolic line arrangement and $\lambda \geq 1$ a fixed scaling factor. As described previously, we need to prove the three components of the statement: $L(\lambda)$ is a valid hyperbolic line arrangement, $L(\lambda)$ is simple and has the same combinatorial descriprion as L.

For the first part, note that l_0 is not changed by the scaling operation. For every other line c, Lemma 3.10 states that there is exactly one circle with center $M(\lambda)$ that is orthogonal to D, which means that $c(\lambda)$ is unambigous and a valid hyperbolic line in the Poincaré disk model.

Afer clarifying the validity of the line arrangement $L(\lambda)$, the next step is to show that the arrangement stays a simple arrangement. This is the result of Lemma 3.13.

Finally, the main property of our scaling operation is left. The combinatorial description of L has to remain identical throughout the scaling operation. For that, in Lemma 3.14, we show that this is the case for every scaling factor. This concludes the proof of Theorem 3.12. □

With that, we now have achieved a valid definition of HyperbolicSimpleStretchability and a method to force hyperbolic line arrangements into arbitrarily small areas where they resemble Euclidean line arrangements closely. This indicates that the problems are equivalent. Indeed, we show the equivalence in the next section.

3.3. HyperbolicSimpleStretchability is $\exists\mathbb{R}$-Complete

In this section, we use our scaling operation to show the main result of this chapter that directly implies the $\exists\mathbb{R}$-completeness of HyperbolicSimpleStretchability:

Theorem 3.3. *Let D be a combinatorial description of a* SimpleStretchability *instance. Then D is realizable by a line arrangement L^e in the Euclidean plane if and only if D is realizable by a line arrangement L^h in the hyperbolic plane.*

We need to show the following two directions: Given a Euclidean line arrangement L^e, we need to transform that into a hyperbolic line arrangement L^h with the same combinatorial description and vice versa. The idea for both directions is identical: We use that, for small areas, the hyperbolic and Euclidean plane are similar. In order to do that, we scale down the original line arrangement via the Euclidean or hyperbolic scaling operation, respectively, until the line arrangement is similar enough to the other geometry. One challenge is that we need to translate hyperbolic lines, which correspond to circles in the Poincaré disk, into Euclidean lines. For that, we use certain tangents that the circle converges towards. As the resulting structures are identical for both directions, we introduce them now.

As a reminder, we represent hyperbolic lines by circles c with center M and radius r. We also use the ray q from O through M and the corresponding angle α to l_0 and intersection point P of q and c, as depicted in Figure 3.4.

In both cases, we use the same structure of Euclidean and hyperbolic lines that we now define: P will be the common point of the hyperbolic line c and the Euclidean line $l^e = l$ (as we only use circles for hyperbolic lines, we drop the additional annotation e for Euclidean lines). We define l as the orthogonal line to q through P, which is the tangent of c in P, and define the corresponding Euclidean line arrangement by the collection of those lines, together with the extension of l_0. This definition is depicted in Figure 3.11.

In order to proceed, we also need notation for intersection points of Euclidean lines. For that, let $l_1 \neq l_2$ be two intersecting lines. We need the following definitions:

- T: the intersection point of l_1 and l_2
- β: the angle of T and l_0 in O

Again, if we need to clarify which lines T and β belong to, we add indices but try to avoid that as much as possible.

Before we start with proving the theorem, we again need geometrical observations to express the intersection points S of hyperbolic lines c_1 and c_2 and T of Euclidean lines l_1 and l_2 using the information we have about c_1, c_2, l_1 and l_2. We do that in the following two lemmas, starting with the intersection point of Euclidean lines. Note that, as both S

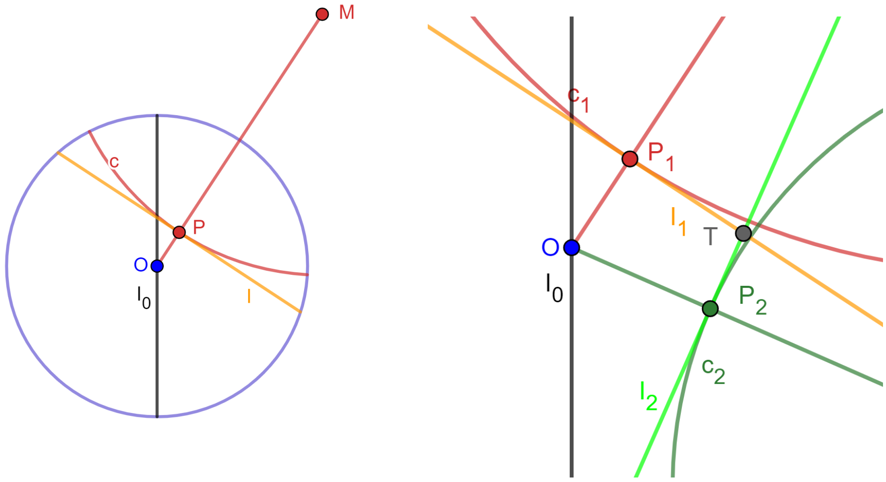

Figure 3.11.: Left: Corresponding lines l and$c$. Right: Definition of point T.

and T are to the right of l_0, their angles are between 0 and π and thus there is a bijective between the angles and their cosines. We thus work with $\cos(\beta)$ and $\cos(\gamma)$ rather than β and γ.

Lemma 3.15. *For any two Euclidean lines l_1, l_2, the intersection point T can be described in polar coordinates $T = (\beta, d_T)$ with*

$$\cos(\beta) = \frac{\sin(\alpha_1)d_{P_2} - \sin(\alpha_2)d_{P_1}}{d(P_1, P_2)}, \quad d_T = \frac{d(P_1, P_2)}{\sin(\alpha_1 - \alpha_2)}.$$

Proof. Let l_1, l_2 be Euclidean lines intersecting in T. The argument is based on the following observation: The points O, P_1 P_2 and T build two rectangular triangles, both with OT as its hypotenuse. From that, we conclude that the four points lie on a circle c_T with center M_T on OT due to Thales' theorem. The origin as well as $P_1 = (\alpha_1, d_{P_1})$ and $P_2 = (\alpha_2, d_{P_2})$ are known and fully define c_T. For readability, we write $d_{P_i} = d_i$ in the scope of this proof. We compute $M_T = (x_M, y_M)$, and consequently the point $T = (x_T, y_T)$. As M_T is the center point of the segment OT, x_T and y_T are given by $x_T = 2x_M$ and $y_T = 2y_M$. The situation is depicted in Figure 3.12.

In order to compute x_M and y_M, we use a system of equalities that describe the center of a circle when given three points $P = (x, y)$ that are situated on the circle. This process gives us cartesian coordinates for T, from which we compute the polar coordinates we need in the following sections. As the process is long and technical, we give the detailed computation in Lemma 6.1 in the appendix. Here, we resume with its result: T can be described by cartesian coordinates (x_T, y_T) with

$$x_T = \frac{\cos(\alpha_1)d_2 - \cos(\alpha_2)d_1}{\sin(\alpha_1 - \alpha_2)}, \quad y_T = \frac{\sin(\alpha_1)d_2 - \sin(\alpha_2)d_1}{\sin(\alpha_1 - \alpha_2)}.$$

From that, we now compute the polar coordinates, angle β and distance d_T, starting with d_T: Due to Pythagoras' theorem, d_T can be obtained from x_T and y_T in the following way:

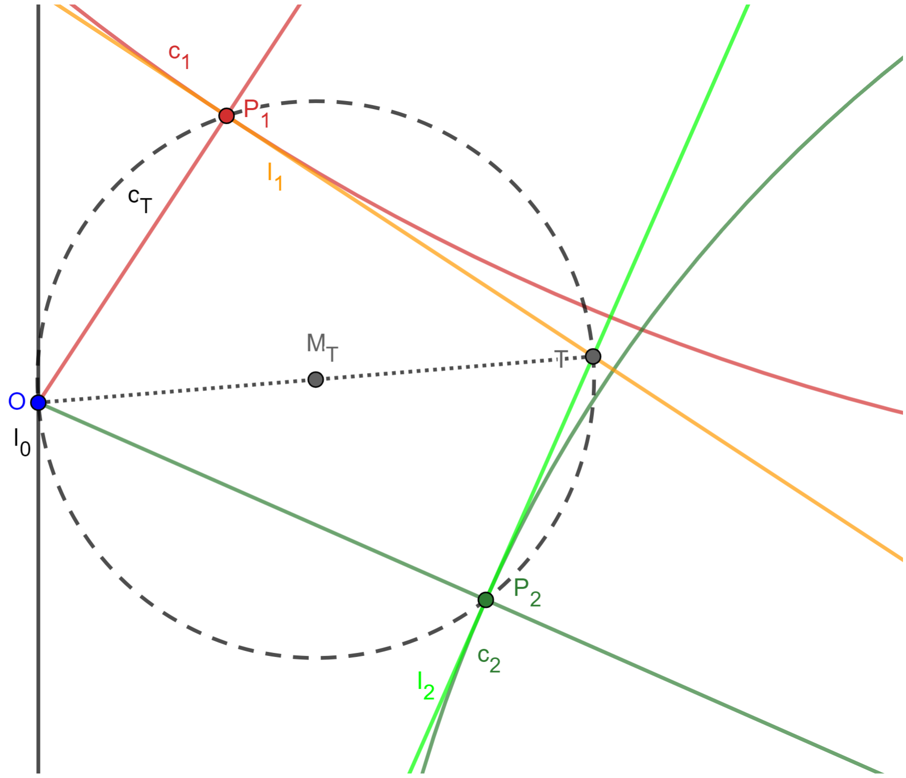

Figure 3.12.: Definition of circle c_T via O, P_1 and P_2.

$$\begin{aligned}
d_T^2 &= x_T^2 + y_T^2 \\
&= \frac{(\cos(\alpha_1)d_2 - \cos(\alpha_2)d_1)^2}{\sin(\alpha_1-\alpha_2)^2} + \frac{(\sin(\alpha_1)d_2 - \sin(\alpha_2)d_1)^2}{\sin(\alpha_1-\alpha_2)^2} \\
&= \frac{\sum_{i=1}^2 d_i^2(\cos(\alpha_i)^2 + \sin(\alpha_i)^2) - 2d_1d_2(\cos(\alpha_1)\cos(\alpha_2) + \sin(\alpha_1)\sin(\alpha_2))}{\sin(\alpha_1-\alpha_2)^2} \\
&= \frac{d_1^2 + d_2^2 - 2d_1d_2\cos(\alpha_1-\alpha_2)}{\sin(\alpha_1-\alpha_2)^2} \\
\Leftrightarrow d_T &= \frac{\sqrt{d_1^2 + d_2^2 - 2d_1d_2\cos(\alpha_1-\alpha_2)}}{\sin(\alpha_1-\alpha_2)} = \frac{d(P_1,P_2)}{\sin(\alpha_1-\alpha_2)}
\end{aligned}$$

We have used the identities $\sin(\alpha)^2 + \cos(\alpha)^2 = 1$ and $\cos(\alpha_1 - \alpha_2) = \cos(\alpha_1)\cos(\alpha_2) + \sin(\alpha_1)\sin(\alpha_2)$. To conclude the proof, we consider the point A on l_0 with $d_A = y_T$. Using this point, we obtain a right triangle ΔOTA with which we can compute $\cos(\beta)$ from d_T and y_T as y_T and d_T are sides of ΔOTA:

$$\cos(\beta) = \frac{y_T}{d_T} = \frac{(sin(\alpha_1)d_2 - \sin(\alpha_2)d_1)\sin(\alpha_1-\alpha_2)}{\sin(\alpha_1-\alpha_2)d(P_1,P_2)} = \frac{sin(\alpha_1)d_2 - \sin(\alpha_2)d_1}{d(P_1,P_2)}$$

□

We need a similar result for the intersection point S of two hyperbolic lines c_1 and c_2. Here, we know that $d_S = d_H - \sqrt{d_H^2 - 1}$ due to Lemma 3.8. Because of that, we need to find a formula for d_H:

Lemma 3.16. *For any two hyperbolic lines c_1, c_2, their intersection point S can be described in polar coordinates $S = (\gamma, d_S)$ with*

$$\cos(\gamma) = \frac{\sin(\alpha_1)d_{M_1} - \sin(\alpha_2)d_{M_2}}{d(M_1, M_2)}, \ d_H = \frac{d_{M_1}d_{M_2}\sin(\alpha_1 - \alpha_2)}{d(M_1, M_2)}$$

and $d_S = d_H - \sqrt{d_H^2 - 1}$.

Proof. Let c_1, c_2 be two hyperbolic lines intersecting in S. The proof to compute the angle γ is based on the triangle that is defined by the points M_1, M_2 and the intersection point of the vertical line through c_2 and the horizontal line through c_1, which we call A. As S lies on the altitude of the triangle ΔOM_1M_2, we need to compute the angle γ of this altitude. This is identical to computing the angle in M_2 of the triangle ΔAM_1M_2 as the two triangles are similar. This triangle has hypotenuse M_1M_2 and the length of the edge AM_2 is given by $\cos(\alpha_1)d_{M_1} - \cos(\alpha_2)d_{M_2}$ as we chose S in a way that this edge represents the differences in y-coordinates of M_1 and M_2. We obtain:

$$\cos(\gamma) = \frac{\cos(\alpha_1)d_{M_1} - \cos(\alpha_2)d_{M_2}}{d(M_1, M_2)}.$$

In order to obtain the distance d_S, we need to compute d_H and d_S follows from Lemma 3.8. For that, we compute the area A of ΔOM_1M_2 in two different ways: On the one hand, we can use $A = (1/2) \cdot d_H \cdot d(M_1, M_2)$. On the other hand, A can be computed by using the vectors $v_1 = OC_1$ and $v_2 = OC_2$ and computing their cross product, which yields the following formula:

$$A = \frac{1}{2}(v_1 \times v_2) = \frac{1}{2}\sin(\alpha_1)d_{M_1}\cos(\alpha_2)d_{M_2} - \sin(\alpha_2)d_{M_2} - \cos(\alpha_1)d_{M_1} = \frac{1}{2}d_{M_1}d_{M_2}\sin(\alpha_1 - \alpha_2).$$

Combining the formulas, we obtain

$$d_H = \frac{d_{M_1}d_{M_2}\sin(\alpha_1 - \alpha_2)}{d(M_1, M_2)}.$$

□

Before we start to show the two directions, we need a few more observation: The last term, $d_H - \sqrt{d_H^2 - 1}$, is not pleasant to work with. However, as we look at the distances asymptotically, we can use the Taylor expansion of the function $f(x) = x - \sqrt{(x^2 - 1)}$ and limit higher orders of summands. This results in:

$$f(x) = x - \sqrt{(x^2 - 1)} = \frac{1}{2x} + O(\frac{1}{x^3}) = \frac{1}{2x}(1 + O(\frac{1}{x^2}))$$

where we use both formulas in different contexts.

