Establishing the Connection Between a Triply Periodic Polyhedron and its Hyperbolic Covering Tessellation

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

Vamsidhar Reddy Kasireddy

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

Dr. Douglas Dunham

July 2016

Acknowledgements

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

I would like to thank my co-advisor Dr. Pete Willemsen and Dr. Marshall Hampton for their support 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. I would also like to thank Lori Lucia, Clare Ford and the entire CS department for their assistance.

I would like to thank my friends and fellow thesis members Nirav Sharda and Swetha Naidu, who contributed to this research, for their ideas and discussions.

Finally, I would like to thank my family and friends for their love and support over the years.

Dedication

Dedicated to

my Family and Friends,

Abstract

The physical world we live in is usually portrayed using the five postulates of Euclidean geometry. Endeavors to infer the fifth postulate from other four led to the discovery of hyperbolic geometry in 19th century. However some facts about hyperbolic geometry, for example the sum of angles in a traingle is less than 180 degrees in hyperbolic geometry, contradict our Euclidean knowledge. These unusual properties in turn intrigued many mathematicians to carry out research on its hyperbolic patterns as they are hard to visualize. Patterns called hyperbolic tessellations are the basis for this research. Research so far on hyperbolic tessellations has led to the development of many applications using various platforms and programming languages, but all the applications generate tessellations based on many input parameters and are difficult for a user who does not have much insight into hyperbolic geomery. This thesis concentrates on developing an application which simplifies this process and as an enhancement helps users to view the three dimensional model of the hyperbolic tessellation that is generated.

The application developed as part of this thesis provides visualization in two sub-screens. The first one contains a hyperbolic tessellation with a selected motif on it, and the second screen contains a three dimensional triply periodic polyhedral model of the hyperbolic tessellation. Users need to specify number of sides of a polygon and number of polygons that meet at each vertex in hyperbolic tessellation. Based on the input, the application generates layers of the hyperbolic tessellation in incremental order in a two-dimensional space and provides a three-dimensional triply periodic polyhedral representation. The visulization was implemented using Unity Pro game engine and C# as the developing language.

Contents

List of Figures

1 Introduction

For a long time art for humans has been an effective way to express and communicate their imaginative skills. Repeated patterns of art, which are called tessellations, have been predominantly used by many civilizations throughout the history, however the study of tessellations in mathematics has a moderately short history. These Patterns studied were Euclidean patterns, which are part of Euclidean Geometry postulated by Euclid a Greek mathematician of Alexandria.

Johannes Kepler a German mathematician of the 17th century was one of the first people to work on regular and semi regular tessellations and document his findings. Around two centuries later work of a Russian mathematician named E.S. Fedorov led to the beginning of formal mathematical study of tessellations.

Be that as it may, the most acclaimed contributor to tessellations was the Dutch craftsmen M. C. Escher. M. C. Escher was considered and greatly respected by many mathematicians for his work on so called impossible constructions. Euclidean tilings intrigued Escher when he visited a fourteenth century Moorish castle Alhambra in Spain, and he was also inspired by H.S.M. Coxeter in generating hyperbolic tessellations. Escher recognized that a two dimensional plane which depicts a hyperbolic tessellation can be used to represent infinity. The Poincaré disk model, also called the conformal disk model, is a model of two-dimensional hyperbolic geometry, which was used to represent hyperbolic tessellations. Euclidean points within a bounding circle represent the hyperbolic points in this model and hyperbolic lines are represented by circular arcs orthogonal to the bounding circle.

Hyperbolic tessellations are the motivation of this research. There are a few computer programs, which right now help to generate tessellations. But there is no computer application available, which extends tessellations into regular skew polyhedrons. Regular skew polyhedrons are a special kind of infinite skew polyhedron and they can be considered as a three-dimensional model or representation of the hyperbolic tessellation on Poincare disk model.

The following chapters will give a detailed description of an application. Which is developed to provide users an interface to not only create hyperbolic tessellation, but also provide a regular triply periodic polyhedral model of that tessellation. This application may appeal to audiences who have little or no knowledge of hyperbolic geometry and tessellations.

Below is a sample picture of the application which shows a {4,6} Tessellation and {4,6|4} Polyhedron

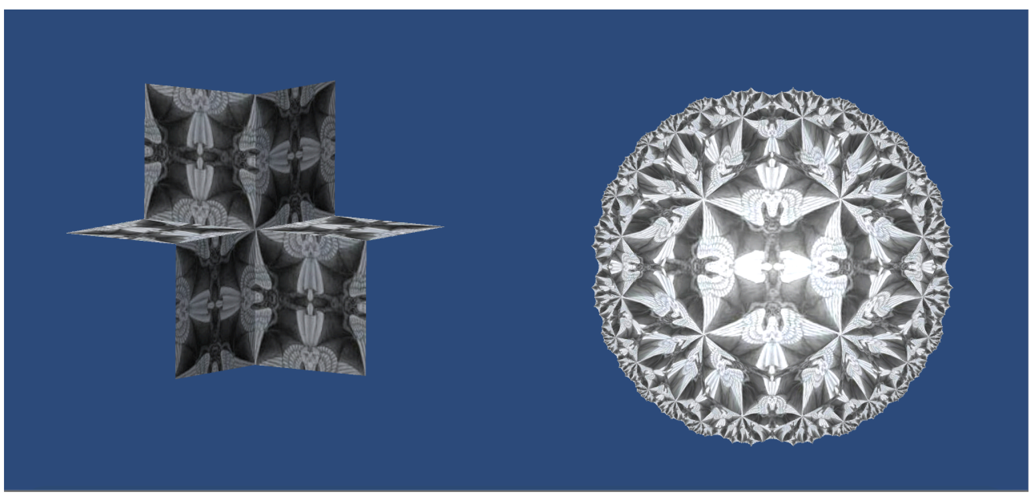


Figure 1.1: A screenshot of the Unity application.

2 Geometry

Geometry is a branch of mathematics which evolved to solve and investigate the practical problems of shapes, sizes and relationships between physical objects. Geometry fundamentally originated from many cultures throughout the history and has been widely used because of its tenets and strategies. Many Greek Scholars such as Thales, Pythagoras and Euclid have contributed an enormous number of principles to the field of geometry, which are considered to be the base for the study of geometry even today. Euclid on the other hand has organized his principles into a logical system of books, which are known as Euclid's Elements.

Based on these principals, we can divide geometry into two different types as Euclidean and Non-Euclidean. Euclidean Geometry deals with the study of points, lines and planes. On the other hand the Non-Euclidean geometry deals with the kind of geometry that is more than two-dimensional. Examples of this kind of geometry are spherical and hyperbolic geometries.

2.1 Euclidean Geometry

The Geometry taught generally in academics is from Euclid's Elements and known as Euclidean Geometry. Euclid was a Greek Mathematician in 3rd century BCE and his seminal work is named Euclid's Elements. Euclid's Elements stands out amongst the most excellent and powerful works of science in the history of humankind. The Elements formed a topic of research in many languages such as Greek, Arabic and many other languages due to its influence on all branches of science. The Elements is sub divided into 13 books of which Book 1 contains Euclid's 23 definitions - 5 postulates including parallel postulate and basic propositions of geometry: the pythagorean theorem, equality of angles and arcs, parallelism and sum of angles in a triangle [8]. Euclidean Geometry is otherwise called an Axiomatic system, which means all the principles and theorems are derived from Euclid's five postulates. The Five postulates are listed below in simple form.

- First Postulate : We can draw a straight line between any two given points.
- Second Postulate : Any terminated Straight Line segment can be extended indefinitely.
- Third Postulate : A Circle can be drawn using a straight line with any given point as center and its length as radius.
- Fourth Postulate : All Right angles are equal.
- Parallel Postulate :If two straight lines in a plane are met by another line, and if the sum of the internal angles on one side is less than two right angles, then the straight lines will meet if extended sufficiently on the side on which the sum of the angles is less than two right angles.

2.1.1 Disagreement on Fifth or Parallel Postulate

For ages Euclidean Geometry was accepted and there was no other geometry. In this process many mathematicians have thought that the parallel Postulate of Euclidean axioms was different from the other four axioms. Numerous scholars for example, Aristotle, attempted to utilize non-rigorous geometrical verification to demonstrate the parallel postulate, however they generally used the postulate itself as part of the demonstration.They thought that parallel postulate was a theorem, not merely a statement and so it was to be proved using the other four postulates. Many mathematicians until the mid 18th century tried proving the parallel postulate and started to address whether the hypothesis was even valid. This prompted the discovery of non-Euclidean geometric principles, for example hyperbolic geometry and spherical geometry.

Ptolemy was the first to attempt to prove the parallel postulate, and was then followed by Proclus, persian mathematician Nasiruddin Tusi and John Wallis. Every one of them accepted that the fifth postulate was not autonomous and they attempted to determine it from the initial four, however this process turned out to go in vain in the light of the fact that they either made some unjustified assumptions or were not able to provide sufficient logical reasoning.

The book " Euclides ab omni naevo vindicatus" which paved way to a new absolute geometry also known as neutral geometry was written by an Italian priest and mathematician Giovanni Saccheri. In this book Giovanni attempted to determine the parallel hypothesize using a mechanism called the Reductio ad absurdum method [9]. He was not successful to contradict his observation but was adamant to trust his outcomes and erroneously attempted to persuade himself that he had without a doubt reached his observation. On a similar line Lambert attempted and left with many questions unanswered.

Prominent mathematician Gauss examined the parallel postulate for quite a while. From his private journal and letters to his associates, it is currently realized that he had for sure found the presence of a geometry that was totally not the same as that of Euclid's and portrays his failure to discover a disagreement to the contradiction of Euclid's fifth postulate. For many reasons Gauss was reluctant in publishing or distributing his work and is reflected by his saying on his seal 'Pauca sed matura' (few yet ready) [9].

Nikolai Lobachevsky a Russian mathematician was one of the first persons to discover non-Euclidean geometry. His work was rejected by the Russian mathematicians. But with the help of a German publisher Lobachevsky's work was recognised by Gauss and commended his work related to the field. Gauss, Lobachevsky, and a Hungarian mathematician Janos Bolyai demonstrated situations where Fifth postulate was invalid which in turn led to the belief in existence of non-Euclidean geometries.

2.2 Non-Euclidean Geometry

During the process of working on alternative geometry, many mathematicians have made several possible theories. But without proper means of publishing their advancements and discoveries, the world didn't know of them until later years. This lead to multiple mathematicians working on same principles without any knowledge of each other at the same time. These consequences lead to the discovery of non-Euclidean geometry four times in a 20-year period.

The first one to discover the non-Euclidean geometry was Carl Friedrich Gauss of Germany. During the same period of time there were other two mathematicians Ivanovitch Lobachevisky from Russia and Janos Bolyai from Hungary who were also working on proving that there is no need of the parallel postulate. They developed a complete and consistent new geometry concept which was non-Euclidean in nature. This was called as hyperbolic geometry.

2.2.1 Hyperbolic Geometry

Hyperbolic geometry is the collection of four postulates of Euclidean geoemtry along with a new postulate called as the hyperbolic axiom. The hyperbolic axiom states that if there exists a line L and a point P not on line L, there can be at least two distinct lines parallel to L passing through point P [9].

It is as shown in the figure below.

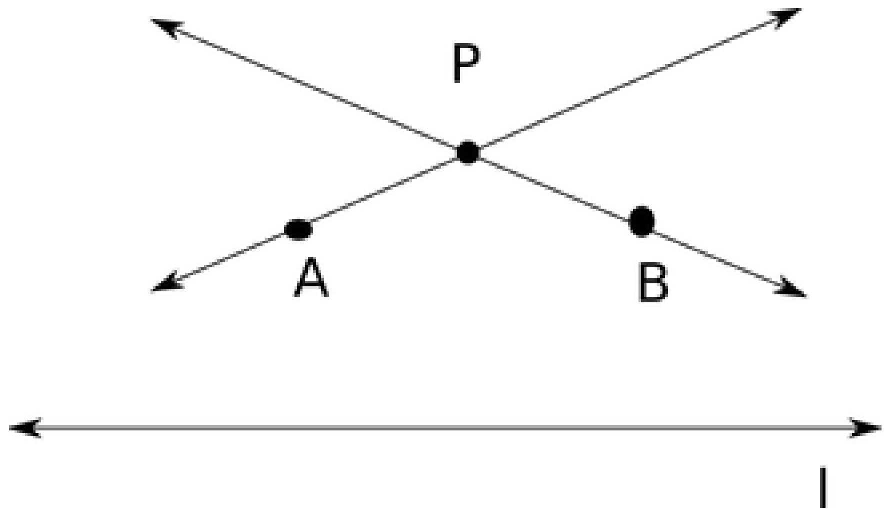


Figure 2.1: Hyperbolic Axiom.

