# VISUALIZING THE MANHATTAN CURVE — JOURNAL

WILLIAM CLAMPITT AND GIUSEPPE MARTONE

Abstract. This notebook will serve as a research journal for William's master thesis project.

Contents



*Date*: November 2024.

## 1. November 19, 2024

### 1.1. **Meeting notes.**

1.1.1. *Counting Problems.* Gauss estimated that the distribution of prime numbers was

$$\#\{p \in P \colon p \leq T\} \sim \frac{T}{\ln T}$$

These type of counting problems are common in fields such as Geometry, Topology, and Dynamical Systems.

Typically, in this context

$$\#\{a \in A \colon a \leq T\} \sim \text{ exponential in } T$$

*Topoligical Entropy.* Typically, topological entropy is calcualted with the below formula:

$$h_{\text{top}}(A) = \lim_{T \to \infty} \frac{1}{T} \ln \#\{a \in A \colon a \leq T\}$$

**Example.**

$$h_{\text{top}}(\mathbb{N}) = 0$$

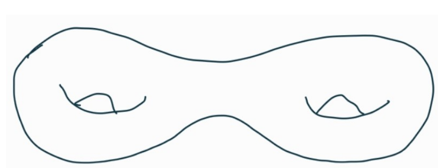

Figure 1. Hyperbolic structure (locally looks like $\mathbb{H}^2$)

*Hilbert Metric First Glance.*

(1) You need a hyperbolic structure on $S$, call it $p$.

(2) The set of closed loops (discrete)
   - Each closed loop has a length w.r.t. the hyperbolic structure (number $> 0$)

We will denote the length of the curve $c$ as $l_p(c)$ w.r.t. $p \in \mathbb{R}$ where $p > 0$.

$$\#\{c\colon l_p(c) < T\} \sim \frac{e^T}{T}$$

*Remark* 1. This was shown by Huber Margulis [William] Is this person the same as Grigory Margulis? Maybe I wrote the name down wrong when I was taking notes.[1]

*Origin of Program Matrices.* The three matrices in the first program are representing a relfection accross the three distinct edges of the triangles that are formed when we stretch the holes of our pants to the boundary of our hyperbolic space.

### 1.2. Post-meeting notes.

#### 1.2.1. *Counting Problems.*

**Example.** The distribution of prime numbers. In 1792 Gauss proposed that

$$\pi(n) \sim \frac{n}{\ln n}$$

but was later refined to

$$\pi(n) \sim \mathrm{Li}(n),$$

where

$$\mathrm{Li}(n) = \int_2^n \frac{\mathrm{d}x}{\ln x}$$

[1]

## 2. December 3, 2024

### 2.1. Meeting notes.

**Summary:**

In this meeting we discussed the problems that I encountered with my program. In summary, my program is using up too much system memory of the computer. This results in the program crashing which prematurely halts the calculation of the words of our alphabet. Because I was not

[1] G: Huber proved it for hyperbolic surfaces. G. Margulis generalized it and his proof is more dynamical

incrementally storing the already computed matrices, the program would not yeild any results if it crashed.

*Potential Solutions to Problem.* I think one way that might be good to reduce the system memory usage of my program would be to store the words of length $n$ in a file, then use those words to generate the words of length $n+1$. The words of length $n$ would then be unloaded from system memory. After the words of length $n$ are unloaded, we can use the words of length $n+1$ to calculated the words of length $n+2$ and so on.

2.2. **Post-meeting notes.** I am attempting to rewrite my current program using Octave while also implimenting my idea about iteratively saving the words of length $n$ each time so that I do not have to store the entirety of the list in system memory at one time. After this is done, the list will need to be sifted through to remove duplicates matrices.

So far it seems like the success of this project would be greatly benifited by reducing the number of matrix multiplications a program would have to do. Currently before any duplicates are removed, generating all of the words up to length $n$ creates $3^{\frac{n(n+1)}{2}}$ matrices. This takes a very long time to compute. My theory though is that this creates a lot of duplicates, especially when we are relitavely close to the origin of our tiling. It would be ideal if we could recognize the conditions that would create a duplicate matrix before having to preform the computation.

From what I understand of what Dr. Martone explained to me in one of our previous meetings is that these matrices represent a reflection across one of the edges of our triangles that are formed by our pairs of pants. I am assuming that the tiling process would begin with a single triangle and then continuously reflect across every edge of each triangle that was formed during this process.