Additionally, we oftentimes have the situation that we compute the distance between two points A_1, A_2 with the law of cosines using O as the third point of the triangle. For points $A_1 = (\alpha_1, d_{A_1}), A_2 = (\alpha_2, d_{A_2})$, this is done in the following way:

$$d(A_1, A_2) = \sqrt{d_{A_1}^2 + d_{A_2}^2 - 2d_{A_1}d_{A_2}\cos(\alpha_1 - \alpha_2)}.$$

If we now scale d_{A_1} and d_{A_2} with a scaling factor x without changing α_1 and α_2, we see that the distance between $A_1(x)$ and $A_2(x)$ changes linearly in x:

$$d(A_1(x), A_2(x)) = \sqrt{d_{A_1}^2x^2 + d_{A_2}^2x^2 - 2d_{A_1}xd_{A_2}x\cos(\alpha_1 - \alpha_2)}$$
$$= x\sqrt{d_{A_1}^2 + d_{A_2}^2 - 2d_{A_1}d_{A_2}\cos(\alpha_1 - \alpha_2)} = x \cdot d(A_1, A_2)$$

At other times, we have a dependence between A_1, A_2 and two other points $B_1 = (\alpha_1, d_{B_1}), B_2 = (\alpha_1, d_{B_2})$ for some scaling factor x and term $t(x)$ of the following kind: $d_{A_i}(x) = (t(x))/(2d_{B_i}x)$ where the term $t(x)$ is identical for both pairs of points. If we insert that into the distance formula of $d(A_1, A_2)$, we obtain:

$$
\begin{aligned}
d(A_1(x), A_2(x)) &= \sqrt{d_{A_1}(x)^2 + d_{A_2}(x)^2 - 2d_{A_1}(x)d_{A_2}(x)\cos(\alpha_1 - \alpha_2)} \\
&= \sqrt{(\frac{t(x)}{2d_{B_1}x})^2 + (\frac{t(x)}{2d_{B_2}x})^2 - 2\frac{t(x)}{2d_{B_1}x}\frac{t(x)}{2d_{B_2}x}\cos(\alpha_1 - \alpha_2)} \\
&= \sqrt{(\frac{d_{B_2}^2}{4d_{B_1}^2 d_{B_2}^2 x^2} + \frac{d_{B_1}^2}{4d_{B_1}^2 d_{B_2}^2 x^2} - \frac{2d_{B_1}d_{B_2}\cos(\alpha_1 - \alpha_2)}{4d_{B_1}^2 d_{B_2}^2 x^2})t(x)^2} \\
&= \frac{1}{2d_{B_1}d_{B_2}x} \cdot \sqrt{d_{B_1}^2 + d_{B_2}^2 - 2d_{B_1}d_{B_2}\cos(\alpha_1 - \alpha_2)} \cdot |t(x)| \\
&= \frac{1}{2d_{B_1}d_{B_2}x} \cdot d(B_1, B_2) \cdot |t(x)|
\end{aligned}
$$

We denote this characterization of $d(A_1(x), A_2(x))$ with $d(A_1, A_2)[x, B_1, B_2, t(x)]$:

$$
d(A_1(x), A_2(x)) := d(A_1, A_2)[x, B_1, B_2, t(x)] = \frac{1}{2d_{B_1}d_{B_2}x} \cdot d(B_1, B_2) \cdot |t(x)|.
$$

3.3.1. Transforming Simple Euclidean Line Arrangements

The direction we start with is to show that for each Euclidean line arrangement L^e, there is a corresponding hyperbolic line arrangement L^h:

Theorem 3.17. *Given an Euclidean line arrangement L^e, there is a hyperbolic line arrangement L^h that has the same combinatorial description.*

Before we start, we need a few assumptions about the Euclidean line arrangement L^e. Similar to our proofs in the previous section, we place the border line l_0 vertically and all intersection points on the right of l_0. We choose an origin O on l_0 that is not part of any additional line. We assume that for all Euclidean lines l, the point P fulfills $d_P < 1$ in order to define corresponding hyperbolic lines in the Poincaré disk model. For that, we can find a translation and rotation to get a line arrangement L^e with those properties without changing the combinatorial description.

Now, we need to transform Euclidean lines l into hyperbolic lines c. For that, we use the transformation we defined at the start of the section: P is the common point of l and c, and c is uniquely defined by α and d_P due to Lemma 3.4.

The main idea of the proof is to use the similarity between hyperbolic and Euclidean space for sufficiently small distances. For that, we need to scale down our Euclidean line arrangement. We describe that scaling process by introducing a scaling factor μ between 0 and 1, and multiply the distances of each Euclidean point to the origin with that distance. This way, the Euclidean line arrangement stays topologically identical. Note that D remains a unit circle and is thus not scaled down. The corresponding hyperbolic lines thus are not scaled down by simply multiplying the distance of each point as well. Instead, our Euclidean scaling operation of Section 3.2.1 resurfaces here. For each μ, the hyperbolic line $c(\mu)$ corresponding to a Euclidean line $l(\mu)$ is defined by $d_P(\mu)$ and α set by $l(\mu)$.

Now that we have clarified the corresponding hyperbolic line arrangement and our way of scaling, we need to show that, if we make the arrangement small enough, L^h has the same combinatorial description as L^e. For that, we first show two convergence results for angle and distance to the origin which we then use to show that the points $S(\mu)$ and $T(\mu)$ converge faster towards each other than any pair of $T_1(\mu), T_2(\mu)$ on the same line $l(\mu)$. From that, we conclude that at some point the hyperbolic arrangement has the same combinatorial descriprion as the Euclidean one.

As a starting point, we think about how $T(\mu) = (\beta(\mu), d_T(\mu))$ behaves when changing the scaling factor. As we use Euclidean scaling, it is not hard to see that $\beta(\mu) = \beta$ remains unchanged and $d_T(\mu) = \mu \cdot d_T$. We can also confirm that by inserting $d_P(\mu) = \mu \cdot d_P$ into Lemma 3.15. Another thing we need to recall is the result of Corollary 3.5: For a Euclidean line l and point P, the distance of the center M of the corresponding hyperbolic line c is given by $d_M = (1 + d_P^2)/2d_P$.

Using that, we now consider how the distance between S and T is influenced by the scaling factor μ. We do that by considering the distances d_S and d_T and the angle (in O) between S and T, starting with the angle $\beta - \gamma(\mu)$:

Lemma 3.18. *Let l_1, l_2 be Euclidean lines, $T = (\beta, d_T)$ their intersection point and $S = (\gamma, d_S)$ the intersection point of the corresponding hyperbolic lines. Then, for every $\epsilon > 0$, there is a minimal scaling factor μ_ϵ such that:*

$$\forall \mu \geq \mu_\epsilon : \cos(\beta - \gamma(\mu)) \geq 1 - \epsilon.$$

Proof. Let l_1, l_2 be a fixed pair of Euclidean lines and $\epsilon > 0$. We prove this lemma by showing that, with decreasing μ, the angle $\gamma(\mu)$ gets arbitrarily close to β. This means that the angle between them, $\beta - \gamma(\mu)$ becomes smaller and smaller and thus its cosine can be bounded by $1 - \epsilon$. For that, we already know that

$$\cos(\beta) = \frac{\sin(\alpha_1)d_{P_2} - \sin(\alpha_2)d_{P_1}}{d(P_1, P_2)}$$

remains constant and have computed

$$\cos(\gamma(\mu)) = \frac{\sin(\alpha_1)d_{M_1}(\mu) - \sin(\alpha_2)d_{M_2}(\mu)}{d(M_1(\mu), M_2(\mu))}$$

in Lemma 3.16. Now, we insert the concrete values for $d_{M_i}(\mu)$: $d_{M_i}(\mu) = \frac{1}{2d_{P_i}\mu}(1 + \Theta(\mu^2))$. Note that the pairs $M_i(\mu)$ and $P_i(\mu)$ fulfill the requirements to use $d(M_1(\mu), M_2(\mu)) = d(M_1, M_2)[\mu, P_1, P_2, 1 + \Theta(\mu^2)]$.

$$\begin{aligned}
\cos(\gamma(\mu)) &= \frac{\sin(\alpha_1)d_{M_1}(\mu) - \sin(\alpha_2)d_{M_2}(\mu)}{d(M_1(\mu), M_2(\mu))} \\
&= \frac{\sin(\alpha_1)\frac{1}{2d_{P_1}\mu}(1 + \Theta(\mu^2)) - \sin(\alpha_2)\frac{1}{2d_{P_2}\mu}(1 + \Theta(\mu^2))}{d(M_1, M_2)[\mu, P_1, P_2, 1 + \Theta(\mu^2)]} \\
&= \frac{(\sin(\alpha_1)\frac{d_{P_2}}{2d_{P_2}d_{P_1}\mu} - \sin(\alpha_2)\frac{d_{P_1}}{2d_{P_1}d_{P_2}\mu})(1 + \Theta(\mu^2))}{\frac{1}{2d_{P_1}d_{P_2}\mu}d(P_1, P_2)|1 + \Theta(\mu^2)|} \\
&= \frac{\frac{1}{2d_{P_1}d_{P_2}\mu}(\sin(\alpha_1)d_{P_2} - \sin(\alpha_2)d_{P_1})(1 + \Theta(\mu^2))}{\frac{1}{2d_{P_1}d_{P_2}\mu}d(P_1, P_2)|1 + \Theta(\mu^2)|} \\
&= \frac{\sin(\alpha_1)d_{P_2} - \sin(\alpha_2)d_{P_1}}{d(P_1, P_2)} \cdot \frac{1 + \Theta(\mu^2)}{|1 + \Theta(\mu^2)|} \\
&= \cos(\beta) \cdot \frac{1 + \Theta(\mu^2)}{|1 + \Theta(\mu^2)|}
\end{aligned}$$

As $(1+\Theta(\mu^2))/(|1+\Theta(\mu^2)|) \to 1$ for $\mu \to 0$, the difference between the two angles indeed becomes arbitrarily small. That means that the cosine of the difference $\cos(\alpha - \gamma(\mu))$ becomes arbitrarily close to one, therefore also greater than $1-\epsilon$ for some μ_ϵ. For every μ that is smaller than μ_ϵ, we can conclude that $\cos(\beta - \gamma(\mu)) \geq 1 - \epsilon$. □

Similarly to the angle, we need a convergence result for the respective distances to the origin. Note that $d_T(\mu) \leq d_S(\mu)$ holds for sufficiently small μ as the Euclidean lines are closer to the origin than the hyperbolic ones. We thus need to bound the quotient $d_S(\mu)/d_T(\mu)$:

Lemma 3.19. *Let l_1, l_2 be Euclidean lines, $T = (\beta, d_T)$ their intersection point and $S = (\gamma, d_S)$ the intersection point of the corresponding hyperbolic lines c_1 and c_2. For every $\epsilon > 0$ there is a maximal scaling factor μ_ϵ such that:*

$$\forall \mu \leq \mu_\epsilon : \frac{d_S(\mu)}{d_T(\mu)} \leq 1 + \epsilon$$

Proof. Let l_1, l_2 be a fixed pair of lines and $\epsilon > 0$. Again, we start with the target distance $d_T(\mu)$. We insert $d_P(\mu) = \mu \cdot d_P$ into the formula from Lemma 3.15 to obtain:

$$d_T(\mu) = \frac{d(P_1(\mu), P_2(\mu))}{\sin(\alpha_1 - \alpha_2)} = \mu \cdot \frac{d(P_1, P_2)}{\sin(\alpha_1 - \alpha_2)} = \mu \cdot d_T$$

We see that $d_T(\mu)$ is linear in μ. Now, we again asymptotically compute the hyperbolic counterpart with $d_M(\mu) = \frac{1}{2d_P\mu}(1+\Theta(\mu^2))$. In order to do that, we start with computing $d_H(\mu)$ and then use $d_S(\mu) = \frac{1}{2d_H(\mu)} + O(\frac{1}{\mu^3})$. Again, we can use $d(M_1(\mu), M_2(\mu)) = d(M_1, M_2)[\mu, P_1, P_2, 1+\Theta(\mu^2)]$.

$$\begin{aligned}
d_H(\mu) &= \frac{d_{M_1}(\mu) d_{M_2}(\mu) \sin(\alpha_1 - \alpha_2)}{d(M_1(\mu), M_2(\mu))} \\
&= \frac{\frac{1}{2d_{P_1}\mu}(1+\Theta(\mu^2)) \frac{1}{2d_{P_2}\mu}(1+\Theta(\mu^2)) \sin(\alpha_1 - \alpha_2)}{d(M_1, M_2)[\mu, P_1, P_2, 1+\Theta(\mu^2)]} \\
&= \frac{\frac{1}{2d_{P_1}d_{P_2}\mu} \frac{1}{2\mu} \sin(\alpha_1 - \alpha_2)(1+\Theta(\mu^2))^2}{\frac{1}{2d_{P_1}d_{P_2}\mu} d(P_1, P_2)|1+\Theta(\mu^2)|} \\
&= \frac{\sin(\alpha_1 - \alpha_2)}{2\mu \cdot d(P_1, P_2)} \cdot \frac{(1+\Theta(\mu^2))^2}{|1+\Theta(\mu^2)|}
\end{aligned}$$

When inserting this into the formula for $d_S(\mu)$, we get the following result:

$$\begin{aligned}
d_S(\mu) &= \frac{1}{2h(\mu)} + O(\frac{1}{h(\mu)^3}) \\
&= \frac{2\mu \cdot d(P_1, P_2)}{2\sin(\alpha_1 - \alpha_2)} \cdot \frac{|1+\Theta(\mu^2)|}{(1+\Theta(\mu^2))^2} + O(\mu^3) \\
&= \mu \frac{d(P_1, P_2)}{\sin(\alpha_1 - \alpha_2)} \cdot \frac{|1+\Theta(\mu^2)|}{(1+\Theta(\mu^2))^2} + O(\mu^3) \\
&= d_T(\mu) \cdot \frac{|1+\Theta(\mu^2)|}{(1+\Theta(\mu^2))^2} + O(\mu^3)
\end{aligned}$$

We use this to compute the quotient $d_S(\mu)/d_T(\mu)$:

$$\frac{d_S(\mu)}{d_T(\mu)} = \frac{d_T(\mu) \cdot \frac{|1+\Theta(\mu^2)|}{(1+\Theta(\mu^2))^2} + O(\mu^3)}{d_T(\mu)} = \frac{|1+\Theta(\mu^2)|}{(1+\Theta(\mu^2))^2} + O(\mu^2)$$

This clearly converges towards 1 for $\mu \to 0$, thus again resulting in a maximal μ_ϵ with $d_S(\mu)/d_T(\mu) \leq 1 + \epsilon$ for every $\mu \leq \mu_\epsilon$, which concludes the proof of the lemma. □

With these convergence results for distance and angle between S and T, we now can prove that they converge faster towards each other than any pair of intersection points T_1, T_2. This is our main tool to show that the combinatorial descriptions of $L^e(\mu)$ and $L^h(\mu)$ are identical for small enough μ.

Lemma 3.20. *Let l be a Euclidean line, T be an intersection point on l with corresponding intersection point S in the hyperbolic line arrangement, and $T' \neq T$ another intersection point on l. For every positive constant k, there is an μ_k such that:*

$$\forall \mu \leq \mu_k : d(T(\mu), T'(\mu)) \geq k \cdot d(T(\mu), S(\mu))$$

Proof. Let $k > 0$, l a fixed line and $T \neq T'$ two intersection points on l. The inequality we need to show is the following (for some μ):

$$\frac{d(T(\mu), T'(\mu))}{d(T(\mu), S(\mu))} \geq k$$

We calculate these distances with the law of cosines (and square the distances):

$$\frac{d_T(\mu)^2 + d_{T'}(\mu)^2 - 2d_T(\mu)d_{T'}(\mu)\cos(\beta(\mu) - \beta'(\mu))}{d_T(\mu)^2 + d_S(\mu)^2 - 2d_T(\mu)d_S(\mu)\cos(\beta(\mu) - \gamma(\mu))} \geq k^2$$

In order to show that inequality, we use a geometric observation for the numerator and the convergence lemmas for the denominator to obtain a term whose numerator is constant and whose denominator is proportional to ϵ. We can then choose ϵ low enough that the inequality is fulfilled.