Using this hyperbolic axiom we can prove properties of hyperbolic geometry. Some of the properties of hyperbolic geometry are:

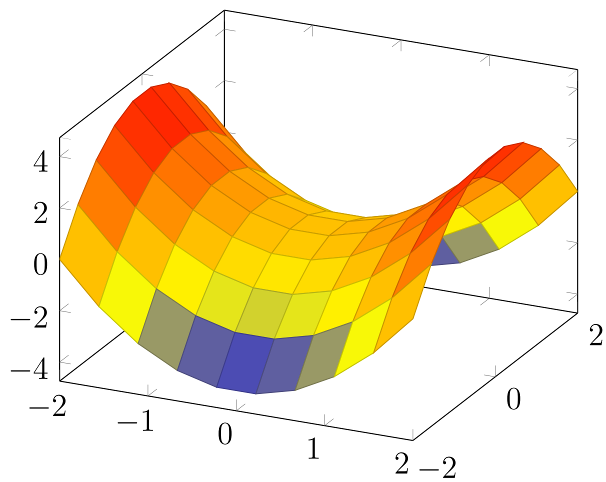
- The sum of all the angles of a triangle is less than 180 degrees.
- If two triangles have the same angles then they are congruent.
- There are no rectangles in hyperbolic geometry.
- All convex quadrilaterals have angle sum less than 360 degrees.

Due to the influence of the Euclidean geometry, the above results seem illogical. But the notions such as sum of angles in triangle will not hold in a geometry like hyperbolic geometry. We give a brief introduction below to help visualize how hyperbolic geometry appears.

Hyperbolic Visualization

Contrasting alternatives to Euclidean geometry have been created mathematically over the course of the past hundred years. They can be distinguished based on the behavior of parallel lines: In Euclidean geometry exactly one line passes through a given point which is parallel to a given line, however in hyperbolic geometry there are numerous such lines. The area of a circle grows exponentially with respect to its radius, where as in Euclidean geometry it grows quadratically.

Hyperbolic geometry, contrary to Euclidean geometry, has a negative curvature like a horse saddle. For example if you start at one end of horse saddle and move towards its opposite end, the saddle curves down and curves up again but when you move from side to side it curves up and then curves down which show its opposite curvatures. Below is a graphical representation of hyperbolic space as a Horse saddle.



Physical Models of Hyperbolic space

Cornell University Mathematician Daina Taimina made models of hyerbolic space that allow us to explore the unique properties of hyperbolic geometry [10]. Dr Taimina was inspired by the paper models designed by geometer William Thurston. As Thurston's model were difficult to make and are fragile, Taimina thought that pith of this development could be executed with knitting or crochet simply by increasing the number of stitches in each row. Below models are to help view the intrinsic properties of hyperbolic space.

Figure 2.2: A Crochet Model of Hyperbolic Plane.

Figure 2.3: A Crochet Model with Geodesics.

Figure 2.4: A Crochet Model of an Ideal Triangle.

2.2.2 Spherical Geometry

Another form of non-Euclidean geometry is spherical geometry. Bernard Reimann and Ludwig Schläfli were the two mathematicians who laid foundation for this kind of geometry. It is defined as the study of figures on the surface of a sphere. The concept of spherical geometry revolves around a sphere, we know that in the actual world lines always intersect other lines. So in case of our spherical geometry, think of a perfect sphere and lines that follow the surface will continue around the sphere and reach the starting point to create a circle. Circles such as this with the maximum length for a given sphere are called great circles. If we consider earth to be a sphere, the equator will be an example of a great circle.

Thus we could determine the properties of spherical geometry as

1. The shortest distance between any two points on this sphere would be a part of one of a greater circle passing through those two points.

2. Any two greater circles on a sphere will intersect in two places hence there can be no parallel lines in spherical geometry.

3. When three curved arcs intersect they would form obtuse summit angles and the sum of these angles would be greater than 180 degrees.

Likewise spherical geometry also contradicts the second postulate of Euclid's elements as lines on the sphere are circles, which have no starting and end points, and they cannot be extended to infinity.

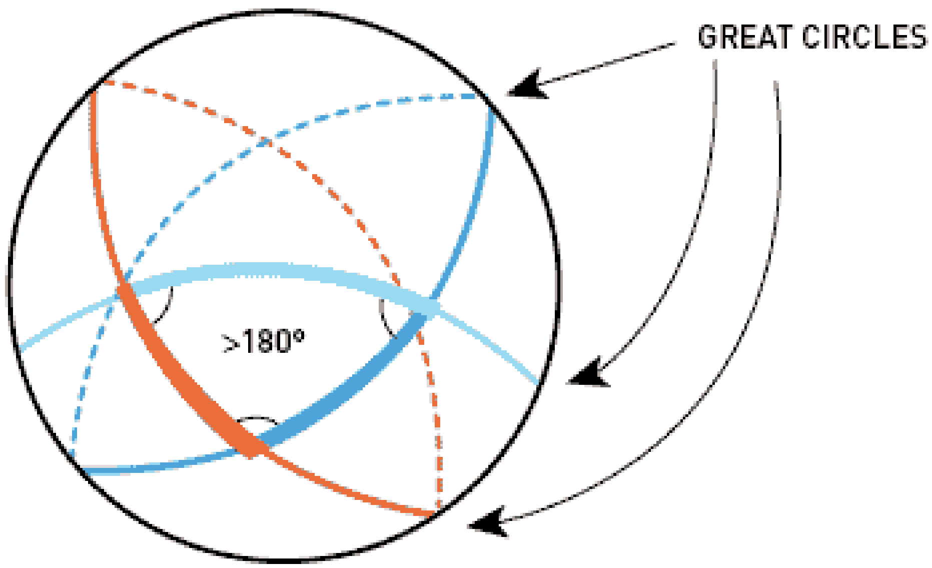


Figure 2.5: Spherical Geometry.

3 Hyperbolic Geometry Models

In diverse areas of mathematics, hyperbolic geometry and its insights have wide applications. The following sections introduce some of the models of hyperbolic geometry, which have been widely used over the course of the time to represent tessellations. The following sections give details about the Klein model, Poincare model, Weierstrass model and the relations between them. [1]

3.1 Beltrami-Klein Model

German mathematician, Felix Klein, proposed Klein model. In the Klein model the hyperbolic plane is a unit disk or can be considered as interior of a Euclidean unit circle.

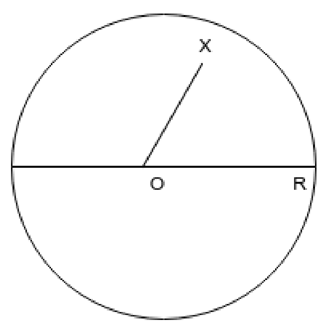


Figure 3.1: Finite Disc of Klein Model

Below is the definition of terms of the Klein Model.

- A point is same as an Euclidean point but it is within the unit circle or in this model the points in interior represent the points on the hyperbolic Plane.

- A line of a hyperbolic plane is an open chord in Klein Model and by definition an open chord is any chord in the circle excluding its endpoints.

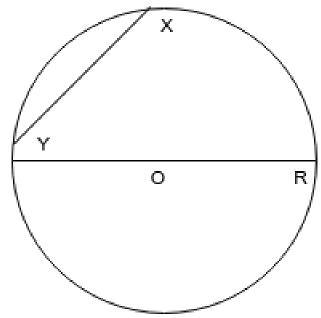


Figure 3.2: A Line in Klein Model

- Lies On retains the Euclidean Property

- Between retains the Euclidean Property

3.1.1 Parallel Lines

There are two kinds of parallel lines in this geometry [11]

- Asymptotically Parallel Lines Two lines do not intersect within the Klein model but they do intersect on its boundary.

- Disjointly Parallel Lines Lines which do not intersect within the model or at the boundary.

3.1.2 Hyperbolic Length

In this model the distance from any point in the disk to the unit circle is infinite. If (x,y) and (u,v) are Euclidean coordinates of two points in the Klein Model, then the hyperbolic distance between them in Klein model is

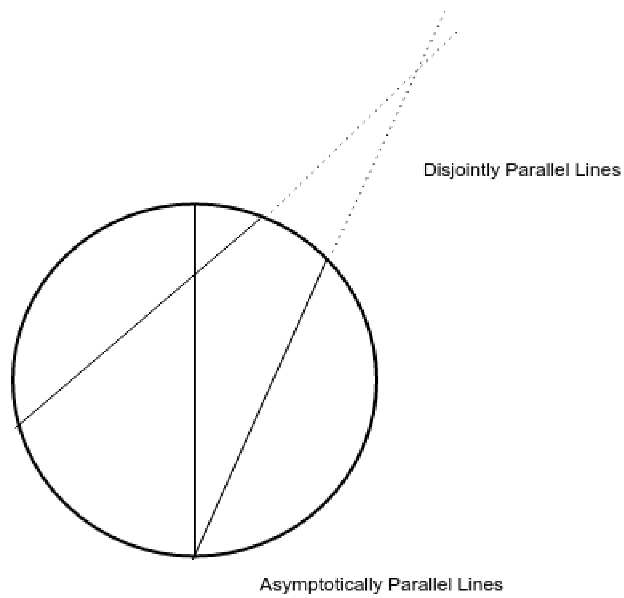


Figure 3.3: Parallel Lines in Klein Model

$$\arccos \frac{(1 - x * u - y * v)}{\sqrt{(1 - x^2 - y^2) * (1 - u^2 - v^2)}}$$

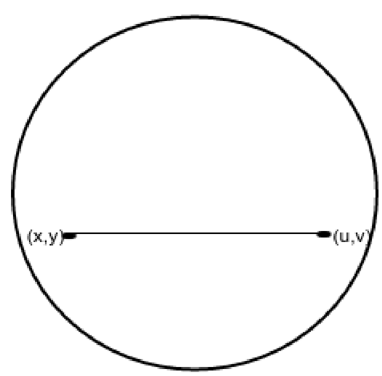


Figure 3.4: Hyperbolic Length in Klein Model

3.1.3 Hyperbolic Angles

A drawback of the Klein Model is that angles in the hyperbolic space get distorted when represented in Klein model, so it is called non-conformal.

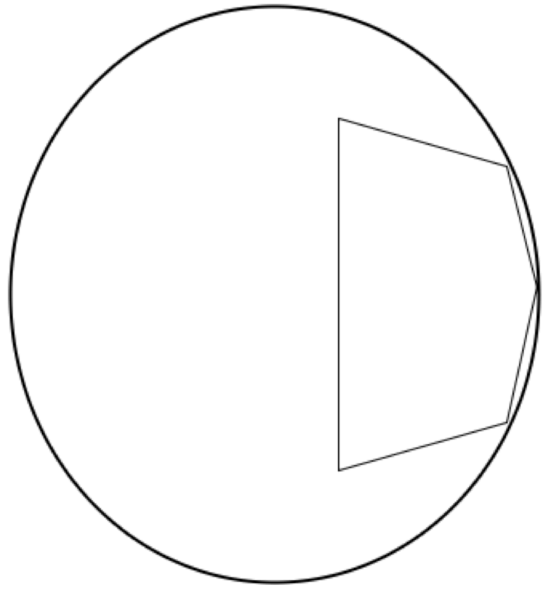


Figure 3.5: A Klein Polygon

3.2 Poincaré Model

Henri Poincaré a French mathematician proposed a model similar to Klein's model, which is finite and represents entire hyperbolic space within the interior of the circle. The terms of Poincaré model are represented as following

- A point is same as an Euclidean point but it is within the unit circle
- A line of hyperbolic plane is defined in two ways. One kind of line is represented by open chords passing through the center of the circle. A second kind of lines are open arcs of circle which are orthogonal to the Unit disc. Angles between the lines can be calcualted with the help of tangents.
- Lies On retains the Euclidean Property
- Between retains the Euclidean Property

3.2.1 Parallel Lines

Euclid's fifth postulate is violated due to open arcs and open chords as shown below.

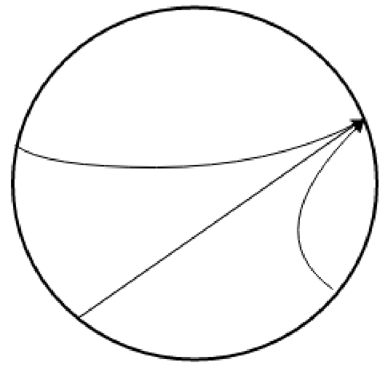


Figure 3.6: Poincaré Lines

3.2.2 Hyperbolic Length

If A and B are two points of Hyperbolic plane on Poincaré disk and P and Q are two points on the boundary then the length between A and B is reprsented by formula

$$Length(A, B) = \lg\left(AP.BQ/BP.AQ\right)$$

3.2.3 Hyperbolic Angles

Angles in the Poincaré disk are same as angles of hyperbolic space which makes this model conformal.

3.3 Weierstrass Model

The Weierstrass Model is an infinite model in hyperbolic geometry. The entire hyperbolic space is represented on the surface of a hyperboloid and looks similar to a cone. The

Klein and Poincaré disk can be obtained by the projections of Weierstrass model.