I am pretty sure that each word we create is representative of a sort of path formed by the reflections about each edge of these triangles. Many of these paths will end up leading to the same place though. [William] I will provide an illustration of my idea here later. If we could find a more efficient way of tiling the space that minimizes the number of paths that lead to the same place in our tiling, then the computation would hopefully become a lot less resource intensive, which would make it more practical.

I suspect that this could be done by using some sort of graph to map out triangular grid that is formed by our tiling. We could hopefully calculate the size of our graph by finding the maximum "radius" that can be reached by words of length $n$, and then using some algorithm to find the shortest amount of reflections that would be required to reach that tile. This would probably be a good place to implement something like Dijkstra's algorithm which finds the shortest path between nodes in a graph. All of this should be able to be done much more efficiently that just brute forcing thousands of matrix multiplications and then sorting out the duplicates.

**TODO List:**

- ☐ Make sure that I am thinking about these matrices in the right way.
- ☐ If I am thinking about this the right way, figure out how we could efficiently encode this tiling as a graph.
- ☐ I would possibly need to prove that two words that lead to the same triangle actually end up being the same matrix.

## 3. December 14, 2024

3.1. **Meeting Notes.** Dr. Martone seems to think that my graph idea is interesting and it worth looking into further. He was concerned about my graph needing to find the shortest Hilbert distance between nodes; not the shortest number of reflections between nodes. My initial though was that I could weight the edges of he graph in such a way that they mimicked the Hilbert distance. This may still be possible, but it would require more thought to get working properly.

3.2. **Post-meeting Notes.** I have spent some time working on this program. My first step is to find a way of encoding the graph of words up to a maximum length of $N$ as an adjacency matrix that the computer can work with. I made some observations about the number of triangles that are added to the tiling when going from a max of $N$ words to a max of $N+1$ words.

| length of words | number of nodes added | sum total of nodes |
|---|---|---|
| 0 | 1 | 1 |
| 1 | 3 | 4 |
| 2 | 6 | 10 |
| 3 | 9 | 19 |
| 4 | 12 | 31 |

This can be extrapolated into the formula

$$\#\ \text{Nodes}(N) = 1 + \sum_{n=1}^{N} 3n = 1 + \frac{3N(N+1)}{2}$$

From the work that I have done so far, I think that the most difficult challenge that I am facing right now is encoding the tiling that we have as an adjacency matrix that the computer can work with. Once we have that, there are many implementations of Dijkstra's algorithm that can either be borrowed or modified to fit our needs. I am currently doing some experimentation with algorithms that will generate the adjacency matrix for a graph that represents a tiling of words up to length $N$.

## 4. January 9, 2025

## 5. Meeting Notes

During this meeting we discussed potential ways of weighting the graph that we worked on last time. After some thinking, Giuseppe thought of an idea to use the cross ratio to find weights for our graph. I have a drawing of the setup we came up with in Figure 2.

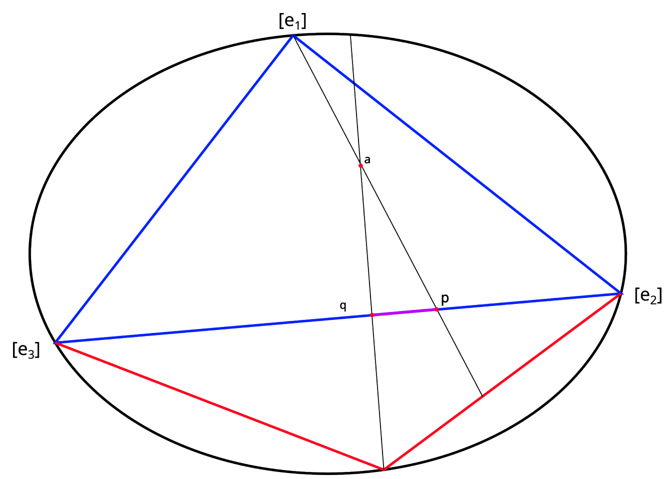


Figure 2. Visualization of method of producing graph weights

## 6. January 14, 2025

6.1. **Meeting Notes.** I had some trouble figuring out the distance discussed in Figure 2. During our meeting today I realized that for some reason I thought that $[e_1], [e_2], [e_3]$ were our reflection matrices, but really they are the standard basis vectors in $\mathbb{R}^3$. After that was sorted, we were able to calculate the length in the case of $R_1$.