For the numerator, first note that $\beta(\mu)$ and $\beta'(\mu)$ remain constant. We can use the for the following observation: For every μ, there is a point A with angle β' that is the closest point to T on the ray with angle β'. This point A is either described by $d_A(\mu) = \cos(\beta - \beta')d_T(\mu)$ if $|\beta - \beta'| \leq \pi/2$, or the origin O. Either way, we can use $d(T(\mu), T'(\mu)) \geq d(T(\mu), A)$ to replace T' in the inequality. In the first case,

$$d(T(\mu), A(\mu)) = d_T(\mu)^2 + \cos(\beta - \beta')^2 d_T(\mu)^2 - 2d_T(\mu)^2\cos(\beta - \beta')^2 = d_T(\mu)^2(1 - \cos(\beta - \beta')^2).$$

In the second case, A is the origin, so $d(T(\mu), A(\mu)) = d_T(\mu)$.

For the denominator, we first fix a $\epsilon > 0$, but not assign a value yet. We use the convergence lemmas to obtain a μ_ϵ for that both $d_S(\mu)/d_T(\mu) \leq 1 + \epsilon$ and $\cos(\beta - \gamma(\mu)) \geq 1 - \epsilon$ are true. Additionally, note that $d_S(\mu) \geq d_T(\mu)$ is true for sufficiently small μ as the Euclidean lines are defined to be closer to the origin than their hyperbolic counterparts. Now, we replace the first $d_S(\mu)$ with $d_S(\mu) \leq (1 + \epsilon)d_T(\mu)$, the second $d_S(\mu)$ directly by $d_T(\mu)$ and $\cos(\beta - \gamma(\mu))$ by $1 - \epsilon$ to obtain:

$$\begin{aligned}
&d_T(\mu)^2 + d_S(\mu)^2 - 2d_T(\mu)d_S(\mu)\cos(\beta - \gamma(\mu)) \leq \\
&d_T(\mu)^2 + (1 + \epsilon)^2 d_T(\mu)^2 - 2d_T(\mu)^2(1 - \epsilon) = \\
&d_T(\mu)^2(1 + (1 + \epsilon)^2 - 2(1 - \epsilon)) = \\
&d_T(\mu)^2(\epsilon^2 + 4\epsilon)
\end{aligned}$$

Together, we obtain the following inequality (We only show $A \neq O$, the other case can be shown similarly):

$$\frac{d(T(\mu), T'(\mu))^2}{d(T(\mu), S(\mu)^2} \geq \frac{d_T(\mu)^2(1 - \cos(\beta - \beta')^2)}{d_T(\mu)^2(\epsilon^2 + 4\epsilon)} = \frac{1 - \cos(\beta - \beta')^2}{\epsilon^2 + 4\epsilon}$$

We can now choose ϵ sufficiently small such that this is larger than k^2 as the numerator is constant and the denominator converges towards 0 for $\epsilon \to 0$. This gives us a μ_k such that for every $\mu \leq \mu_k$:

$$\frac{d(T(\mu), T'(\mu))^2}{d(T(\mu), S(\mu)^2} \geq \frac{1 - \cos(\beta - \beta')^2}{\epsilon^2 + 4\epsilon} \geq k^2 \Leftrightarrow \frac{d(T(\mu), T'(\mu))}{d(T(\mu), S(\mu)} \geq k.$$

□

This allows us to characterize intersection points between lines that are not l_0. For the border line, we could repeat the argumentation in a slightly adjusted way to account for l_0 being an origin line. However, that would be repetitive and there is an easier way to deal with the intersection points on l_0. As an origin line can be interpreted as a circle with infinite radius, we can change that radius to a finite one in a way that the properties of the line arrangement remain identical. This way, Lemma 3.20 is also applicable for the intersection points with l_0.

Using that, we can prove Theorem 3.17:

Proof of Theorem 3.17. Let L^e be a Euclidean line arrangment. We have described how the corresponding hyperbolic line arrangement L^h is built and how we scale down L^e using Euclidean scaling. We know that scaling the Euclidean line arrangement does not change its combinatorial description. What we need to prove here is that there is a scaling factor μ such that the resulting hyperbolic line arrangement $L^h(\mu)$ has the same combinatorial description as L^e. For that, we need to show that for each Euclidean line l, the order of intersections on l is identical to the one on c.

Our main idea is to show that, if we fix a line l and two intersection points T, T' and where T is closer to l_0 than T', that there is a $\mu_{(T,T')}$ such that for every $\mu \leq \mu_{(T,T')}$ the same holds true for the corresponding points on $c(\mu)$, $S(\mu)$ and $S'(\mu)$. This gives us an upper bound for μ for every pair of intersection points on every line. If we choose a scaling factor that stays under all those upper bounds, every property is fulfilled and thus the hyperbolic arrangement has the correct combinatorial description.

All that is left now is to argue that for every line l and pair of intersection points T and T' where T is closer to l_0, such an $\mu_{(T,T')}$ exists. Our tool for that is Lemma 3.20. There, we have shown that $T(\mu)$ and $S(\mu)$ converge faster towards each other than $T(\mu)$ and $T'(\mu)$. This means that there is a scaling factor μ for which $S(\mu)$ has to be closer to l_0 than $S'(\mu)$ on $c(\mu)$, dictated by the same being true for $T(\mu)$ and $T'(\mu)$. This is the case as the distance between $S(\mu)$ and $T(\mu)$ as well as between $S'(\mu)$ and $T'(\mu)$, respectively, is small enough in comparison to $d(T(\mu), T'(\mu))$ that $S'(\mu)$ cannot be closer to l_0 than $S(\mu)$. □

3.3.2. Transforming Simple Hyperbolic Line Arrangements

We now consider the other direction of Theorem 3.3: Starting with a hyperbolic line arrangement L^h. we transform that into an equivalent Euclidean line arrangement L^e.

Theorem 3.21. *Given a hyperbolic line arrangement L^h, there is a Euclidean line arrangement L^e that has the same combinatorial description.*

This time, we start with a hyperbolic line arrangement in our usual setting: l_0 as the vertical origin line in the Poincaré disk and all intersection points to its right. Our task here is to transform the circles c that represent the hyperbolic lines into Euclidean lines l in a way that the combinatorial description is identical. For that, we again use P as the common point of c and l and define l as the orthogonal line to q through P. This is the transformation we described at the start of this section. However, the speed at which the points P and intersection points S and T converge towards the origin differs from the other direction as we use the hyperbolic scaling operation from Section 3.2 this time.

At this point, ideally we would use Theorem 3.17 to show the other direction as well: If we could prove that, for every given hyperbolic line arrangement L^h, there is a scaling factor λ such that the corresponding Euclidean arrangement $L^e(\lambda)$ could have also been created by Euclidean scaling, we would prove Theorem 3.21. However, this is not trivial to prove as the Euclidean scaling could need a smaller scaling factor than given by the hyperbolic scaling at every point. Instead, we execute a similar plan as in the last section with minor differences at a few key points.

We again start by confirming what happens to α, d_P and d_M for a given hyperbolic line c. From the definition of the hyperbolic scaling operation, we know that $\alpha(\lambda) = \alpha$ remains untouched and M is scaled back in linear fashion, thus $d_M(\lambda) = \lambda \cdot d_M$. For d_P, we use Lemma 3.6 and the property $d_P = d_M - r$:

$$d_P(\lambda) = d_M(\lambda) - \sqrt{d_M(\lambda)^2 - 1} = \lambda \cdot d_M - \sqrt{(\lambda \cdot d_M)^2 - 1}$$

As we described previously, we can alter that to obtain the following representation: $d_P(\lambda) = \frac{1}{2\lambda d_P}(1 + O(\frac{1}{\lambda^2}))$. Note that the second part again convergence to 1 for $\lambda \to \infty$.

Now we can start by introducing the adjusted convergence lemmas for $\beta(\lambda), \gamma, d_S(\lambda)$ and $d_T(\lambda)$:

Lemma 3.22. *Let c_1, c_2 be hyperbolic lines intersecting in S and T the intersection point of the corresponding Euclidean lines. Then for every $\epsilon > 0$ there is a minimal scaling factor λ_ϵ such that:*

$$\forall \lambda \geq \lambda_\epsilon : \cos(\beta(\lambda) - \gamma(\lambda)) \geq 1 - \epsilon.$$

Proof. We again know from Lemma 3.8 that $\cos(\gamma)(\lambda) = \cos(\gamma)$, and in Lemma 3.16 we computed its value:

$$\cos(\gamma) = \frac{\sin(\alpha_1)d_{M_1} - \sin(\alpha_2)d_{M_2}}{d(M_1, M_2)}.$$

For $\cos(\beta(\lambda))$, we insert $d_P(\lambda) = \frac{1}{2\lambda d_M}(1 + O(\frac{1}{\lambda^2}))$ into the formula of Lemma 3.15:

$$\begin{aligned}
\cos(\beta(\lambda)) &= \frac{\sin(\alpha_1(\lambda))d_{P_2}(\lambda) - \sin(\alpha_2(\lambda))d_{P_1}(\lambda)}{d(P_1(\lambda), P_2(\lambda))} \\
&= \frac{\sin(\alpha_1)\frac{1}{2\lambda d_{M_2}}(1 + O(\frac{1}{\lambda^2})) - \sin(\alpha_2)\frac{1}{2\lambda d_{M_1}}(1 + O(\frac{1}{\lambda^2}))}{d(P_1, P_2)[\lambda, M_1, M_2, 1 + O(\frac{1}{\lambda^2})]} \\
&= \frac{\frac{1}{2d_{M_1}d_{M_2}\lambda}(\sin(\alpha_1)d_{M_1} - \sin(\alpha_2)d_{M_2})(1 + O(\frac{1}{\lambda^2}))}{\frac{1}{2d_{M_1}d_{M_2}\lambda}d(M_1, M_2)|1 + O(\frac{1}{\lambda^2})|} \\
&= \cos(\gamma) \cdot \frac{1 + O(\frac{1}{\lambda^2})}{|1 + O(\frac{1}{\lambda^2})|}
\end{aligned}$$

Again, this means that $\beta(\lambda)$ and γ become arbitrarily close to each other. This allows us to conclude that for rising scaling factor λ, $\cos(\beta(\lambda) - \gamma)$ converges towards 1, which in turn yields that for every $\epsilon > 0$, there is a λ_ϵ such that for every $\lambda \geq \lambda_\epsilon : \cos(\beta(\lambda) - \gamma) \geq 1 - \epsilon$ is true. □

Again, we need a similar convergence result for the distance to the origin. In contrast to the last section, we slightly adjust the statement:

Lemma 3.23. *Let c_1, c_2 be hyperbolic lines intersecting in S and T the intersection point of the corresponding Euclidean lines l_1 and l_2. For every $\epsilon > 0$ there is a minimal scaling factor λ_ϵ such that:*

$$\forall \lambda \geq \lambda_\epsilon : \frac{d_T(\lambda)}{d_S(\lambda)} \geq 1 - \epsilon.$$

Proof. We again start with calculating $d_S(\lambda)$: From Lemma 3.16 we know that this is given by $d_S(\lambda) = d_H(\lambda) - \sqrt{d_H(\lambda)^2 - 1}$ with

$$d_H(\lambda) = \frac{d_{M_1}(\lambda) d_{M_2}(\lambda) \sin(\alpha_1 - \alpha_2)}{d(M_1(\lambda), M_2(\lambda))} = \lambda \cdot \frac{d_{M_1} d_{M_2} \sin(\alpha_1 - \alpha_2)}{d(M_1, M_2)}.$$

We use the characterization $d_H - \sqrt{d_H^2 - 1} = \frac{1}{2d_H} + O(\frac{1}{d_H^3})$ to achieve the following term for $d_S(\lambda)$:

$$d_S(\lambda) = \frac{d(M_1, M_2)}{2\lambda d_{M_1} d_{M_2} \sin(\alpha_1 - \alpha_2)} + O(\frac{1}{\lambda^3})$$

Now we compute $d_T(\lambda)$ by inserting $d_P(\lambda) = \frac{1}{2\lambda d_M}(1 + O(\frac{1}{\lambda^2}))$ into the formula of Lemma 3.15.

$$\begin{aligned} d_T(\lambda) &= \frac{d(P_1(\lambda), P_2(\lambda))}{\sin(\alpha_1 - \alpha_2)} \\ &= \frac{d(P_1, P_2)[M_1, M_2, (1 + O(\frac{1}{\lambda^2})]}{\sin(\alpha_1 - \alpha_2)} \\ &= \frac{d(M_1, M_2)|1 + O(\frac{1}{\lambda^2})|}{2\lambda d_{M_1} d_{M_2} \sin(\alpha_1 - \alpha_2)} \end{aligned}$$

Notice that, for $t = \frac{d(M_1, M_2)}{2d_{M_1} d_{M_2} \sin(\alpha_1 - \alpha_2)}$, t/λ is part of both $d_S(\lambda)$ and $d_T(\lambda)$. This term is cut when computing the quotient, leaving

$$\frac{d_S(\lambda)}{d_T(\lambda)} = \frac{1}{|1 + O(\frac{1}{\lambda^2})|} + \frac{\lambda O(1/\lambda^3)}{t|1 + O(\frac{1}{\lambda^2})|} = \frac{1}{|1 + O(\frac{1}{\lambda^2})|} + \frac{O(1/\lambda^2)}{|1 + O(\frac{1}{\lambda^2})|}$$

as t is constant, converging towards 1 for $\lambda \to \infty$. As $d_S(\lambda)/d_T(\lambda)$ converges towards 1, so does $d_T(\lambda)/d_S(\lambda)$. This allows us to conclude that, for every $\epsilon > 0$, there is an initial λ_ϵ such that

$$\forall \lambda \geq \lambda_\epsilon : \frac{d_T(\lambda)}{d_S(\lambda)} \geq (1 - \epsilon).$$

□

Again, these convergence lemmas now help us in the following way:

Lemma 3.24. *Let c be a hyperbolic line and $S \neq S'$ two different intersection points on c with lines, and T the intersection point on the corresponding Euclidean line l. For every positive constant k, there is a λ_k such that:*

$$\forall \lambda \geq \lambda_k : d(S(\lambda), S'(\lambda)) \geq k \cdot d(S(\lambda), T(\lambda))$$

Proof. Let $k > 0$ be fixed, pick a hyperbolic line c and fix an intersection point S on c with T being the respective intersection point on the corresponding Euclidean line l. Let S' be another intersection point on c. This time, we need to show the following inequality:

$$\frac{d(S,S')}{d(S,T)} \geq k$$

We again do that by representing the distances with the law of cosines and replacing terms until we have a constant numerator and a demoninator dependant on ϵ so that we can choose ϵ sufficiently small that the inequality is true. Let ϵ thus be fixed, but we do not choose its value yet. For the numerator, we can again use the argumentation of Lemma 3.20 to obtain a point A and

$$d(S(\lambda), S'(\lambda))^2 \geq d(S(\lambda), A(\lambda))^2 = d_S^2(1 - \cos(\gamma - \gamma')^2).$$

For the demoninator, we again use the convergence lemmas. Again, note that $d_T(\lambda) \leq d_S(\lambda)$ for sufficiently small λ:

$$d(S(\lambda), T(\lambda))^2 = d_S^2 + d_T^2 - 2d_S d_T \cos(\gamma - \beta(\lambda)) \leq d_S^2 + d_S^2 - 2d_S^2(1-\epsilon)^2 = d_S^2(4\epsilon - 2\epsilon^2)$$

Using this, we obtain:

$$\frac{d(S,S')}{d(S,T)} \geq \frac{d_S^2(1 - \cos(\gamma - \gamma')^2)}{d_S^2(4\epsilon - 2\epsilon^2)} = \frac{1 - \cos(\gamma - \gamma')^2}{4\epsilon - 2\epsilon^2}$$

Now, we can find a value for ϵ such that the inequality is true. The convergence lemmas then give us the required constraints for λ. □

Again, we consider l_0 in the hyperbolic line arrangement to be represented by a circle c_0 with sufficiently large radius in order to apply Lemma 3.24 to intersection points on l_0 as well. As we have identical results as in the other direction, the proof for Theorem 3.21 is similar to the one of Theorem 3.17:

Proof of Theorem 3.21. This time, we start with a simple hyperbolic line arrangement L^h that we scale down using hyperbolic scaling. For pairs of intersection points S, S', we again conclude from Lemma 3.24 that there is a scaling factor $\lambda_{(S,S')}$ such that for every $\lambda \geq \lambda_{(S,S')}$, $S(\lambda)$ being closer to l_0 than $S'(\lambda)$ implies the same for their Euclidean counterparts $T(\lambda)$ and $T'(\lambda)$. Thus, for a λ^* that fulfills all the requirements imposed by pairs (S, S'), $L^e(\lambda^*)$ is equivalent to L^h. □

With this, the proof of Theorem 3.3 is done: Theorem 3.17 directly proves one of the implications while Theorem 3.21 proves the other one. As exactly the same instances of SimpleStretchability are realizable in Euclidean and in hyperbolic space, we obtain two simple reductions from EuclideanSimpleStretchability to HyperbolicSimpleStretchability and the other way around, achieving $\exists\mathbb{R}$-hardness and membership: We do not transform the instances represented as combinatorial descriptions at all. Theorem 3.3 now ensures the correctness of the reduction. Thus, HyperbolicSimpleStretchability is also $\exists\mathbb{R}$-complete and our starting point for the reductions in the next chapters.

Corollary 3.25. HyperbolicSimpleStretchability *is* $\exists\mathbb{R}$*-complete.*

3.4. More on Stretchability

In this last section of this chapter, we offer a few further thoughts on the Stretchability problem. First, we still need to proof a claim from earlier in the chapter: For each simple hyperbolic line arrangement, there is a hyperbolic line arrangement with similar description where the border line l_0 can be added. After that, we examine hyperbolic line arrangements that are not simple. Specifically, we investigate how they behave when they are scaled via the hyperbolic scaling operation and draw conclusions for the general Stretchability problem.

As stated, we start with including the border line in a simple hyperbolic line arrangement:

Lemma 3.26. *Let L be a simple hyperbolic line arrangement without border line l_0. There is a hyperbolic line arrangement L' that is homeomorphic to L and includes a border line l_0 that divides the hyperbolic plane in a way that all intersection points between lines other than l_0 are in the same half plane of l_0.*

Proof. Let L be a simple hyperbolic line arrangement without border line. The main idea is to use our scaling method to move all intersection points towards the origin, then add a line l_0 that is further apart from the origin than every intersection point. Then all intersection points are in the same half plane, the one that includes the origin.

In order to define l_0, we define the concept of a *convergence line* g for a hyperbolic line c that is the line through O that is orthogonal to q (visualized in Figure 3.13). Note that, with hyperbolic scaling, g is invariant to the scaling and we have shown in the last section that c converges towards g for $\lambda \to \infty$. We take this collection of convergence lines as the base of our argumentation. It is not hard to see that we can add a hyperbolic line l_0 fulfilling the requirements for the border line to this collection: We assume that none of the convergence lines is vertical as we could choose the location of the Poincaré disk in a different way to prevent that. We need to choose two ideal endpoints on the outer disk to define l_0. For that, we choose two points that are to the left of the vertical line through O, but to the right of any endpoint of a convergence line in the left half of D, one on the top and one on the bottom (depicted in Figure 3.13). We claim that, for a large enough λ, the line l_0 defined by those points (which we do not scale) intersects all other lines and divides the plane in a way that every other intersection point is in the same half plane.

For the first claim, that l_0 intersects every $c(\lambda)$ in $L(\lambda)$ for some $\lambda \geq 1$, we argue that l_0 intersects every convergence line. As the lines of the arrangement get more and more indistinguishable from their convergence lines for increasing λ, there is an λ_1 such that for every $\lambda' \geq \lambda_1$, l_0 intersects $c(\lambda)$. Now take the convergence line for any c. We know that it cannot be vertical, thus it has one endpoint to the left of the vertical line through the origin and one endpoint to the right. That means that l_0 splits those endpoints into different half planes as we chose l_0 to split the endpoints in the left half of the Poincaré disk from those in the right half. Thus l_0 intersects the convergence line.

The second claim, that for some λ every intersection point $S(\lambda)$ is on the same side of l_0, is not hard to see. For increasing λ, the intersection points have a maximum possible distance from O of $\lambda - \sqrt{\lambda^2 - 1}$ (follows from Lemma 3.8) while l_0 has a fixed distance from O, thus there is a λ_2 so that for every $\lambda' \geq \lambda_2$ all intersection points are in the same half plane as the origin.

If we now choose any λ^* with $\lambda* \geq \lambda_1$ and $\lambda* \geq \lambda_2$, the line arrangement $L(\lambda^*)$ is homeomorphic to L and allows the inclusion of the border line l_0. ☐

This concludes our research on SimpleStretchability, we now shift our focus to general line arrangements. The two requirements that simple line arrangements have to fulfill are

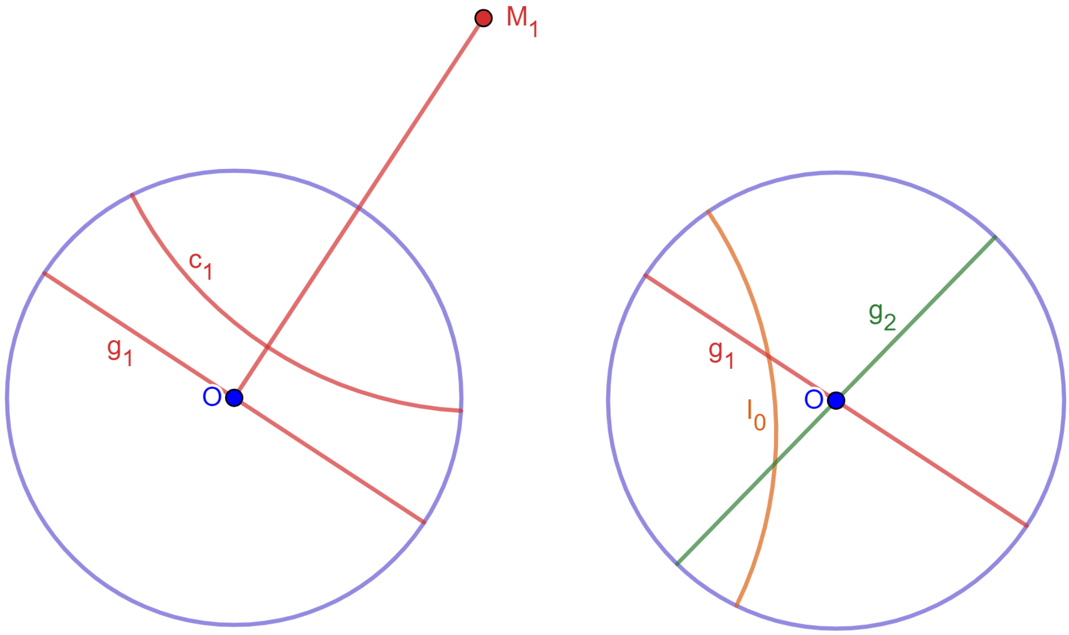

Figure 3.13.: Left: Definition of the convergence line g_1. Right: Adding l_0 into an arrangement of convergence lines.

that there are no parallel lines and that no more than two lines intersect in each point. We examine what happens when we lift those requirements in two steps: First, we define the intermediate version of *non-parallel line arrangements* where we still require every pair of lines to intersect, but allow three or more lines to intersect in the same point. Then, we look at how general line arrangements with no requirements behave when being sujected to the hyperbolic scaling operation, and think about the general problem of STRETCHABILITY in its Euclidean and hyperbolic variants. As we see in the following theorems, non-parallelity is the foundation for the correctness of the hyperbolic scaling operation: For non-parallel line arrangements, the scaling operation still works while for general line arrangements the combinatorial description may change when scaling the arrangement.

Theorem 3.27. *Let L be a non-parallel hyperbolic line arrangement. Then for every scaling factor $\lambda \geq 1$, the line arrangement $L(\lambda)$ has the same combinatorial description as L.*

Proof. We have already seen what happens if multiple lines intersect in the same point in the proof of Lemma 3.13, so we shortly recall the argument: The distance function for intersection points we computed in Lemma 3.8 is injective and only depends on the intial height h. Thus, if three or more points intersect in L, their initial height is identical, so they intersect in the same point in $L(\lambda)$ for every $\lambda \geq 1$. □

Now we also drop the requirement of non-parallelity and examine general hyperbolic line arrangements. Similar to simple line arrangements, the decision problem STRETCHABILITY is defined in the following way:

STRETCHABILITY:
Input: Combinatorial description D of a pseudoline arrangement with border line l_0.
Problem: Does a line arrangement L exist that realizes D?

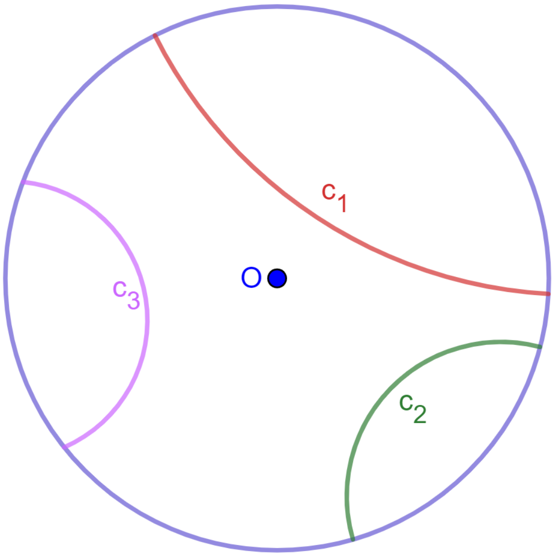

Figure 3.14.: A hyperbolic line arrangement into which no border line can be placed.

By defining what kind of line arrangements we use, we get the problem variants HYPERBOLICSTRETCHABILITY and EUCLIDEANSTRETCHABILITY. Note that we still require the existance of a border line l_0 that intersects all other lines, although the other lines do not have to pairwise intersect anymore. With that, we cannot represent every hyperbolic line arrangement anymore, as the example depicted in Figure 3.14 shows: For a representation via pseudoline arrangement, we need to be able to place a border line l_0 into a homeomorphic line arrangement that intersects every other line. In this case, with three lines c_1, c_2 and c_3 who each divide the plane in a way that the other two lines are in the same half plane, this is not possible. A border line l_0 cannot cross the lines c_i twice, but has only two ideal points on the boundary of the Poincaré disk D. It thus can only intersect at most two of the three lines. This opens the question if there is a better way to define a decision problem about hyperbolic line arrangements that includes all hyperbolic line arrangements. However, in this thesis, we focus on the problem defined by pseudoline arrangements.

In its Euclidean case, the problem is known to be $\exists\mathbb{R}$-complete [Mat14]. The questions we answer now are the following: Is the problem $\exists\mathbb{R}$-complete in hyperbolic space as well? And do we find a similar equivalence result to SIMPLESTRETCHABILITY?

We start with the question of equivalence of the Euclidean and hyperbolic variants. As there are infinitely more parallel lines than in Euclidean geometry, it intuitively seems likely that the same argumentation as for simple line arrangements cannot suffice here. Indeed, we find that only lines that have a Euclidean sense of parallelity stay parallel for every scaling factor while other parallel lines intersect at some point:

Theorem 3.28. *For two hyperbolic lines c_1 and c_2, $c_1(\lambda)$ and $c_2(\lambda)$ are parallel for every $\lambda \geq 1$ if and only if their corresponding Euclidean lines l_1 and l_2 are parallel.*

Proof. Let c_1 and c_2 be two hyperbolic lines and l_1, l_2 their corresponding Euclidean lines. On the one hand, if l_1 and l_2 are parallel, then M_1, M_2 and O are collinear. This is also true for $M_1(\lambda)$ and $M_2(\lambda)$ for every $\lambda \geq 1$ as we do not change their angles. This means that $H(\lambda) = O$ for every $\lambda \geq 1$, and thus $c_1(\lambda)$ and $c_2(\lambda)$ do not intersect due to Lemma 3.7.

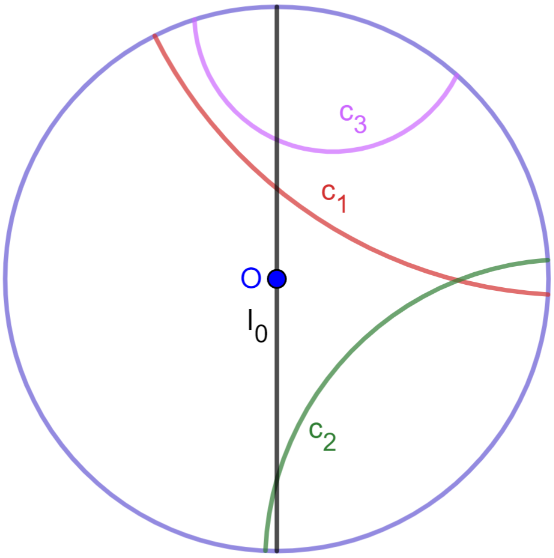

Figure 3.15.: A hyperbolic line arrangement with no Euclidean counterpart.

On the other hand, if the lines c_1 and c_2 do not intersect for every $\lambda \geq 1$, M_1, M_2 and O have to be collinear. If not, then d_H would be linearly scaled with λ due to Lemma 3.11. For some λ^*, $d_H(\lambda^*) > 1$ would be true and thus $c_1(\lambda^*)$ and $c_2(\lambda^*)$ would intersect due to Lemma 3.7. The collinearity of M_1, M_2 and O induces that l_1 and l_2 are parallel. $\square$

This result indicates that, unlike the specialized SimpleStretchability version, the general Stretchability problem is not equivalent in hyperbolic and Euclidean geometry. Indeed, this is not hard to see: When we take three lines in Euclidean space, two of which are parallel ($l_1||l_3$), the line l_2 is either parallel to both l_1 and l_3 or to none of the lines. In hyperbolic space however, c_2 can be parallel to c_3 while intersecting c_1 as parallelity is not transitive here. This is depicted in Figure 3.15 and yields a combinatorial description that is realizable in hyperbolic space but not in the Euclidean plane.

While the problem is not equivalent to its Euclidean counterpart, it still is $\exists\mathbb{R}$-hard as well. This is the case as there is an easy reduction from HyperbolicSimpleStretchability: We get a simple pseudoline arrangement P as input which obviously is a general pseudoline arrangement. If there is a line arrangement that stretches P, that line arrangement has to be simple because P is simple. Thus, if we can solve the general hyperbolic stretchability problem, we can solve HyperbolicSimpleStretchability which we have shown to be $\exists\mathbb{R}$-complete in Corollary 3.25.

This concludes our research of hyperbolic line arrangements and offers a new insight for the main result of this chapter, Theorem 3.3. In a way, it is surprising that a problem is equivalent in hyperbolic and Euclidean space when hyperbolic space is much larger and offers many more possibilities for line arrangements. However, constraining the line arrangements to be non-parallel negates those differences, which do exist for general line arrangements. Parallelity thus indeed is what differentiates hyperbolic and Euclidean line arrangements, when restricting this aspect, hyperbolic line arrangements cannot use the additional space and become equivalent to their Euclidean counterparts.

4. Unit Disk Graphs

Now that we have proven the equivalence of simple hyperbolic and Euclidean line arrangements, we consider the consequences on geometric $\exists\mathbb{R}$-complete problems that are proven to be $\exists\mathbb{R}$-hard by reduction from EUCLIDEANSIMPLESTRETCHABILITY. In this chapter, we start with the most fitting candidate: Recognizing unit disk graphs. We choose this problem because its hyperbolic variant is already the focus of research: Introduced by Papadopoulos et al. [PKBV10] in a probabilistic setting, hyperbolic unit disk graphs were found to have many interesting properties. As random graphs, they have a highly hierarchical structure and serve as a well-fitting testing ground for algorithms defined for large networks. Additionally, Dohse [Doh22] shows that the recognition problem for hyperbolic unit disk graphs is NP-hard and in $\exists\mathbb{R}$, leaving the question of $\exists\mathbb{R}$-hardness open.

The main goal of this chapter is to answer that question affirmatively. For that, we first recall the existing proof ideas for recognizing Euclidean and hyperbolic unit disk graphs. The membership proofs use a reduction to ETR to show $\exists\mathbb{R}$-membership. Introduced by Erickson et al. in "Smoothing the gap between NP and ER" [EVDHM22], there is a new technique to proof $\exists\mathbb{R}$-membership. We need to show that a verification algorithm on a real RAM model exists. As this was not established when the existing membership proofs were made, we outline the existing proofs $\exists\mathbb{R}$-membership for both EUDGs and HUDGs and add the new versions as our own contribution. We further present the $\exists\mathbb{R}$-hardness proof for recognizing EUDGs by Schaefer [Sch09] and Matoušek [Mat14] as we use it for our main contribution: Using that and the equivalence of Theorem 3.3, we show that recognizing HUDGs is also $\exists\mathbb{R}$-hard.

We now start with presenting the existing results and the new membership proof for the recognition problem of Euclidean unit disk graphs before we do the same for hyperbolic unit disk graphs. Finally, we also consider the complexity of a special case of hyperbolic unit disk graphs: Strongly hyperbolic unit disk graphs, introduced by Bläsius et al. [BFKS21].

4.1. Euclidean Unit Disk Graphs

As mentioned previously, Euclidean unit disk graphs are one of the geometric problems that are complete for $\exists\mathbb{R}$:

Theorem 4.1. *Recognizing EUDGs is $\exists\mathbb{R}$-complete.*

The proof consists of two parts. The first part is showing that recognizing EUDGs is part of $\exists\mathbb{R}$, which we do in two different ways as described above. The second part is a reduction from EuclideanSimpleStretchability to Recog(EUDG) to show that it is $\exists\mathbb{R}$-hard.

Lemma 4.2. *Recognizing EUDGs is in* $\exists\mathbb{R}$

This result is implied in multiple papers but, as far as we know, was never actually proven formally. We do that now using both ETR formulas and a verification algorithm on a real RAM model. For both variants, the first steps are identical: Each vertex is represented by a pair of variables denoting the coordinates of the center of the unit disk. Then all we need are checks for each edge and non-edge of the graph that ensure that the distance between adjacent vertices is at most 1 and the distance between non-adjacent vertices is greater than 1. In Euclidean space, the distance between points A and B can be expressed by $\sqrt{(B_x - A_x)^2 + (B_y - A_y)^2}$.

Proof by ETR formula. The first kind of membership proof now constructs an ETR formula that is solvable if and only if we find an embedding of unit disks that represents the input graph. For an input graph $G = (V, E)$, we use n pairs of variables (x_i, y_i) that represent the embedding of vertex v_i. Now, each of the distance checks for is represented by an inequality in the form we described previously. One additional hurdle is that square roots are not allowed in ETR formulas. We can solve this by squaring the inqualities. The resulting checks are, for pairs of vertices v_i, v_j: $(x_i - x_j)^2 + (y_i - y_j)^2 \leq 1$ for edges and $(x_i - x_j)^2 + (y_i - y_j)^2 > 1$ for non-edges. Combining those checks with logical ands yields the complete formula. ☐

Formulating the verification algorithm is a bit easier than that:

Proof by verification algorithm. We interpret the input of n pairs of real variables (x_i, y_i) again as the coordinates of the vertices v_i. The algorithm now computes all the distance checks: For edges v_iv_j, $(x_i - x_j)^2 + (y_i - y_j)^2 \leq 1$ and for non-edges $(x_i - x_j)^2 + (y_i - y_j)^2 > 1$ have to hold, respectively. If one of them fails, the solution is denied, else accepted. ☐

That concludes the first part of the proof. For the $\exists\mathbb{R}$-completeness, we follow the argumentation of Kang and Müller [KM12]. Their proof generalizes the problem to higher dimensions, we only consider the proof for dimension 2 which is the problem of recognizing EUDGs. Note that we use orientated combinatorial descriptions that characterize the regions of a pseudoline arrangement in this chapter.

Lemma 4.3. *[[KM12]] Recognizing EUDGs is* $\exists\mathbb{R}$*-hard.*

As mentioned previously, Kang and Müller reduce EuclideanSimpleStretchability to recognizing EUDGs and use the oriented combinatorial description as input. As a first step, they adjust the problem in order to simplify talking about the input: They only include sign vectors without $0s$ in it, meaning only the sign vectors from the inside of the regions. Kang and Müller show that this is still as complex as the general version.

The transformation is the following: Let D be a SimpleStretchability-instance (simplified combinatorial description of sign vectors). For each sign vector σ in D, which corresponds to a region in a pseudoline arrangement, we have a vertex c_σ. For each line l_i we have two vertices a_i and b_i. They idea is that a_i and b_i are embedded in a way that the perpendicular bisector between the two points can be used as the line l_i in a line arrangement that corresponds to D. To ensure that the line l_i splits the plane into the

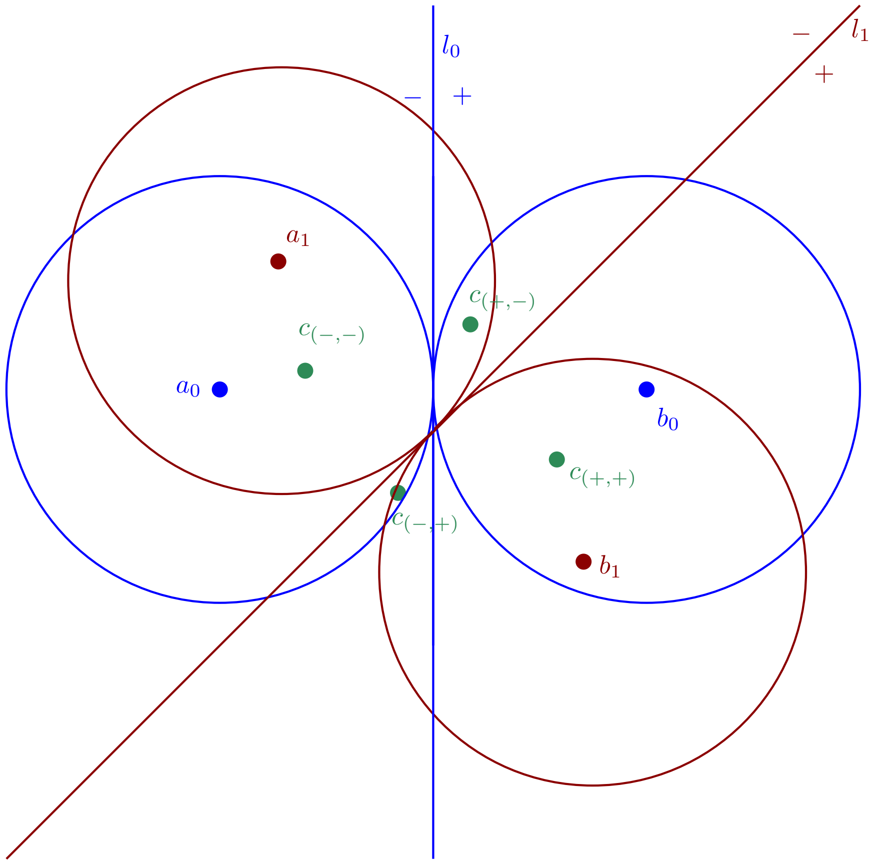

Figure 4.1.: A line arrangement of three lines with its region vertices c_σ (red) and a pair of vertices a_i, b_i (blue).

correct regions, we need the following adjacencies: We have three cliques A, B, C and no edges between A and B, while for each i, the neighbourhoods of a_i and b_i partition C into two parts. This last requirement ensures the required property. The transformation thus transforms a combinatorial description D into a graph $G_D = (V, E)$. This is shown in Figure (not yet). The definition of graph G_D is the following:

Definition 4.4 (Graph G_D)**.** $G_D = (V_D, E_D)$ *with*

$$\begin{aligned} V_D &= \{a_i, b_i \mid 0 \leq i \leq n\} \cup \{c_\sigma \mid \sigma \in D\} \\ E_D &= \{a_i a_j \mid i \neq j\} \cup \{b_i b_j \mid i \neq j\} \\ &\quad \cup \{a_i c_\sigma \mid \sigma_i = -\} \cup \{b_i c_\sigma \mid \sigma_i = +\} \\ &\quad \cup \{c_\sigma c_{\sigma'} \mid \sigma \neq \sigma'\} \end{aligned}$$

G_D is visualized in Figure 4.1.

Kang and Müller now show $\exists\mathbb{R}$-hardness of recognizing EUDGs with the following equivalence: D is realizable in Euclidean space if and only if G_D is an EUDG. We motivate both directions in the following lemmas:

Lemma 4.5. *[[KM12]] For any given combinatorial description D: If D is realizable, then G_D is an EUDG.*

Proof Sketch. The proof is very technical, so we only provide a short overview. We start with a line arrangement L that is a realization of a combinatorial description D. Kang and Müller first transform this line arrangement into a more suitable form: They show that

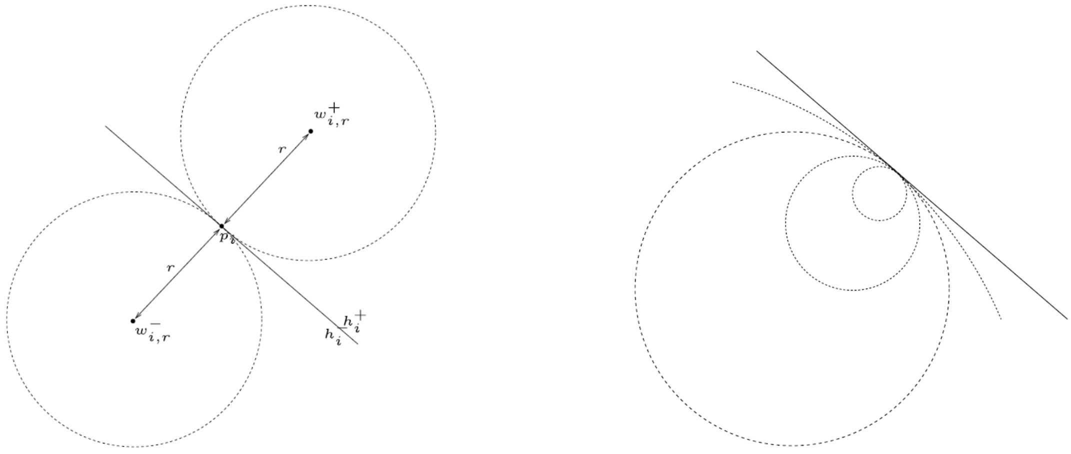

Figure 4.2.: Left: Placement of the vertices a_i and b_i. Right: Motivation that a radius exists such that the circle includes all region vertices.

there is a point O and an ϵ-disk C_ϵ with center O such that every region of L intersects C_ϵ and every line l_i has a slope, defined by vector v_i, that is close to $e_1 = (1, 0)$ ($||e_1 - v_i|| \leq \epsilon$). With that, they find locations for the region points c_σ inside of C_ϵ so that c_σ is situated in the region with sign vector σ. Additionally, every line l_i has a segment on the inside of C_ϵ where a point P_i is fixed. The vertices a_i and b_i now are placed in a way that the segment between them is orthogonal to l_i (shown in Figure 4.2) and they are centers of circles that include exactly the c_σ with $\sigma_i = -$ for a_i and $\sigma_i = +$ for b_i. There is a radius r such that this condidition is fulfilled for every i (also visualized in Figure 4.2). The c_σ all form a clique as they are situated inside of C_ϵ, and the a_i and b_i respectively form cliques as well as every l_i has a slope that is close to e_1 and thus the vertices are near each other. The embedding of G_D is the described placement with the coordinates scaled down by r to achieve unit disks. □

Now we only need the other direction to show that recognizing EUDGs is $\exists\mathbb{R}$-complete:

Lemma 4.6. *[[KM12]] For a combinatorial description D, if there is an embedding of G_D into Euclidean space, then D is realizable.*

Proof. The main idea of the proof is to use the perpendicular bisector between the vertices a_i and b_i as the line l_i. We show that, for the resulting collection of lines L, its combinatorial description, consisting of the sign vectors of the regions, is D.

We start with an embedding f_e for G_D. We need to show that every region defined by D is a region in L. As $n + 1$ lines can only divide the plane into a fixed amount of regions and D consists of the maximum amount of sign vectors (if not, the arrangement not simple), it is sufficient to prove that L implements every region defined by D. We start with a region from D, which we identify with its sign vector σ. There is a corresponding vertex c_σ in G_D, which is placed into the plane at the point $f_e(c_\sigma)$. We argue that $f_e(c_\sigma)$ is not a part of any line l_i and has sign vector σ in the line arrangement L.

For the first point, if $f_e(c_\sigma)$ would be part of a line l_i, then the distance from $f_e(c_\sigma)$ to $f_e(a_i)$ and $f_e(b_i)$ would be identical as l_i is the perpendicular bisector between them, thus c_σ would need to be adjacent to either both a_i and b_i or none of them. However, it is adjacent to exactly one of them, which is a contradiction.

For the second point, we construct the sign vector σ' of $f_e(c_\sigma)$ in L. For coordinate i, $\sigma'_i = -$ exactly when $c_\sigma a_i \in E$ and $\sigma'_i = +$ otherwise. However, $c_\sigma a_i \in E$ is only the case if $\sigma_i = -$ which means that $\sigma'_i = \sigma_i$. This concludes the proof that L realizes the combinatorial description D and thus, if G_D is an EUDG, then D is realizable. $\square$

With these lemmas completing the reduction, we now can prove Lemma 4.3:

Proof of Lemma 4.3. We already know that EUCLIDEANSIMPLESTRETCHABILITY, even in the simplified version of the input, is $\exists\mathbb{R}$-complete. We use a reduction to RECOG(EUDG) to show $\exists\mathbb{R}$-hardness of the latter problem. The transformation is defined in Definition 4.4. To ensure that this transformation does not change the validity of the instances, we show equivalence of input D and transformed graph G_D in Lemma 4.5 and Lemma 4.6. $\square$

4.2. Hyperbolic Unit Disk Graphs

As we have seen in the previous chapter, the problem SIMPLESTRETCHABILITY is equivalent when changing the underlying geometry from Euclidean to hyperbolic. In this chapter, we achieve a similar result for the problem of recognizing unit disk graphs. Although the graph classes of EUDGs and HUDGs are different, we can find a reduction from HYPERBOLICSIMPLESTRETCHABILITY to recognizing HUDGs that is very similar to the reduction of the Euclidean variants we showed in the last section. This means that recognizing HUDGs is also $\exists\mathbb{R}$-complete:

Theorem 4.7. *Recognizing HUDGs is $\exists\mathbb{R}$-complete.*

Again, we start with membership which was first shown by Dohse [Doh22] via building an ETR formula. We recall that proof shortly and add the verification algorithm.