The mathematical equation of a hyperboloid of two sheets is as described below

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

Here X is vector(x,y,z) which lies on the hyperboloid of two sheets, and the lower sheet points is reflection of the upper sheet. If only upper sheet of the plane is considered then the new equation is

$$< X, X >= -k^2 andz > 0$$

- A point is this model satisfies above upper sheet equation.
- A line can be obtained by a section where a plane passing through the origin intersects the hyperboloid and is given by <X,L> =0 where L is a h-normal to the intersecting plane.
- If a point satisfies the equation <X,L> =0 and Z>0 then it is said to lie on Line L.
- Any line which can be expressed as a linear combination of two points is said to be in between those points.

3.4 Isomorphism in Hyerbolic Models

The Weierstrass model is isomorphic to Klein and Poincaré models, as it can be used to generate both the Klein and Poincaré models. To calculate the properties of the Klein and Poincaré models generally, the Weierstrass model is used.

3.4.1 Weierstrass - Poincaré Model

The stereographic projection of Weierstrass model on to the xy-plane is given by

$$[x, y, z] = 1/(z+1)[x, y, 0]$$

The Poincaré to Weierstrass inverse projections is as below

$$[x, y, 0] = 1/(1 - x^2 - y^2)[2x, 2y, 1 + x^2 + y^2]$$

3.4.2 Weierstrass - Klein Model

Stereographic projection of the Weierstrass model onto z=1 plane gives the Klein Model and is given by

$$[x, y, z] = [x/z, y/z, 1]$$

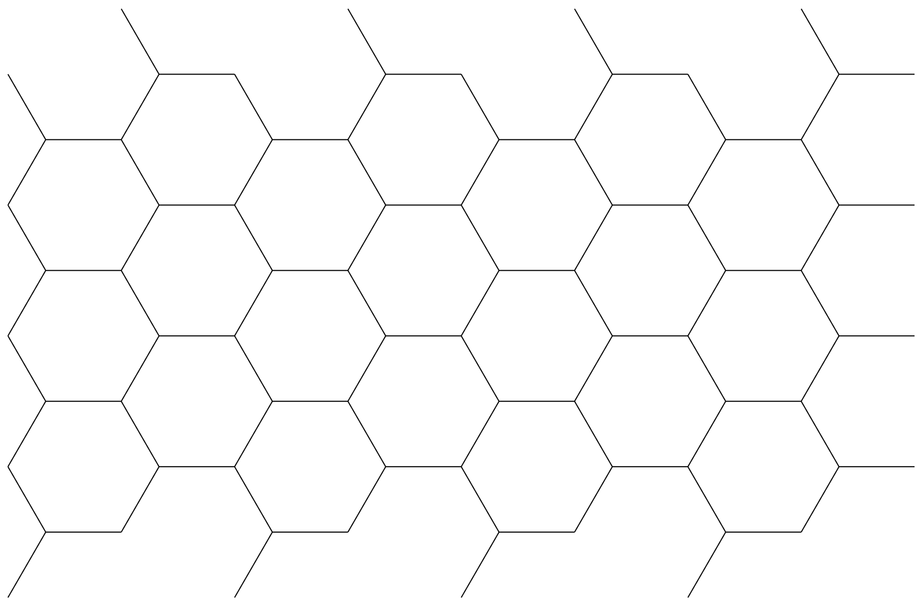
The inverse projection is given by

$$[x, y, 1] = 1/\sqrt{1 - x^2 - y^2}[x, y, 1]$$

4 Tessellation

A tessellation is an arrangement of geometric shapes on a surface such that there are no overlaps and no space between the geometric shapes. For example, tiles in a bathroom or a honeycomb pattern in a bees nest are some of the tessellations which you can see around.

Below is a sample of tessellation in a Honeycomb.



The tessellations mentioned above are basically in a Euclidean plane. But tessellations are also possible in non-Euclidean geometries like Hyperbolic geometry and Spherical geometry.

Tessellations can be represented using the Schläfli symbol. The Schläfli symbol {p,q} is a simple way of representing a Tessellation, where p is the number of sides of a regular polygon and q is the number of polygons which meet at each vertex. [6]

4.1 Hyperbolic Tessellations

The Poincareé disk model or the Klein model can be used for tiling in hyperbolic plane. A Tessellation to be represented in Hyperbolic plane it should satisfy the condition (p-2)(q-2)>4. Otherwise it is Euclidean or Spherical. Figure 4.1 shows the regular hyperbolic tessellation {4,6}. [6]

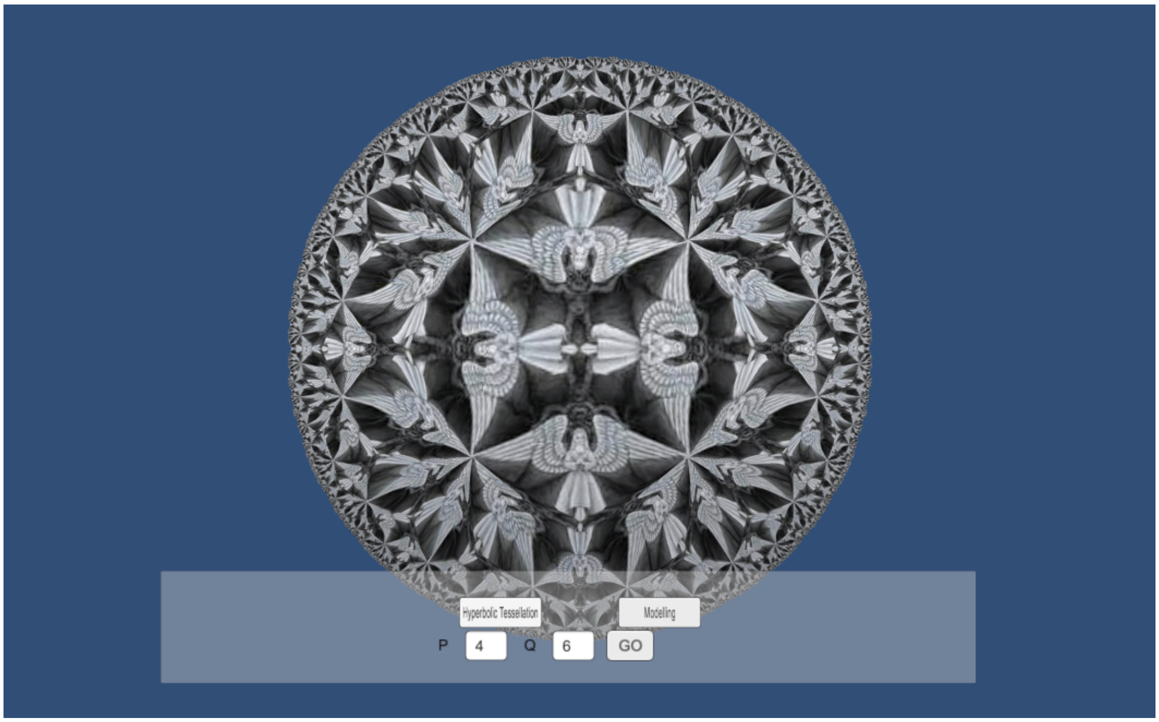


Figure 4.1: Hyperbolic Tessellation

4.2 Infinite Skew Polyhedron

An infinite skew polyhedron has regular polygon faces, a non-planar vertex figure and repeats infinitely in three independent directions [14]. As these polyhedra have negative angle defects at their vertices they have been called hyperbolic tessellations [6].

One of the important parts of this thesis are a special kind of infinite skew polyhedron called Regular skew polyhedra, which were discovered by John Petrie [14]. H.S.M Coxeter used the modified Schläfli symbol $\{p,q|n\}$ to denote them, indicating there are q regular p-gons around each vertex and regular n-gonal holes [2, 3]. Figure 4.2 is the regular skew polyhedra $\{4,6|4\}$ which was developed as part of this thesis.

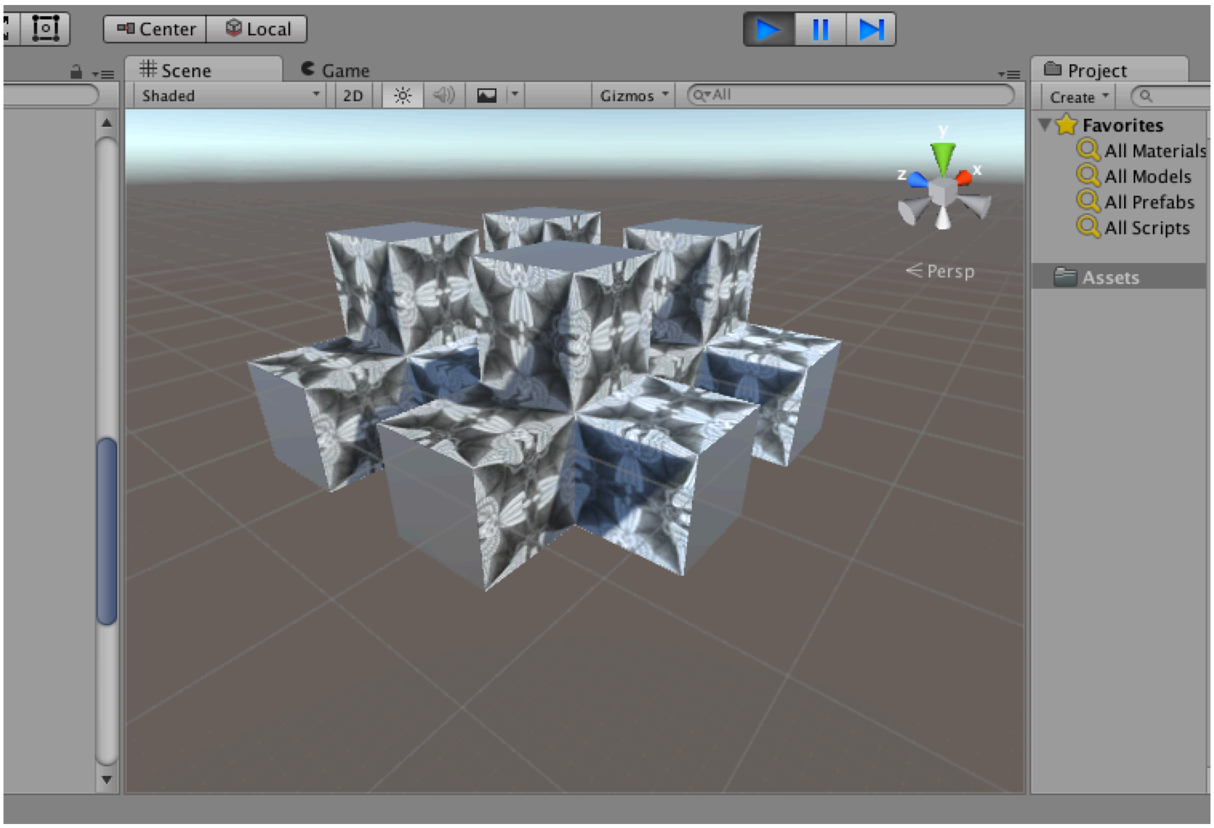


Figure 4.2: An Infinite Skew Polyhedron

5 Implementation

The environment in which this thesis was implemented is a tool called Unity 3D pro game engine and C-sharp is the programming language. Unity is a cross platform game engine which is developed to target multiple platforms such as mobile devices, web browsers, desktops and consoles [13]. Below sections give an in detail an explanation about a concept called procedural mesh generation in Unity and how it was effectively used to generate hyperbolic tessellations and regular skew polyhedra.

5.1 Procedural Mesh Generation in Unity

Unity uses a mesh to display any kind of visual object. A mesh is a construct used by graphics hardware to draw stuff [12], it contains a collection of vertices that define points in 3D space and a set of triangles that are formed by the connections among these points. Triangles form the basis for any type of mesh generated in unity, for example even a straight line in unity is collection of small triangles that may be no large than a pixel, which are arranged in a specific way so that they a represent a straight line.

Any entity in Unity is referred as a game object, which can be assigned an individual name. Mesh Filter, Mesh Renderer are the components of a game object which draw a mesh based on the input vertices and triangles. Triangles are defined as an array of vertex indices and each triangle has three vertices. In order to generate a mesh we need to define the three vertices which form the triangle. Once an array of triangles are defined they are assigned as mesh triangles and the mesh is displayed.

Below is a code snippet which will generate a quad which is mix of two triangles followed by the screenshot of quad generated in Unity.

```

private void meshGenerator {

GetComponent<MeshFilter>().mesh = mesh;

vertices= new Vector3[4];
for(int i=0;i<4;i++) {
vertices[i] = new Vector3(i,i);
}
mesh.vertices = vertices;

int[] triangles = new int[6];
triangles[0]=0;
triangles[1]=triangles[4]=2;
triangles[2]=triangles[3]=1;
triangles[5]=3;

mesh.triangles = triangles;
}
```

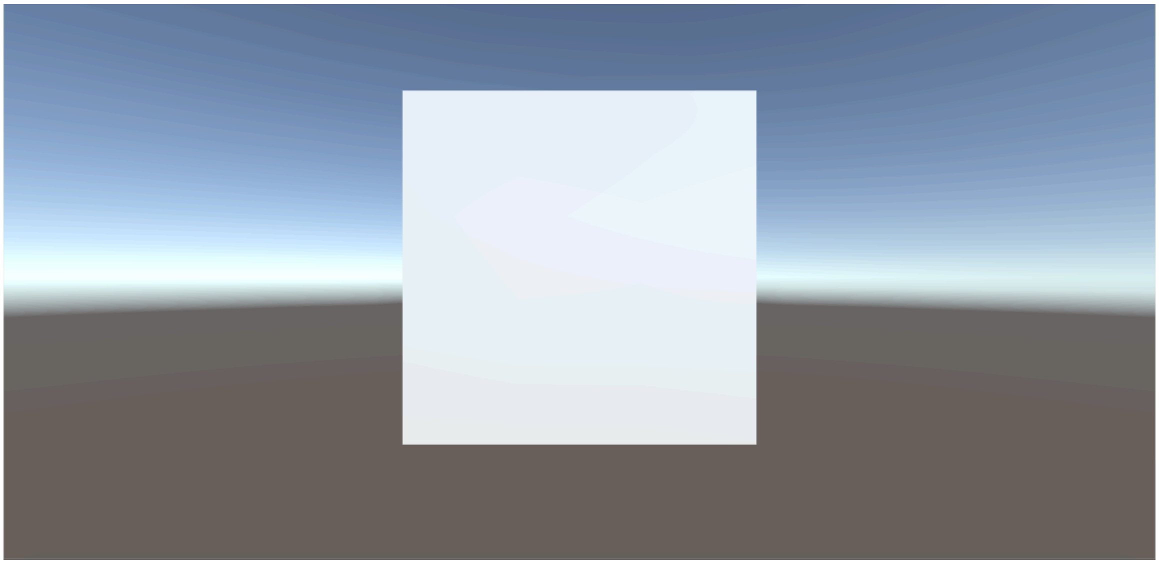


Figure 5.1: A Unity Quad

5.2 Hyperbolic Tessellations on the Ponicaré Disk

Based on Dr. Dunham [7, 5] and H.S.M. Coxeter [4], we identified two main parts of generating a Hyperbolic Tessellation. The first is to generate central regular polygon at the origin of the Poincaré disk. The second is to reflect the central polygon across the edges so that they fill the Poincaré disk. In the sections below, we briefly describe about the steps taken to generate the hypebolic tessellation and give in detail the approach about how we used to implement it in Unity. For detailed algorithms readers are encouraged to look at [7, 5].

5.2.1 Central polygon

As every mesh in Unity is made up of triangles, the central polygon is also made up of a set of triangles. The initial triangle consists of the origin of the Poincaré disk as one of the vertices, the center of an edge of polygon is another vertex and the third and final vertex is the vertex of a polygon.

Let P in Figure 5.2 be the center of Poincaré disk, Q in Figure 5.2 is a vertex of polygon and R in Figure 5.2 is center of an edge in the polygon. Based on [4] we know that any chord passing through points P and Q or chord PQ makes an angle of Π/p with respect to chord PR, where p is number of sides of polygon. And any chord passing through points Q and R or chord QR makes an angle of Π/q with respect to chord PQ, where q is the number of polygons meeting at a vertex of polygon.

Based on the above angles and a series of geometric operations, we were able to calculate the distance between points A and B, from there it was simply a matter of placing vertices every $2\Pi/p$ radians at a distance PQ from the center P to form the complete center polygon.

Below digaram gives a brief idea about the approach followed to generate the center polygon.

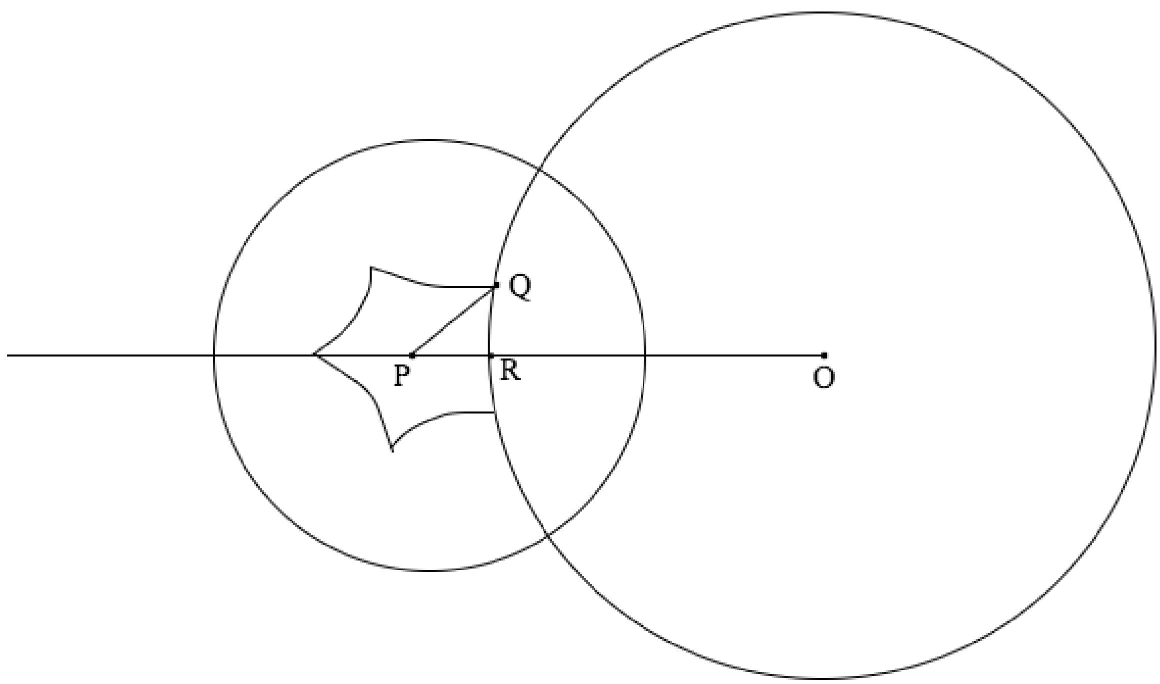


Figure 5.2: Fundamental Region

5.2.2 Generation of Remaining Polygons

We generated the remaining polygons on Poincaré disk by using inversion across the edges of polygons. Once the central polygon is generated, the next step is to invert the polygon across each of its edges to generate adjacent polygons. The process is to find a circle which precisely defines one of the edges of the central polygon. Let O be the center of the circle which defines an edge of a polygon, then a ray which passes from O to a vertex P of the polygon will also contain the inverted point P'of the polygon, where $\mathbf{OP.OP'=r^2}$. Using a series of geometric operations we can calculate the vertex of the polygon, which is adjacent to the edge defined by the circle with center O. When we repeate the steps mentioned above we get all the vertices of polygon adjacent to the edge.

The diagram illustrates the approach

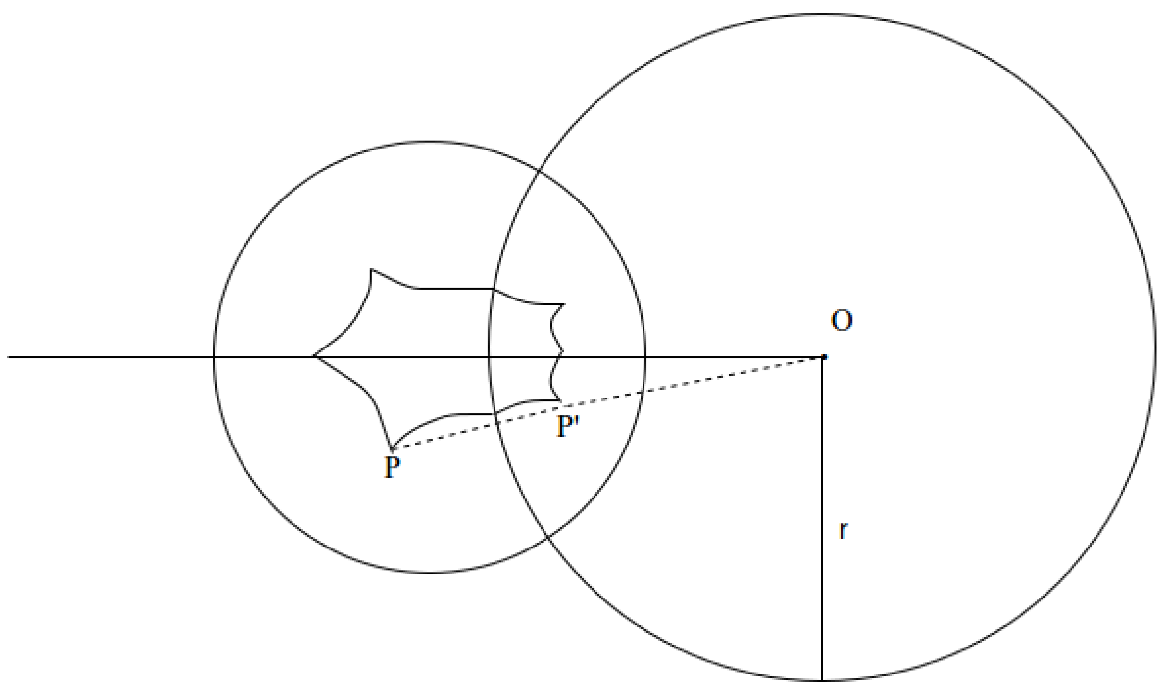


Figure 5.3: Generation of the Remaining Polygons

5.2.3 Visualization of Hyperbolic Lines

The above methods generated vertices of polygons on Poincaré disk. But when we define lines in between those vertices in Unity, they display straight lines between the vertices of polygon, which are not hyperbolic lines. Hyperbolic lines in general are orthogonal to Poincaré disk but not straight lines. Figure 5.4 shows the outcome.

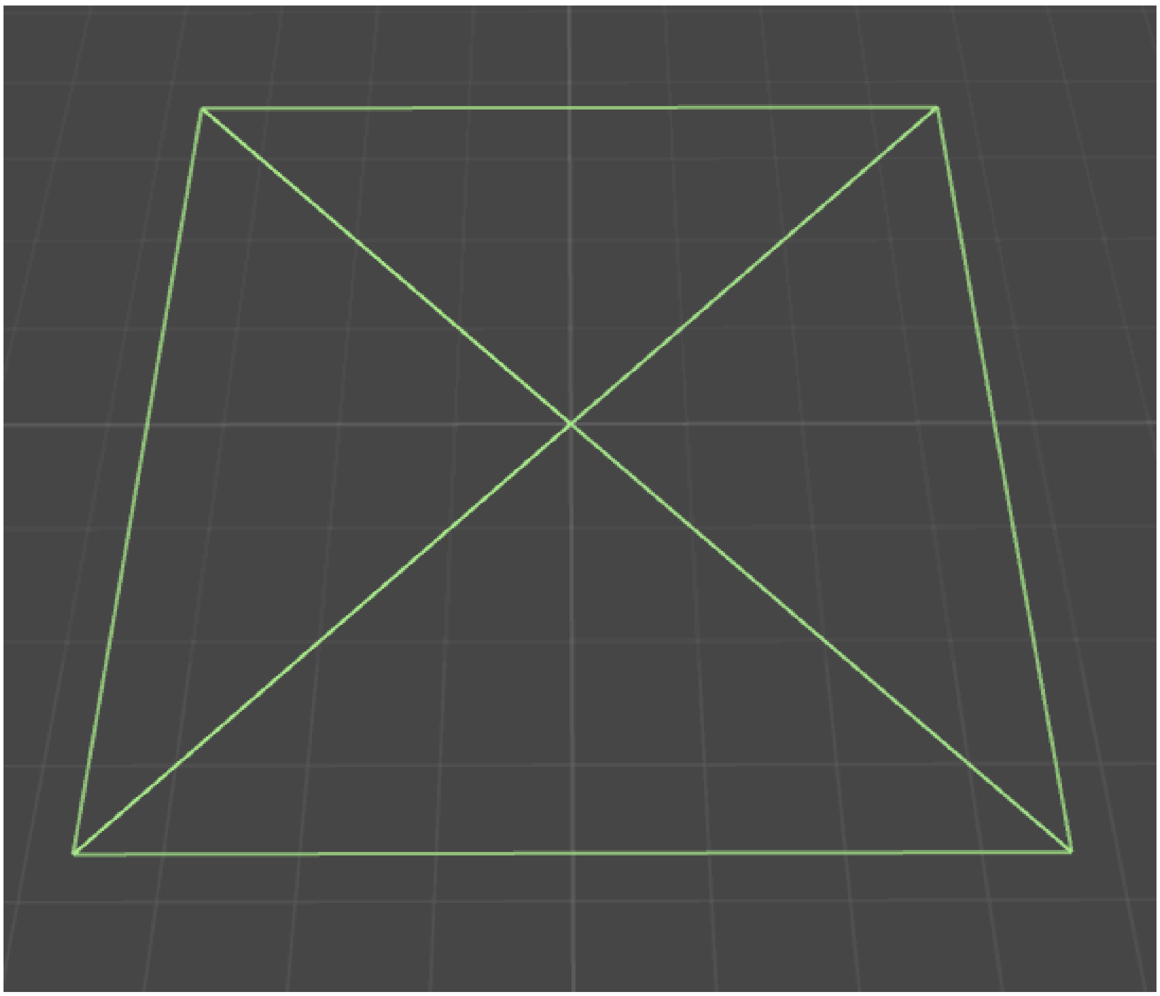


Figure 5.4: Straight Lines

To display each edge of polygon as a hyperbolic line, we took each Poincaré polygon onto the Klein disk. Then subdivided each edge of the polygon in the Klein disk to a certain number of parts.We then projected them back onto the Poincaré disk and generated the polygon. Below are the sample scripts which convert a point in Poincaré to Klein and Klein to Poincar'e. Once the polygon edges are subdivided and converted back to the Poincaré model we generate the mesh required to display the hyperbolic tessellation.

```

public Vector3 PoincaretoKlein(Vector3 point){

return Vector3(2*point.x/(1+point.x*point.x+point.y*point.y),2*point.y/(1+point.x*point.x+point.y*point.y),0);
}

public Vector3 KleintoPoincare(Vector3 point){
return Vector3(Point.x/(1+sqrt(1-point.x*point.x-point.y*point.y)),Point.x/(1+sqrt(1-point.x*point.x-point.y*point.y)),0);
}
```

5.2.4 Mesh Generation of Hyperbolic Tessellation

Every mesh in Unity is a set of triangles, so even a hyperbolic tessellation is a made up of triangles. Rendering in Unity is one sided, which means generally a mesh is only rendered on one side or can be viewed from one side. As we got the sub divided vertices of polygons from above steps, we tried to triangulate the points. But we found that a few triangles are not rendered from one side. The reason was, previously there were only two triangles for a polygon. After subdividing, the number of points which Unity should render increased. But the order in which they should be rendered was not specified.

To form the triangles out of the these points we calculated the center of each polygon. The sum of vertices divided by number of sides of the polygon gave us the center of the polygon. Once we got the center of the polygon we assigned triangles, which have subdivided points as two vertices and the third vertex is the center of the polygon. Figure 5.5 shows the mesh topology, where we can see center of the polygon is the third vertex for every triangle.

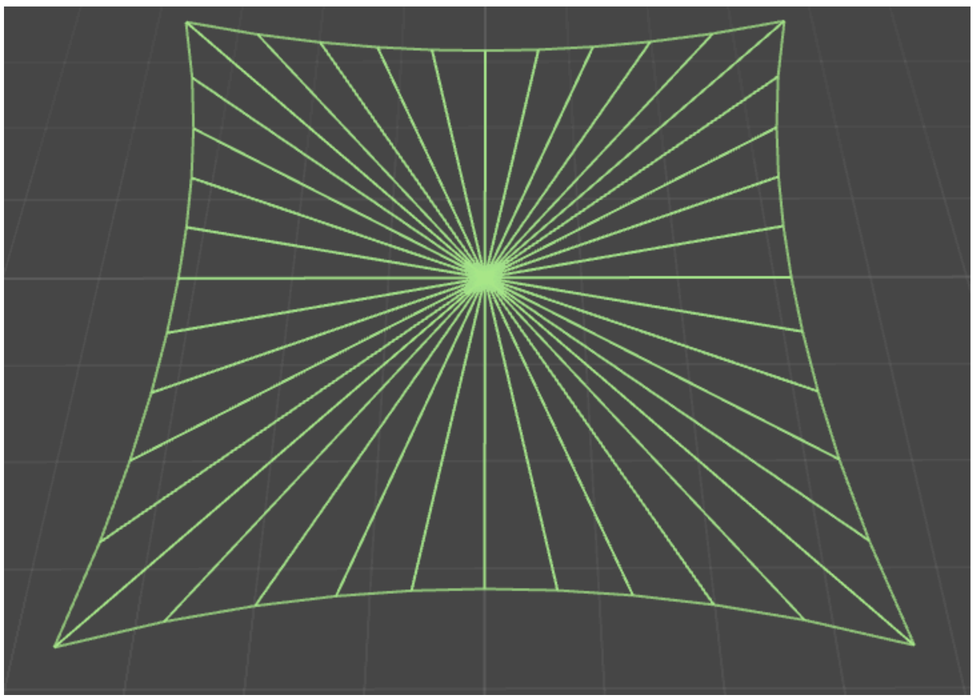


Figure 5.5: Hyperbolic Lines

5.2.5 Texture for a Hyperbolic Tessellation

Once the Mesh is generated a texture is attached as material to the mesh. Texture is attached to every pixel of the polygon. Figure 5.6 of a hyperbolic tessellation.

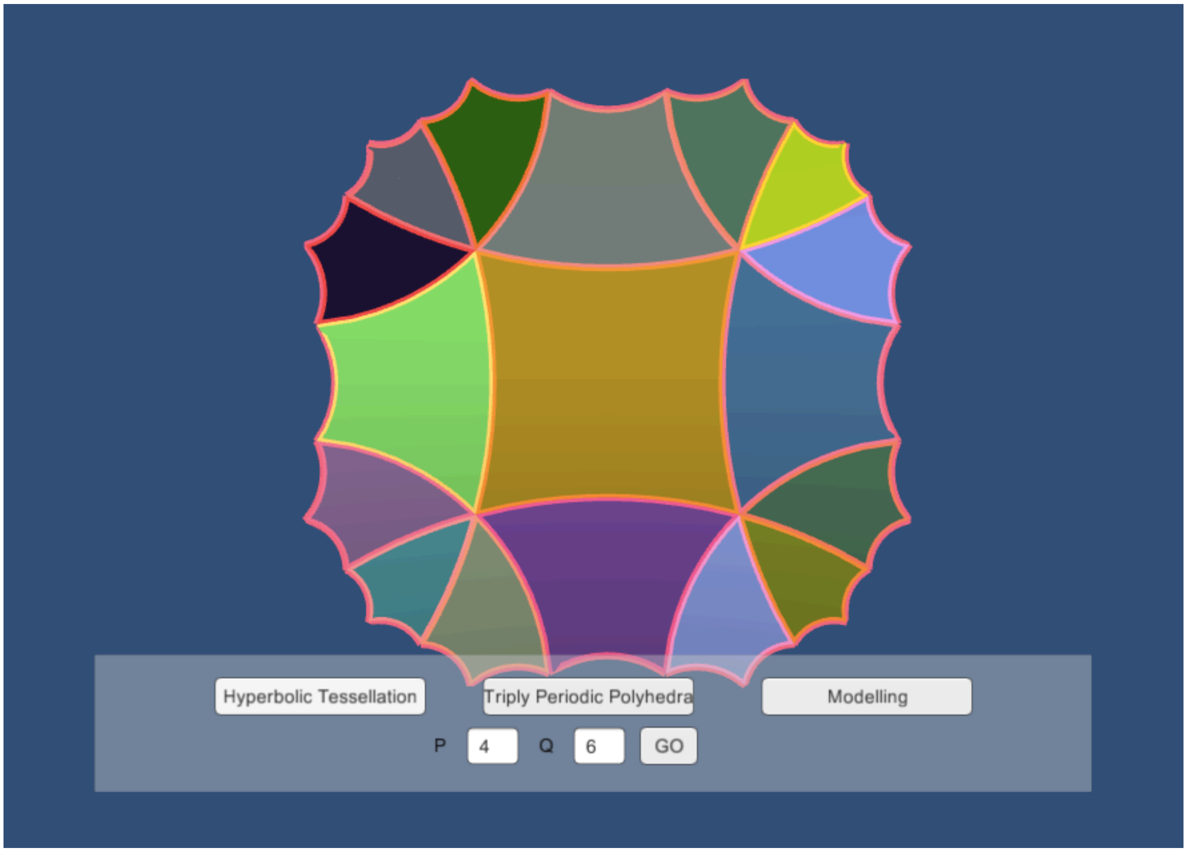


Figure 5.6: A Unity Tessellation

5.3 Triply Periodic Polyhedron

As explained by Dr. Dunham [6], triply periodic polyhedra are related to hyperbolic tessellations since they have negative angle defects at their vertices. H.S.M. Coxeter used the modified Schläfli symbol $\{p,q|n\}$ to denote them, indicating that there *q regular p-gons* around each vertex and regular *n-gonal* holes [2, 3]. Figure 5.7 is a screenshot of pattern $\{4,6|4\}$ generated in Unity.

5.3.1 Generation of Triply Periodic Polyhedron

This thesis concentrates mainly on the relation between a hyperbolic tessellation and its corresponding triply periodic polyhedron. We show the triply periodic polyhedron as

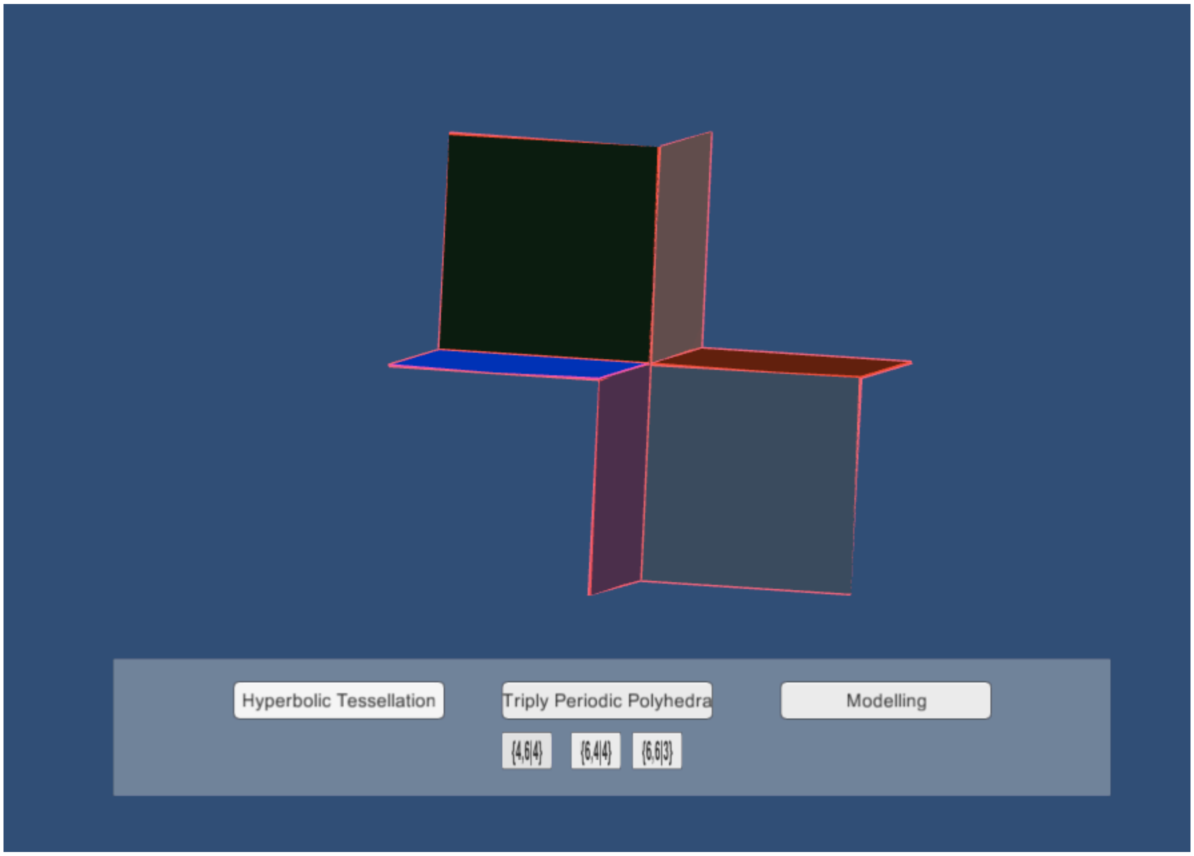


Figure 5.7: Triply Periodic Polyhedron

part of modelling part in the application, where user is able to move a sphere over a hyperbolic tessellation and can find another sphere automatically moving over the triply periodic polyhderon.

A vertex of the central polygon in the hyperbolic tesselation is the center for triply periodic polyhedron. Polygons around the vertex are represented in the triply periodic polyhedron model. The generation of triply periodic polyhedron involves two steps. The first step is to generate the central polygon which represents the central polygon of the hyperbolic tessellation, and in second step all the remaining polygons which share the same vertex are generated.

The edges of polygons in the hyperbolic tessellation are curved, as they are represented on a Poincaré disk. But the polygons in triply periodic polyhedra have straight lines, so in order to generate the first polygon, the central polygon of hyperbolic tessellation was projected onto a Klein disk. As polygons in the Klein disk have straight edges, the first polygon was literally a Klein model representaion.

Once the first polygon was generated, the next step involved the generation of polygons sharing the vertex. After a series of geometric steps a $\{4,6|4\}$ was generated and Figure 5.7 shows the triply periodic polyhedron.

5.4 Interaction Between Hyperbolic Tessellation and Triply Periodic Polyhedron

The application developed as part of this thesis has a section for modelling between hyperbolic tessellation and triply periodic polyhedron. The user will be able to move a point using a mouse on the hyperbolic tessellation and a corresponding point will move on the respective tile in the triply perioidic polyhedron. A hyperbolic tessellation on a Poincaré disk is two-dimensional representaion, where as a triply periodic polyhedron is a three-dimensional structure. Area of polygons on Poincaré disk decrease when moving from center to boundary of disk, where as polygons in triply periodic polyhedron are of the same size even though they represent the polygons on Poincaré disk. As stated in above two points, direct representation of point between hyperbolic tessellation and triply periodic polyhedra involves some series of geometrical calculations.

To achieve the modelling, three steps are required to generate the representation of the point on hyperbolic tessellation. In the first step, the Poincaré disk is projected onto the Weierstrass model. As the Weierstrass model is an infinite model of hyperbolic geoemtry, it is used to represent the infinite space which Poincaré represents. In the second step if the polygon is not the central polygon of Poincaré model, it is translated and rotated to the center of the Weierstrass model using below matrices which are suggested in [7, 5] . In the third and final step, it is projected back on to the Klein disk. The distance between the offset of the polygon in the triply periodic polyhedron and the point on polygon is calculated and then represented on the specific polygon on the triply periodic polyhedron.

The tranformation used in the program are represented by 3X3 matrices.

$$ReflectPgonEdge := \begin{bmatrix} -cosh2q & 0 & sinh2q \\ 0 & 1 & 0 \\ -sinh2q & 0 & cosh2q \end{bmatrix}$$

$$ReflectEdgeBisector := \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$ReflectHypotenuse := \begin{bmatrix} cos(2\Pi/p) & sin(2\Pi/p) & 0 \\ sin(2\Pi/p) & -cos(2\Pi/p) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where

$$coshq = cos(\Pi/q)/sin(\Pi/p)$$

$$cosh2q = 2 * coshq * *2 - 1$$

$$sinh2q = sqrt(cosh2q * *2 - 1)$$

The code snippet below projects a Poincaré point onto the Weierstrass model.

```
public Vector3 poincaretohyperboloid(Vector3 poincarepoint){
```

```
x = poincarepoint.x;
y = poincarepoint.y;
divisor = (1-x*x-y*y);
return new Vector3(2*x/divisor,2*y/divisor,(1+x*x+y*y)/divisor);
}
```

The code snippet below projects a Weierstrass model point onto the Klein model.

```
public Vector3 hyperboloidtoklein(Vector3 hyperboloidpoint){

return new Vector3(hyperoboloidpoint.x/hyperboloidpoint.z,hyperboloidp
}
```

Figure 5.8 shows the final output. A point on hyperbolic tessellation and its respective point on triply periodic polyhedron.

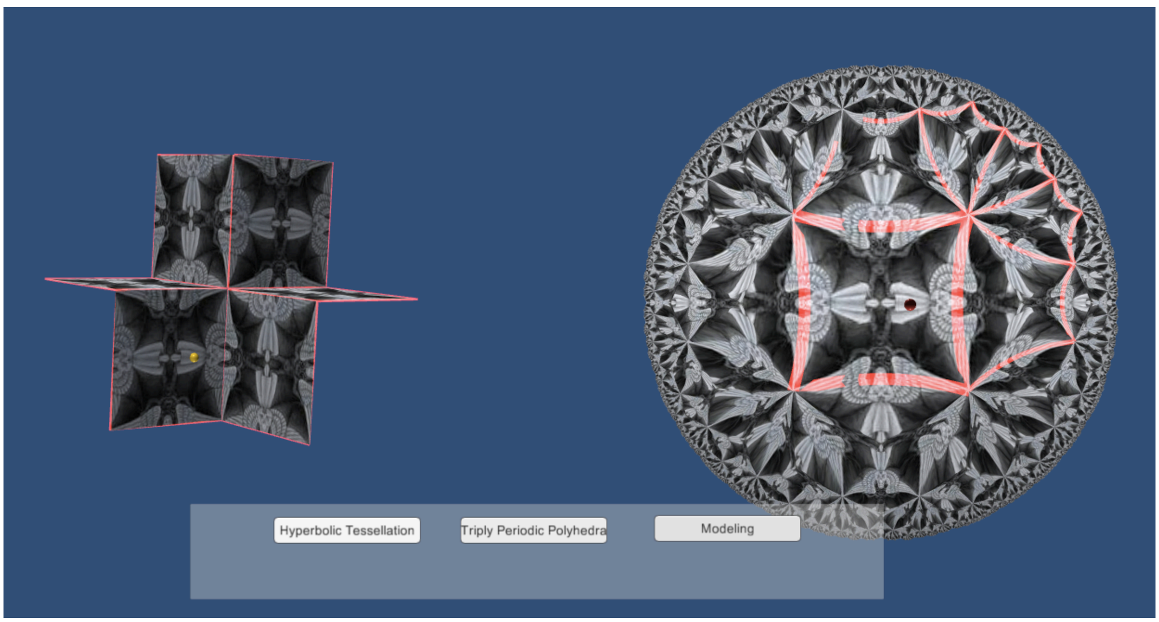


Figure 5.8: Triply Periodic Polyhedron and Hyperbolic Tessellation

6 User Interface

The user interface provided by the program allows the user to perform various functionalities such as creating new hyperbolic tessellations, a triply periodic polyhedron, and interaction between a triply periodic polyhedron and its respective hyperbolic tessellation. The program can be used on any platform including personal computers, mobiles and tabs, which is one of the main advantages of using Unity pro.

When the application is started, the user interface will have three clickable buttons at the bottom of the application as shown in the Figure 6.1. The first one is *Hyperbolic Tessellation* button, which upon clicked opens further options. The second one is *Triply Periodic Polyhedron* display button. Final button *Modelling* allows user to establish the connection between a triply periodic polyhedron and its hyperbolic covering tessellation.

Figure 6.1: The User Interface upon application execution

Once the user clicks or selects the *Hyperbolic Tessellation* button, the user is presented with two input boxes. The first box is the number of edges p a p-gon should have in the hyperbolic tessellation. The second box is number of p-gons q meeting at each vertex in the tessellation. Figure 6.2 shows $\{6,4\}$ tessellation. Note :- In a hyperbolic tessellation (p-2)(q-2)>4, so the application won't generate if the values do not satisfy the condition

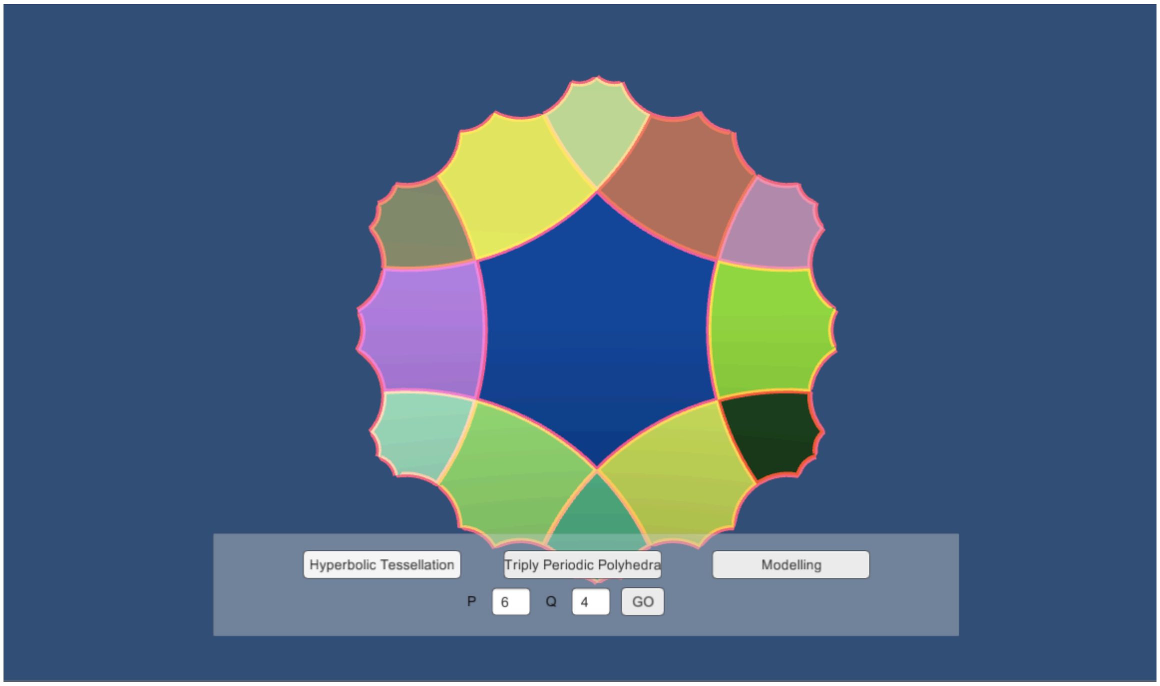


Figure 6.2: The Hyperbolic Tessellation Interface

If the user selects the *Triply Periodic Polyhedron* button, the application will display other clickable buttons. The user can select the available triply periodic polyhedrons and the application will display them. Figure 6.3 displays $\{4,6|4\}$ triply periodic polyhedron.

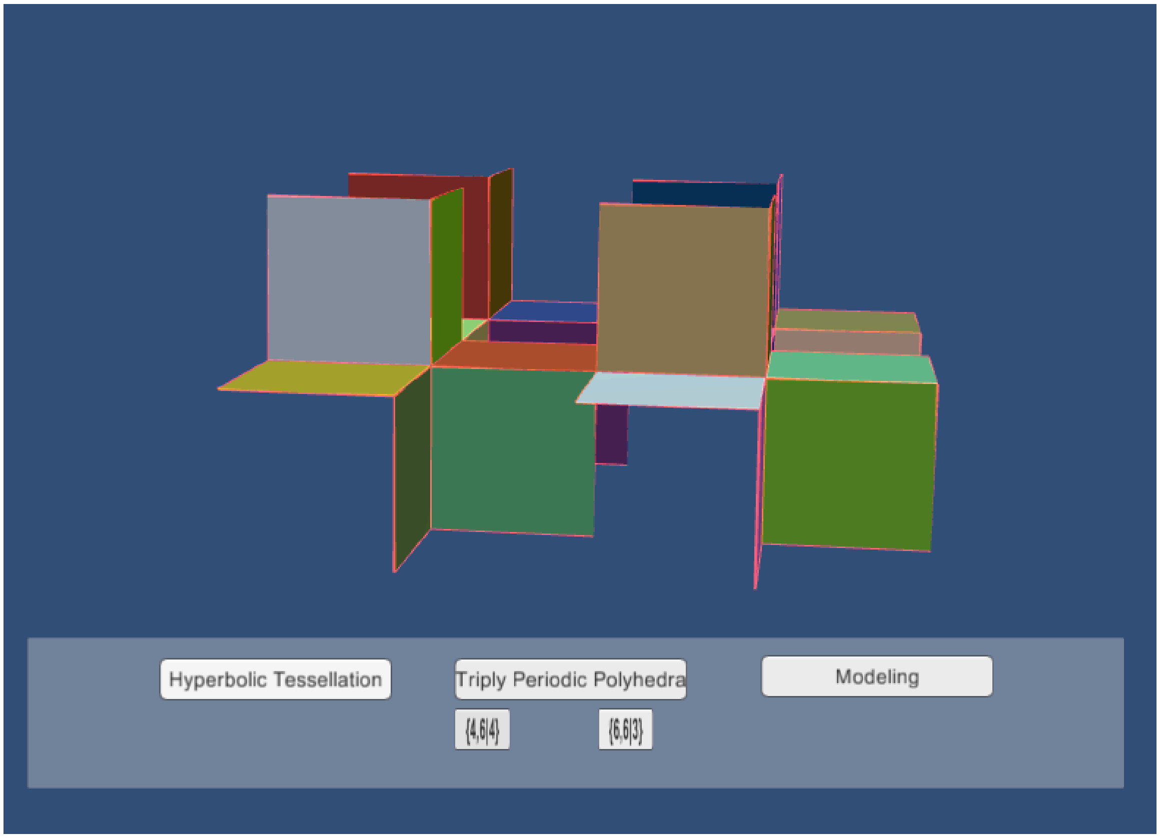


Figure 6.3: Interface displaying Triply Periodic Polyhedron

The third button *Modelling* allows the user to establish the connection between a triply periodic polyhedron and its hyperbolic covering tessellation. Upon selecting the button, the application displays the $\{4,6|4\}$ triply periodic polyhedron on the left side of the screen and its $\{6,4\}$ hyperbolic tessellation on the right side of the screen. The user with the help of a mouse, can move a point on the hyperbolic tessellation and a corresponding point will move on the respective tile in triply periodic polyhedron. Edges of the polygons both in the triply periodic polyhedron and the hyperbolic tessellation are highlighted or get bright when the mouse pointer is on the respective polygon. The Point on the hyperbolic tessellation, which can be moved is displayed in *red* color and its corresponding point in the triply periodic polyhedron point is *yellow* in color. Figure 6.4 shows the hyperbolic tessellation with red point on right of the screen and triply periodic polyhedron with yellow point on left of the screen. The User can see the edges of the polygon are highlighted when the point is above the polygon.

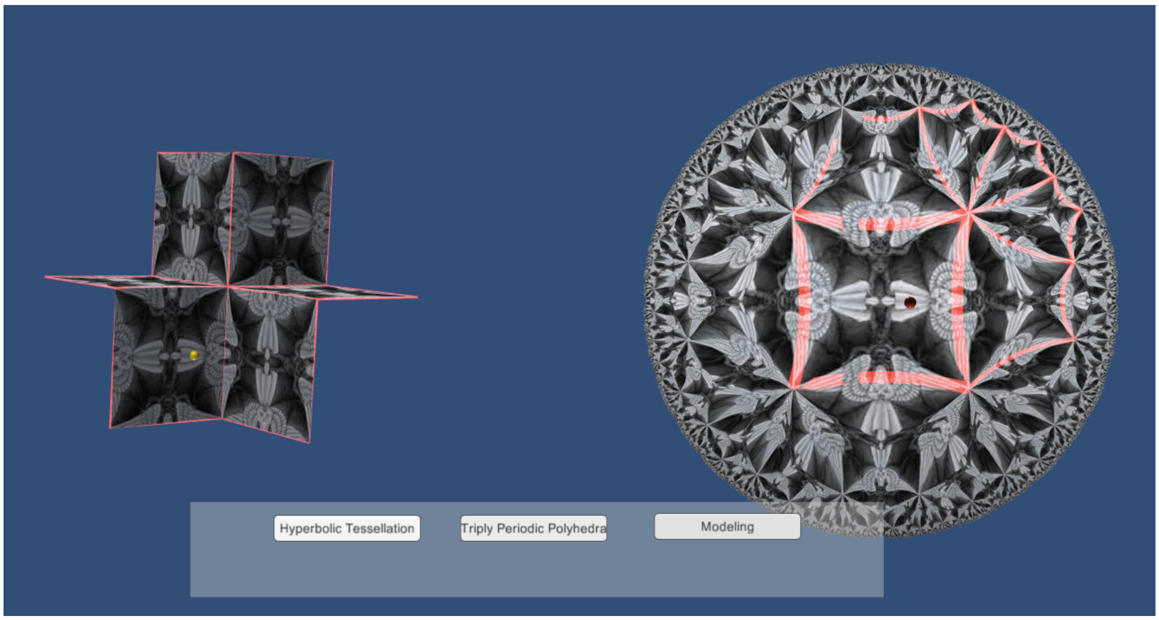


Figure 6.4: Modelling between Triply Periodic Polyhedron and its corresponding Hyperbolic Tessellation

7 Results

This chapter contains the outcome of the application developed as part of this thesis. This application generates hyperbolic tessellations, triply periodic polyhedrons, and establishes a connection between a triply periodic polyhedron and its respective hyperbolic tessellation. This program is based on an algorithm by Dr. Dunham [5]. Previous applications developed on this concept generate tessellations based on many input parameters, but this application simplified that process. The user can enter number of edges of a p-gon and number of p-gons meeting at a vertex and it generates the respective hyperbolic tessellation. It allows user to view the triply periodic polyhedron seperately. It also allows user to visualize the triply periodic polyhedron and its respective hyperbolic tessellation in a single screen, where user can model the movement of a point on hyperbolic tessellation and the application will move a point in its respective tile on triply periodic polyhedron. The Figures below are some of the screenshots from this application.