**The General Case for Distance.** Since $v_2$ is the plane spanned by $v_1$ and $v_4$, $v_2 = a_1v_1 + a_4v_4$.

Similarly, since $v_3$ is the plane spanned by $v_1$ and $v_4$, $v_3 = b_1v_1 + b_4v_4$.

The distance between $[v_2]$ and $[v_3]$ would be given by

$$d([v_2], [v_3]) = \ln\left(\mathrm{Cr}\left[v_1, v_2, v_3, v_4\right]\right)$$

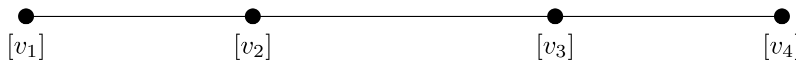


Figure 3. General setup for distance.

Insert picture of tree here

Figure 4. Tree structure created by the reflections of starting triangle.

The cross ratio would be given by

$$\mathrm{Cr}\left[v_1, v_2, v_3, v_4\right] = \frac{b_4}{a_4} \cdot \frac{a_1}{b_1}$$

**Claims:**

- $\mathrm{Cr}\left[v_1, v_2, v_3, v_4\right] > 1$
- It doesn't depend on the choice of coordinates. In other words, if we rescale $v_1, v_2, v_3, v_4$ independently, the cross ratio does not change.
- For any invertible $3\times 3$ matrix $A$, we have $\mathrm{Cr}\left[Av_1, Av_2, Av_3, Av_4\right] = \mathrm{Cr}\left[v_1, v_2, v_3, v_4\right]$.

6.2. **Post-Meeting Notes.** I was able to calculate the other two distances on my own after our meeting last Tuesday. From my calculations, each edge of our graph for $R_1, R_2$, and $R_3$ should be weighted as $\ln\left(\frac{t+2}{2t^2+t}\right)$

## 7. January 17, 2025

7.1. **Meeting Notes.** During this meeting, we discovered that our method of tiling was not accurate to what is actually going on. Instead of a regular graph that can connect back on itself, we really have more of a tree structure.

7.2. **Post-Meeting Notes.**

## 8. January 24, 2025

8.1. **Meeting Notes.**

8.2. **Post-Meeting Notes.**

## 9. Program Explanation

After we discovered that our tiling was more reminiscent of a tree structure, I rewrote the code to generate an adjacency matrix that reflects this tree. At its current stage, my adjacency matrix only represents one of the main branches of the tree and generates $N$ layers of nodes.

After the adjacency matrix is generated, the program encodes the edges of the tree with the reflections that they represent. A 1 is used for $R_1$, a 2 for $R_2$, and a 3 for $R_3$.

I then use Dijkstra's algorithm to find the the path between the root node and any other node in branches below it. My current implementation uses redundant calculations and could be further optimized. The implementation I used for Dijkstra's algorithm can be found at.

After it generates a path from the root node to another node in the tree, my program multiplies the matrices together that would form the path from the root node to the node that we are currently working with. Once this is done, we find the singular values of the matrix and assign the "length" of that matrix to be $l = \ln\left(\frac{\sigma_3}{\sigma_1}\right)$. The program then looks at the value of $l$ and adds a tally to a list that keeps track of how many of these $l$ values fall in the interval $[n, n+1)$.

I made the margins of the paper wider so that it would be easier to read the todo notes on the side.

## 10. Introduction

## 11. (Topology) Surfaces with punctures

A 2-manifold is a topological space that is second countable, Hausdorff, and locally Euclidean of dimension 2.

Quickly recall topological space, second countable and Hausdorff (without definition environment)

In definition environment, define locally Euclidean of dimension $d$

Example, the sphere is locally Euclidean of dimension 2

In definition environment, define surfaces

In an example environment, talk about $S_{0,3}$

In an example environment, discuss $\mathbb{RP}^2$.

- Definition is the space of line passing through the origin in $\mathbb{R}^3$.
- It will be convenient for us to describe $\mathbb{RP}^2$ as a disjoint union of an affine patch and the projection of a plane passing through the origin.
- A *plane* in $\mathbb{R}^3$ is a 2-dimensional vector subspace of $\mathbb{R}^3$. An *affine plane* $A$ is a translate of a plane $P$ by a non-zero vector $v$ not in $P$. Add the description as an affine plane as a set

$$A = \{x \in \mathbb{R}^3 \mid x = v + w, \text{ for some } w \in P\}$$

- Define what $\mathbb{P}(P)$ is (or $\pi(P)$).
- