An Interactive Java Program to Generate Hyperbolic Repeating Patterns Based on Regular Tessellations Including Hyperbolic Circles and Horocycles

A THESIS
SUBMITTED TO THE FACULTY OF THE
GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Shiva Kumar Chittamuru

IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE
Dr. Douglas Dunham
July, 2015

Acknowledgements

I would like to sincerely thank my advisor, Dr. Douglas Dunham for providing constant support and invaluable guidance during the course of my thesis.

I would like to thank my thesis defense committee members Dr. Peter Willemsen and Dr. Steven Trogdon for their enthusiasm and encouragement. I would like to thank my graduate instructors Dr. Hudson Turner, Dr. Ted Pedersen, and Dr. Henry Wang for sharing their vast knowledge and wide area of expertise. I would also like to thank Lori Lucia and Clare Ford and the entire CS department for their assistance.

I would like to thank my friends and fellow thesis members Sakethram Karumuri and Mounika Alla, who contributed to this research, for their ideas, discussions and extensive effort.

Finally, I would like to thank my amazing parents and my brother for their unceasing love, support, encouragement and motivation I have gotten over the years.

Dedicated to

my mom,

Mrs. Aruna Chittamuru,

my dad,

Mr. Seetha Ramaiah Chittamuru,

and my brother,

Sharath Kumar Chittamuru

Abstract

Artists have created various repeating patterns and they have been absorbed into various cultures across the world as art. Notable are the works of the Dutch artist M.C. Escher who was most likely the first to create such artistic patterns. In an age without technology, he mastered these designs through the tedious and time consuming handiwork. Later, several attempts were made by the researchers to generate these patterns through sub-patterns, called motifs, repetitively through a computer algorithm. Currently there are two programs – a C++ program and a Java application that recursively replicate and create these hyperbolic tessellations.

The focus of this paper is to enhance and extend the existing Java application by providing interactive capabilities and adding more complex hyperbolic curves namely hyperbolic circles and horocycles. It enables user to load an existing data file or create a new data file that defines the sub-pattern and other required information pertaining to the design. The program will take that information and generate the repeating hyperbolic pattern for the user. The user can also modify the pattern by using the rich user interface and save it as a data file.

Contents

List of Figures

Chapter 1

Introduction

Creating artistic and intricately entwined geometrical patterns and designs has been a significant part of human history. Every civilisation, irrespective of the demographic, has integrated these designs into their establishments as observed from the rule and ruins of the Sumerians, Chinese, Egyptians, Greek, Arabic, Byzantine, Japanese, Moors, Persians, and Romans. During their times, the architecture of their buildings, floors, walls, and even the ceilings with mosaics in geometric patterns made of hardened clay was inspired by the beauty of the symmetrical and asymmetrical designs. Many mathematicians have explored these patterns, designed, represented and captured their essence through various geometries, notably the Euclidean geometry and the non-Euclidean geometries comprised of hyperbolic and spherical[4]. Escher's Circle Limit patterns are some such examples of hyperbolic geometry[5].

In what Escher identified as a milestone in the development of his career, he explored the concept of representing infinity on a two-dimensional plane by transferring his work from the focus of Euclidean geometry to hyperbolic tessellations. The resulting designs followed the notion of Schläfli form of $\{p, q\}$, where "p" denotes a regular p-sided polygon, and "q" specifies the number of them that meet at each vertex. They applied symmetric transformations to their patterns to obtain new patterns, thereby modifying the symmetry of the original pattern and sometimes even projecting it onto entirely different geometry to create a completely new pattern. This motivated other artists to create several patterns with several objects to form tillings on planes. This is the motivation for my research.

There are currently few programs that create repeating hyperbolic patterns. The existing algorithms to draw regular tessellations of a hyperbolic plane were initially implemented in C by using the X/Motif interface development system[1][6], and thus were not portable across platforms. To achieve portability and to incorporate several extensible features, a new algorithm in Java has been implemented by Vejendla[8] while also providing a rich user interface. The proposed thesis will enhance and extend this existing application by providing better interactive capabilities and adding special curves such as hyperbolic circles and horocycles. The research will use Poincaré Disk Model and Weierstrass Model of hyperbolic geometry as the basis for pattern representations and calculations respectively.

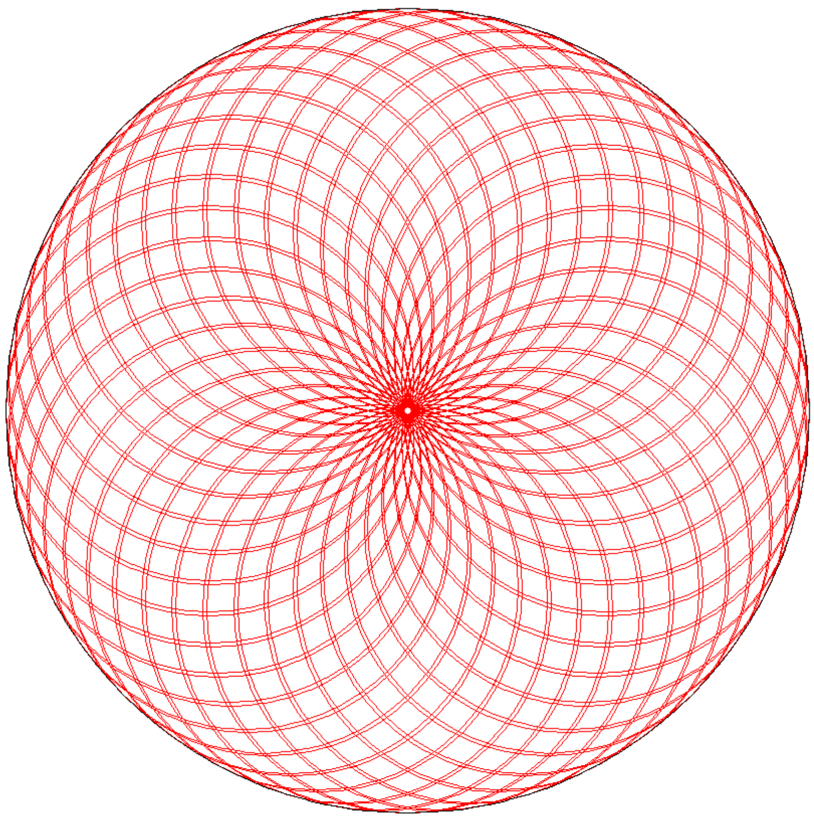

Figure 1.1: An example of program output

Chapter 2

Types of Geometries

The word “geometry” is a branch of Mathematics originated from the Greek word geometrein which means “to measure earth.” It deals with study and questions of shapes and sizes of objects of various dimensions. This study with a set of rules and procedures has helped the people of ancient history with their everyday life and has been significantly progressing thus far. Modern geometry deals with much more complex problems such as the study of differential geometry and gravitational fields[13]. The common areas of geometry used for artistic purposes are: Euclidean or Planar Geometry (which includes shapes such as lines, circles, and squares), Spherical Geometry (which includes shapes such as spherical lines and spherical polygons), and Hyperbolic Geometry (which includes shapes such as hyperbolic lines and hyperbolic curves).

2.1 Euclidean Geometry

Euclidean geometry is a widely used geometrical system which was named after Euclid, a Greek mathematician from 300 BC. It is an axiomatic system which means that all the theorems are derived from a finite number of axioms. Euclid postulated five axioms for the basis of this geometrical approach which are listed as follows:

1. Any two points can be joined by a straight line segment.
2. Any straight line segment can be extended indefinitely in a straight line.
3. Given any straight line segment, a circle can be drawn having the line segment as the radius and one endpoint as the center.
4. All right angles are congruent.
5. If two lines intersect a third line in such a way that the sum of the inner angles on one side is less than that of the two right angles, then the two lines inevitably must intersect

each other on that side if extended far enough. This axiom is also known as the parallel postulate.

The adjective “Euclidean” was not necessary for many centuries because no other geometry had been conceived. Mathematicians perceived the parallel postulate as a special postulate, inconsistent with the first four postulates. Since neutral geometry, consisting of Euclid’s first four axioms, did not in itself imply the parallel postulate, they believed that there must be a different geometry that was based on the first four axioms and the negation of the parallel postulate. They did not doubt its correctness, but they thought that it was a theorem that can be proved rather than an axiom. At the end of 18th century, many mathematicians attempted to prove the inconsistency of negation of parallel postulate with the first four postulates. They never reached to the contradiction but this idea led to the discovery of complete and consistent non-Euclidean geometries.

2.2 Non-Euclidean Geometry

During a time when communication was not as extensive as now, the information about a possible discovery or a proposition of an idea were lost in the times, only to be stumbled upon later. Carl Friedrich Gauss and Ferdinand Karl Schweikart were two such individuals who worked out the essentials of an alternative geometry but did not publish their findings for the world to know. After span of several years, two mathematicians, János Bolyai and Nikolai Ivanovich Lobachevsky, unknown to the other individually published their study of Non-Euclidean geometry.

When Gauss was introduced to the work of Bolyai, he found an intersection of his postulates, which he never published for the fear of controversy, with Bolyai’s findings. It was Gauss who indeed coined the term ‘non-Euclidean’ postulates which later was to become synonymous with “Hyperbolic Geometry”. Bolyai’s work however concluded with an opinion stating that it is not possible to decide through mathematical reasoning alone if the geometry of the physical universe is Euclidean or non-Euclidean.

Bernhard Riemann, in the second half of 19th century, founded the field of Riemannian geometry, discussing in particular the ideas now called manifolds, Riemannian metric, and curvature. He constructed an infinite family of geometries which are not Euclidean by giving a formula for a family of Riemannian metrics on the unit ball in Euclidean space. The simplest of these is called elliptic geometry and it is considered to be a non-Euclidean geometry due to its lack of parallel lines. By formulating the geometry in terms of a curvature tensor, Riemann allowed non-Euclidean geometry to be applied to higher dimensions.

2.2.1 Hyperbolic Geometry

Hyperbolic geometry, one of the most useful non-Euclidean geometries, is the geometry discovered by Bolyai, Gauss, Lobachevsky, and Schweikart. It is the geometry of hyperbolic space with a constant negative curvature. An example of hyperbolic plane is a saddle-shape plane (as shown in Figure 2.1).

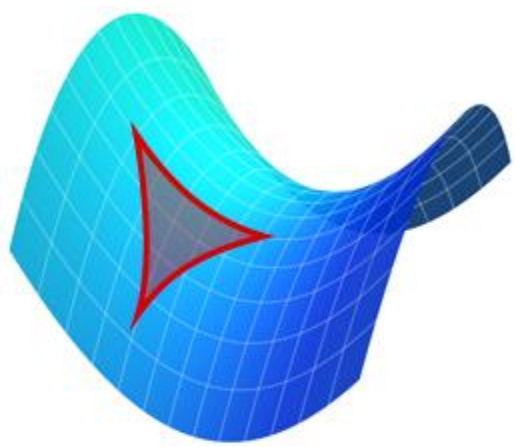

Figure 2.1: A saddle representing hyperbolic plane

In this geometry, all of Euclid's axioms hold with the exception of the parallel postulate. Instead of the parallel postulate, an axiom called the Hyperbolic axiom is used. This axiom states that if there exists a line L and a point P not on the line L, there are at least two distinct lines (A and B) parallel to L passing through P (as shown in Figure 2.2) .

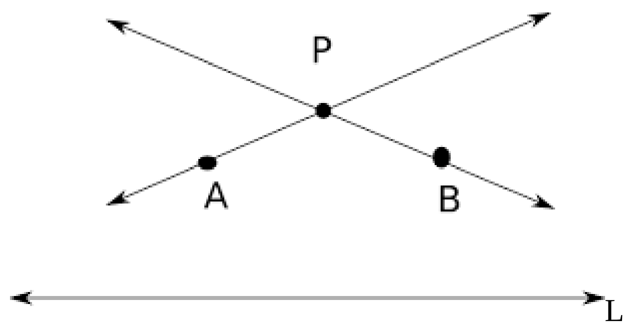

Figure 2.2: Hyperbolic Axiom

This axiom is used to prove and obtain many other useful properties of hyperbolic geometry. Some of the properties of hyperbolic geometry are:

1. The sum of the angles of a triangle is less than 180◦.
2. Rectangles do not exist.
3. All convex quadrilaterals have angle sum less than 360◦.
4. If two triangles are similar, they are congruent, that is, if the angles of the triangles are equal, so are their sides [9].

There are several models to represent hyperbolic geometry such as the Poincaré Disk, the Klein and the Weierstrass models. These models are discussed extensively in the next chapter.

2.2.2 Spherical Geometry

Spherical geometry is another type of non-Euclidean geometry defined as the study of figures on the surface of a sphere, and it can be viewed as the 3-dimensional version of Euclidean geometry. Bernhard Riemann and Ludwig Schläfli are the pioneers of this geometry. The basic element of spherical geometry is the sphere, a three-dimensional surface made up of the set of all points in space at a given distance from a fixed point called the center. If we take an arbitrary plane and sphere lying in the plane, there are three possibilities for their intersections. First, the plane and the sphere never intersect. Secondly, the plane may lay tangent to the sphere forming one distinct point of intersection. In the final case where the plane passes through the sphere, the intersection forms a circle. There is a unique case within the final case, that is, if the plane passes through the center of the sphere, the circle formed is known as a *great circle*. Great circles are the largest circles that can be drawn on a sphere and are defined as the shortest distance between two given points. For example, equator and longitudinal lines of the earth are great circles and they divide sphere into two hemispheres.

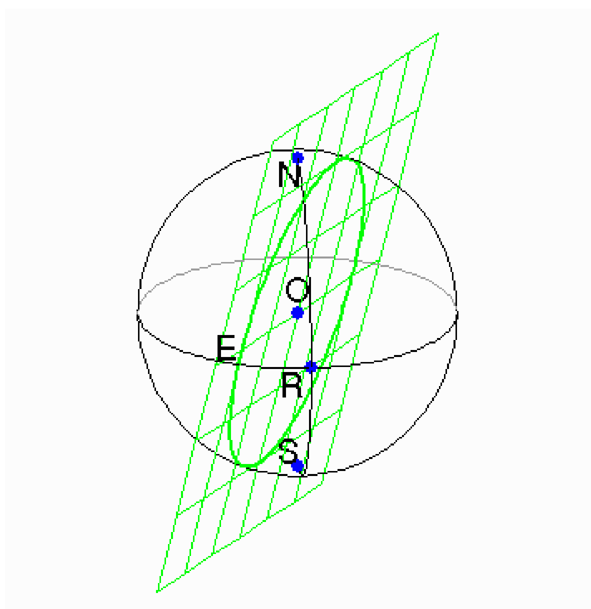

Figure 2.3: Great lines and antipodal points

In the Figure 2.3**,** point O represents the sphere's center, and the great circle is represented by circle E. The points N and S represent the two antipodal points. *Antipodal points* are the points formed by the intersection of a sphere and a line that passes through the center of the sphere. One recognizable example of antipodal points would be the north and south poles of the earth[10].

Two practical applications of the principles of spherical geometry are navigation and astronomy. As the shortest distance from one point to another point on a sphere is along the arc of a great circle, it is interesting to know that the shortest flying distance from Florida to Philippine Island is a path across Alaska. This is because Florida, Alaska, and the Philippines lie on the same great circle and so are collinear in spherical geometry.

The following are some of the properties of Spherical geometry:

1. There are no straight lines in spherical geometry and hence there are no parallel lines.
2. Any two lines intersect in two diametrically opposite points are called *antipodal points*. Any two points that are not antipodal points determine a unique line.
3. When three curved arcs intersect one another, a spherical triangle is formed. A spherical triangle is any three-sided region enclosed by sides that are the arcs of great circles. The angle sum of a spherical triangle is greater than 180° and less than 540°.
4. Two spherical triangles are not only similar, but congruent if they share same angles.

Chapter 3

Models of Hyperbolic Geometry

A *model* is used to give meanings to all the objects in hyperbolic space in such a way the axioms become true statements. The hyperbolic geometry models are classified into finite and infinite models based on their representations. Some models of hyperbolic geometry can represent hyperbolic objects in a finite portion of Euclidean 2-space. The Poincaré Disk Model and the Beltrami-Klein Model are the finite hyperbolic geometry models, and the Weierstrass Model is an infinite model for hyperbolic geometry that is embedded in Euclidean 3-space. The finite models have boundaries which play a key role in the definition or representation of parallel lines and other geodesics in hyperbolic space. The process of projecting the objects of one model onto the other model is called isomorphism. The following sections describe the finite and the infinite models and the isomorphism between them in detail.

3.1 The Poincaré Disk Model

Of what is seemingly identified as one of the easiest models for hyperbolic geometry, the Poincaré Disk Model, was created by the famous French Mathematician, physician and philosopher, Henri Poincaré. It is also accepted as the Poincaré ball model (in three or more dimensions) or the Conformal disk model (in two dimensions). A model of n-dimensional hyperbolic geometry, this model is *conformal*, with the hyperbolic plane being depicted as points lying inside a ball or an n-dimensional disk. This representation is in turn a n-dimensional hyperbolic geometry wherein all the points of this particular geometry lie in the n-dimensional Euclidean geometry. The disk is called the Poincaré disk and it can also be referred as the bounding circle which is an imaginary circle or the circle at infinity. From this point. the terms “Poincaré disk” and “bounding circle” will be used interchangeably.

A "point" in this model is represented as any point in the Euclidean unit circle which is centered at the origin. It is defined as the 2-tuple (x, y), such that $x^2 + y^2 < 1$. The points on the circumference of the bounding circle are at infinity and are referred as ideal points.

A “line” is represented in two ways, either as a diameter which is an open chord passing through centre of the disk, or as an open arc[9] that is orthogonal to the boundary of the circle (as shown in the Figure 3.1).

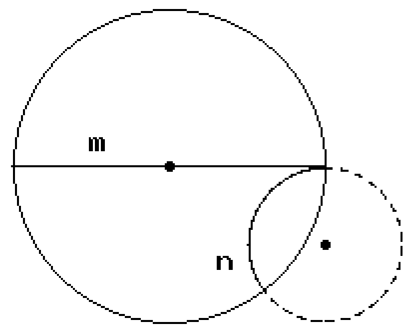

Figure 3.1: Lines in Poincaré Disk Model

The terms "lie on" and "between" retain their Euclidean meanings. As it is a conformal model, measuring the angle between the tangents of these intersecting "lines" at their point of intersection will give the angle between these two lines.

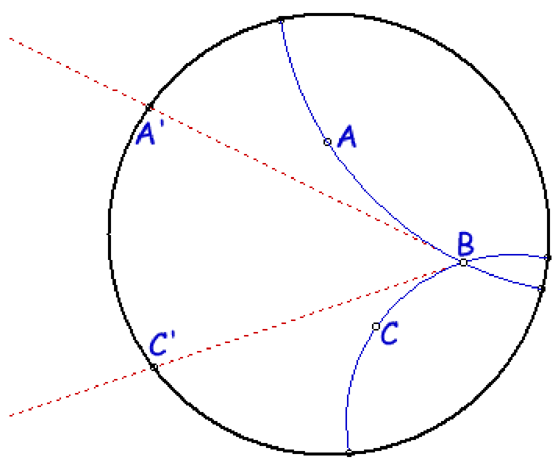

Figure 3.2: Angle between two intersecting lines in Poincaré Disk Model

In this model, a Euclidean circle represents[14]:

1. a hyperbolic circle if it is entirely inside the unit disk.
2. a horocycle if it is inside the unit disk except for one point where it is tangent to the unit disk.
3. a hyperbolic line if it cuts the unit disk orthogonally.
4. an equidistant curve if it cuts the unit disk non-orthogonally in two points.

3.2 The Beltrami-Klein Model

Proposed by the German mathematician, Felix Klein, this model is similar to the Poincaré Disk Model in many ways except that it is not a conformal model as the Disk Model. This means that the angles are not accurately represented by this model.

A "point" on this plane, as in Poincaré Disk Model, has the same meaning as in Euclidean geometry. A "line" in this model is an open chord on the unit circle, where the open chord is any chord of the circle with its end points removed (as shown in the Figure 3.2). The terms "lies on" and "between" yet again maintain their Euclidean meanings[9].

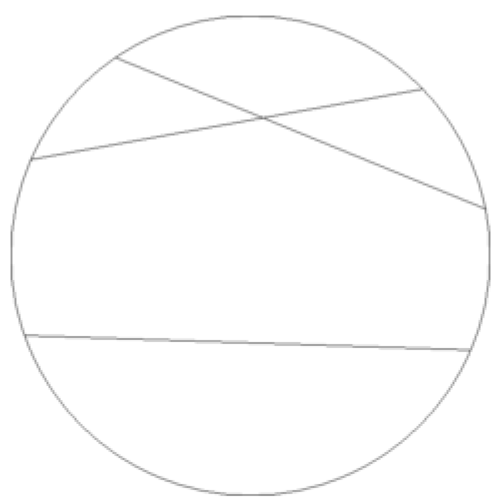

Figure 3.3: Lines in Klein Model

3.3 Comparison of the Klein and Poincaré Disk

With almost similar attributes between them, these models represent hyperbolic space on the unit circle.

When a vector p is used to represent a point on the Poincaré Disk Model, the same point on the Klein Model (say vector k) can be calculated as

$$k = \frac{2p}{1 + p\,.p}$$

And conversely, a point p on the Poincaré Disk Model can be calculated from the point k on the Klein Model as follows:

$$p \; = \frac{k}{1 + \sqrt{1 - k\,.k}} = \frac{(1 - \sqrt{1 - k\,.k})\;k}{k\,.k}$$

Between any two ideal points, a line in the Klein Model is represented as an open chord between these points and in Poincaré Disk Model, it is represented as a circular arc on the two-dimensional space which is generated by the vectors of these ideal points, orthogonal to the boundary of the circle.