Lemma 4.8 ([Doh22])**.** *Recognizing HUDGs is in $\exists\mathbb{R}$.*

In this proof, we need to use hyperbolic distance formulas in an ETR formula. For that, we introduced the hyperboloid model of hyperbolic geometry in Chapter 2. As a reminder, here the points of the hyperbolic plane are represented in $\mathbb{R}^3$ as points on the surface of a hyperboloid with $z^2 - x^2 - y^2 = 1$ defining the hyperboloid, and $z > 0$ as we only use the upper sheet. The distance between two points u, v now is given by $d_h(u, v) = \operatorname{arcosh}(B(u, v))$ where B is the Minkowski bilinear form $B((u_x, u_y, u_z), (v_x, v_y, v_z)) = u_z v_z - u_x v_x - u_y v_y$.

Proof by ETR formula [Doh22]. We again start with an input graph $G = (V, E)$. We have a few differences to the Euclidean case: As the radius R is not fixed, it needs to be encoded in a variable x_R. We now need three variables (x_i, y_i, z_i) for each vertex v_i and additional checks to ensure that the vertices are placed onto the hyperboloid. We cannot express the distance formula directly in an ETR formula as arcosh is not allowed. However, it is monotone and thus can be left out of the checks because $\operatorname{arcosh}(B(u, v)) \leq R \Leftrightarrow B(u, v) \leq \cosh(R)$ (thus we do not have the actual radius R used for the embedding in x_R, but it can be recovered by computing $\operatorname{arcosh}(x_R)$). To recap, we need a variable x_R for the radius, variables x_v, y_v, z_v and checks $z_v^2 - x_v^2 - y_v^2 = 1$ and $z_v > 0$ for each vertex v. For each pair of vertices $u \neq v$, we need distance checks $z_u z_v - x_u x_v - y_u y_v \leq x_R$ if $uv \in E$ and $z_u z_v - x_u x_v - y_u y_v > x_R$ if $uv \notin E$. Combining all these checks with logical ands again yields the complete formula. As each of those components has fixed length, the formula has polynomial size. $\square$

The verification algorithm can be described in the following way:

Proof by verification algorithm. We start with an input graph $G = (V, E)$ and use three variables (x_i, y_i, z_i) for each vertex v_i. We again need an additional variable for our radius R. As the real-RAM model also cannot compute arcosh, we use the same idea as in the last proof to encode arcos(R) in the variable x_R. The verification algorithm now needs to check if each of the vertices is correctly placed on the hyperboloid, and if the distance checks hold true for each pair of vertices. □

Our main contribution in this chapter is the proof that recognizing HUDGs is not only in $\exists\mathbb{R}$, but − similar to recognizing EUDGs − also $\exists\mathbb{R}$-hard:

Theorem 4.9. *Recognizing HUDGs is $\exists\mathbb{R}$-hard.*

The main idea is that due to SimpleStretchability being equivalent in Euclidean and hyperbolic space, the Euclidean reduction can be used to achieve a hyperbolic reduction as well. We first state the necessary results we need for the reduction. The first one is our main result from the last chapter:

Theorem 3.3. *Let D be a combinatorial description of a* SimpleStretchability *instance. Then D is realizable by a line arrangement L^e in the Euclidean plane if and only if D is realizable by a line arrangement L^h in the hyperbolic plane.*

This ensures that we can start the same way as in the Euclidean case: With a combinatorial description we want to realize in Euclidean space. For the next step, we need the Euclidean reduction from EuclideanSimpleStretchability to recognizing EUDGs we presented in the last section. From that, we conclude that there is a poly-time transformation from EuclideanSimpleStretchability to Recog(EUDG) that transforms a combinatorial description D into an graph G_D such that D is realizable in the Euclidean plane if and only if G_D as an EUDG.

This gives us a Euclidean unit disk graph, but our reduction needs to end with a hyperbolic unit disk graph. The following result from Dohse [Doh22] and Bläsius et al. [BFKS21] bridges that gap:

Theorem 4.10 ([Doh22], [BFKS21])**.** *Every Euclidean unit disk graph G is also a hyperbolic unit disk graph.*

With this, we now reduce HyperbolicSimpleStretchability to Recog(HUDG). We start with a combinatorial description of an orientated hyperbolic line arrangement D. This description also fits with an Euclidean line arrangement, so we perform the same transformation as in the Euclidean proof to obtain a graph G_D as defined in Definition 4.4. This will be our HUDG.

The last thing we need to show now is that a drawing of G_D in the hyperbolic plane emits a hyperbolic line arrangement of D the same way that it does in Euclidean space, which we prove in the following lemma:

Lemma 4.11. *Let D be a combinatorial description and G_D the corresponding graph defined in Definition 4.4. If G_D is a HUDG, we can extract a hyperbolic line arrangement L^h that realizes D in hyperbolic space.*

Proof. Let D be a combinatorial description, G_D the corresponding graph and f_h an unit disk embedding of G_D. We begin with extracting a hyperbolic line arrangement L^h from the graph and then show that the line arrangement indeed realizes the same combinatorial description D that was used to define G_D. The extraction part is easy, we perform the same process as in the Euclidean case: For every pair of vertices a_i and b_i, we consider the perpendicular bisector of the segment between $f_h(a_i)$ and $f_h(b_i)$ and use it as line l_i. L^h is now defined as the set of all those lines l_i. We know that this process gives us a line arrangement with combinatorial description D in the Euclidean case. For the hyperbolic case, we show the correctness step by step.

The first thing to show is that we indeed get a simple hyperbolic line arrangement. The main tool to show that are the adjacencies of the vertices c_σ with either a_i or b_i. Assume that either there are lines that do not intersect or three or more lines intersect in the same point. In both cases, the lines of L^h do not divide the hyperbolic plane in the maximum amount of regions. This means that there have to be two vertices $c_\sigma, c_{\sigma'}$ with $\sigma \neq \sigma'$ that are placed in the same region. Note that no vertex c_σ can be placed onto one of the l'_i as it would have the same distance to both $f_h(a_i)$ and $f_h(b_i)$ and thus would need to be either adjacent to both or neither vertices, but is adjacent to exactly one of them. As $\sigma \neq \sigma'$, there is at least one coordinate k with $\sigma_k \neq \sigma'_k$. But then c_σ and $c_{\sigma'}$ are placed on the same side of the perpendicular bisector l_k. The two vertices thus have to be both adjacent to either a_k or b_k, which corresponds to $\sigma_k = \sigma'_k$ which is not the case. This shows that L^h indeed is a simple hyperbolic line arrangement.

Now that we know that L^h indeed is a hyperbolic line arrangement, we consider its combinatorial description. Here, the same argumentation as in the Euclidean case works: The region vertices c_σ ensure that each of the sign vectors σ are correctly represented by unique regions. We conclude that L^h indeed realizes the combinatorial description D. □

Now we have all the tools to prove Theorem 4.9, stating that the problem of recognizing hyperbolic unit disk graphs is $\exists\mathbb{R}$-hard:

Proof of Theorem 4.9. In this proof, we reduce HYPERBOLICSIMPLESTRETCHABILITY in poly-time to RECOG(HUDG). In the previous chapter, we showed that HYPERBOLIC-SIMPLESTRETCHABILITY is $\exists\mathbb{R}$-complete, thus such a reduction shows that recognizing hyperbolic unit disk graphs is indeed $\exists\mathbb{R}$-hard. Our idea here is to show that the Euclidean transformation also works for the hyperbolic variant.

The Euclidean proof of Kang and Müller, which we outlined in the previous section, starts with a EUCLIDEANSIMPLESTRETCHABILITY instance D. This is also a valid instance of HYPERBOLICSIMPLESTRETCHABILITY, as we argued in the last chapter. As the problems are equivalent, interpreting a combinatorial description as a Euclidean instance as opposed to a hyperbolic one does not change its outcome, thus we can indeed start in the same way as the Euclidean proof. From that, we obtain a graph G_D that is a Euclidean unit disk graph if and only if D was indeed realizable in Euclidean space. We now show that the same holds true for hyperbolic space: G_D also is a hyperbolic unit disk graph if and only if D is realizable in hyperbolic space.

For the one direction, we start with a yes-instance D of HYPERBOLICSIMPLESTRETCHABILITY. This is also a yes-instance of EUCLIDEANSIMPLESTRETCHABILITY due to Theorem 3.3, which means that the corresponding graph G_D is a Euclidean unit disk graph. As a result of Theorem 4.10, every Euclidean unit disk graph, which includes G_D, is also a hyperbolic unit disk graph. This concludes the first implication: If D is a yes-instance, then G_D is a HUDG.

For the other direction, we start with the case that, for a HYPERBOLICSIMPLESTRETCHABILITY instance D, G_D is a hyperbolic unit disk graph. In Lemma 4.11 we have shown that a hyperbolic line arrangement then can be extracted, which means that D is also a yes-instance. This proves the second implication.

With these two implications, we have shown that the same transformation as in the Euclidean proof also works in hyperbolic space, thus recognizing HUDGs also is $\exists\mathbb{R}$-complete. □

If we take a closer look at the proof, we showed that we do not transform onto any HUDGs, but onto a subclass that encapsulates line arrangements. This subclass is hard to recognize because it implicitly stretches the line arrangement and is identical in Euclidean and hyperbolic space as we formulate in the following corollary:

Corollary 4.12. *For any combinatorial description D of a pseudoline arrangement, G_D is a Euclidean unit disk graph if and only if G_D is a hyperbolic unit disk graph.*

The proof here is not hard: On the one hand, every EUDG is also an HUDG. On the other hand, if G_D is a HUDG, it implicitly realizes a hyperbolic line arrangement. This means that there is a Euclidean line arrangement with combinatorial description D which in turn results in every G_D also being an EUDG.

4.3. Strongly Hyperbolic Unit Disk Graphs

In the last section, we have shown that $\exists\mathbb{R}$-hardness of HUDGs can be shown similar to its Euclidean variant. However, there is a special subset of HUDGs that typically have highly non-Euclidean structure: *strongly hyperbolic unit disk graphs*, first regarded by Bläsius et al. [BFKS21]. This special class of unit disk graphs has an additional constraint for the hyperbolic embedding: The ground space radius can only be as large as the threshold distance. An example of such a graph is visualized in Figure 4.3.

Definition 4.13 (Strongly Hyperbolic Unit Disk Graphs)**.** *A graph $G = (V, E)$ that can be embedded into the hyperbolic plane with an embedding f_h such that there is a threshold distance R that fulfills: For each $i \neq j \in V \times V$: $d_h(f_h(i), f_h(j)) \leq R \Leftrightarrow ij \in E$ and there is a point P with $\forall v \in V : d_h(f_h(v), P) \leq R$.*

We now discuss if this specialization of HUDGs also is $\exists\mathbb{R}$-complete. As a start, we give a reduction from the problem of recognizing SHUDGs to recognizing HUDGs by adding an additional vertex to the input graph that is adjacent to every other vertex. This vertex can be interpreted as the center of the ground space disk in the strongly hyperbolic case and any embedding with such a vertex is automatically a strongly hyperbolic embedding.

Lemma 4.14. *A graph $G = (V, E)$ is a strongly hyperbolic unit disk graph if and only if $G' = (V', E')$ with $V' = V + \{x \mid x \notin V\}$ and $E' = E + \{xv \mid v \in V\}$ is a hyperbolic unit disk graph.*

Now we can focus on the $\exists\mathbb{R}$-completeness of recognizing strongly hyperbolic unit disk graphs. The last lemma already shows that the problem is in $\exists\mathbb{R}$ as we have a reduction to recognizing HUDGs and thus to $\exists\mathbb{R}$. For the hardness, we consider the same proof idea as earlier: If we can show that for any SimpleStretchabilityinstance D and the corresponding graph G_D defined in Definition 4.4, G_D is a strongly hyperbolic unit disk graph if and only if the line arrangement is stretchable, then we have shown that recognizing SHUDGs is $\exists\mathbb{R}$-hard. We use Lemma 4.14 to simplify the process: Instead of asking whether G_D is a SHUDG, we can ask if G'_D with the added vertex is an HUDG. It could be the case that this further simplifies to the question if G'_D is an EUDG as in Corollary 4.12, though we did not manage to show that result.

For the rest of the section, we introduce approaches to answer to this question, though we do not answer it definitively. Again, if we find that G'_D is indeed an EUDG, then we

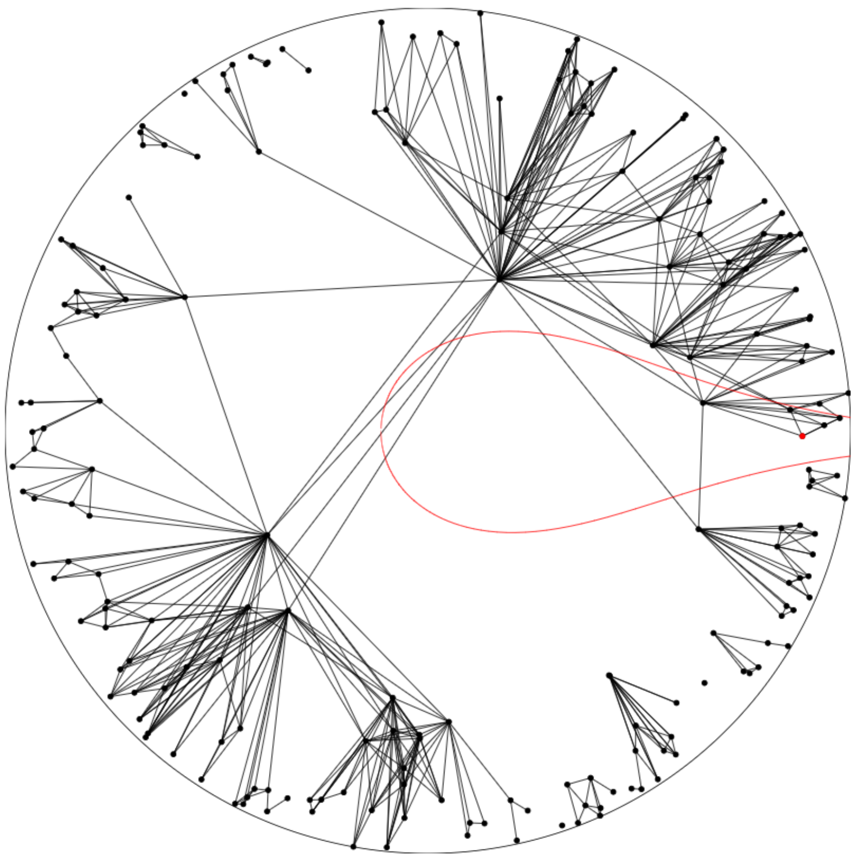

Figure 4.3.: A strongly hyperbolic unit disk graph.

would gain a reduction from HyperbolicSimpleStretchability the same way as for normal HUDGs: We could transform a combinatorial description D into G'_D which also is a strongly hyperbolic unit disk graph if it is a Euclidean one, and as any embedding of G'_D gives us a line arrangement for D due to Lemma 4.11.

As a starting point, we recall the transformation of Kang and Müller. They embed the graph G_D in way that is If we can show that this embedding of G_D already leads to a correct embedding of G'_D, we would show that our problem is $\exists\mathbb{R}$-hard. However, that is not the case: The distances between the a_i and the origin are too large.

We could try to fix this issue by simply moving the vertices within the circle around O while maintaining the same radius and hoping that the adjacencies do not change. That is not successful as there are special instances of Recog(EUDG) that require double exponential precision, as proposed by McDiarmid and Müller [MM13]. As the vertices need to be precisely placed, we only have very little slack move them and still represent the same unit disk graph. With this in mind, it is obvious that the transformation from Kang and Müller does not lead to an embedding of G'_D for every combinatorial description D as the required change of the radius does not work for these special instances. However, there could be other ways to embed G'_D even for the special line arrangements.

At this point, we did not reach further conclusions. However, we have assembled a handful of approaches that could work to show that recognizing SHUDGs is $\exists\mathbb{R}$-complete:

- Show that G'_D is a Euclidean unit disk graph for every stretchable D.
- Show that G'_D is a hyperbolic unit disk graph for every stretchable D.
- Try a different reduction, from either HyperbolicSimpleStretchability, Recog(HUDG) or a completely different problem.

If none of these approaches work, one could try to show that the reduction by Kang and Müller does not work for this problem by finding a hyperbolic line arrangement with combinatorial description D such that G_D is no strongly hyperbolic unit disk graph. While this does not disprove $\exists\mathbb{R}$-hardness of recognizing SHUDGs, it at least is an indicator that the problem might not be as hard as its general version.

5. ∃ℝ-Hardness Proof Framework

The definition of hyperbolic unit disk graphs, their interesting and useful properties and the ∃ℝ-completeness of the recognition problem is, in our minds, motivation for trying the same approach for more geometric ∃ℝ-complete problems. In order to start this process and give an indication how it could look like, we define the hyperbolic version of the recognition problem for segment graphs and consider its complexity. For that, we now generalize the ∃ℝ-completeness proof from the last chapter in order to be applicable for other geometrical problems and apply it on the segment graph recognition problem.

We first recall the proof structure we used for Recog(HUDG): We used that the Euclidean problem version is ∃ℝ-hard by reduction from EuclideanSimpleStretchability, that EuclideanSimpleStretchability and HyperbolicSimpleStretchability are equivalent, that every EUDG is also an HUDG and that the extraction process for the line arrangements used in the Euclidean hardness proof also works in the hyperbolic context. The proof then follows the following argumentation: We start with an instance D of HyperbolicSimpleStretchability. As per Theorem 3.3, we can replace it with an instance of EuclideanSimpleStretchability. We then use D in the Euclidean hardness proof to obtain a graph G_D. If D is realizable in hyperbolic space, then also in Euclidean space, which means that G_D is an EUDG and subsequently also a HUDG. On the other hand, the process of extracting a line arrangement out of an embedding of G_D also works in the hyperbolic context. This concludes the hardness proof because if G_D is a HUDG, then D is realizable in hyperbolic space.

Now we use the same idea on other problems. Obviously, this only works for problems where hyperbolic geometry is applicable, and that are shown to be ∃ℝ-complete by reduction from SimpleStretchability. As a first step, we visualize the argumentation:

$$\begin{tikzcd}
\textsc{EuclideanSimpleStretchability} \arrow[r, phantom, "\overset{(a)}{\Leftrightarrow}"] \arrow[d, "(b)"'] & \textsc{HyperbolicSimpleStretchability} \\
\text{Euclidean problem version} \arrow[r, "(c)"] & \text{Hyperbolic problem version} \arrow[u, "(d)"']
\end{tikzcd}$$

The equivalence (a) is given by Theorem 3.3. For the problems we consider in this thesis, the reduction (b) is already established, in case of the unit disk graphs by Kang and Müller [KM12]. The transformation (c) is an argument that establishes that the Euclidean yes-instances of the problem are also hyperbolic yes-instances as in small areas, hyperbolic and Euclidean space become similar. For unit disk graphs, this result is shown by Dohse [Doh22] and Bläsius et al. [BFKS21]. For the last arrow (d), we have to argue that any hyperbolic yes-instance of the problem induces a realization of the initial combinatorial description the same way it does in the Euclidean case. For that, the concepts used to show that have to work in hyperbolic geometry. This is the case for unit disk graphs.

In general, we need the following results for the framework to be applicable:

(a) Given by Theorem 3.3.

(b) Geometric reduction from EuclideanSimpleStretchability.

(c) Every Euclidean yes-instance is also a hyperbolic yes-instance.

(d) A hyperbolic line arrangement is extractable from the hyperbolic yes-instance that is the result of the transformation given in (b).

In the rest of this chapter, we first show another application of this proof framework for the problem of recognizing intersection graphs of segments. Matoušek [Mat14] and Schaefer [Sch09] show that this problem is ∃ℝ-complete in Euclidean space, which implements the reduction (b). We complete the proof by implementing the (c) and (d) arguments. Finally, we introduce a few problems whose hyperbolic variants likely can successfully be subjected to this proof framework.

5.1. Intersection Graphs of Segments

The problem we use to show how to apply the proof framework is the problem of recognizing segment graphs. A *segment graph*, short for intersection graph of segments, is a graph G whose vertices can be represented by segments of lines in a given plane where two segments intersect if and only if their corresponding vertices are adjacent. The Euclidean version of the problem can be formalized in the following way:

Recog(ESEG):
Input: Graph $G = (V, E)$
Problem: Can V be represented by a set S of $|V|$ segments in the Euclidean plane such that two segments intersect if and only if the corresponding vertices are adjacent?

The problem is considered in the context of ∃ℝ by Schaefer [Sch09] and Matoušek [Mat14]. They show that recognizing Euclidean segment graphs is ∃ℝ-complete via geometric reduction from EuclideanSimpleStretchability, which indicates that a ∃ℝ-hard hyperbolic variant might exist. Indeed, we now define the hyperbolic variant and use our proof framework to show ∃ℝ-hardness.

5.1.1. Hyperbolic Segment Graphs

The only thing we need to change from the Euclidean definition is the plane in which the segments are placed. Segments of lines can be used in the hyperbolic plane in the same way as in the Euclidean plane, so we define the hyperbolic version of the problem:

Recog(HSEG):
Input: Graph $G = (V, E)$
Problem: Can V be represented by a set S of $|V|$ segments in the hyperbolic plane such that two segments intersect if and only if the corresponding vertices are adjacent?

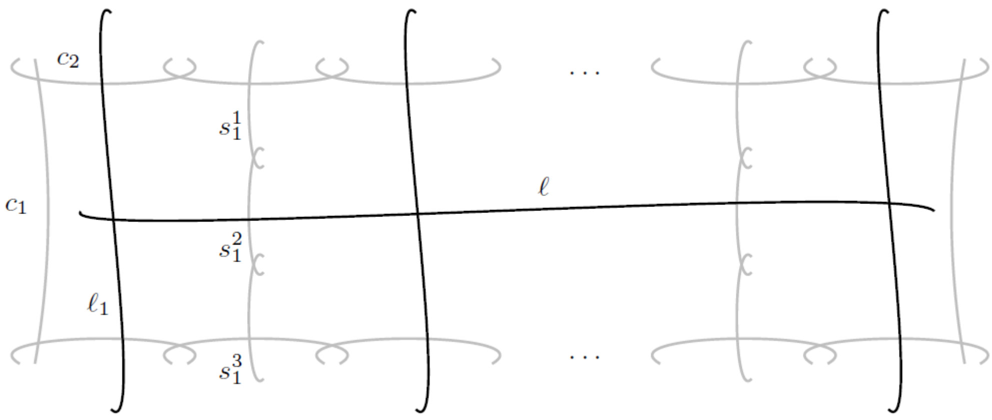

Figure 5.1.: The ordering gadget for RECOG(ESEG) (Source: [Sch09])

In the case of hyperbolic unit disk graphs, there are graphs that can be represented by hyperbolic unit disks but not by any Euclidean unit disk representation. We do not give a similar result for hyperbolic segment graphs, so in theory the graph class could be identical to Euclidean segment graphs. However, even if the classes are equivalent, this proof still holds merit as an example of applying the proof framework.

We now use the proof framework to show that this problem is $\exists\mathbb{R}$-hard. For that, we formulate and prove the three necessary steps: Giving the Euclidean reduction from EuclideanSimpleStretchability, showing that every Euclidean segment graph is also a hyperbolic one, and showing that we can extract a hyperbolic line arrangement from every hyperbolic segment graph that is built from a pseudoline arrangement in the way we define now.

5.1.2. Euclidean Hardness Proof

The first step (b) is to recapitulate the Euclidean hardness proof:

Theorem 5.1 ([Sch09], [Mat14])**.** *Recognizing Euclidean segment graphs is $\exists\mathbb{R}$-complete.*

Proof Sketch. We focus on the hardness proof. For that, we give the geometric reduction introduced by Schaefer [Sch09] as it is more easily understandable. The reduction can of course be formalized to receive a combinatorial description D as input and define a graph G_D from that.

The reduction starts with a pseudoline arrangement P. We add two additional border lines l_u and l_b to form a triangle around the intersection points together with l_0. Then we cut every pseudoline just outside of this outer triangle to transform the pseudolines into segments. Additionally, we add so-called *ordering gadgets* for each line which are displayed in Figure 5.1. The graph G_D is now defined by interpreting each segment, including the ordering gadgets, as a vertex and adding edges for each intersection. The ordering gadgets ensure that the order of intersections is forced in every realization of G_D. $\square$

The proof gives us a graph G_D for each combinatorial description D, where G_D is a Euclidean segment graph if and only if D is realizable in Euclidean space.

5.1.3. Transforming Euclidean Segment Graphs

We now need to show step (c), which requires us to show that the class of Euclidean segment graphs is a subset of the hyperbolic segment graphs:

Theorem 5.2. *Every Euclidean segment graph is also a hyperbolic segment graph.*

Proof. Similar to the proofs in Section 3.3, the idea here is to reduce the area in which the segments lie. Then we can find a transformation from Euclidean to hyperbolic segments that represent the same graph. We start with a Euclidean segment graph G and a corresponding segment representation S^e. We assume that the segments in S^e are non-parallel. This is possible because we may always move the endpoints of the segments in a small local surrounding and thus can always avoid parallelity. Our idea is to use Euclidean scaling to reduce the area in which the segments are situated and then build the hyperbolic segment representation S^h. As a reminder, we have already done something similar in the proof of Theorem 3.17 where we scaled down Euclidean line arrangements until the corresponding hyperbolic line arrangements had the same combinatorial description. We use that result by transforming S^e into a Euclidean line arrangement, use Theorem 3.17 to get a hyperbolic line arrangement with identical combinatorial description and extract a hyperbolic segment representation of G.

In order to transform S^e into a line arrangement L^e, we first transform each of the segments s into their respective lines l (each segment is part of exactly one line). Additionally, we add two lines l_l and l_r for each s that intersect l each in one of the endpoints of s. Now, if two segments intersect, then the intersection point of the corresponding lines lies between the intersection points with l_l and l_r on both lines. If two segments do not intersect, either the corresponding lines do not intersect or the intersection point is outside of the segment defined by $l_l and l_r$ on at least one of the lines.

This is also the property we use to extract the hyperbolic segment represenation after applying Theorem 3.17 to L^e. We obtain a hyperbolic line arrangement L^h with identical combinatorial description. We now define the hyperbolic segments as the segments between the intersection points of c with c_l and c_r. As the combinatorial description of L^e and L^h is identical, the property ensures that the hyperbolic segments intersect if and only if the Euclidean segments intersected. Thus, S^h also is a segment representation of G, and G is a hyperbolic segment graph if it is a Euclidean one. □

We thus now know that for each realizable combinatorial description D of a simple hyperbolic line arrangement, the corresponding graph G_D is a Euclidean segment graph and thus a hyperbolic segment graph. This concludes one direction of the reduction.

5.1.4. Reconstructing Hyperbolic Line Arrangements

The remaining step (d) requires us to show that we can define a simple hyperbolic line arrangement L^h with combinatorial description D from each hyperbolic segment representation of G_D.

Theorem 5.3. *Let D be an instance of* HyperbolicSimpleStretchability. *Then, if G_D is a hyperbolic segment graph, then D is realizable in hyperbolic space.*

Proof. The proof relies on the ordering gadgets introduced in the proof of Theorem 5.1. In order to see that the ordering gadgets indeed force the order of intersections on each line to be correct, even for hyperbolic lines, we first need to understand how they work. We explain them with no distinction of the underlying geometry to show that they work for both hyperbolic and Euclidean segment graphs, as the needed concepts can be used in both geometries.

First, recall the geometric transformation of a pseudoline arrangement D into the graph G_D. The segment s corresponding to one of the initial lines l is completely surrounded by other segments that are part of the ordering gadgets, with each intersection point on s having its own cage, a simple polygon consisting of segments from the ordering gadgets. If there would be a realization of G_D with a different order of intersections, the segment representing the initial line would need to cross the ordering gadget of other intersections at some point. This is not possible without adding additional edges to the graph, thus the order of intersections remains untouched. This means that for any realization of G_D, a hyperbolic line arrangement realizing D can be extracted by extending the segments corresponding to the original lines included in D. □

This was the last step we need to complete our proof framework. We obtain a reduction from HyperbolicSimpleStretchability to Recog(HSEG) similarly to the reduction to Recog(HUDG):

Theorem 5.4. *Recognizing hyperbolic segment graphs is* $\exists\mathbb{R}$*-hard.*

Proof. We have filled the steps (a) - (d). The transformation of instances is the same one as used in the Euclidean case. Theorems 5.1 and 5.2 and show that, when we start with an instance of HyperbolicSimpleStretchability, the Euclidean transformation gives us an equivalent instance of recognizing Euclidean segment graphs, and every Euclidean segment graph is also a hyperbolic one. On the other hand, Theorem 5.3 shows the other direction: If we have a hyperbolic embedding of our segment graph G_D, we can reconstruct a hyperbolic line arrangement that realizes D. Together, this completes the reduction from HyperbolicSimpleStretchability to Recog(HSEG) and the application of the framework. □

5.2. More Potential Problems with Hyperbolic Variants

In the previous section, we showed how to apply our proof framework to another Euclidean $\exists\mathbb{R}$-complete geometric problem. Recognizing segment graphs is only one of the many possible problems. Now we list a few other candidates that could also have an interesting $\exists\mathbb{R}$-complete hyperbolic variant. An overview over more $\exists\mathbb{R}$-complete problems in Euclidean geometry and thus potential candidates for this process can be found in [Bie20].

Intersection graphs of convex sets

Again Schaefer [Sch09] also introduces the generalization of recognizing intersection graphs of convex sets as another $\exists\mathbb{R}$-complete geometric problem:

RECOG(CONV):
Input: Graph $G = (V, E)$
Problem: Can V be represented by a set C of $|V|$ convex sets in the Euclidean plane such that two convex sets intersect if and only if the corresponding vertices are adjacent?

The hyperbolic problem version is apparent: We place the convex sets in the hyperbolic plane instead of the Euclidean one. As the transformation Schaefer proposes is similar to the one for segment graphs, again with ordering gadgets that force a correct order of intersections, the framework should be applicable for this problem as well.

Unit Ball Graphs

Another generalization of a problem we have already seen in this paper is the graph problem of recognizing unit ball graphs. In this problem, we are given a graph $G = (V, E)$ and have to place vertices into d-dimensional space for a given dimension d such that the distance distance of two vertices is less than 1 if and only if they are adjacent. Unit disk graphs are the special case of this for $d = 2$. For a given d-dimensional space $\mathcal{P}$, with $\mathcal{P} = \mathbb{R}^d$ or $\mathcal{P} = \mathbb{H}^d$, the problem can be defined in the following way:

RECOG(UBG):
Input: Graph $G = (V, E)$
Problem: Is there an embedding $f : V \to \mathcal{P}$ that fulfills $\{u, v\} \in E \Leftrightarrow d(f(u), f(v)) \leq 1$?

In the Euclidean version, Kang and Müller [KM12] find a reduction from the recognition problem of unit disk graphs to higher dimension, so the problem is ∃ℝ-complete. Additionally, Kisfaludi-Bak [KB20] studies the graph class of hyperbolic ball graphs and finds many similarities to its Euclidean counterpart. This leads us to believe that this generalization of unit disk graphs is also an interesting problem to consider in this setting.

Linkage Realization

Another problem that could be considered in a hyperbolic context is the problem of LinkageRealizability. Here, we are given a graph $G = (V, E)$ and a length function $l : E \to \mathbb{R}_{>0}$ that fixes the length of every edge in a straight-line drawing of G:

LinkageRealizability:
Input: Graph $G = (V, E)$, function $l : E \to \mathbb{R}_{>0}$.
Problem: Is there a straight-line drawing of G where every edge e has length $l(e)$?