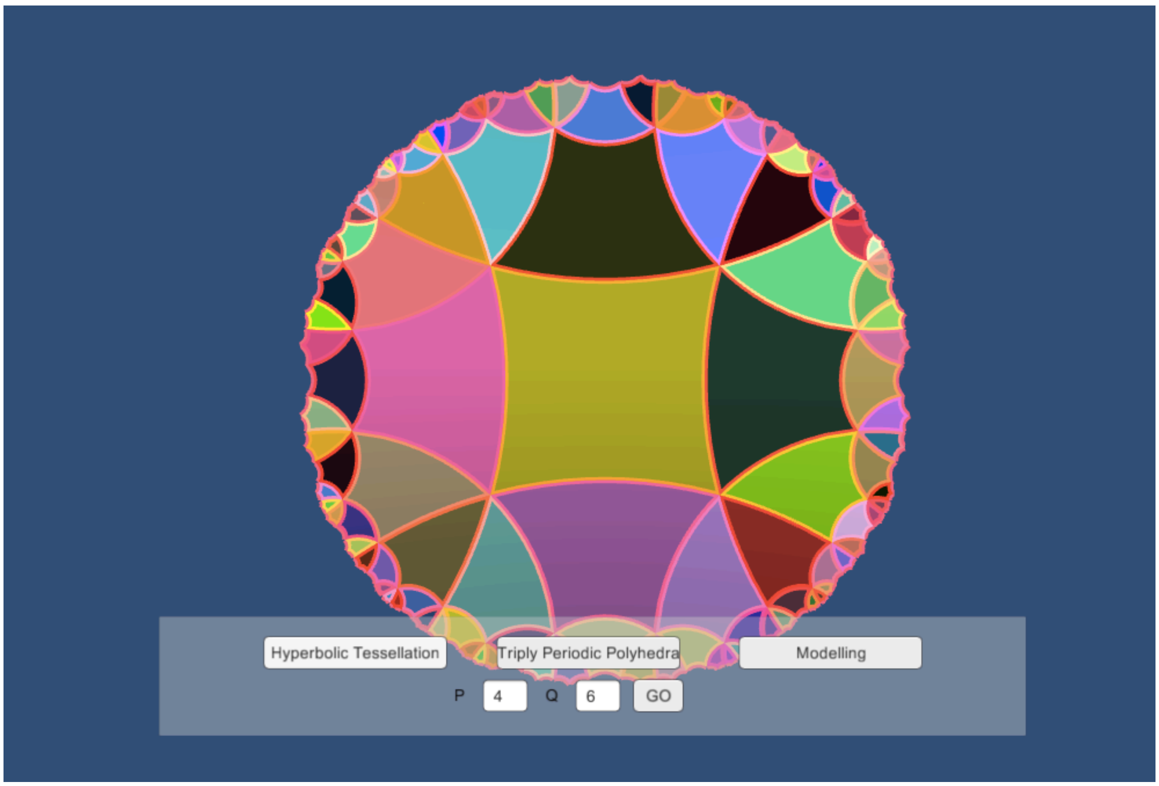


Figure 7.1: The {4,6} Hyperbolic Tessellation

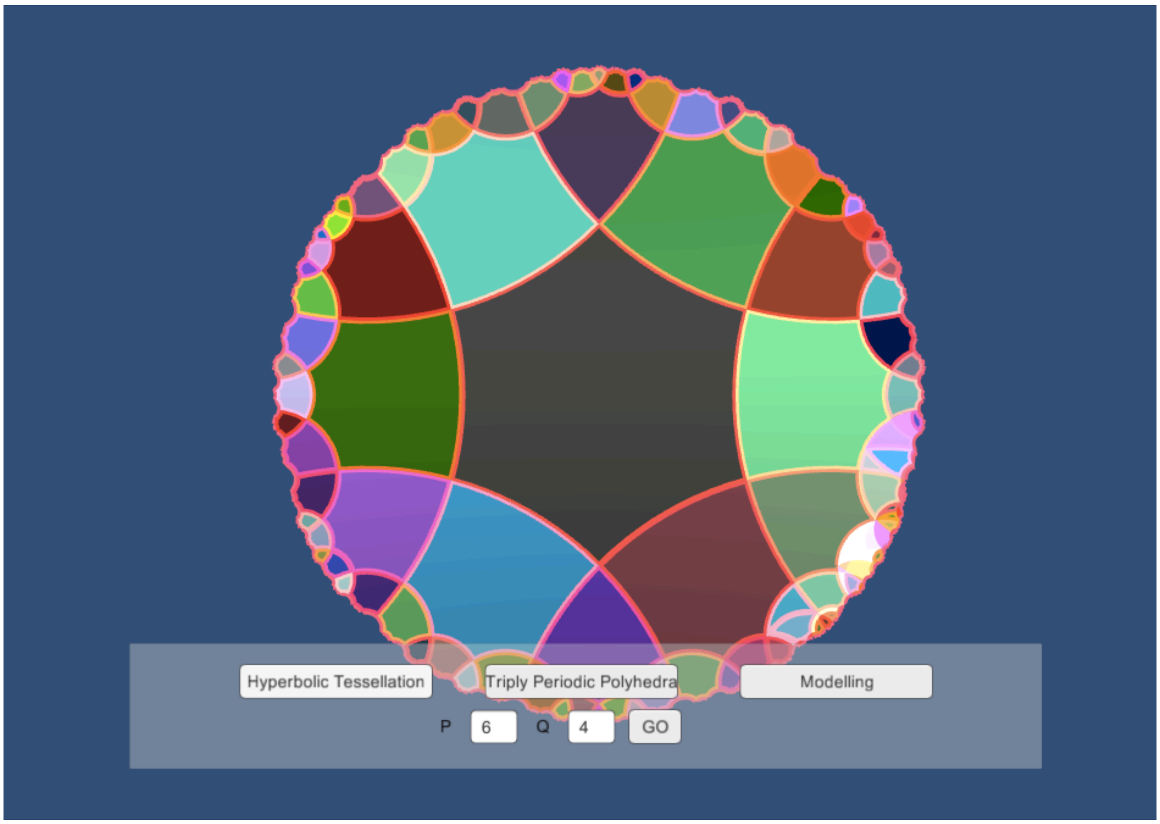


Figure 7.2: The {6,4} Hyperbolic Tessellation

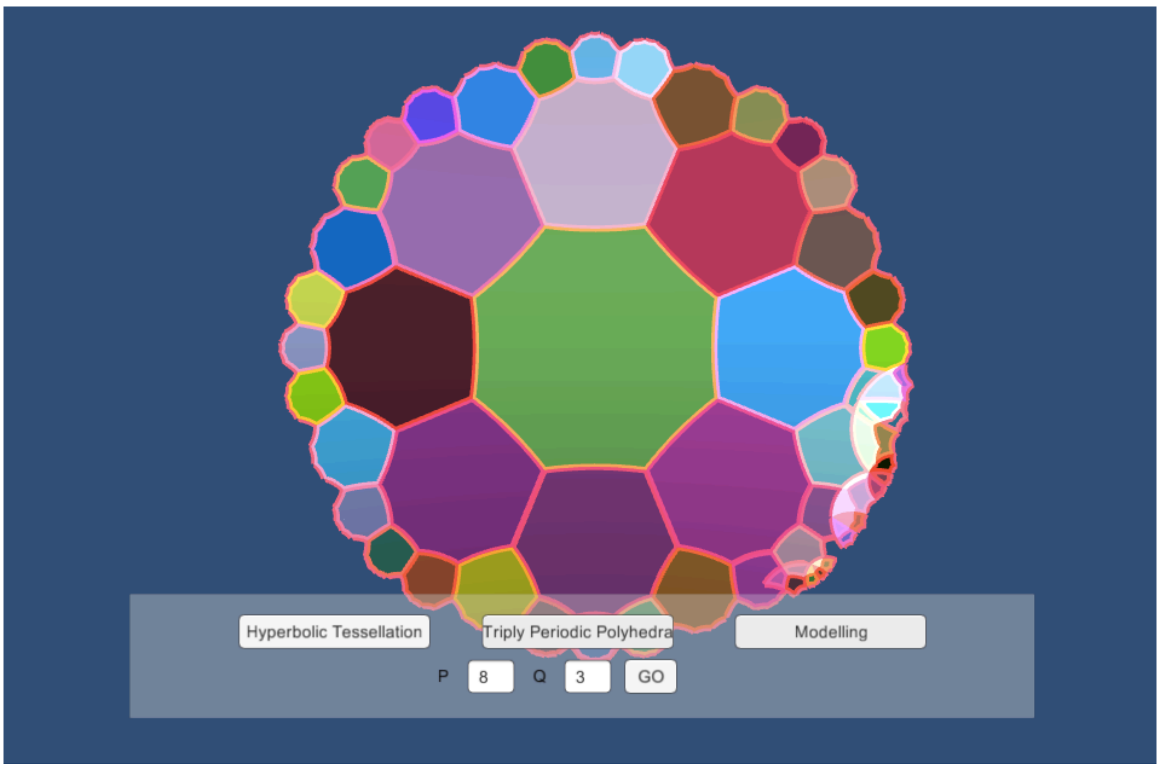


Figure 7.3: The {8,3} Hyperbolic Tessellation

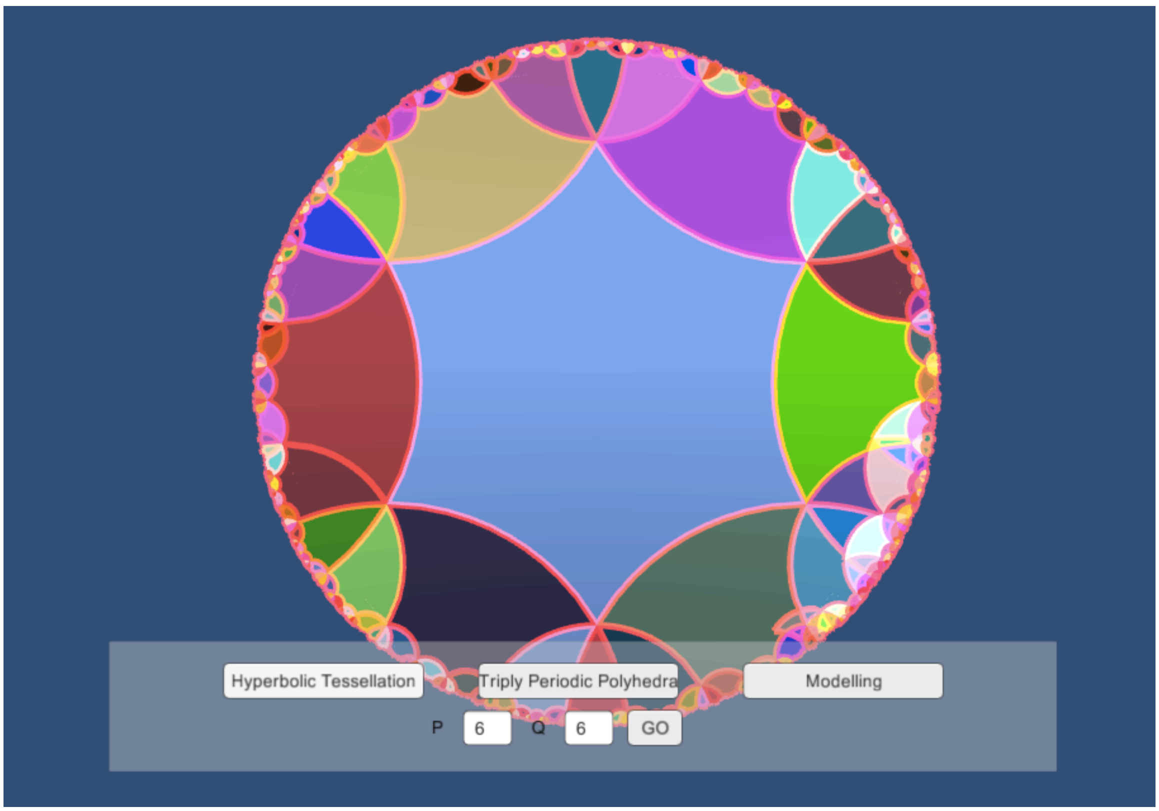


Figure 7.4: The {6,6} Hyperbolic Tessellation

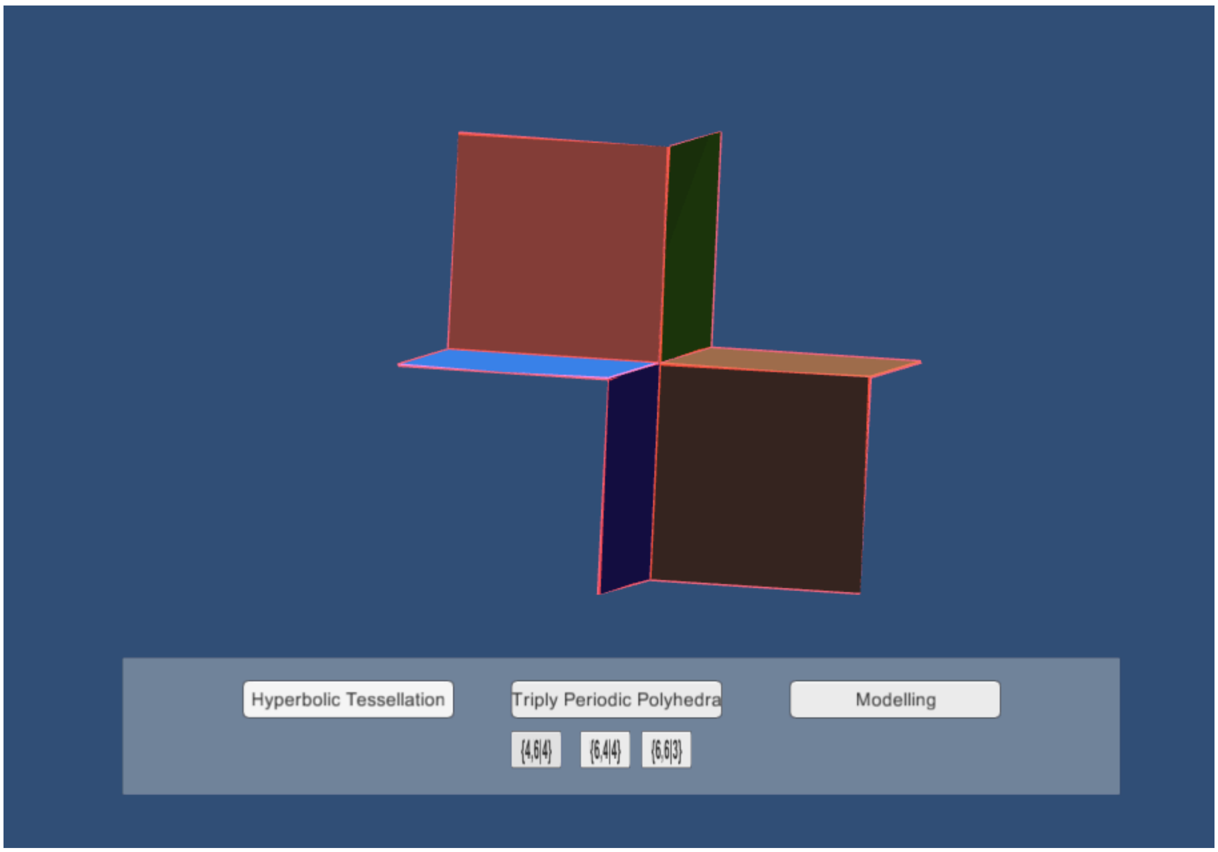


Figure 7.5: The {4,6|4} Triply Periodic Polyhedron

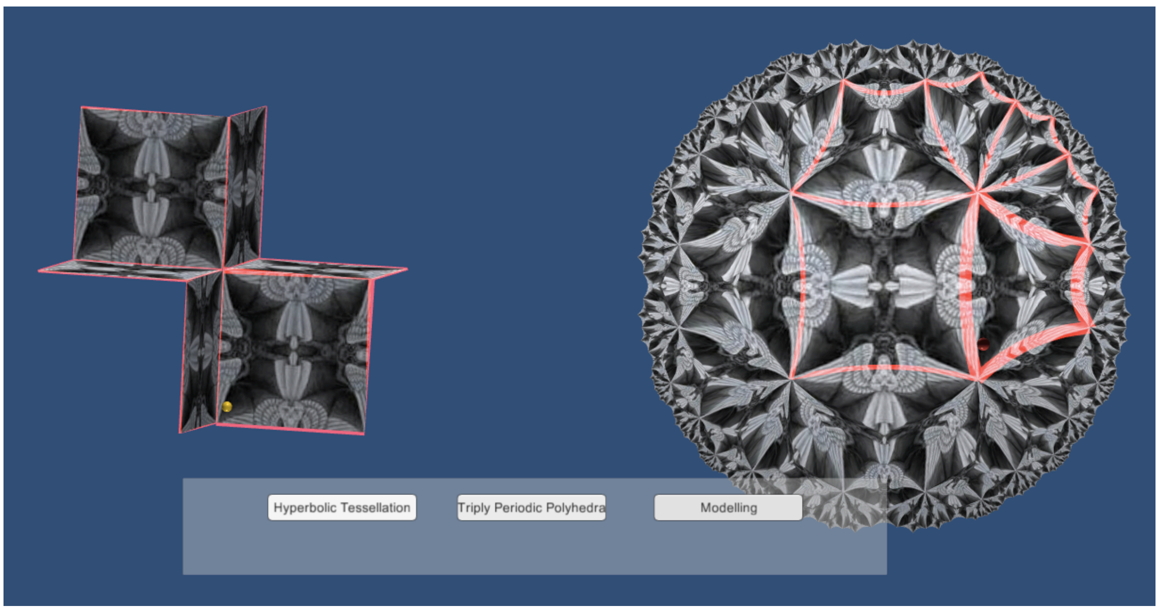


Figure 7.6: Establishing a Connection between Triply Periodic Polyhedron and Hyperbolic Tessellation

8 Conclusions

This research focuses on establishing a connection between a triply periodic polyhedron and hyperbolic tessellation. The application provides interactive capabilities to the user to generate the hyperbolic tessellations and the triply periodic polyhedra. The application is portable and compatible for multiple devices. It was tested for various hyperbolic tessellations, triply periodic polyhedra, modelling the point movement between the triply periodic polyhedron and hyperbolic tessellation. The results were as expected.

Further Enhancements can be made to this application. The user can be allowed to generate an infinite triply periodic polyhedron pattern. Generalize the pattern repitition algorithm for all the hyperbolic tessellations. The user can utilize the Unity program, to generate a 3-dimensional model of the triply periodic polyhedron using a 3-dimensional printer.

Another direction to take would be to allow the user to manifest the triply periodic polyhedron as an architectural model building and visualize using head mounted display.

Bibliography

[1] K. Bhumakar. *Interactive Visualization of Hyperbolic Geometry using the Weierstrass Model*. 2006 (cit. on p. 13).

[2] H. Coxeter. *Regular Polytopes*. third edition Dover Publications, 1973. ISBN: 0-486-61480-8 (cit. on pp. 22, 31).

[3] H. Coxeter. *The Beauty of Geometry : Twelve Essays*. Dover Publications, 1999. ISBN: 0-486-40919-8 (cit. on pp. 22, 31).

[4] H. Coxeter. ``The trigonometry of hyperbolic tessellations''. In: (1997) (cit. on pp. 25, 26).

[5] D. Dunham. ``Hyperbolic Symmetry''. In: *Computers & Mathematics with Applications 12.1*. 1986, pp. 139–153. DOI: 10.1016/0898-1221(86)90147-1. URL: http://www.sciencedirect.com/science/article/pii/0898122186901471 (cit. on pp. 25, 34, 41).

[6] D. Dunham. ``Patterned Triply Periodic Polyhedra''. In: 2012. URL: https://www.d.umn.edu/~ddunham/bridges12.pdf (cit. on pp. 20, 21, 31).

[7] D. Dunham, J. Lindgren, and D. Witte. ``Creating repeating hyperbolic patterns''. In: *SIGGRAPH '81 Proceedings of the 8th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 1981, pp. 215–223. ISBN:

0-89791-045-1. DOI: 10.1145/800224.806808. URL: http://dl.acm.org/citation.cfm?id=806808 (cit. on pp. 25, 34).

[8] D. E.Joyce. *Euclid's Element*. Worcester, MA, 1996 (cit. on p. 4).

[9] M. J. Greenberg. *Euclidean and Non-Euclidean Geometries*. New York: W.H. Freeman and Company, 1994 (cit. on pp. 5, 6).

[10] *Hyperbolic space Crochet Models*. http://theiff.org/oexhibits/oe1e.html. Accessed: 2016-06-06 (cit. on p. 8).

[11] *Parallel Lines in Hyperbolic Geometry*. http://planetmath.org/parallellinesinhyperboli Accessed: 2016-06-06 (cit. on p. 14).

[12] *Procedural Mesh*. http://catlikecoding.com/unity/tutorials/procedural-grid. Accessed: 2016-06-10 (cit. on p. 23).

[13] *Unity Game Engine*. https://en.wikipedia.org/wiki/Unity_(game_engine). Accessed: 2016-06-06 (cit. on p. 23).

[14] *Wikipedia entry for "Infinite skew polyhedron"*. http://en.wikipedia.org/wiki/Infinite_skew_polyhedron. Accessed: 2016-06-10 (cit. on pp. 21, 22).