A projection, which is a ray from the center of the circle passing through a point from one model to the other model from the centre of disk, establishes a relation between these two models.

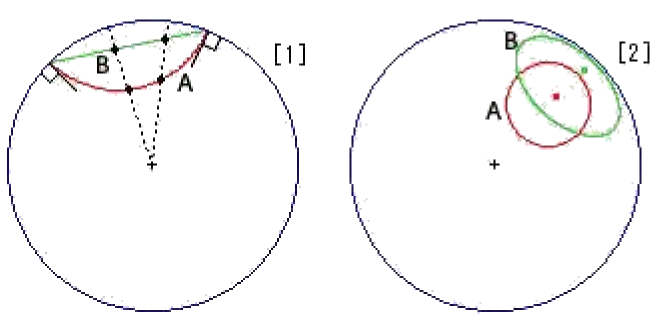

Figure 3.4: Relationship between the Klein Model and the Poincaré Disk Model

In Figure 3.3, the Poincaré and Klein disks are superimposed on each other in which red figures belong to Poincaré Disk Model and green figures belong to Klein Model. [1] shows the relationship of a hyperbolic line and [2] shows the relationship of a hyperbolic circle with a radius of 1 unit.

3.4 The Weierstrass Model

This model is the infinite model of hyperbolic geometry represented on the surface in Euclidean space rather than a disk, unlike Poincaré Disk and Klein models. This entire model is the hyperbolic space represented on the surface of a hyperboloid. As the hyperboloid is a 3-dimensional cone-like structure, this model turns beneficial in obtaining the other two finite models through the projections. The “lines” in this plane are intersections of the surface with planes through the origin.

A hyperboloid is represented mathematically as described by the equation

$$< X, X > = x^2 + y^2 - z^2 = -k^2$$

where *X* is a vector *(x, y, z)*. This divides the hyperboloid into 2 sheets: upper sheet and the lower sheet[11]. As these points on the lower sheet are a reflection of the upper sheet, the lower sheet is discarded. And to denote this upper sheet, a newer mathematical equation

$$< X, X > = -K^2 \quad and\ z > 0$$

is obtained and a point in this model is to satisfy this equation. A line is that section where a plane passing through the origin intersects the hyperboloid. A point L "lies on" the plane if it satisfies the equation

$$< X, L > = 0\ and\ z > 0.$$

And any line R that can be expressed as a linear combination of P and Q is said to be in "between" the points P and Q.

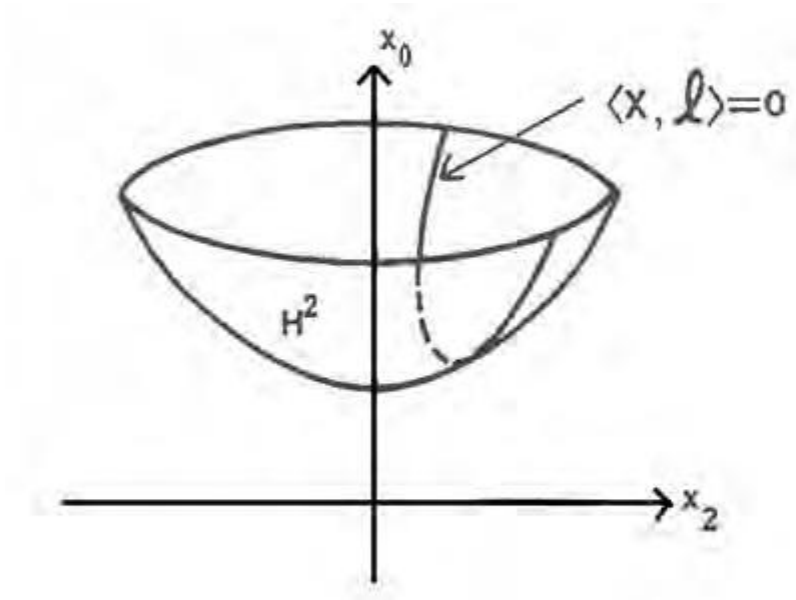

Figure 3.5: A "line" in Weierstrass Model

3.5 Isomorphism

When a model is mapped to another model and it still preserves its structure, an isomorphism is said to exist between these models. The Weierstrass Model can be projected to obtain the Klein and the Poincaré Disk Models, thus this model is isomorphic to the other two models. The Weierstrass Model is used when doing all computations for the transformations and the Poincaré Disk Model is used for displaying the results[9].

3.5.1 Weierstrass - Poincaré Disk Model Isomorphism

The Poincaré Disk Model is obtained by stereographic projection of the 3D Weierstrass Model onto the x-y plane. The projection is towards the point (0, 0, -1). It is given by

$$[x, y, z] \rightarrow \frac{1}{z+1}[x, y, 0]$$

The Poincaré to Weierstrass inverse projection is given by

$$[x, y, 0] \rightarrow \frac{1}{(1 - x^2 - y^2)}[2x, 2y, 1 + x^2 + y^2]$$

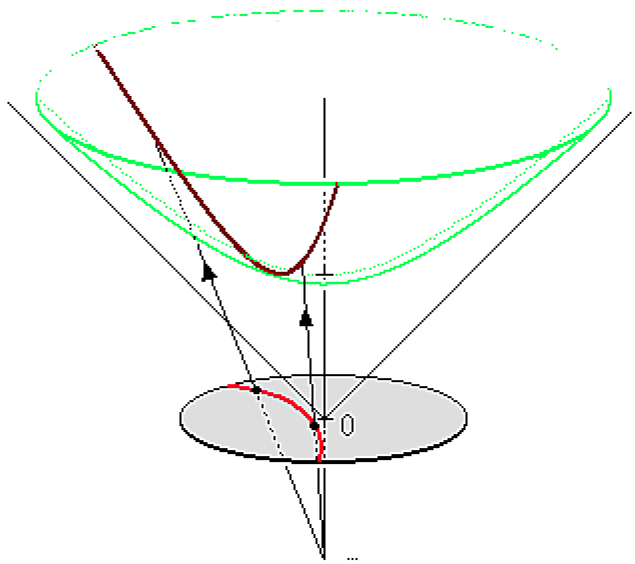

Figure 3.6: The Hyperboloid Model to represent a line in Weierstrass and Poincaré Disk Model

In the Figure 3.6, the red circular arc represents the line in the Poincaré Disk Model. It is projected onto the green hyperboloid and forms a brown arc which represents the line in the Weierstrass Model.

3.5.2 Weierstrass - Klein Model Isomorphism

The Klein Model is obtained by stereographic projection of the Weierstrass Model onto the z=1 plane. The projection is directed towards the point (0, 0, 0). It is given by

$$[x, y, z] \rightarrow [\frac{x}{z}, \frac{y}{z}, 1]$$

The Klein to Weierstrass inverse projection is given by

$$[x, y, 1] \rightarrow \frac{1}{(1 - x^2 - y^2)} [x, y, 1]$$

Chapter 4

Special Curves

4.1 Hyperbolic Circles

Euclid's third axiom states that a circle can be drawn with any center and any radius. A circle in the hyperbolic plane is the locus of all points at a fixed distance from the center, similar to that in the Euclidean plane[15]. Hence, the hyperbolic circle still holds good with Euclid's third axiom. Hyperbolic circles and Euclidean circles differ in centres, radii, circumferences and area except on very small scales where hyperbolic geometry approximates Euclidean geometry. Furthermore, a hyperbolic circle differs from a Euclidean circle since a hyperbolic circle cannot be drawn through every three non-collinear points. Figure 4.1 shows a hyperbolic circle with red point as a center, blue point as a point on the circle and red line as a radius.

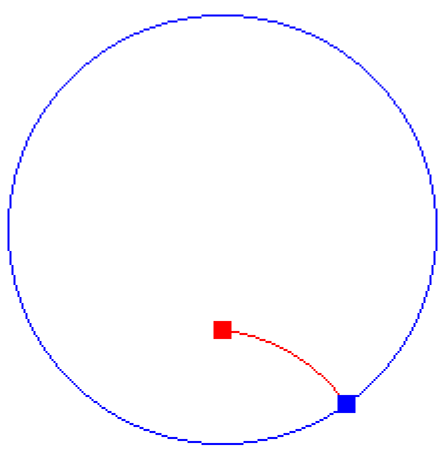

Figure 4.1: A Hyperbolic Circle

In the Poincaré Disk Model, a Euclidean circle represents a hyperbolic circle if it is entirely inside the unit disk[14]. If the center is the origin, the hyperbolic centre is same as the Euclidean center. However, as we move towards the edge of the disk, the Euclidean distances are distorted in the Poincaré disk and centers of hyperbolic circles are not preserved (as shown in the Figure 4.2), albeit all points on the circumference of the circle are at the constant hyperbolic distance from the center of the circle (as shown in the second Figure 4.3). Therefore, a hyperbolic circle is a Euclidean circle with a different center and distorted radii.

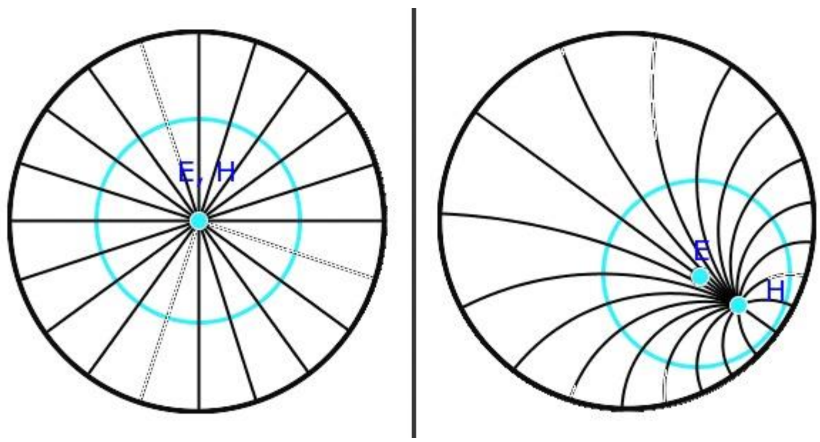

Figure 4.2: Hyperbolic circle with Euclidean and hyperbolic centers

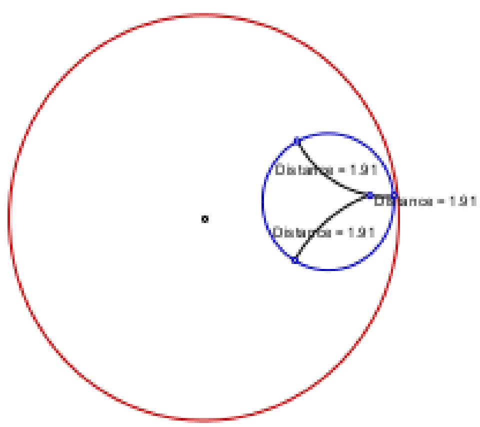

Figure 4.3 Hyperbolic circles with distorted radii in the Poincaré Disk Model

If two or more hyperbolic circles have a common center, they are concentric and are called concentric hyperbolic circles (as shown in the Figure 4.4).

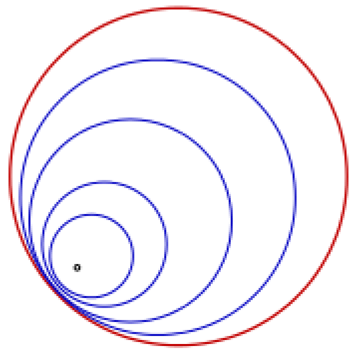

Figure 4.4 Concentric hyperbolic circles

The following are some of the properties of hyperbolic circles:

1. There is a unique Euclidean circle with a given center which passes through a given point. Similarly, there is a unique hyperbolic circle with a given center which passes through a given point.
2. A Euclidean circle and a hyperbolic circle have different centers and radii. However, a unique radius is associated with each point of a hyperbolic circle similar to that of a Euclidean circle.
3. A hyperbolic line and a hyperbolic circle meet in at most two points.
4. Two distinct hyperbolic circles meet in at most two points.
5. If a hyperbolic line contains an interior point P of a hyperbolic circle, then the hyperbolic line and the hyperbolic circle meet in two distinct points, with P between these points.
6. A hyperbolic circle with center C and radius r is symmetric about the hyperbolic line l if and only if C lies on l.
7. If a hyperbolic line l meets a hyperbolic circle in a single point P, then l is a hyperbolic tangent to the circle at P. If P lies on the hyperbolic circle with center

C and radius r then there is a unique hyperbolic tangent to the circle at P, and the tangent is perpendicular to the hyperbolic line PC.

8. A chord of a hyperbolic circle is a segment joining its two points. The radius bisects a chord of a hyperbolic circle at right angles.
9. The hyperbolic circumference and area of a circle are always greater than the Euclidean circumference and area of the same circle for radius $r > 0$ respectively[15].
10. Any three non-collinear points determine a Euclidean circle but they do not necessarily determine a hyperbolic circle.

4.2 Horocycles

In hyperbolic geometry, a horocycle is a curve whose normals converge asymptotically. It is also called a oricycle or a oricircle or a circle limit. Its center is an ideal point which is at infinity in the hyperbolic plane. It can also be described as the limit of the circles that share a tangent in a given point, as their radii go towards infinity. Such a circle of infinite radius in Euclidean geometry would be a straight line, but it is a curve in hyperbolic geometry. Figure 4.5 demonstrates a blue horocycle formed by the convergence of red normals to an upper central point.

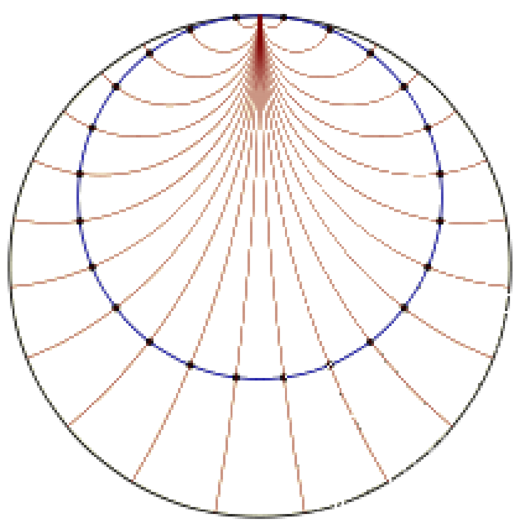

Figure 4.5 - A Horocycle

In the Poincaré Disk Model, horocycles are represented by Euclidean circles that are internally tangent to the bounding circle. Let *l* be a diameter of the bounding circle *b* whose interior represents the hyperbolic plane, and let Q be its center. Let *d* be a hyperbolic circle with Euclidean center R and hyperbolic center P. As the circle *d* is pulled towards the ideal point S, its hyperbolic center P overlaps with S forming a horocycle *h*. Now the center of this horocycle is P' or S its Euclidean center is pulled up to the midpoint of SQ. This is clearly demonstrated in Figure 4.6.

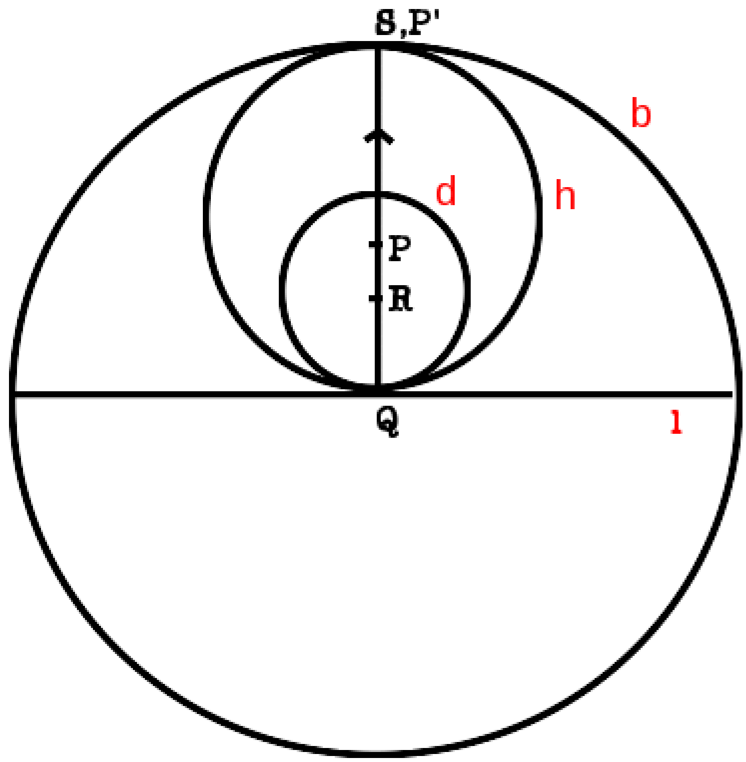

Figure 4.6: A Horocycle in the Poincaré disk

Two or more horocycles tangent to the bounding circle are concentric and are called concentric horocycles (as shown in the Figure 4.7). They have a common centre, an ideal point, which is at infinity and it resides on the edge of the bounding circle.

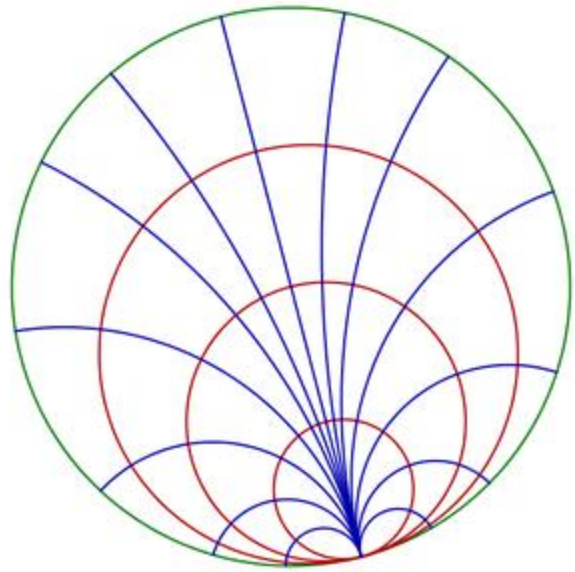

Figure 4.7: Concentric Horocycles

The following are some of the properties of horocycles[14]:

1. There is a unique circle with a given center which passes through a given point. Similarly, there is a unique horocycle with a given direction which passes through a given point.
2. Two concentric circles have no common point. Similarly, two codirectional horocycles have no common point.
3. A unique radius is associated with each point of a circle. Similarly, a unique radius is associated with each point of a horocycle even though the center appears very close to the edge of Poincaré disk.
4. No line can meet a horocycle in more than 2 points.
5. A tangent to a horocycle at a point on the horocycle is defined to be the line through the point is perpendicular to the radius associated with the point.
6. A chord of a horocycle is a segment joining its two points. The radius bisects a chord of a horocycle at right angles.

Chapter 5

Hyperbolic Patterns

5.1 Tessellations

A tessellation or tiling is defined as the pattern of shapes that fit together perfectly. It is a pattern that covers a surface or a plane is covered with no overlaps or gaps. It is a symmetric design featuring objects, animals, people, etc., which can fit together in repetitive patterns like simple jigsaw puzzles. Figure 5.1 shows some examples of these tessellations.

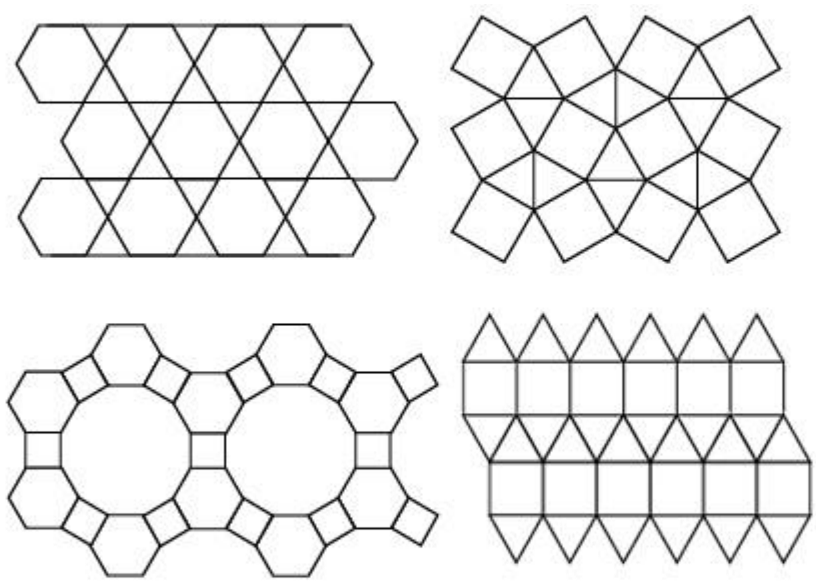

Figure 5.1 Euclidean Tessellations

5.2 Repeating Hyperbolic Patterns

A repeating pattern is formed by replicating a sub-pattern called motif, copies of which may or may not interlock. The focus of this thesis is on repeating hyperbolic patterns that include special curves. Regular tessellations {p, q} of the hyperbolic plane are good examples of repeating hyperbolic patterns. The notation {p, q} denotes that the pattern is composed of regular p-sided polygons which meet q congruent copies at a vertex. While the patterns can be semi-regular and irregular, the scope of this thesis is limited to regular hyperbolic patterns.

The necessary and sufficient condition for a tessellation {p, q} to be in hyperbolic plane is $(p-2)(q-2) > 4$. Spherical tessellations result in $(p-2)(q-2) < 4$, and tessellations with $(p-2)(q-2) = 4$ are Euclidean tessellations. Figure 5.2 is an example a repeating hyperbolic pattern generated by the program.

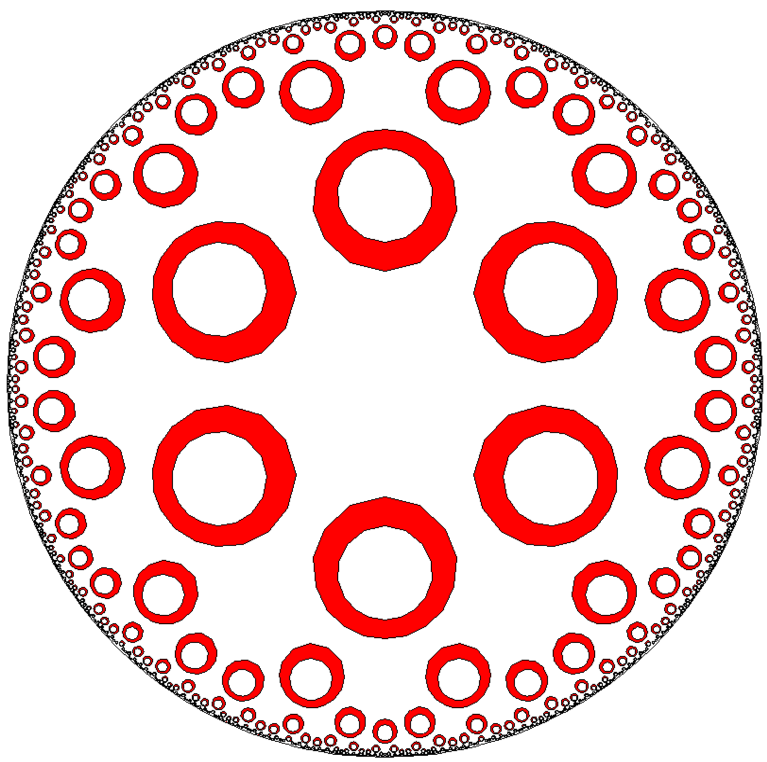

Figure 5.2: A computer generated version of a pattern based on {6, 4} tessellation

5.2.1 Symmetry Groups

An isometry that transforms a pattern onto itself is known as a symmetry operation, or symmetry for short. This is a distance preserving mapping between two metric spaces. The set of all these symmetry operations are collectively known as the symmetry group. The fixed lines of reflection in a tessellation {p, q} are mirrors or lines of symmetry[2]. They divide each p-gon into 2p right triangles with acute angles π/p and π/q. Owing to the hyperbolic geometrical principles, they do not add up to 2π. The symmetry group of the tessellation {p, q} is denoted by [p, q]. The reflections across the sides of each triangle generate this symmetry group. Figure 5.3 is an example of the symmetry group [6, 4] of the regular tessellation {6, 4}. It also shows the lines of symmetry for the central p-gon.

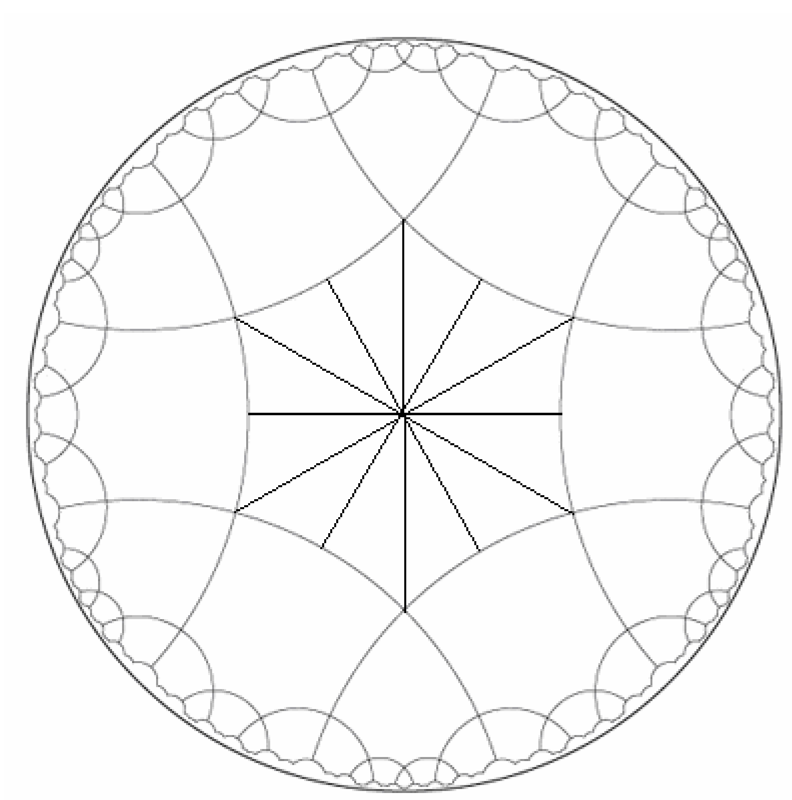

Figure 5.3: A pattern with [6, 4] symmetry group

The symmetry group [p, q]+, is a subgroup of symmetry group [p, q] of index 2[12] and can be used to generate repeating patterns in two approaches. As this is a subgroup of [p, q] of index 2, it has half as many symmetries from the group [p, q]. The first approach to generate the patterns would include all symmetries from [p, q] which are the result of applying an even number of reflections from [p, q]. The second approach would be applying any two of the following three rotations: 180, 360/p, and 360/q degrees about the corresponding vertices of the right triangle formed by the mirrors. Figure 5.4 shows a pattern generated using [5, 4]+ symmetry group.

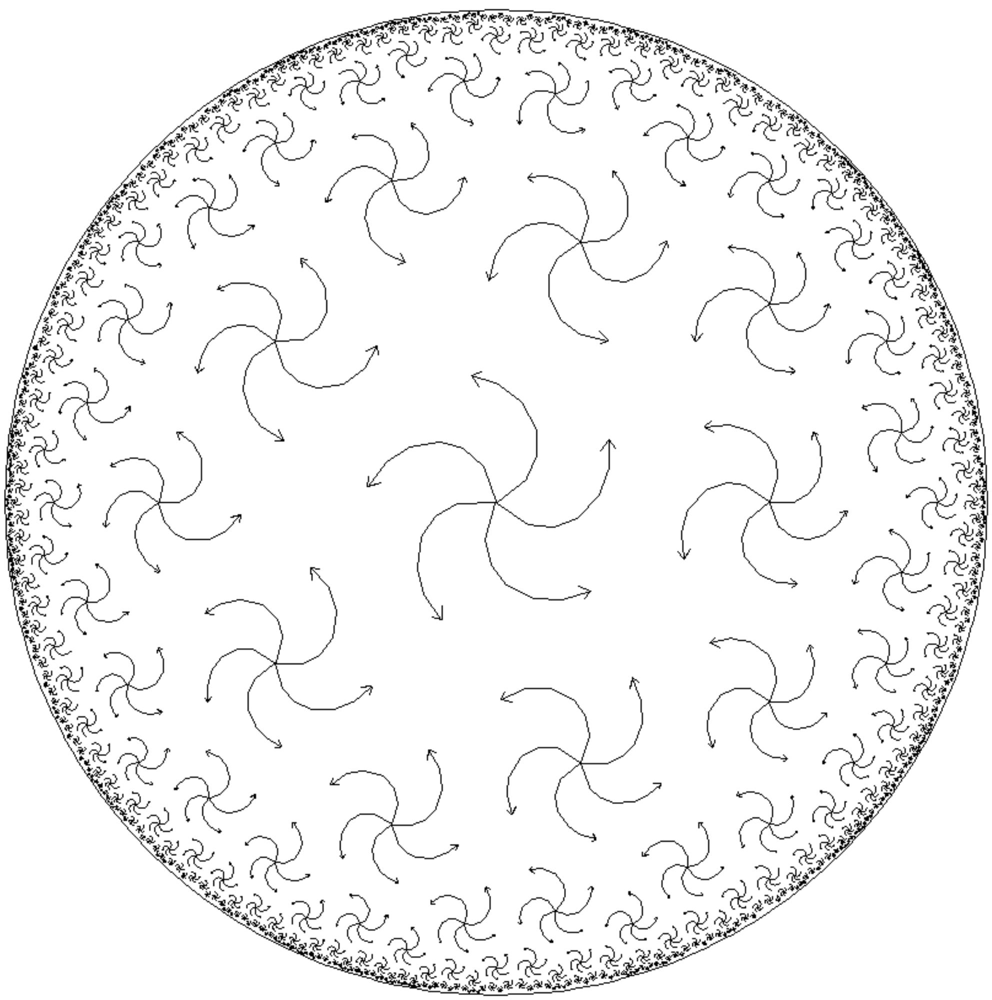

Figure 5.4: A pattern with [5, 4]+ symmetry group

Another variant, a symmetry group [p+, q], is also a subgroup of [p, q] of index 2[12]. A repeating pattern can be generated using this symmetry group by rotations of 2**π**/p about the center of p-gon and reflections across the edges of the regular tessellation {p, q}. Figure 5.5 shows a pattern generated using [5+, 4] symmetry group. It can be noticed that that Figure 5.4 has all the arrows oriented in a counterclockwise direction, whereas Figure 5.5 has arrows oriented in both clockwise and counterclockwise directions.

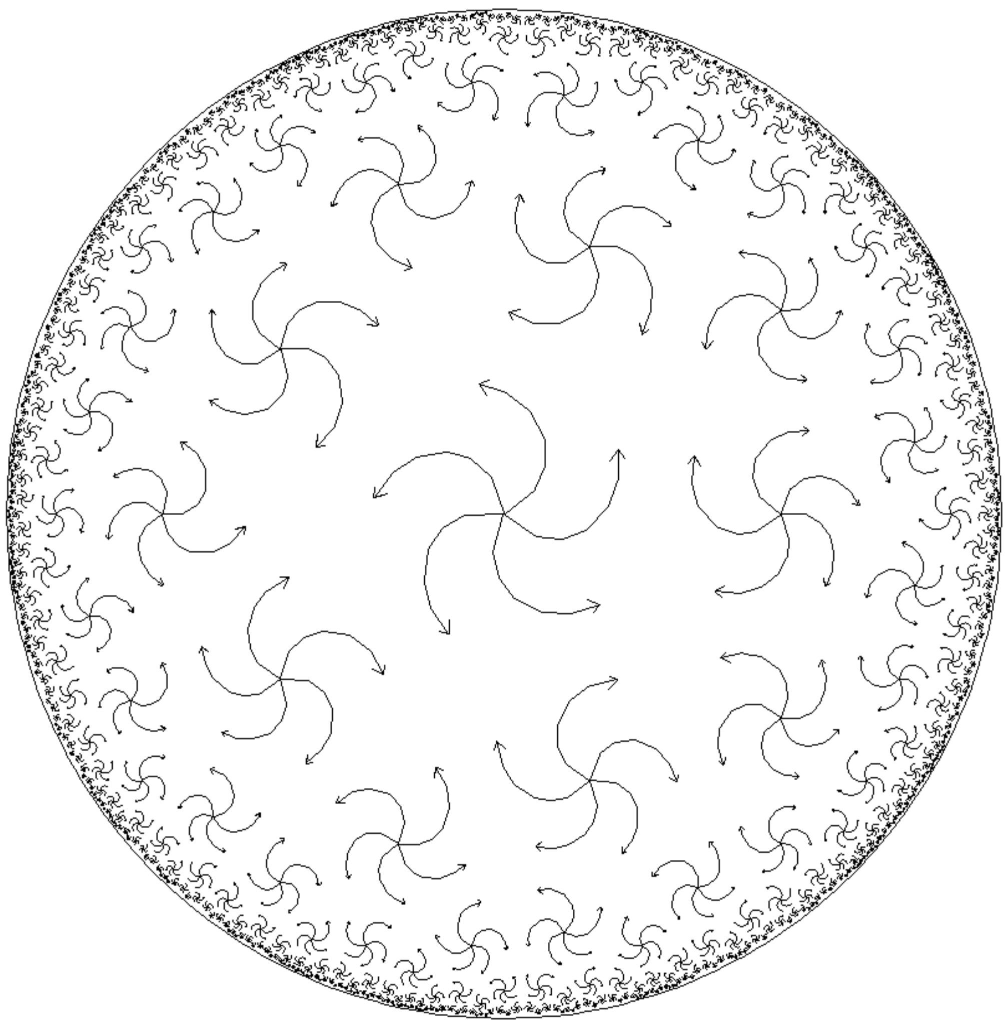

Figure 5.5: A pattern with [5+, 4] symmetry group

5.2.2 Motif

A sub-pattern that is used generate repeating patterns is known as a motif. As defined by Dunham [12], if the hyperbolic plane is covered without overlapping the transformed copies of a connected set under elements of a symmetry group, that set is called fundamental region for the symmetry group. If the motif covers the entire fundamental region then it results in the repeating hyperbolic pattern which will be interlocking. Figure 5.6 is an example of an interlocking pattern, whereas Figure 5.4 and Figure 5.4 are examples of non-interlocking repeating hyperbolic patterns. A fundamental region with dark boundaries is also marked in the Figure 5.6.

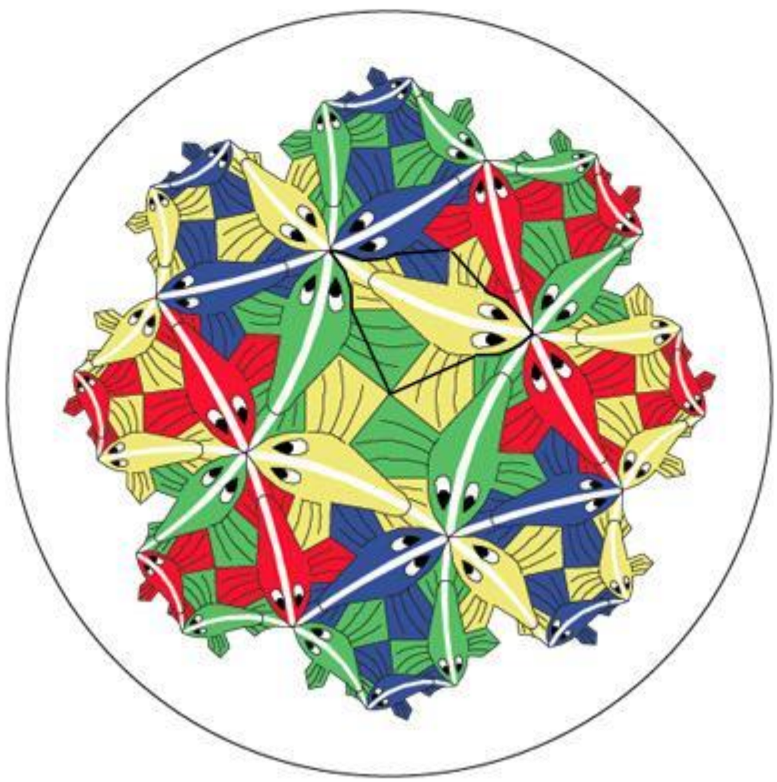

Figure 5.6: An interlocking repeating hyperbolic pattern

5.3 Repeating Pattern Generation Algorithm

The algorithm designed by Dunham [3] creates the repeating hyperbolic patterns by replicating the motifs. Vejendla[8] implemented this algorithm and created a portable Java application. The focus of this thesis is to enhance and extend that application by adding special hyperbolic curves. The algorithm is culminated in a 2-step procedure as described below.

The first step of this process involves the creation of the *central p-gon pattern* by replicating the fundamental region. This is done by rotating the motif around the p-gon center and reflecting it across the diameters and perpendicular bisectors of edges until the p-gon is filled with copies of the motif. As the p-gons of a {p, q} tessellation are arranged in layers, these p-gon patterns which reside on the first layer of the repeating hyperbolic pattern may protrude from the p-gon as long as there are corresponding indents resulting in interlocking final pattern. Figure 5.7 shows the creation of the first layer of the repeating pattern, the central p-gon pattern

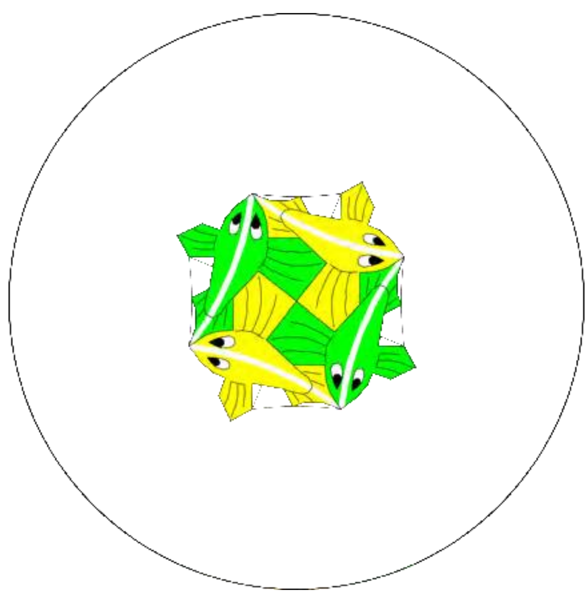

Figure 5.7 The replication of the fundamental region to create the central p-gon pattern.

The second step comprises the replication of the p-gon pattern in order to obtain the complete repeating pattern. As the number of transformations is fewer compared to that of replication of the individual motifs, this algorithm is simpler and is less prone to roundoff errors. The layers are generated recursively. The first step of this algorithm generates the first layer of the pattern and $(k+1)^{st}$ layer consists of all the p-gons sharing an edge or vertex with k^{th} layer. Figure 5.8 shows the extension of the p-gon pattern from layer 2 to layer 3. The p-gon labelled 1 in layer 2 is first rotated about vertex A to draw p-gon labelled 2 in layer 3. It is rotated about B to get p-gons labelled 3 and it is again rotated about C to get p-gons labelled 4. This is explained in detail in the next section.

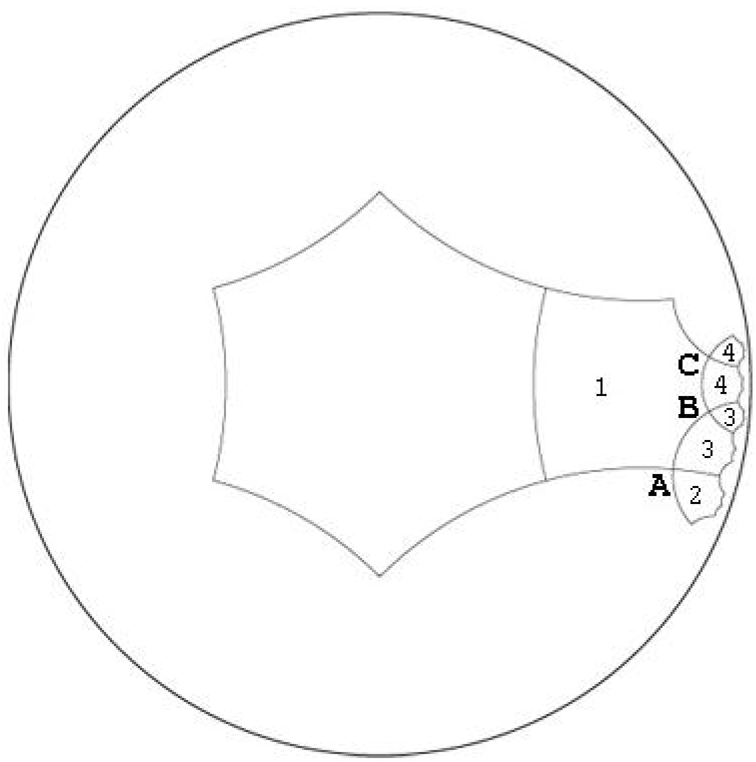

Figure 5.8: Extending the p-gon pattern from layer 2 to layer 3

A symmetry operation on all the points of the fundamental region results in the transformation of a p-gon from one-layer to the next. Each point is projected onto the hyperboloid of the Weierstrass Model by the inverse projection. The point is then transformed to a new location by taking the product of the vector representing the coordinates with the 3x3 Lorentz Matrix representing the symmetry operation. It is then projected back to the Poincaré Disk Model using the transformation provided by in Section 3.5.

5.4 Implementation of the Replication Algorithm

This section describes the recursive replication algorithm that is used by the pattern generation algorithm given by Dunham[3]. The iterations over each vertex of the p-gon in the k-layer results in the $(k+1)^{st}$ layer, which shares that vertex with the k^{th} layer. For each vertex, it calculates the number of polygons that need to be drawn in the $(k+1)^{st}$ layer from that vertex. This number is either q - 2 or q - 3, depending on the exposure of the vertex. Then, the algorithm is recursively called for the vertices of the p-gons just drawn in the $(k+1)^{st}$ layer. As shown in Figure 4.9, after p-gon labeled 1 is drawn, the replication algorithm is called for each of its “exposed” vertices A, B and C. The algorithm is then recursively called for all the exposed vertices of the new p-gons.

5.5 Implementation of Hyperbolic Circles and Horocycles

This section describes the implementation of special hyperbolic curves such as hyperbolic circles and horocycles. As mentioned earlier, a circle in the hyperbolic plane is the locus of all points at a fixed hyperbolic distance from the center. A horocycle is a Euclidean circle that is internally tangent to the bounding circle. The approach used to construct these curves is discussed below.

A hyperbolic circle is generated when a user clicks two points to define its center and radius. The program ensures that the hyperbolic circle does not cross the bounding circle by continually prompting the user for a valid entry. The implementation of hyperbolic circles has been amended in order to eliminate bugs that were found in the earlier program. This method took care of all cases pertaining to the bounding circle. The earlier program did not offer this kind of error checking which resulted in circles crossing the bounding circle. However, this error checking alone does not entirely solve the problem.

A circle is replicated and transformed to create various patterns. The earlier program does these transformations by transforming the center as well as point on the circle. When the number of layers in the pattern is increased, these transformations resulted in circles crossing the bounding circle. This is due to the precision errors in the Weierstrass Model calculations used to find the transformed center and a point on the circle, basically transforming the Euclidean center rather than the hyperbolic center. This was identified and resolved in the current program by using a different approach. When the user clicks two points, the program identifies four other points on the circle by moving along positive and negative X-Y coordinate axes from the center. After every transformation, the new center is calculated using these four points on the circle. The new center is then calculated by finding the point of intersection of the perpendicular bisectors to the chords formed by joining any two pairs (from four points). A hyperbolic circle can now be generated using the new center and any point on the circle. This approach eliminated previous errors and always resulted in circles residing within the bounding circle.

A horocycle is also a Euclidean circle that appears to touch the bounding circle in Poincaré Disk Model. It can be constructed in different ways but the current program allows the user to click only one point to define the Euclidean center of the horocycle. Only one horocycle can be drawn in the bounding circle with a given center, which is also an internal tangent to the bounding circle. The user does not have to click the second point (i.e. the point on the circle) to generate a horocycle as it is calculated automatically by normalizing the Euclidean center vector of the horocycle. Now, the obtained second point resides on the bounding circle at infinity. Once the program calculates these two points, it follows a similar approach that used to draw a hyperbolic circle. The first point is the Euclidean center of the horocycle, while the second point is the hyperbolic center, since the center of horocycle lies at infinity, i.e., on circumference of the bounding circle.

Chapter 6

Graphical User Interface

The Java program provides a graphical user interface that allows the user to perform various functionalities such as creating new patterns, modifying existing patterns, replicating the patterns, and saving a data file.

When the program is executed, the user will be presented a main interface as shown in the Figure 6.1. It contains a drawing canvas on which patterns are generated, notes section that guides the user about next steps, a colour combo box to choose a required colour combination for the patterns, a menu bar with menu options such as File, Edit, Tools etc. and other necessary options. The following section further elaborates more about each of these menu options.

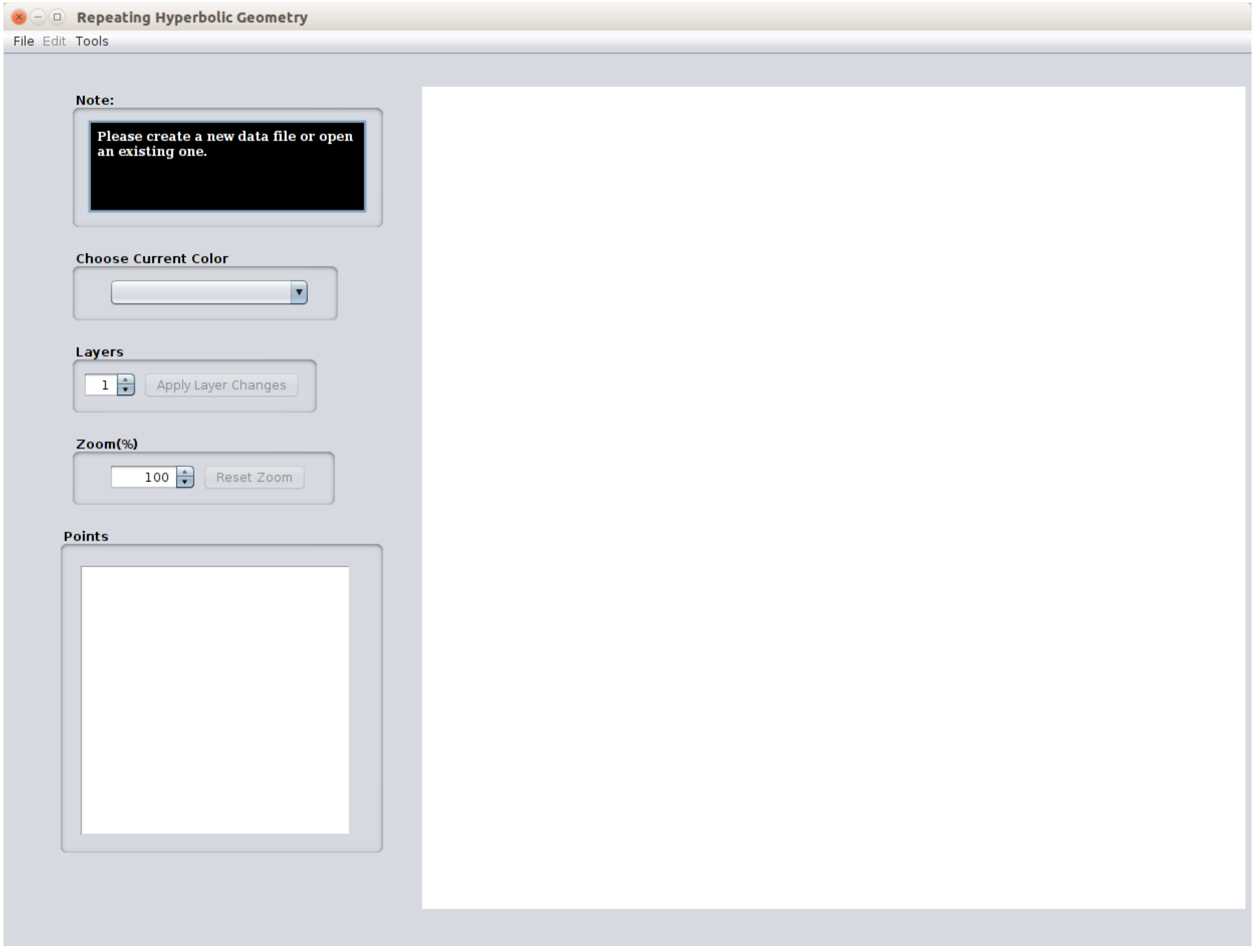

Figure 6.1: Interface displayed to the user when the program is executed.

The “File” menu has the following menu items: new, open, save and exit (as shown in the Figure 6.2). The “New” menu item is used to create a new file. It opens a dialog box which allows the user to enter the number of sides of the central polygon (p), the number of polygons meeting at each vertex (q), the number of different sides in the central p-gon that are used to create a fundamental region, the maximum number of colours, the type of reflection symmetry, and the transformation. Figure 6.3 shows the interface with a dialog box that accepts required parameters for creating a {8,3} hyperbolic pattern.

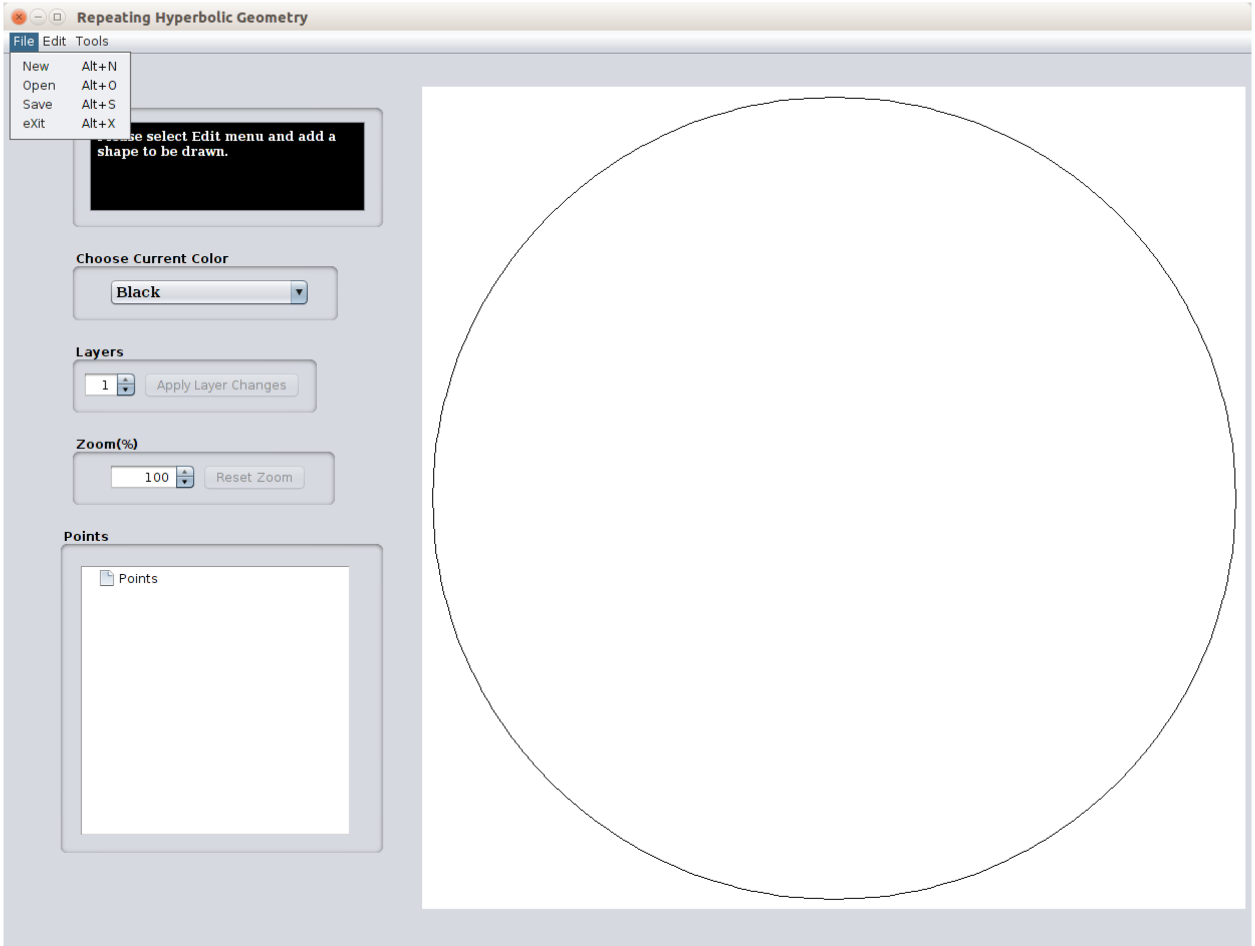

Figure 6.2: Interface showing File menu

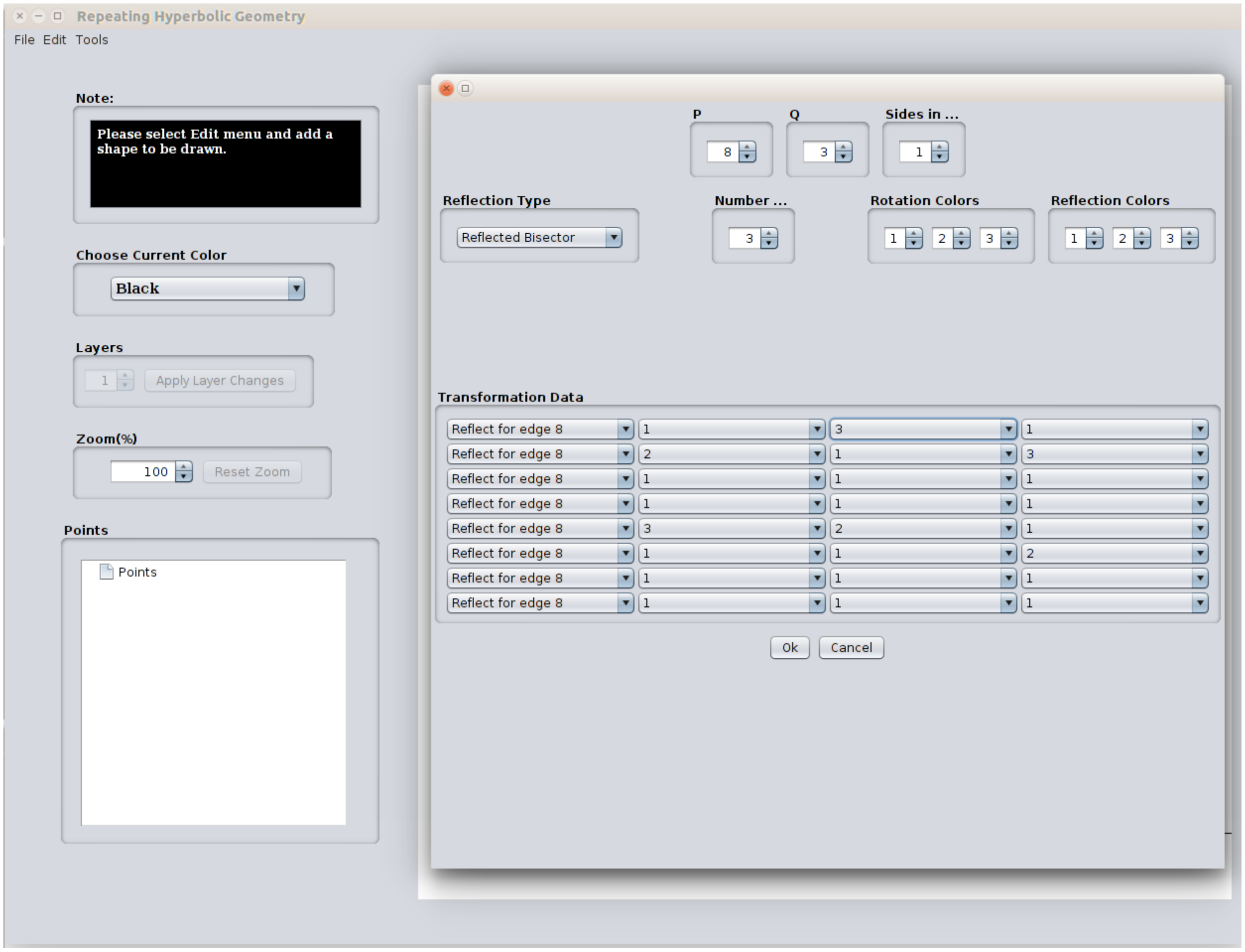

Figure 6.3: Dialog box that accepts parameters to create a new pattern

The “Open” menu item is used to open existing data files. User can then remodify the pattern represented by respective data file and save the changes by using “Save” menu item. The interface prompts the user to save the changes when the application is exited in case the changes are not saved. Figure 6.4 shows the functionality of Open menu item to open an existing data file.

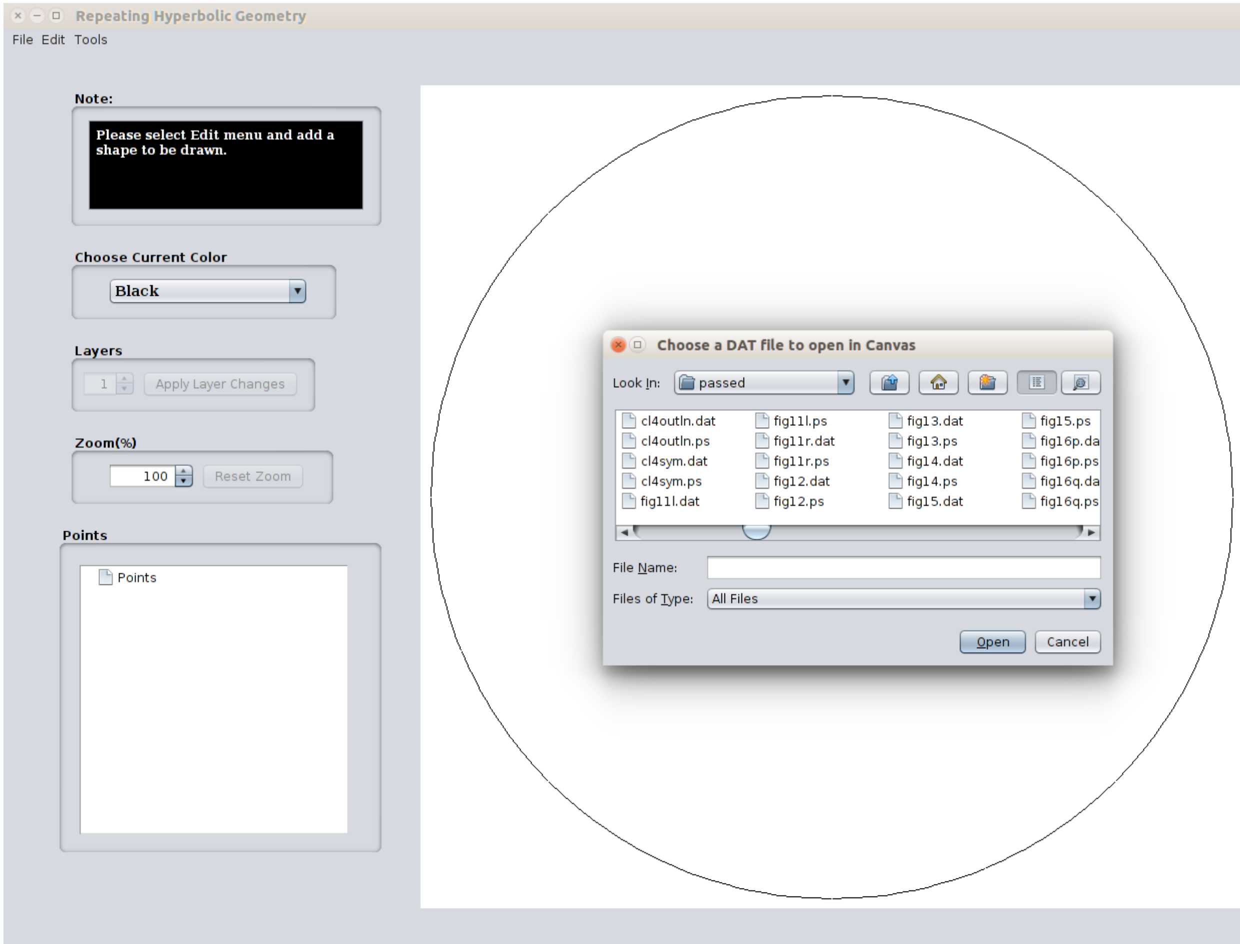

Figure 6.4 Dialog box to open an existing data file

The "Edit" menu is used to add various points to draw a variety of patterns using hyperbolic shapes such as circles, filled circles, filled polygons, filled p-gons, polylines, hyperbolic lines and line segments, equidistant curves, and horocycles (as shown in Figure 6.5). Every shape has a specified set of points that can be added interactively. Each point has an X-coordinate value and Y-coordinate value scaled according to the size of the drawing canvas. The Edit menu also provides undo and redo options for the user to undo or redo the list of actions.

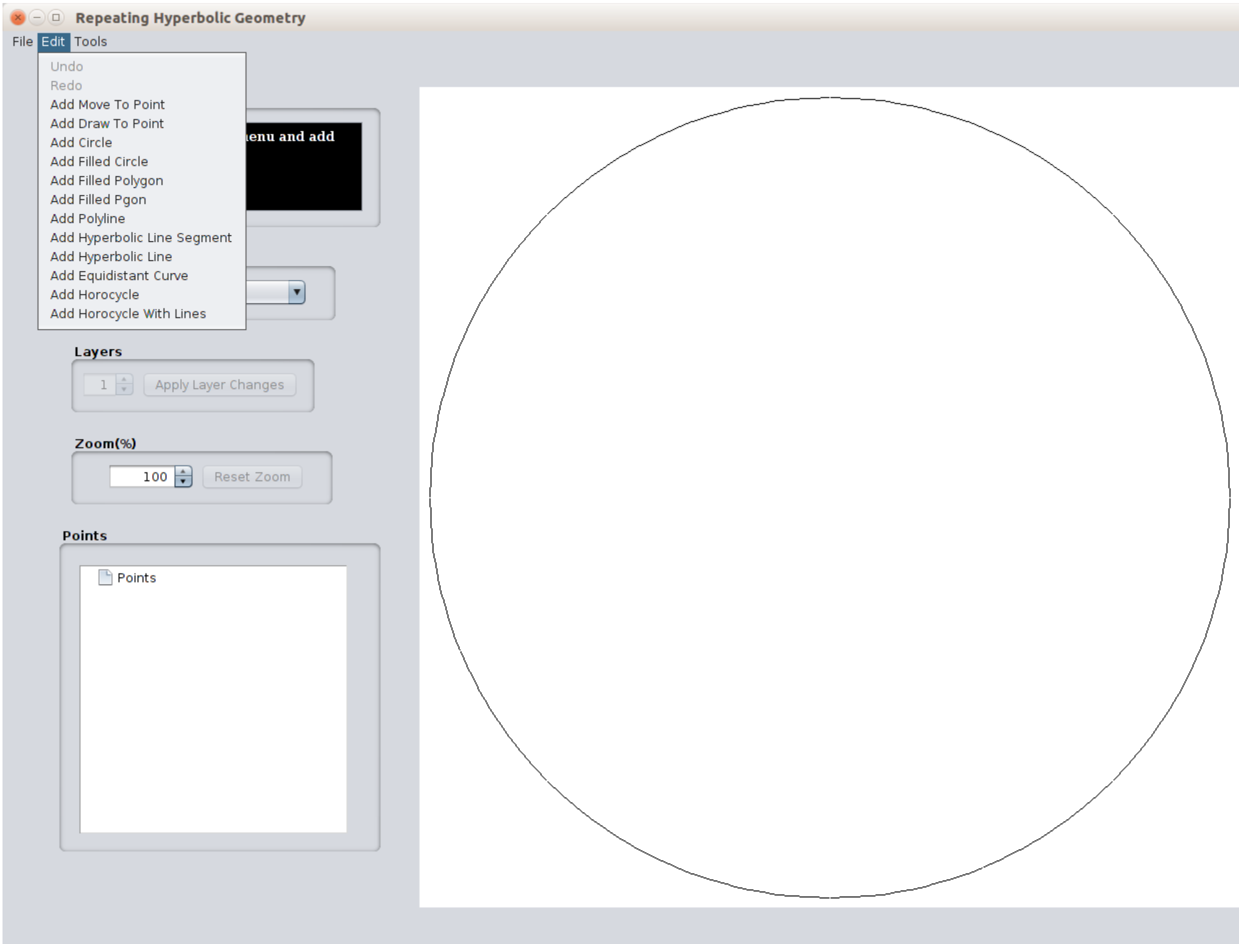

Figure 6.5: Interface showing Edit menu

The interface restricts the user from adding the points outside the bounding circle as it represents the circle at infinity (which also represents the Poincaré disk). If a point is added outside the bounding circle, the user is prompted with a warning (as shown in the Figure 6.6).

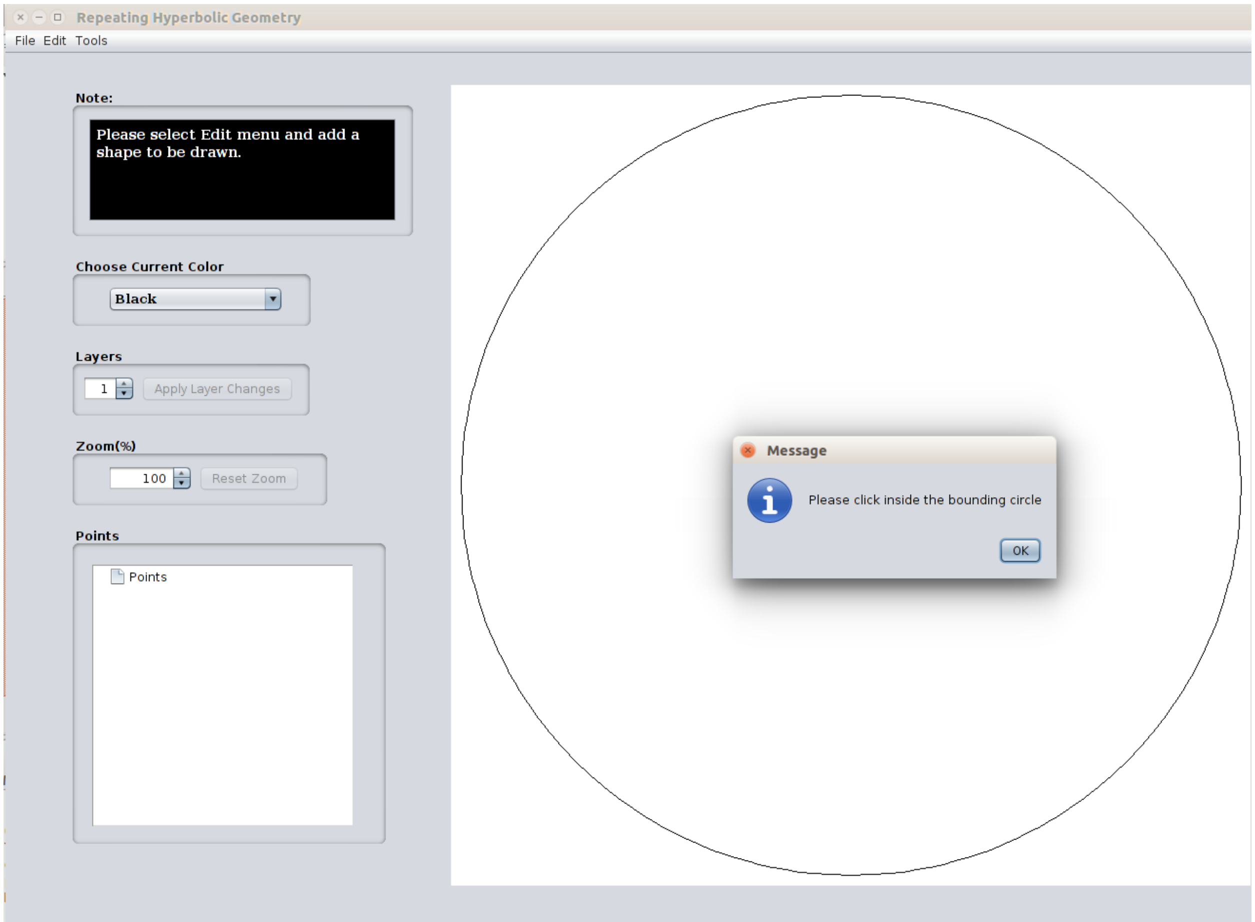

Figure 6.6: Warning that prompts the user to click inside the bounding circle

Hyperbolic special curves can be drawn in the bounding circle by selecting a respective menu item in the Edit menu and adding specific number of points needed to draw that curve. For example, two points, that is, a center and a radius, are needed to draw a circle or a filled circle. Figure 6.7 shows an interface with the "Add Circle" menu item that allows the user to add a circle in the drawing canvas. Similarly, Figure 6.8 shows an interface with the "Add Filled Circle" menu item that allows the user to add a filled circle.

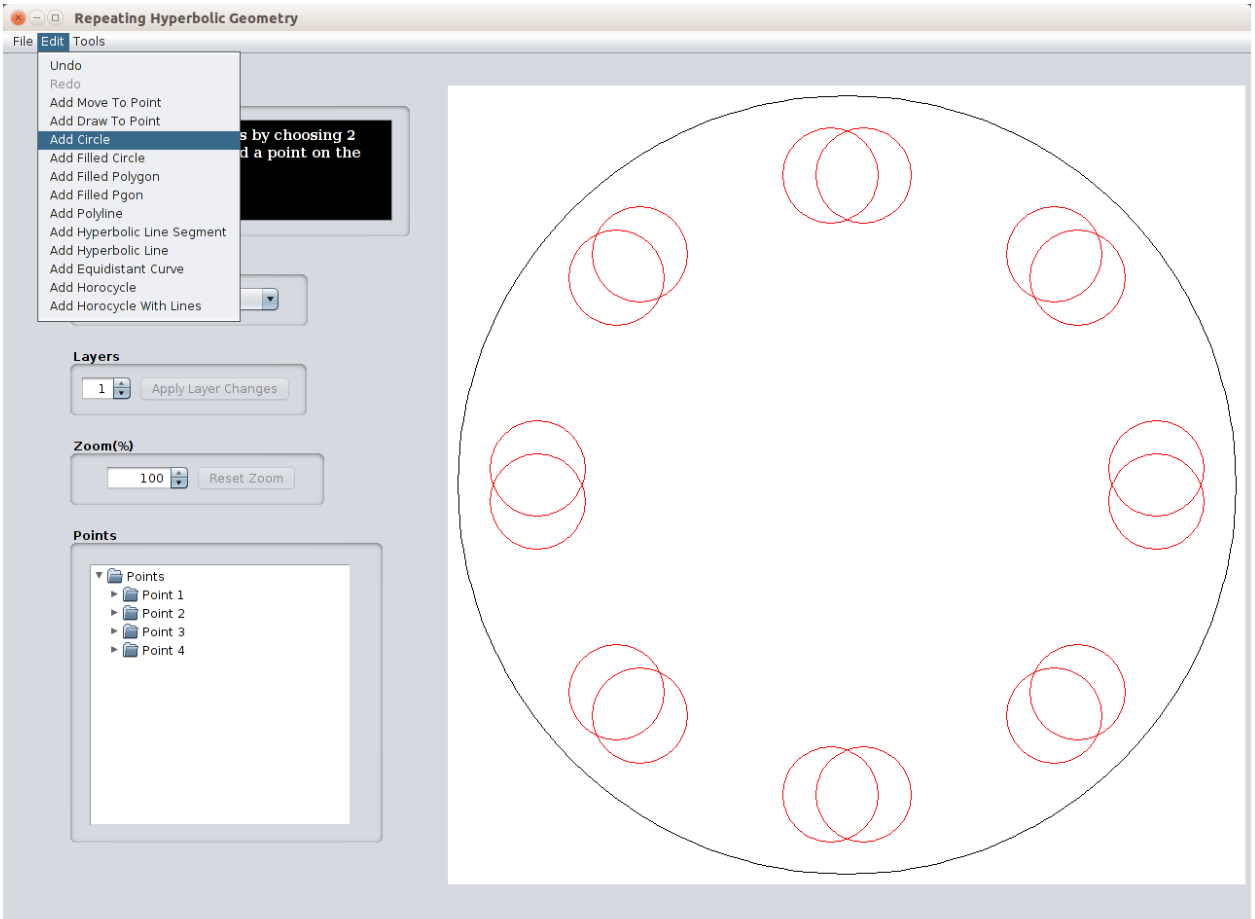

Figure 6.7: Interface with hyperbolic circles

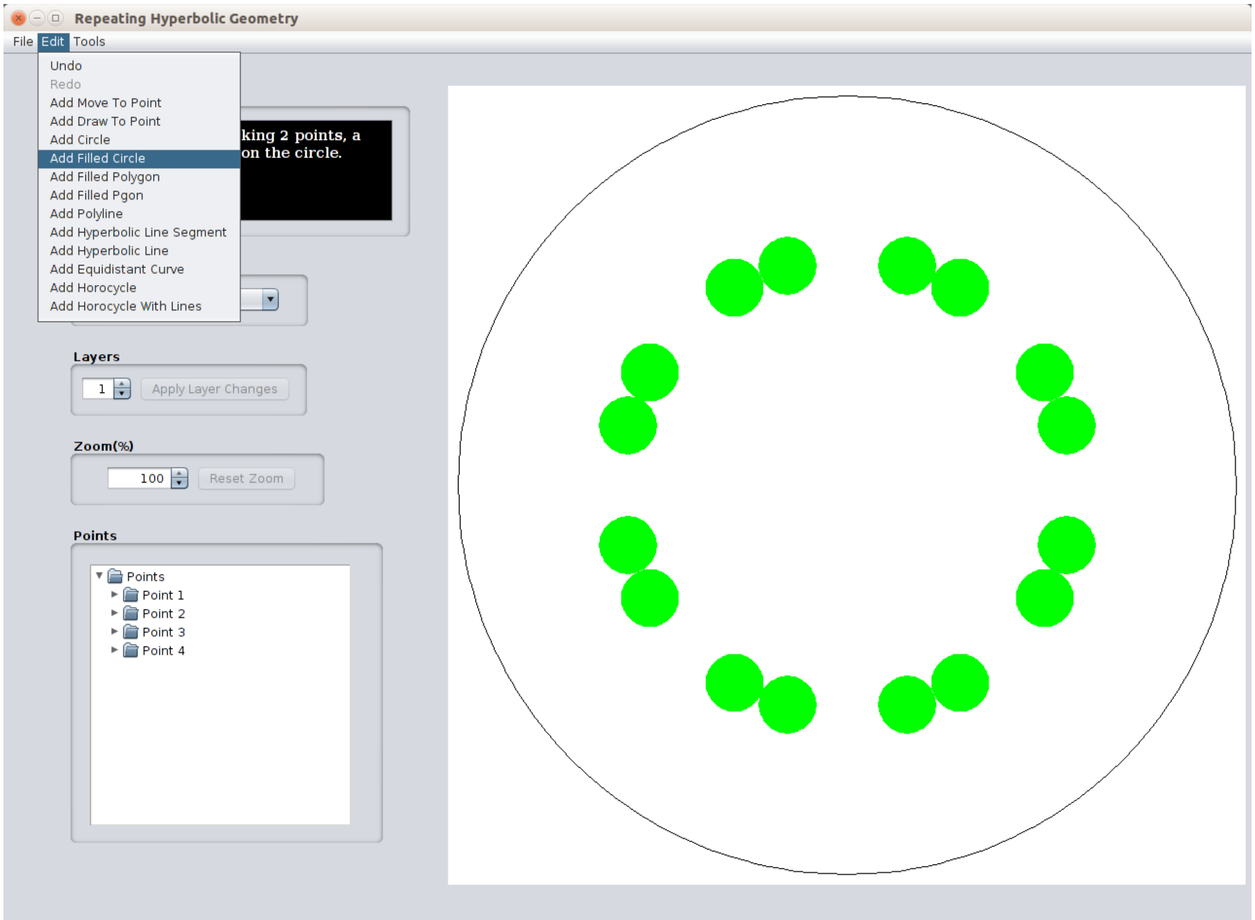

Figure 6.8: Interface with hyperbolic filled circles

The created hyperbolic circles must not overlap the bounding circle as by the definition they all must lie within the bounding circle. So, the interface restricts the user from adding the points in a way that creates any such circles (as shown in the Figure 6.9).

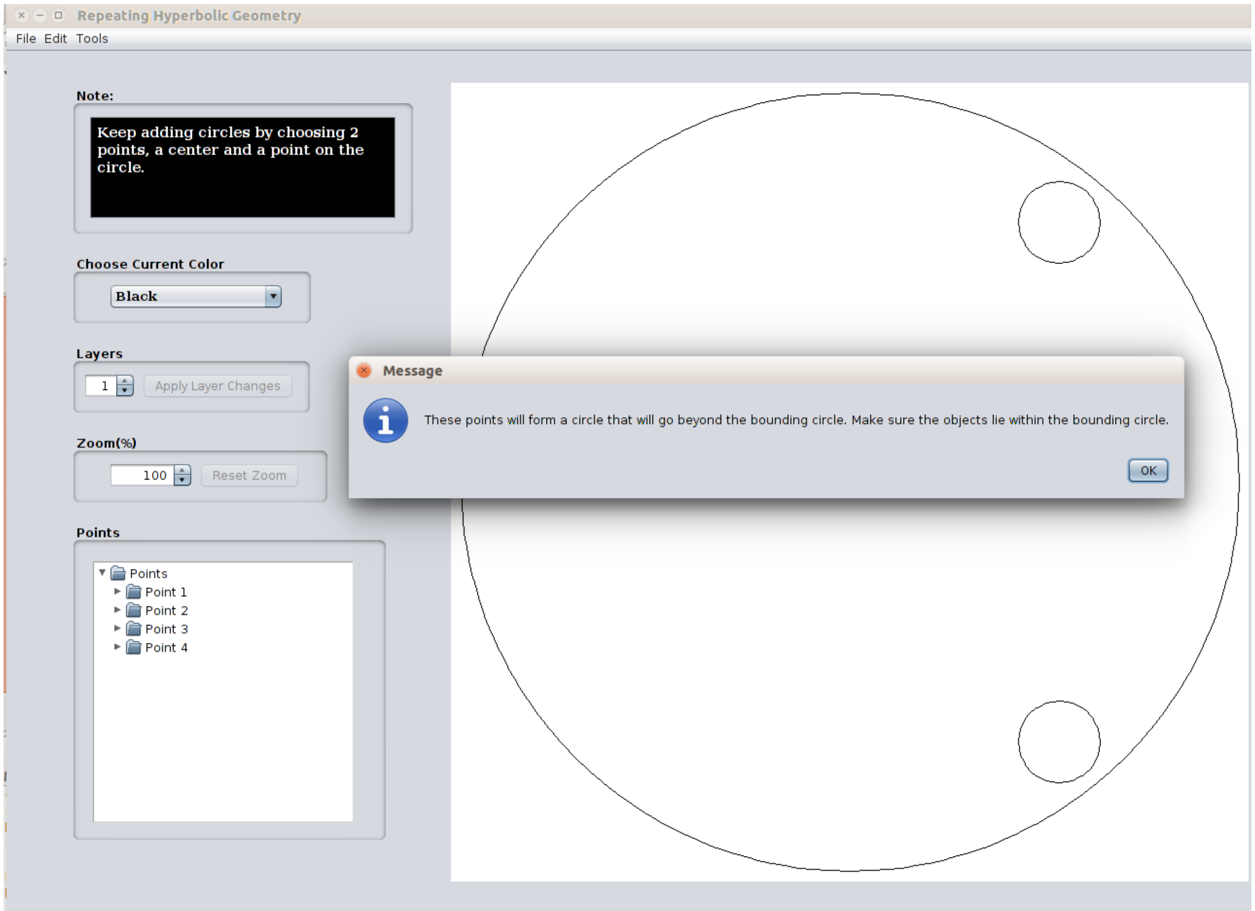

Figure 6.9: Warning that prompts the user to create circles only within the bounding circle.

A horocycle is another hyperbolic special curve whose center is at infinity. Therefore, the center lies on the edge of the bounding circle. For the user to create a horocycle, only one point, that is, a Euclidean center is needed in the bounding circle, which is then normalized onto the edge of the bounding circle. Figure 6.10 shows an interface with the "Add Horocycle" menu item that allows the user to add a horocycle in the drawing canvas. It also shows red and green concentric horocycles that have a common center.

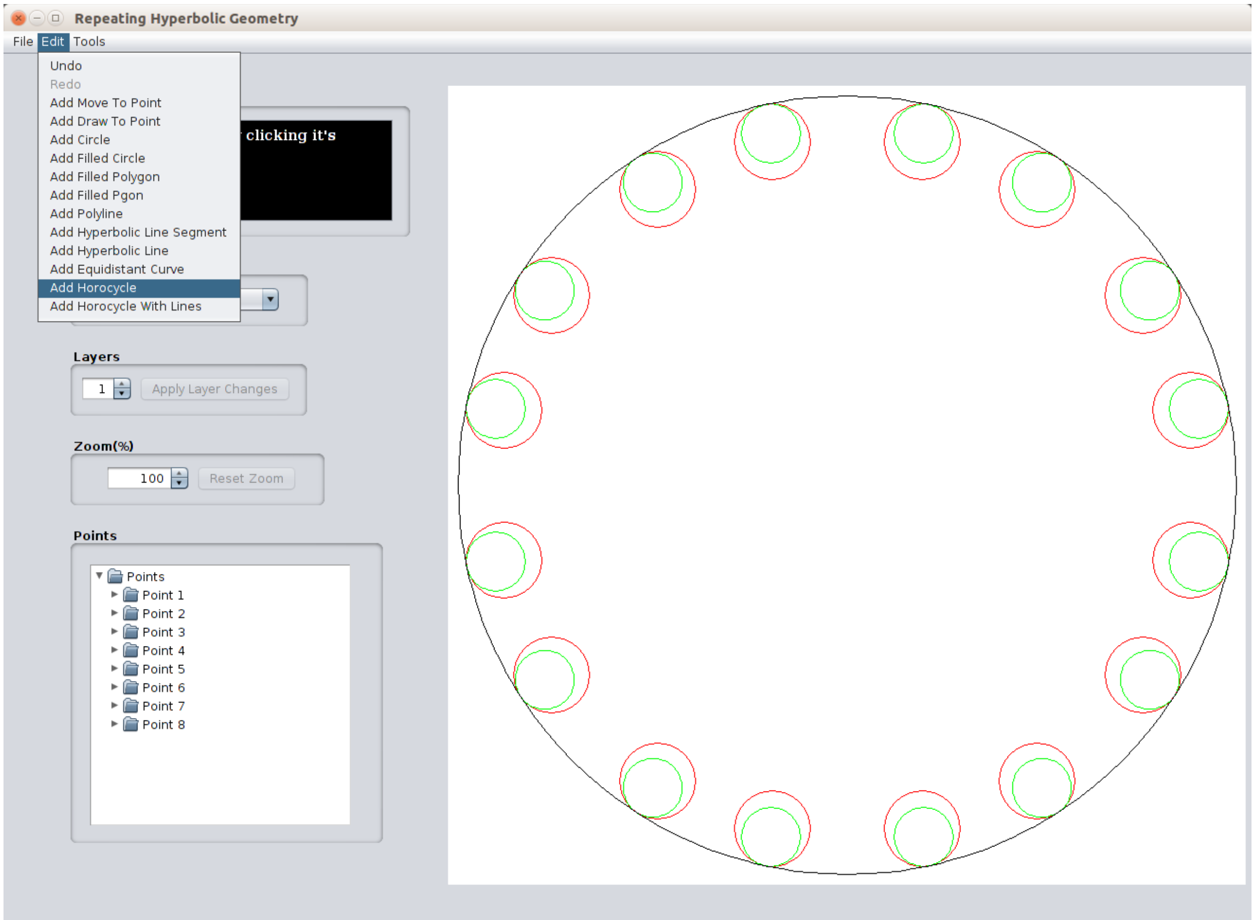

Figure 6.10: Interface with horocycles

A horocycle is a curve whose normals converge asymptotically. It can also be described as the limit of the circles that share a tangent in a given point, as their radii go towards infinity. Figure 6.11 shows the "Add Horocycle with Lines" menu item that allows the user to add a horocycle along with asymptotic convergence of its normals. Here, normals are represented by hyperbolic lines.

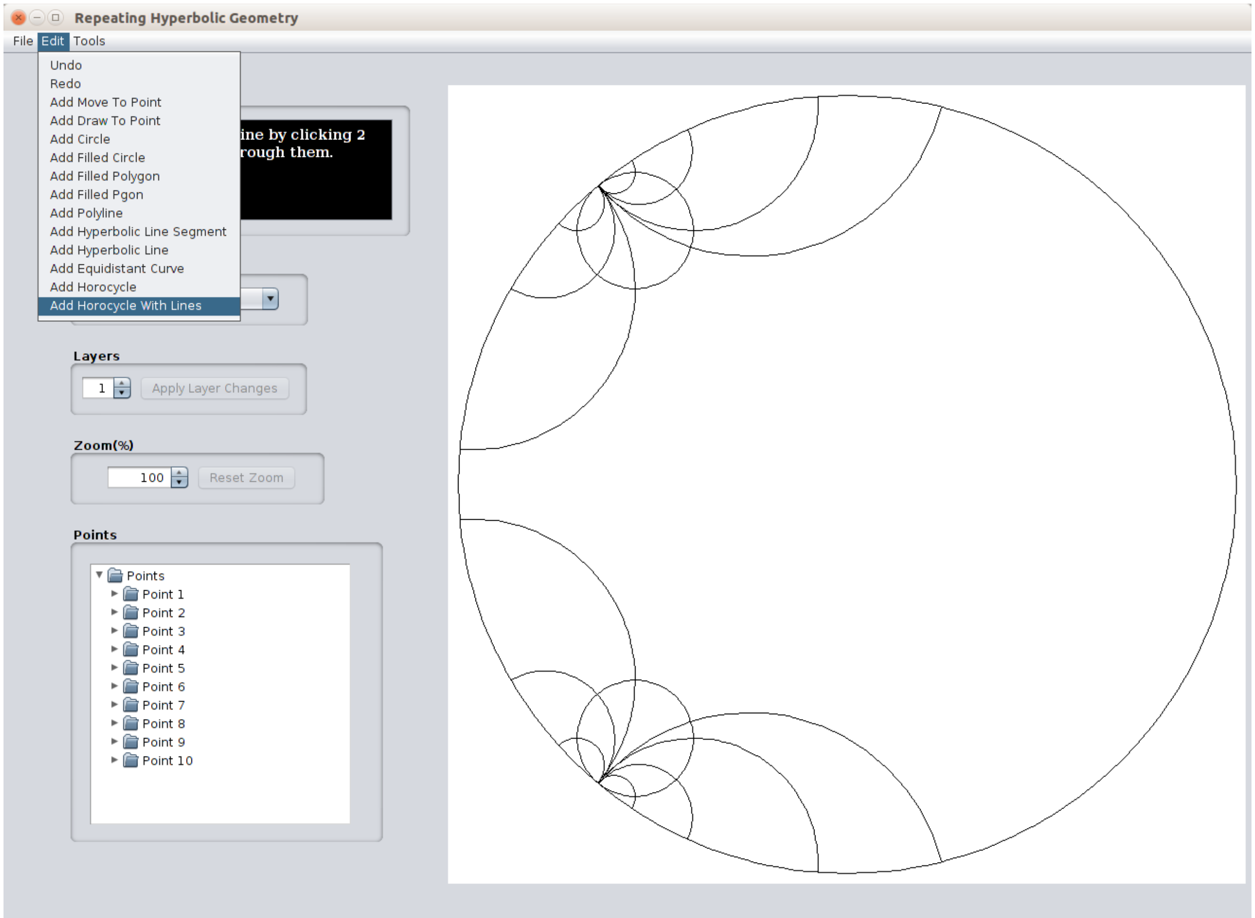

Figure 6.11: Interface with horocycles and its normals

The scroll pane at the top left part of the interface is a “Note” section that guides the user with next set of actions by providing the necessary information in the text area. And the scroll pane to the bottom left is a “Points” section that lists the properties of the points that are used to draw the hyperbolic shapes. The points are branched in the form of a tree, and on expanding those points, the properties of the points such as x-coordinate, y-coordinate, color and type of the shape are listed. The “Color Panel” section has a combo box that has a list of colors supported by a data file. Only a specified number of colors are listed depending on the maximum number of colors parameter inputted by the user in the dialog box while creating a new pattern. Figure 6.12 highlights these three sections.

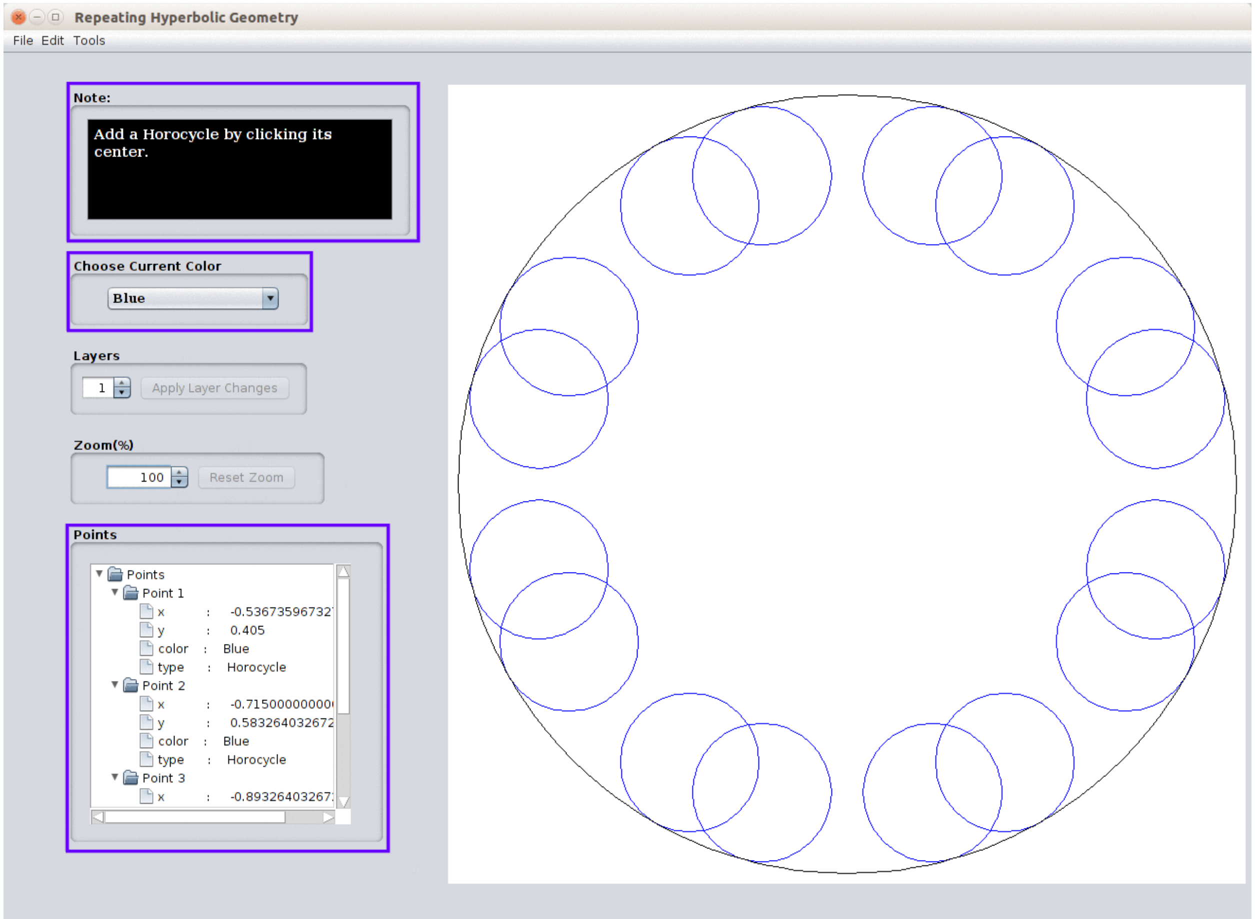

Figure 6.12: Interface showing Note, Color and Points sections

The Zoom Panel is used for displaying and setting a zoom level to have a closer look at the pattern. Figure 6.13 shows a part of Escher's Circle Limit II pattern zoomed to 150%.

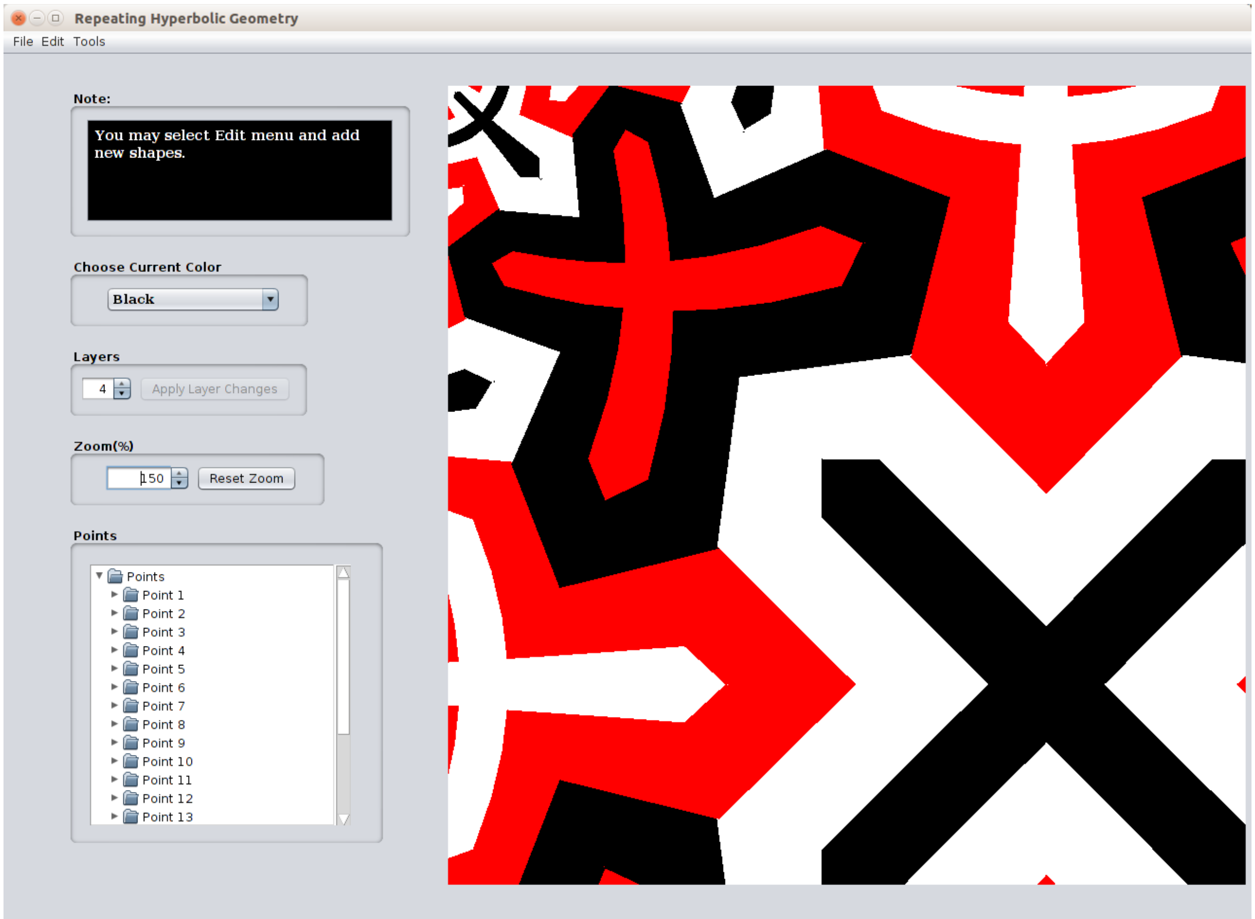

Figure 6.13: Escher's Circle Limit II zoomed to 150%

The Layers Panel indicates the number of layers central p-gon must be replicated in order to generate a repeating pattern. The first layer of the pattern is the central p-gon pattern, which is the fundamental region. As we increase the number of layers, the central p-gon is replicated and the pattern is expanded towards the edge of the bounding circle.

Figure 6.14 shows a 4-layered expanded pattern of Escher's Circle Limit II based on {8,3} tessellation.

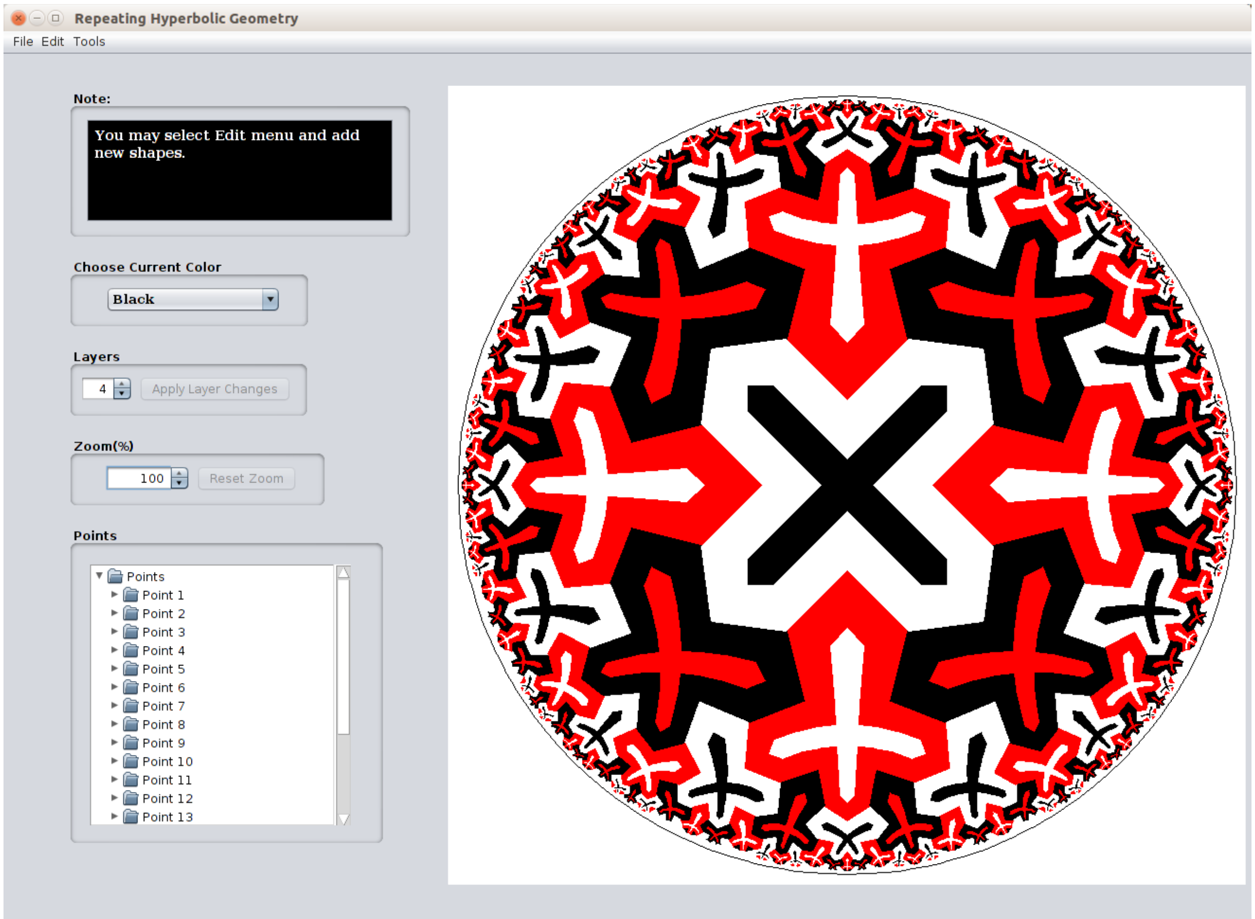

Figure 6.14: Escher's Circle Limit II Pattern with 4 layers

Chapter 7

Results

This section demonstrates the results generated by the enhanced version of Java application. This application not only generates Escher circle limit patterns and its variations but also patterns with special hyperbolic curves. As mentioned earlier, this program is based on an algorithm by Dr. Dunham[3]. It was originally written in C and used the Motif framework for the Graphical User Interface (GUI), which restricted it to run only on the Linux/Unix platform. It was later written in C++ by Becker[7] using Qt framework (available on standard platforms such as Windows, Linux/Unix, and Mac) to make the code less error-prone and easier to read. It was then converted to Java program, written by Vejendla[8], to make it portable to any platform and provide a user-friendly interface. The current enhanced version is an extension of this Java program to enable better user interaction and provide support for special curves. The backward compatibility of this application is also tested and verified by running it on previous data files. The figures listed below are some of the screenshots from this application.

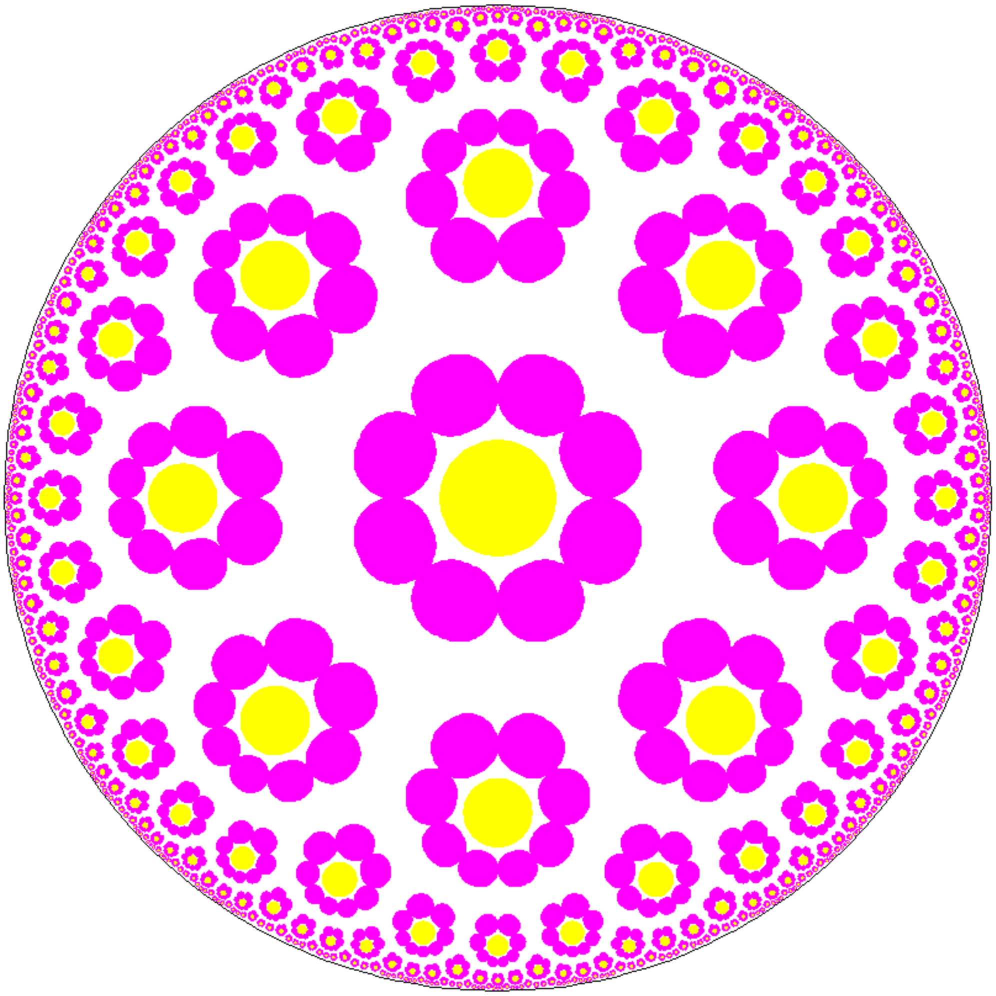

Figure 7.1: A hyperbolic lotus pattern based on {8, 3} regular tessellation with 4 layers drawn using hyperbolic filled circles.

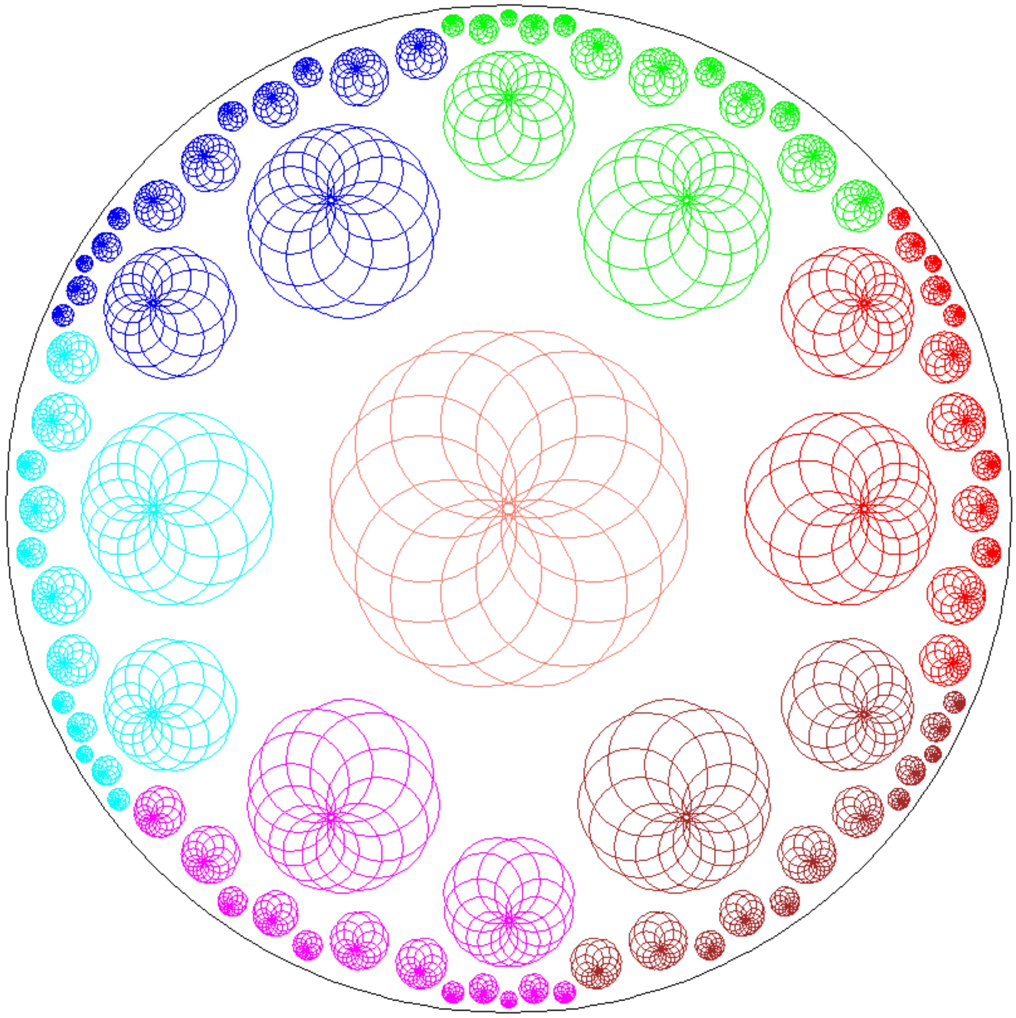

Figure 7.2: A hyperbolic torus pattern based on {6, 4} regular tessellation with 3 layers drawn using hyperbolic circles.

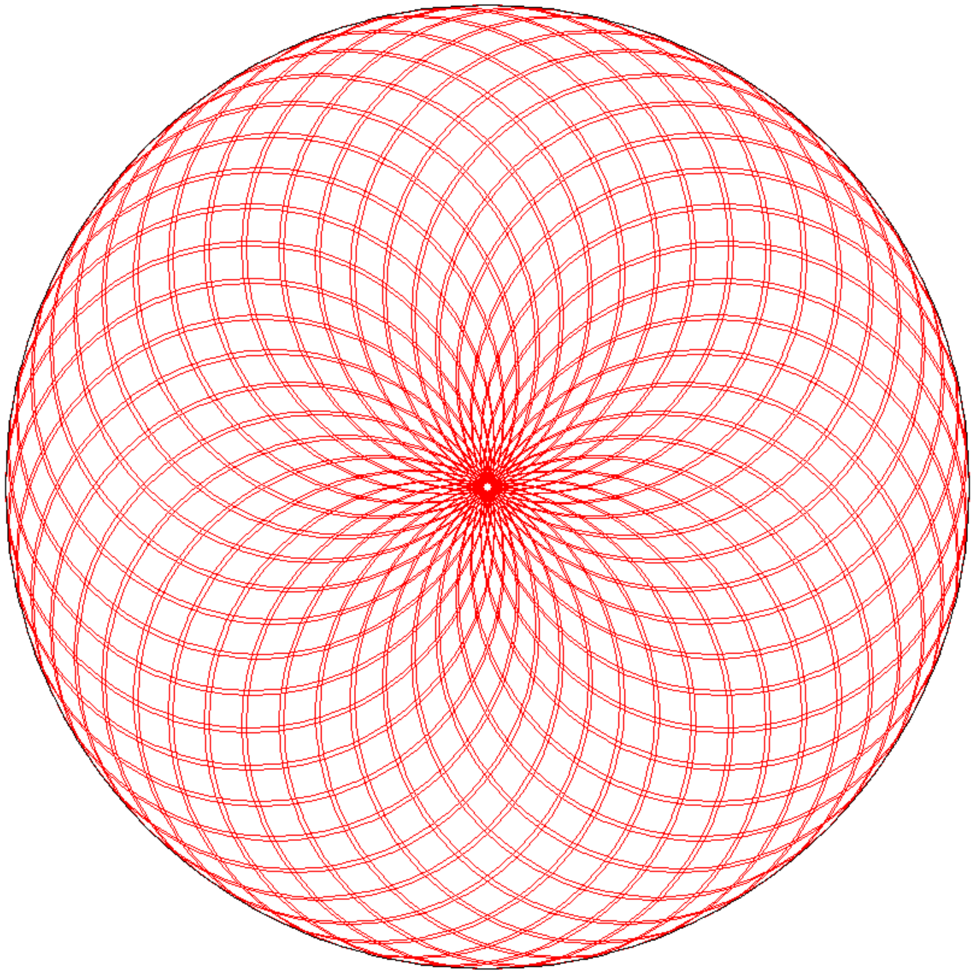

Figure 7.3: A torus based on {40, 20} tessellation with 1 layer drawn using horocycles.

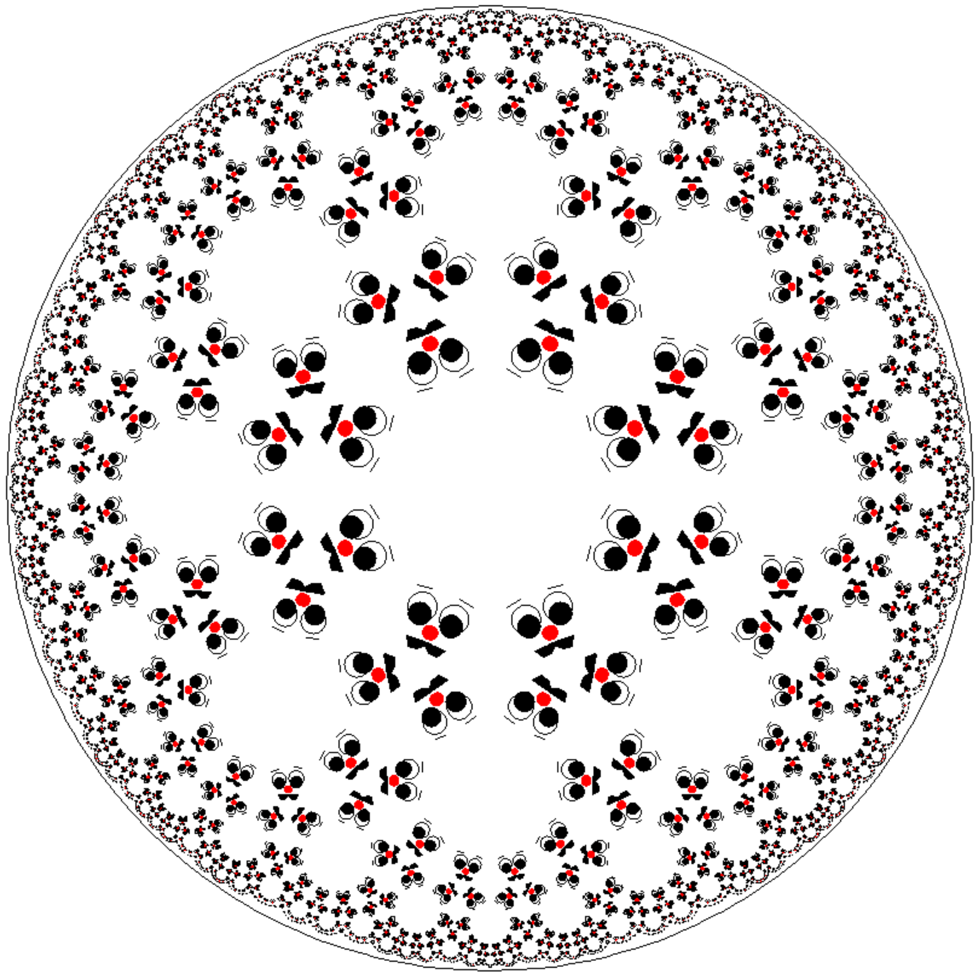

Figure 7.4: A hyperbolic pattern of jokers based on {8, 3} regular tessellation with 4 layers drawn using hyperbolic line segments, filled polygons, circles, and filled circles.

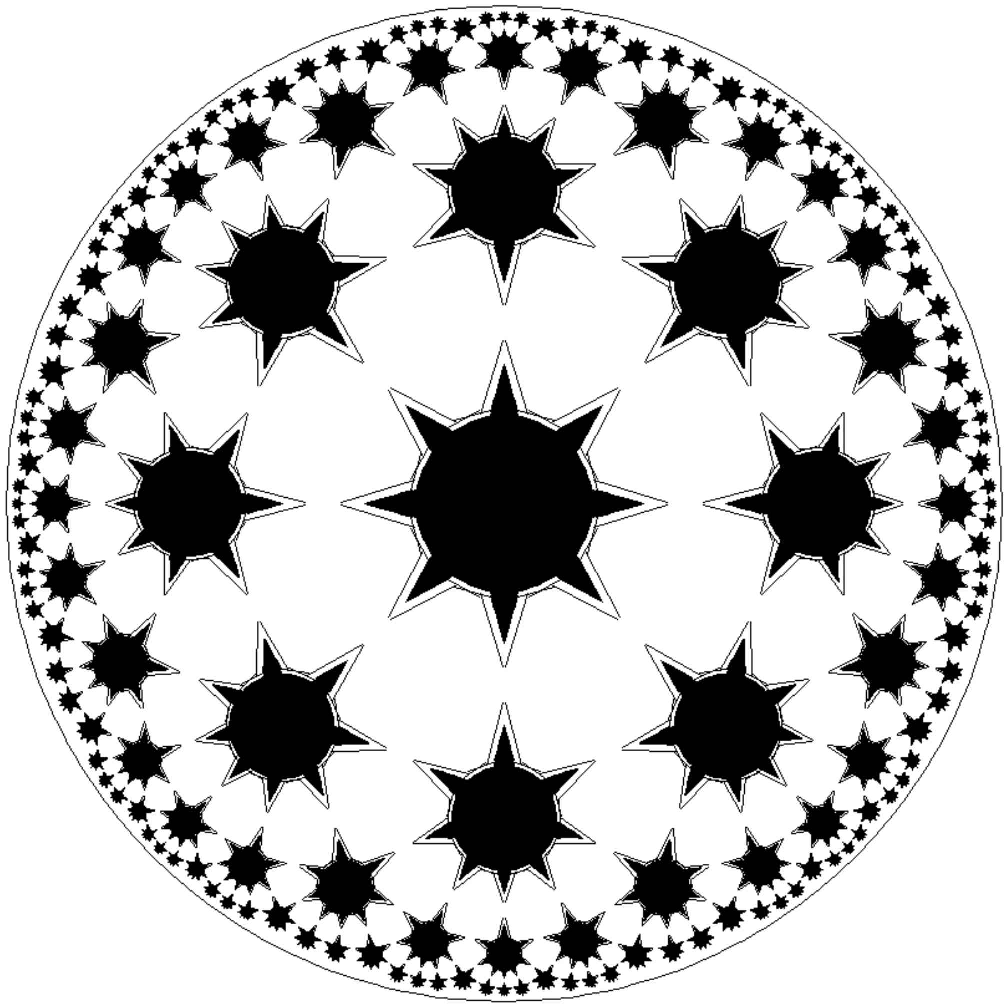

Figure 7.5: A hyperbolic pattern of mines based on {8, 3} regular tessellation with 4 layers drawn using hyperbolic polylines, filled polygon and filled circles.

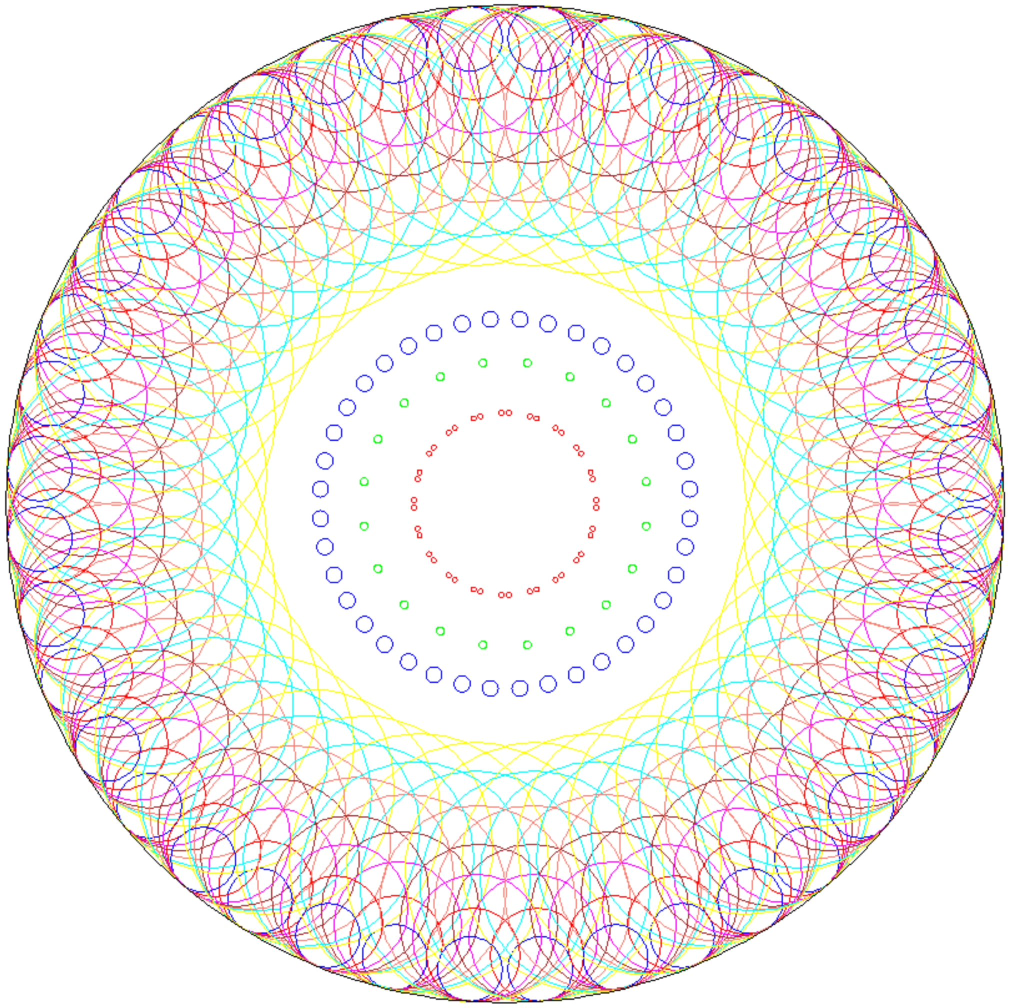

Figure 7.6: A hyperbolic pattern based on {20, 10} regular tessellation with 4 layers drawn using hyperbolic circles and horocycles.

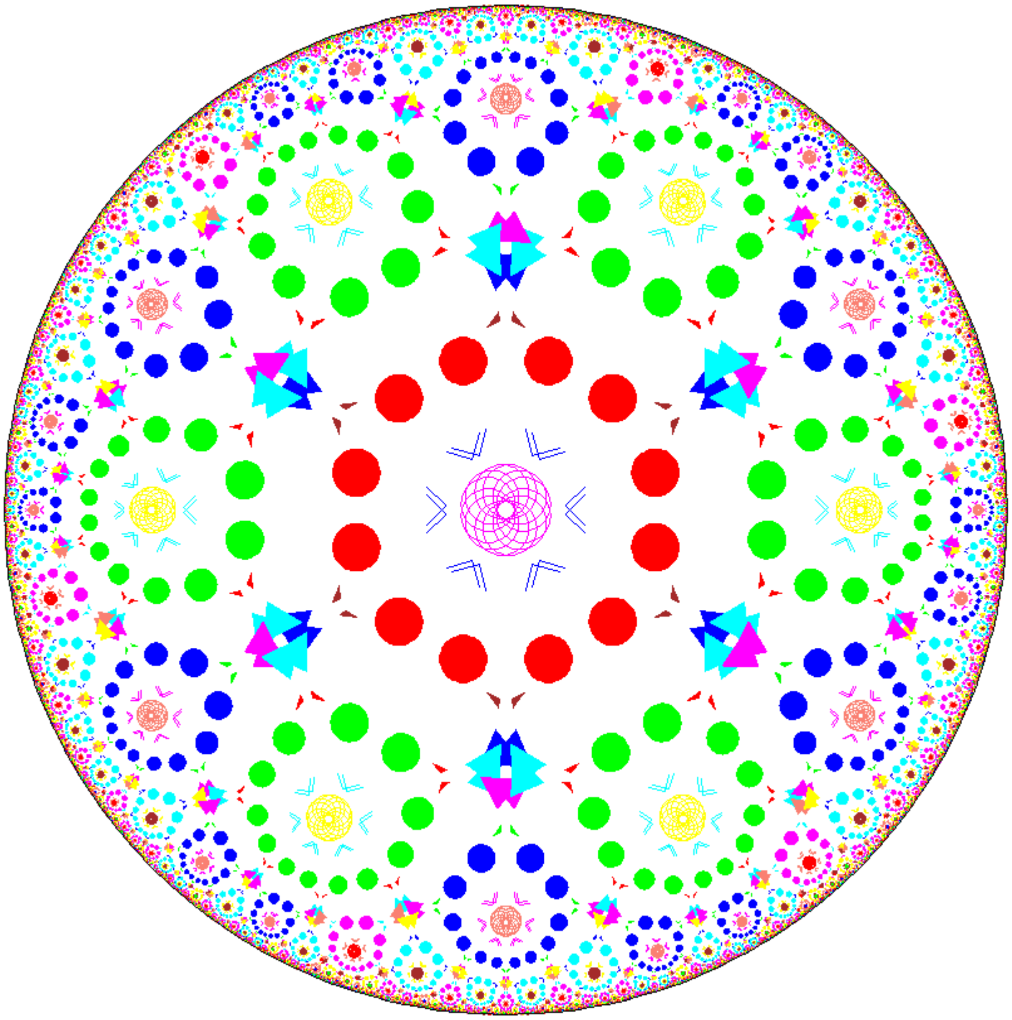

Figure 7.7: A hyperbolic pattern based on {6, 4} regular tessellation with 4 layers drawn using hyperbolic polylines, filled polygons, and filled circles.

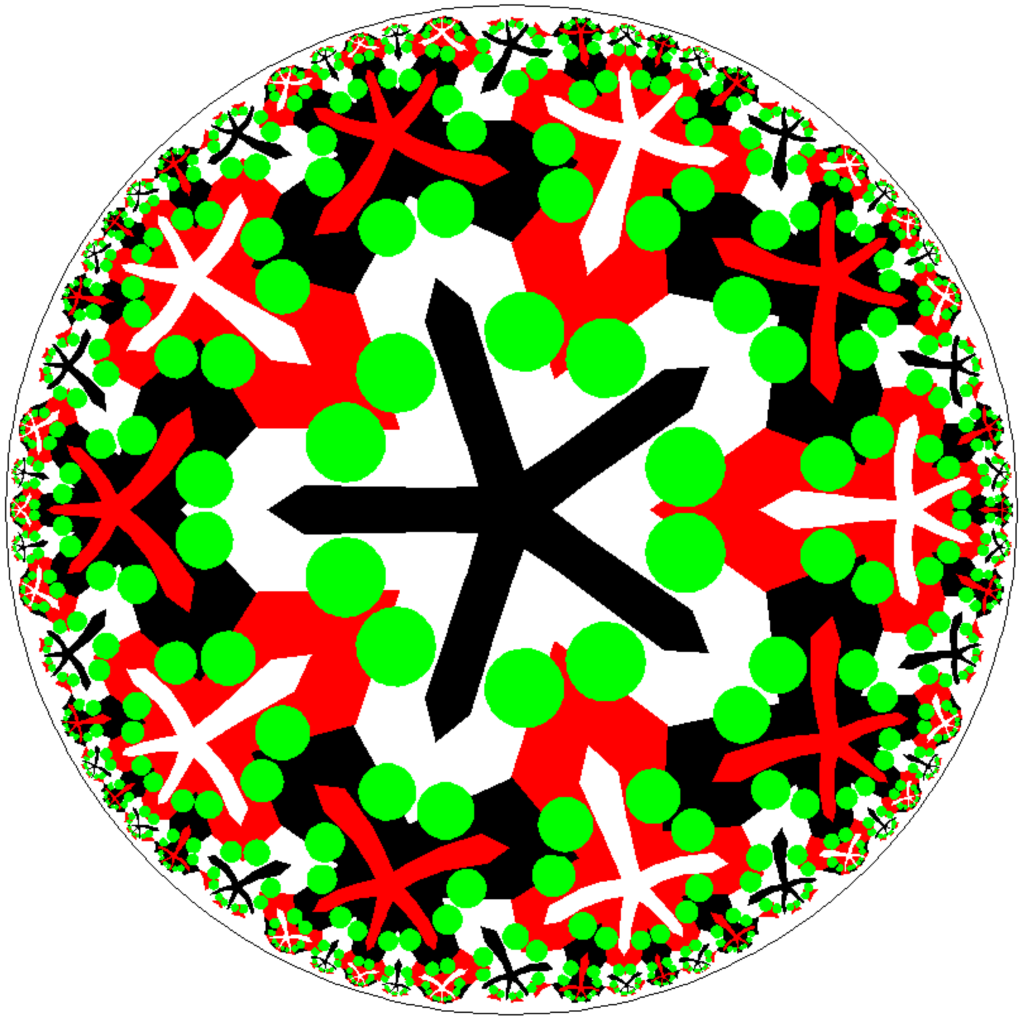

Figure 7.8: A hyperbolic pattern based on {10, 3} regular tessellation with 3 layers drawn using hyperbolic filled polygons, and filled circles.

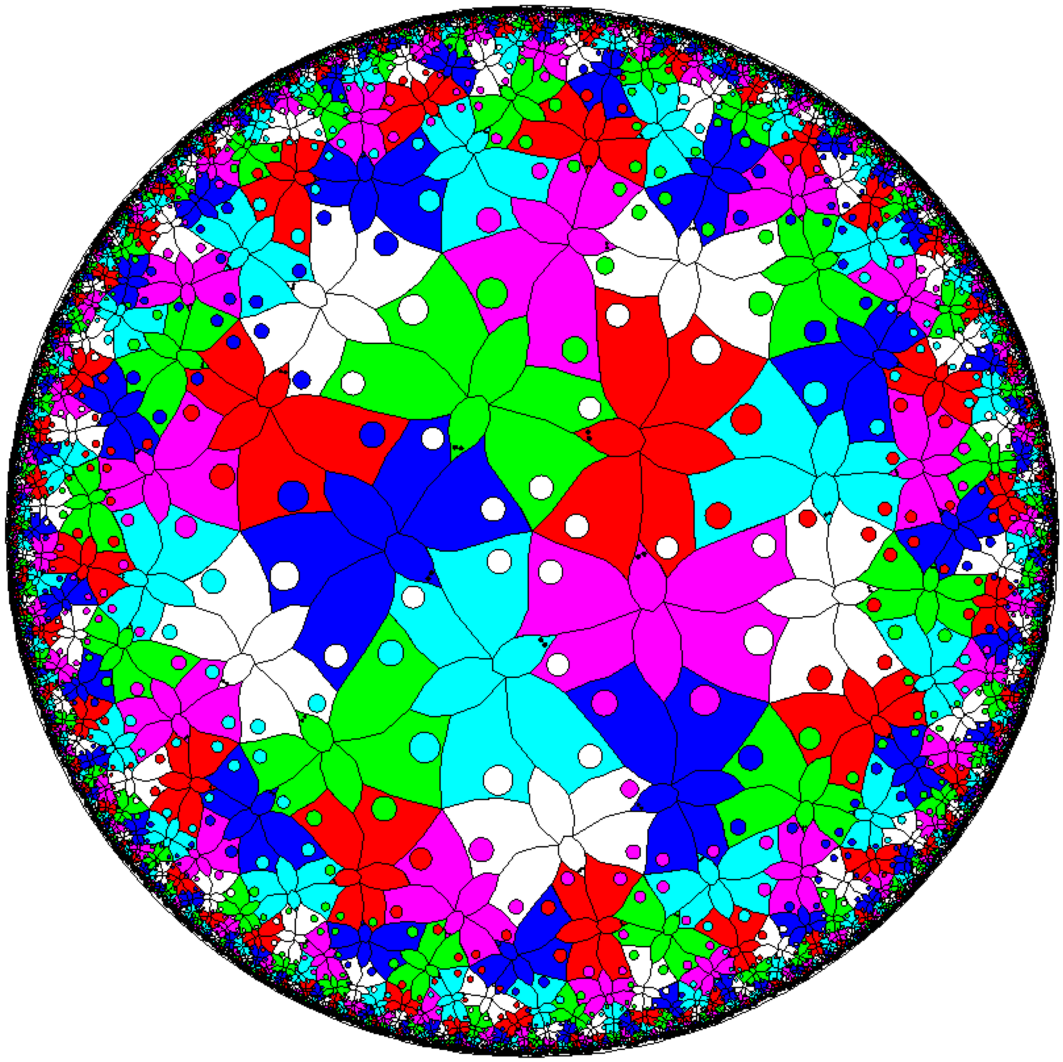

Figure 7.9: A Hyperbolic Butterfly Pattern Based on {5, 5} tessellation with 4 layers.

Chapter 8

Conclusion

This research focuses on enhancing and extending an existing Java application that generates repeating hyperbolic patterns based on {p, q} regular tessellations, where “p” denotes a p-sided regular polygon meeting “q” other p-sided regular polygons at each vertex. This extended version provides interactive capabilities to the user and also generates patterns that include special curves such as hyperbolic circles and horocycles. The Poincaré Disk Model of hyperbolic geometry was used for representing these patterns and displaying them in a finite area. The Weierstrass Model of hyperbolic geometry was used for all calculations related to transformations.

The application is portable and compatible across multiple platforms. It was tested using various data files and the results were as expected. It provides support to data files that contain information about special curves.

Chapter 9

Future Work

More enhancements can be made to this Java application that generates repeating hyperbolic patterns. The user interaction can be further improved in order to allow the user to select and drag objects to a certain position. The program could allow either p or q to be infinite, that is, to have an infinite number of edges or have all vertices on the bounding circle.

The replication algorithm could be further extended to generate hyperbolic patterns based on semi-regular and irregular tessellations. Furthermore, when layers are increased or decreased, the current algorithm repeats the pattern generation process. To avoid these repetitions, dynamic programming could be implemented thereby reducing time complexity.

Another direction to take would be to allow different models for representing hyperbolic patterns. For example, Poincaré Disk Model can be replaced with Klein Model (another finite model of hyperbolic geometry) for displaying the patterns in a finite area. Also, research could be done to display the patterns in 3-D by graphically representing the Weierstrass Model using OpenGL.

References

[1] Dunham, Douglas, John Lindgren, and David Witte. "Creating repeating hyperbolic patterns." ACM SIGGRAPH Computer Graphics. Vol. 15. No. 3. ACM, 1981.

[2] Dunham, Douglas. "Hyperbolic symmetry." Computers & Mathematics with Applications 12.1 (1986): 139-153.

[3] Dunham, Douglas. "An algorithm to generate repeating hyperbolic patterns." the Proceedings of ISAMA (2007): 111-118.

[4] Abhijit Parsekar. A Unified data representation and Visualization of Patterns based on regular tessellations in the three classical geometries. Master's Thesis, University of Minnesota Duluth 2002.

[5] D. Dunham. Artistic patterns in hyperbolic geometry. In Bridges 1999 Conference Proceedings, pages 239 - 249, 1999.

[6] D. Dunham. Creating regular repeating hyperbolic patterns. In Fifth Mathematics and Design Conference Proceedings, pages 15 - 20, 2007.

[7] Christopher D. Becker. Creating Repeating Hyperbolic Patterns Based on Regular Tessellations. Master's Thesis, University of Minnesota Duluth 2012.

[8] Vejendla, Maneesha. *Generating Repeating Hyperbolic Patterns based on Regular Tessellations using an Applet*. Diss. UNIVERSITY OF MINNESOTA, 2013.

[9] Marvin Jay Greenberg. Euclidean and Non - Euclidean Geometries Second Edition. W.H. Freeman and Company, New York, 1980.

[10] John C. Polking. http://math.rice.edu/~pcmi/sphere/sphere.html

[11] Faber Richard L. (1983). Foundations of Euclidean and Non - Euclidean Geometry. Marcel Dekker Inc, New York.

[12] Dunham, D. 1986, Hyperbolic symmetry, Computers and Mathematics with Applications, Volume 12B, pages 139 - 153.

[13] Luis Ollrtegui. Different kinds of Geometry. http://www.ehow.com/info_8774739_different-kinds-geometry.html

[14] Davis C. Royster. Hypercycles and Horocycles,
http://www.msc.uky.edu/droyster/courses/spring04/classnotes/Chapter%2010.pdf

[15] Andrew G. Bennett,
http://www.maa.org/publications/periodicals/loci/joma/hyperbolic-geometry-circles

[16] Hyperbolic geometry. (2015, July 5). In Wikipedia, The Free Encyclopedia. Retrieved 15:00, July 7, 2015, from
https://en.wikipedia.org/w/index.php?title=Hyperbolic_geometry&oldid=670052722

[17] Poincaré disk model. (2015, May 25). In Wikipedia, The Free Encyclopedia. Retrieved 15:35, July 7, 2015, from
https://en.wikipedia.org/w/index.php?title=Poincar%C3%A9_disk_model&oldid=663955784

[18] Horocycle. (2015, June 17). In Wikipedia, The Free Encyclopedia. Retrieved 15:39, July 7, 2015, from
https://en.wikipedia.org/w/index.php?title=Horocycle&oldid=667164125

Appendix

Data File Format

This section explains the format of the input data files for this program. Here is a sample data file, "*p2045.dat*" that creates a fish pattern based on a {4, 5} tessellation.

```
4 5 1 0 6 0
1 4 3 2 6 5
1 2 3 4 5 6
1 1 5 4 3 2 6
1 1 6 2 3 4 5
1 1 5 4 3 2 6
1 1 6 2 3 4 5
148
0.000000e+00 0.000000e+00 2 4 3
-2.522164e-02 -1.551347e-02 2 5 3
-5.236417e-02 -2.368675e-02 2 5 3
-8.576716e-02 -2.073424e-02 2 5 3
-1.010776e-01 -2.068176e-02 2 5 3
-8.179112e-02 -6.246868e-02 2 5 3
-5.700511e-02 -8.967406e-02 2 5 3
-2.817910e-02 -1.159125e-01 2 5 3
8.568493e-03 -1.449260e-01 2 5 3
6.064025e-02 -1.674427e-01 2 5 3
1.006000e-01 -1.836826e-01 2 5 3
1.457868e-01 -1.994115e-01 2 5 3
1.760064e-01 -2.247228e-01 2 5 3
2.004755e-01 -2.522416e-01 2 5 3
2.167416e-01 -2.739743e-01 2 5 3
```

2.511121e-01 -2.738278e-01 2 5 3

2.814111e-01 -2.814111e-01 2 5 3

2.798014e-01 -2.502713e-01 2 5 3

2.708078e-01 -2.171263e-01 2 5 3

2.880714e-01 -1.964981e-01 2 5 3

3.092706e-01 -1.678899e-01 2 5 3

3.278329e-01 -1.357145e-01 2 5 3

3.367552e-01 -9.238492e-02 2 5 3

3.483841e-01 -5.483702e-02 2 5 3

3.668022e-01 -7.574818e-03 2 5 3

3.559745e-01 6.283626e-03 2 5 3

3.425676e-01 1.454246e-02 2 5 3

2.835251e-01 9.468863e-03 2 5 3

2.592636e-01 0.000000e+00 2 5 3

2.347223e-01 -9.729257e-03 2 5 3

1.720502e-01 -1.598787e-02 2 5 3

1.570840e-01 -7.017827e-03 2 5 3

1.449260e-01 8.568492e-03 2 5 3

1.159125e-01 -2.817910e-02 2 5 3

8.967409e-02 -5.700511e-02 2 5 3

6.246868e-02 -8.179113e-02 2 5 3

2.068177e-02 -1.010775e-01 2 5 3

2.073424e-02 -8.576716e-02 2 5 3

2.368676e-02 -5.236417e-02 2 5 3

1.551347e-02 -2.522164e-02 2 5 3

0.000000e+00 0.000000e+00 2 6 3

0.000000e+00 0.000000e+00 1 1 3

-2.522164e-02 -1.551347e-02 1 2 3

-5.236417e-02 -2.368675e-02 1 2 3

-8.576716e-02 -2.073424e-02 1 2 3

-1.010776e-01 -2.068176e-02 1 2 3

-8.179112e-02 -6.246868e-02 1 2 3

-5.700511e-02 -8.967406e-02 1 2 3

-2.817910e-02 -1.159125e-01 1 2 3

8.568493e-03 -1.449260e-01 1 2 3

6.064025e-02 -1.674427e-01 1 2 3

1.006000e-01 -1.836826e-01 1 2 3

1.457868e-01 -1.994115e-01 1 2 3

1.760064e-01 -2.247228e-01 1 2 3

2.004755e-01 -2.522416e-01 1 2 3

2.167416e-01 -2.739743e-01 1 2 3

2.511121e-01 -2.738278e-01 1 2 3

2.814111e-01 -2.814111e-01 1 2 3

2.798014e-01 -2.502713e-01 1 2 3

2.708078e-01 -2.171263e-01 1 2 3

2.880714e-01 -1.964981e-01 1 2 3

3.092706e-01 -1.678899e-01 1 2 3

3.278329e-01 -1.357145e-01 1 2 3

3.367552e-01 -9.238492e-02 1 2 3

3.483841e-01 -5.483702e-02 1 2 3

3.668022e-01 -7.574818e-03 1 2 3

3.559745e-01 6.283626e-03 1 2 3

3.425676e-01 1.454246e-02 1 2 3

2.835251e-01 9.468863e-03 1 2 3

2.592636e-01 0.000000e+00 1 2 3

2.347223e-01 -9.729257e-03 1 2 3

1.720502e-01 -1.598787e-02 1 2 3

1.570840e-01 -7.017827e-03 1 2 3

1.449260e-01 8.568492e-03 1 2 3

1.159125e-01 -2.817910e-02 1 2 3

8.967409e-02 -5.700511e-02 1 2 3

6.246868e-02 -8.179113e-02 1 2 3

2.068177e-02 -1.010775e-01 1 2 3

2.073424e-02 -8.576716e-02 1 2 3

2.368676e-02 -5.236417e-02 1 2 3

1.551347e-02 -2.522164e-02 1 2 3

0.000000e+00 0.000000e+00 1 2 3

3.261841e-01 -5.793457e-02 1 8 3

3.220662e-01 -5.854071e-02 1 8 3

3.261841e-01 -5.793457e-02 1 3 3

3.187276e-01 -6.250624e-02 1 3 3

3.271498e-01 -1.076295e-01 1 9 3

3.098909e-01 -1.169108e-01 1 10 3

2.946654e-01 -1.169023e-01 1 10 3

2.761998e-01 -1.093106e-01 1 10 3

2.642670e-01 -9.981766e-02 1 10 3

2.526320e-01 -8.667453e-02 1 10 3

2.419422e-01 -7.385034e-02 1 10 3

2.372214e-01 -5.894766e-02 1 10 3

2.342312e-01 -4.147891e-02 1 10 3

2.328204e-01 -2.829623e-02 1 10 3

2.413640e-01 -1.310931e-02 1 11 3

2.419422e-01 -7.385034e-02 1 9 3

2.273731e-01 -9.567055e-02 1 10 3

2.094576e-01 -1.142995e-01 1 10 3

1.942030e-01 -1.209717e-01 1 10 3

1.812062e-01 -1.254654e-01 1 10 3

1.732525e-01 -1.165812e-01 1 10 3

1.706710e-01 -1.043848e-01 1 10 3

1.707905e-01 -9.269763e-02 1 10 3

1.734902e-01 -8.255267e-02 1 10 3

1.773035e-01 -7.447228e-02 1 10 3

2.021144e-01 -6.631199e-02 1 10 3

2.233415e-01 -6.040012e-02 1 10 3

2.372214e-01 -5.894766e-02 1 10 3

2.419422e-01 -7.385034e-02 1 11 3

3.339163e-01 -7.970977e-02 1 9 3

3.262820e-01 -8.609628e-02 1 10 3

3.148216e-01 -8.832532e-02 1 10 3

3.041866e-01 -8.577594e-02 1 10 3

2.874104e-01 -7.531657e-02 1 11 3

3.511967e-01 -1.520289e-02 1 9 3

3.374544e-01 -6.390858e-03 1 10 3

3.204852e-01 -3.157738e-03 1 10 3

2.846035e-01 -2.274535e-03 1 10 3

2.725238e-01 -1.446575e-02 1 10 3

2.624111e-01 -2.481543e-02 1 11 3

2.708078e-01 -2.171263e-01 1 1 3

2.364777e-01 -2.235421e-01 1 2 3

1.921248e-01 -2.102864e-01 1 2 3

2.167416e-01 -2.739743e-01 1 9 3

2.100366e-01 -2.396576e-01 1 10 3

1.921248e-01 -2.102864e-01 1 10 3

1.651436e-01 -1.874755e-01 1 10 3

1.348666e-01 -1.758531e-01 1 10 3

1.015717e-01 -1.682420e-01 1 10 3

7.055474e-02 -1.594849e-01 1 10 3

4.069599e-02 -1.426619e-01 1 11 3

6.313552e-02 -1.657161e-01 1 9 3

4.069599e-02 -1.426619e-01 1 10 3

2.319284e-02 -1.184682e-01 1 10 3

1.546654e-02 -1.044683e-01 1 10 3

3.559359e-03 -8.577734e-02 1 10 3

-6.901448e-03 -7.628068e-02 1 10 3

-2.075610e-02 -6.956738e-02 1 10 3

-3.060265e-02 -8.993067e-02 1 10 3

-2.817910e-02 -1.159125e-01 1 11 3

2.319284e-02 -1.184682e-01 1 9 3

4.751632e-02 -1.161689e-01 1 10 3

8.087387e-02 -1.074058e-01 1 10 3

1.115339e-01 -8.768092e-02 1 10 3

1.495975e-01 -5.721005e-02 1 10 3

2.347223e-01 -9.729257e-03 1 11 3

Figure A.1 shows the pattern generated using the above data file:

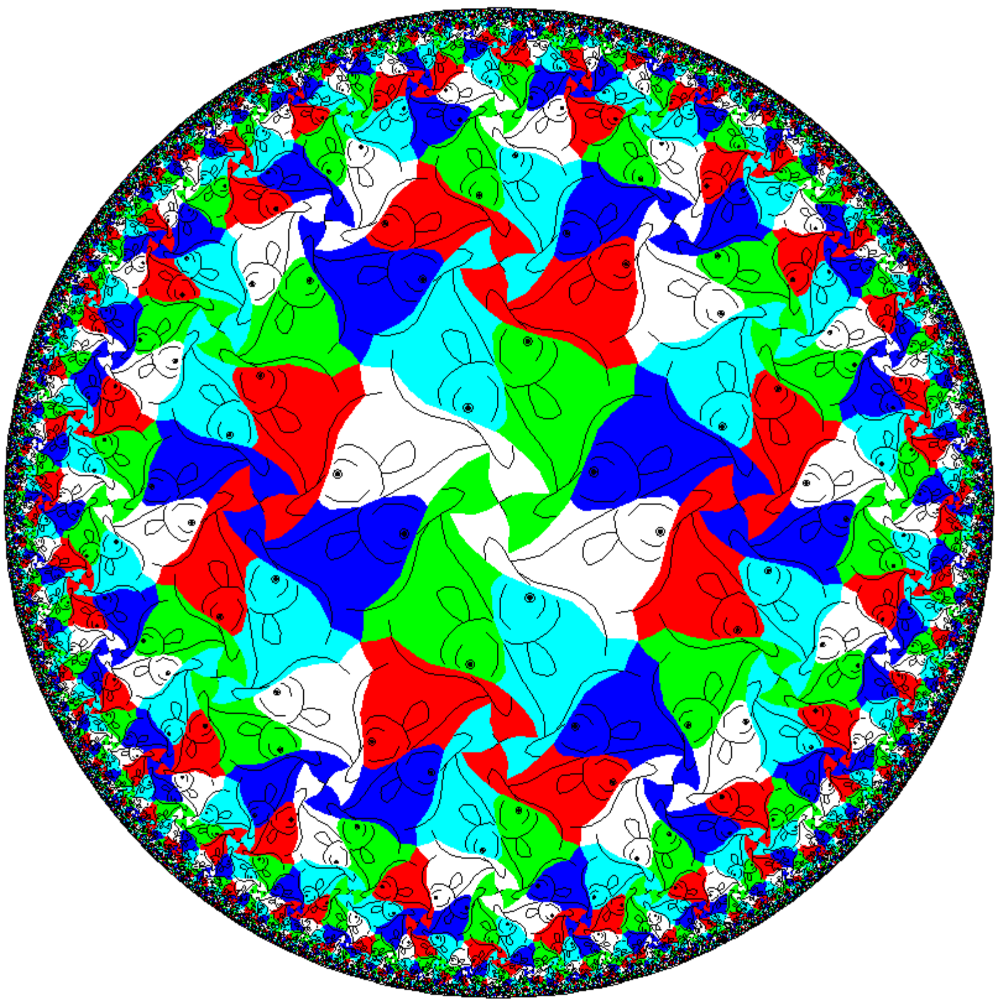

Figure A.1: A hyperbolic fish pattern based on {4, 5} tessellation

In the first line, 4 5 1 0 6 0:

• The first number is the value of p, i.e. p = 4 in this case. The central polygon in Figure A1 is an 4-gon.

• The second number is the value of q, i.e. q = 5 in this case (this pattern is based on the tessellation {4, 5}). The 6-gon in Figure A.1 meets five other 4-gons at each vertex.

• The third number, 1 in this case, is the number of "different" sides of the central p-gon that are used to form the fundamental region (the other sides of the fundamental region are two radii from the center to two vertices of the central p-gon separated by $2 * (2 * \pi/p)$). This number must divide p, and p divided by this number is the number of copies of the motif that appears in the central p-gon.

• The fourth number is not used and is there merely to maintain compatibility with older versions of the program.

• The fifth number, 6 in this case, must be the highest "color" number of the colors used. The color numbers are:

  – 1 Black
  – 2 White
  – 3 Red
  – 4 Green
  – 5 Blue
  – 6 Cyan
  – 7 Magenta
  – 8 Yellow
  – 9 Salmon
  – 10 Brown

• The sixth number, 0 in this case, indicates the kind of reflection symmetry the pattern has within the central p-sided polygon:

  – 0 indicates that there is no reflection symmetry (only rotation symmetry).
  – 1 indicates that there is reflection symmetry across the perpendicular bisector of one of the edges of the p-gon.

– 2 indicates that there is reflection symmetry across a radius (from the center to a vertex of the p-sided polygon).

The second line, 1 4 3 2 6 5, is the color permutation induced by rotating by $2 * (2 * \pi/p)$ (i.e., the third number of line 1 times $2 * (2 * \pi/p)$. Note that this is the "array" representation of permutations (not the "mathematical" one using cycles): the values listed are the values of perm[1], perm[2], etc.

The third line, 1 2 3 4 5 6, is the color permutation induced by the reflection, if the sixth number of line 1 is 1 or 2 (it is just the identity, if the sixth number is 0).

The next p lines consist of a first number followed by a color permutation. The first number of the first of these lines indicates which edge (edge 1 in this case) of the transformed p-gon should lie next to edge 1 of the central p-gon. In general, if this first number is positive, the transformed p-gon is rotated into position; if the number is negative, a reflection is used to move the transformed p-gon into position. Note that the edges are numbered from 1 to p, not from 0 to p-1, so that the edges can be assigned an unambiguous sign (i.e. 0 is not used as +0 = -0). The next six numbers, 1 5 4 3 2 6 (perm[1]=1, perm[2]=2, perm[3]=4, etc.), define the color permutation that will be induced when we go across this edge. The initial color permutation is always assumed to be the identity permutation.

The first number of the second of these lines indicates which edge (edge 1 in this case) of the transformed p-gon should lie next to edge 2 of the central p-gon. In this case, the color permutation is 1 6 2 3 4 5. This pattern continues for two more lines.

The next line consists of a single number, the number of points that make up the motif. It is 148 in this case. Following that line are 148 lines of five numbers each; each line specifies one point. Each line has the following format: x-coordinate y-coordinate color point-type number-of-layers.

• The x-coordinate and y-coordinate are within the central p-gon (and hence the unit circle).

• The color is one of the color numbers discussed previously.

• The point-type is one of:

- – 1 “Move To”
- – 2 “Draw To”
- – 3 “Circle” (there must be two of these in succession)
- – 4 Start a (Euclidean) “Filled Polygon”
- – 5 Continue a (Euclidean) “Filled Polygon”
- – 6 End a (Euclidean) “Filled Polygon”
- – 7 “Hyperline” (there must be two of these in succession)
- – 8 “Filled Circle” (there must be two of these in succession)
- – 9 Start a (Euclidean) “Polyline”
- – 10 Continue a (Euclidean) “Polyline”
- – 11 End a (Euclidean) “Polyline”
- – 12 Start a (hyperbolic) “Filled p-gon”
- – 13 Continue a (hyperbolic) “Filled p-gon”
- – 14 End a (hyperbolic) “Filled p-gon”
- – 20 “Equidistant Curve” (there must be three of these in succession)
- – 21 “Horocycle”
- – 22 “Horocycle With Lines”
- – 23 “Hyperbolic Line Segment”
- – 24 “Hyperbolic Line”

• The number-of-layers is not used and is there merely to maintain compatibility with older versions of the program.