The drawing can of course be done by placing the vertices in either the Euclidean or hyperbolic plane and using the corresponding distance functions to determine if l has been fulfilled. Schaefer [Sch13] shows that this problem is ∃ℝ-complete in its Euclidean variant. For the geometric transformation, he uses Peaucellier linkages, depicted in Figure 5.2, to force points to be on straight lines and he represents line arrangements as graphs using those linkages. The behaviour of Peaucellier linkages in hyperbolic space has been researched by Kourganoff [Kou16], who presents a hyperbolic Peaucellier linkage with a slightly different composition than in the Euclidean case. Using those would require a direct reduction from HyperbolicSimpleStretchability, but his ideas could potentially also be used to show the necessary steps in our proof framework. Either way, the problem is easily definable in a hyperbolic way and seems to allow a hardness proof, which indicates it could be interesting for further research.

Optimal Curve Straigthening

The last problem we consider is the problem CurveToPolygon, defined by Erickson [Eri19]:

CurveToPolygon:
Input: Self-intersecting closed curve γ, integer m.
Problem: Can γ be continuously deformed into a polygon P with at most m vertices without changing the pattern of intersections?

The closed curve γ and the polygon P can be used both in Euclidean and hyperbolic space, defining the two problem versions. We include this problem because it is a geometric problem, but not a graph problem. Also, the reduction by Erickson uses line arrangements in a very direct way: He starts with a different representation of pseudoline arrangements, wire diagrams (depicted in Figure 5.3). He cuts the pseudolines outside of the intersections

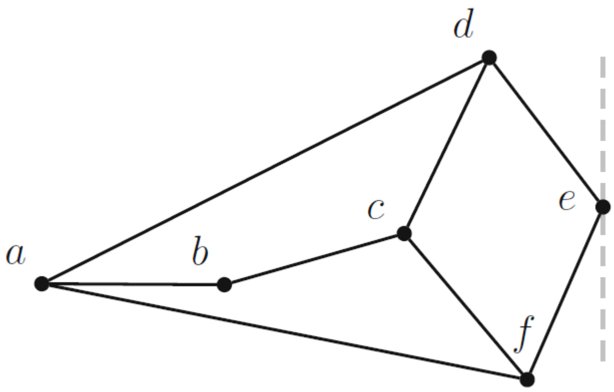

Figure 5.2.: A Peaucellier linkage for LinkageRealizability (Source: [Sch13])

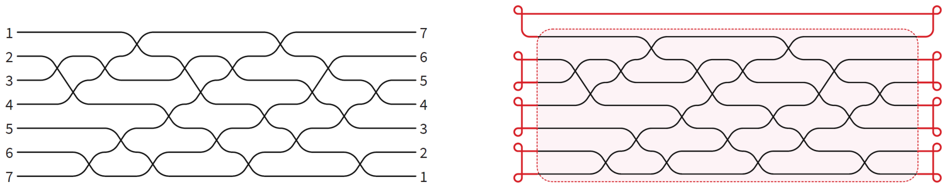

Figure 5.3.: Left: Wire diagram of the pseudoline arrangement. Right: The resulting CurveToPolygon instance. (Source: [Eri19])

and combines neighbouring ends to form a closed curve γ. As he only allows 4 vertices for each pseudoline, any polygon P that is isotopic to γ has to use up the available vertices outside of the intersection points (depicted in Figure 5.4). The polygon thus directly implements a line arrangement that stretches the pseudoline arrangement, which should also work in the hyperbolic case.

These are some of the problems that could be considered in search for problems with interesting properties of their hyperbolic variants. In general, we think that the results for hyperbolic unit disk graphs and the hardness proof framework we established should encourage similar approaches for more geometric problems. It is unclear if the hyperbolic variants often differ from their Euclidean counterparts as it is the case for unit disk graphs, or if the equivalence of hyperbolic and Euclidean variants seen for simple line arrangements is more common. Nevertheless, this could be an interesting field of research that can and should be explored further.

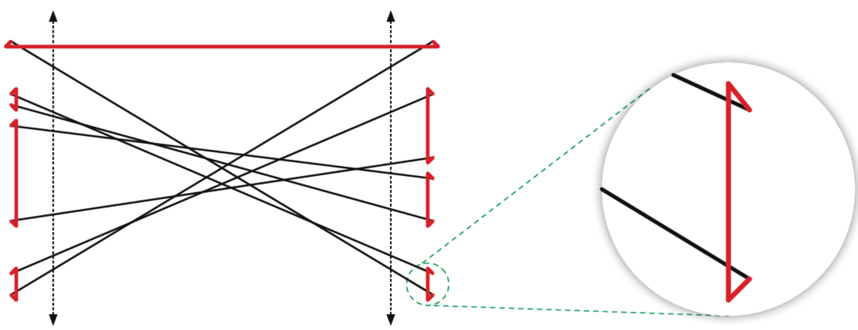

Figure 5.4.: The resulting polygon that realizes the pseudoline arrangement. (Source: [Eri19])

6. Conclusion

In this thesis, we examined geometrical problems that are complete for the complexity class $\exists\mathbb{R}$. Until now, those problems were shown to be $\exists\mathbb{R}$-complete when the underlying geometry was assumed to be Euclidean. We defined problem variants for considering hyperbolic geometry instead and showed that they are as hard as their Euclidean counterparts. In Chapter 3, we first examined the SimpleStretchability problem and showed that the Euclidean and hyperbolic versions are equivalent. In order to do that, we established a way of scaling hyperbolic line arrangements that does not change their combinatorial descriptions. This allowed us to bring the intersection points of the line arrangement to a sufficiently small area to allow us to define an equivalent Euclidean line arrangement.

In Chapter 4, we considered the problem of recognizing unit disk graphs. We recalled the existing proofs for $\exists\mathbb{R}$-completenss of the Euclidean and $\exists\mathbb{R}$-membership of the hyperbolic problem version. Using the Euclidean hardness proof as well as the equivalence of Euclidean and hyperbolic SimpleStretchability, we showed that recognizing hyperbolic disk graphs is also $\exists\mathbb{R}$-hard and thus has the same complexity as its Euclidean counterpart.

We then used the idea for this proof to define a framework that allows similar proofs for other geometric $\exists\mathbb{R}$-complete problems in Chapter 5. We defined the hyperbolic variant of recognizing segment graphs and applied the framework to show $\exists\mathbb{R}$-hardness. Finally, we gave a description of additional candidates of geometric $\exists\mathbb{R}$-complete problems that could have interesting hyperbolic versions and can likely be subjected to this framework.

Open Questions

In Chapter 3, we wrote about the general stretchability problem and its hyperbolic variant. Unlike the special case of simple line arrangements, we showed that there are pseudoline arrangements that are stretchable in the hyperbolic but not in the Euclidean plane. This could be studied further in order to understand what the differences between pseudoline arrangements realizable in hyperbolic and Euclidean space are. Additionally, we motivated that our version of defining the problem via the border line l_0 excludes hyperbolic line arrangements which leaves the question if there is a better way to define HyperbolicStretchability.

In Chapter 4, we showed that recognizing hyperbolic unit disk graphs is $\exists\mathbb{R}$-complete, but found no complexity result for the problem of recognizing strongly hyperbolic unit disk graphs. We offered first thoughts and attempts to solve that problem but could find neither a reduction to show $\exists\mathbb{R}$-hardness nor a proof that the problem is not $\exists\mathbb{R}$-hard. This line of research could be continued to find an answer for the complexity of recognizing strongly hyperbolic unit disk graphs.

The last chapter, Chapter 5, offers another interesting complex of open problems in this thesis. We introduced a framework to show $\exists\mathbb{R}$-hardness for hyperbolic variants of geometric $\exists\mathbb{R}$-complete problems and gave an overview over potential candidates where this framework could be applied. More research could be done if hyperbolic variants can be defined for other problems. If yes, do they define interesting and relevant problem classes different from their Euclidean counterparts, as in the case of unit disk graphs? Or are they equivalent as the variants of SimpleStretchability? We believe that there are interesting results to be found in this area.

Bibliography

[BFKS21] Thomas Bläsius, Tobias Friedrich, Maximilian Katzmann, and Daniel Stephan. Routing in strongly hyperbolic unit disk graphs. *arXiv preprint arXiv:2107.05518*, 2021.

[Bie20] Nicholas Bieker. Complexity of graph drawing problems in relation to the existential theory of the reals. Bachelor's thesis, Karlsruhe Institute of Technology (August 2020), 2020.

[Bru] Arndt Bruenner. https://www.arndt-bruenner.de/mathe/scripts/kreis3p.htm.

[CYRL19] Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32, 2019.

[DKM02] Andreas Dress, Jack H Koolen, and Vincent Moulton. On line arrangements in the hyperbolic plane. *European Journal of Combinatorics*, 23(5):549–557, 2002.

[Doh22] Emil Dohse. On hyperbolic unit disk graphs. Master's thesis, KIT, 2022.

[Eri19] Jeff Erickson. Optimal Curve Straightening is $\exists\mathbb{R}$-Complete. *arXiv preprint arXiv:1908.09400*, 2019.

[EVDHM22] Jeff Erickson, Ivor Van Der Hoog, and Tillmann Miltzow. Smoothing the gap between np and er. *SIAM Journal on Computing*, (0):FOCS20–102, 2022.

[KB20] Sándor Kisfaludi-Bak. Hyperbolic intersection graphs and (quasi)-polynomial time. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1621–1638. SIAM, 2020.

[KM12] Ross J Kang and Tobias Müller. Sphere and dot product representations of graphs. *Discrete & Computational Geometry*, 47(3):548–568, 2012.

[Kou16] Mickaël Kourganoff. Universality theorems for linkages in homogeneous surfaces. *Geometriae Dedicata*, 185(1):35–85, 2016.

[Mat14] Jiri Matousek. Intersection graphs of segments and $\exists\mathbb{R}$. *arXiv preprint arXiv:1406.2636*, 2014.

[MM13] Colin McDiarmid and Tobias Müller. Integer realizations of disk and segment graphs. *Journal of Combinatorial Theory, Series B*, 103(1):114–143, 2013.

[Mnë88] Nikolai E Mnëv. The universality theorems on the classification problem of configuration varieties and convex polytopes varieties. In *Topology and geometry—Rohlin seminar*, pages 527–543. Springer, 1988.

[PKBV10] Fragkiskos Papadopoulos, Dmitri Krioukov, Marián Boguná, and Amin Vahdat. Greedy forwarding in dynamic scale-free networks embedded in hyperbolic metric spaces. In *2010 Proceedings IEEE Infocom*, pages 1–9. IEEE, 2010.

[Sch09] Marcus Schaefer. Complexity of some geometric and topological problems. In *International Symposium on Graph Drawing*, pages 334–344. Springer, 2009.

[Sch13] Marcus Schaefer. Realizability of graphs and linkages. In *Thirty Essays on Geometric Graph Theory*, pages 461–482. Springer, 2013.

[Tar98] Alfred Tarski. A decision method for elementary algebra and geometry. In *Quantifier elimination and cylindrical algebraic decomposition*, pages 24–84. Springer, 1998.

Appendix

A. Computing the Intersection Point of Euclidean Lines

Lemma 6.1. *For given points $O = (0, 0)$, $P_1 = (\alpha_1, d_1)$ and $P_2 = (\alpha_2, d_2)$ that uniquely define a circle c with center M, the point T that is on the diameter of c that includes O can be described in coordinates by $T = (x_T, y_T)$ with*

$$x_T = \frac{\cos(\alpha_1)d_2 - \cos(\alpha_2)d_1}{\sin(\alpha_1 - \alpha_2)}, \; y_T = \frac{\sin(\alpha_1)d_2 - \sin(\alpha_2)d_1}{\sin(\alpha_1 - \alpha_2)}.$$

Proof. The main idea of this proof is to use a system of equations that describe the center of a circle when given three points on its boundary to compute the cartesian coordinates of T, as they are given by $x_T = 2x_M$ and $y_T = 2y_M$. In order to obtain this system of equations, we follow the argumentation of Brünner [Bru]:

In general, the points (x, y) of circle c with center $M = (x_M, y_M)$ and radius r can be described via $(x - x_M)^2 + (y - y_M)^2 = r^2$ because of Pythagoras' theorem. Using variables $A = x_M^2 + y_M^2 - r^2$, $B = 2x_M$ and $C = 2y_M$, this yields the system of equations from the website for three points $(x_i, y_i), i = 1, 2, 3$ on the boundary of C:

$$\begin{aligned}
&(1)\; A - x_1B - y_1C = -x_1^2 - y_1^2\\
&(2)\; A - x_2B - y_2C = -x_2^2 - y_2^2\\
&(3)\; A - x_3B - y_3C = -x_3^2 - y_3^2
\end{aligned}$$

In our case, those points are $O = (0, 0)$ as well as $P_i = (\sin(\alpha_i)d_i, \cos(\alpha_i)d_i)$. Note that $B = x_T$ and $C = y_T$. For A, we can insert O into one of the equations to see that $A = 0$ holds true. Together, this simplifies the system of equations to just the following two (inserting $P_i = (\sin(\alpha_i)d_i, \cos(\alpha_i)d_i)$):

$$\begin{aligned}
&(1)\; \sin(\alpha_1)d_1x_T - \cos(\alpha_1)d_1y_T = -(\sin(\alpha_1)^2d_1^2 + \cos(\alpha_1)^2d_1^2)\\
&(2)\; \sin(\alpha_2)d_2x_T - \cos(\alpha_2)d_2y_T = -(\sin(\alpha_2)^2d_2^2 + \cos(\alpha_2)^2d_2^2)
\end{aligned}$$

As $\sin(\alpha)^2 + \cos(\alpha)^2 = 1$ and by dividing by d_1 and d_2, respectively, we obtain:

$$\begin{aligned}
&(1)\; \sin(\alpha_1)x_T + \cos(\alpha_1)y_T = d_1\\
&(2)\; \sin(\alpha_2)x_T + \cos(\alpha_2)y_T = d_2
\end{aligned}$$

We now resolve the system of equations two times to obtain formulas for both x_T and y_T. We start by solving the first equation for x_T and the second one for y_T:

$$x_T = \frac{d_1 - \cos(\alpha_1)y_T}{\sin(\alpha_1)},\ y_T = \frac{d_2 - \sin(\alpha_2)x_T}{\cos(\alpha_2)}$$

To compute x_T, we insert the formula for y_T from the second equation into the first one:

$$\begin{aligned}
x_T = \frac{d_1 - \cos(\alpha_1)y_T}{\sin(\alpha_1)} &= \frac{d_1}{\sin(\alpha_1)} - \frac{\cos(\alpha_1)}{\sin(\alpha_1)} \cdot \frac{d_2 - sin(\alpha_2)x_T}{\cos(\alpha_2)} \\
&= \frac{d_1\cos(\alpha_2) - \cos(\alpha_1)d_2 + \cos(\alpha_1)\sin(\alpha_2)x_T}{\sin(\alpha_1)\cos(\alpha_2)} \\
&\Leftrightarrow x_T(\sin(\alpha_1)\cos(\alpha_2) - \cos(\alpha_1)\sin(\alpha_2)) = d_1\cos(\alpha_2) - \cos(\alpha_1)d_2 \\
&\Leftrightarrow x_T\sin(\alpha_1 - \alpha_2) = \cos(\alpha_1)d_2 - \cos(\alpha_2)d_1
\end{aligned}$$

In a similar fashion, we can insert the first equation into the second one to obtain

$$\sin(\alpha_1 - \alpha_2)y_T = \sin(\alpha_1)d_2 - \sin(\alpha_2)d_1.$$

$\square$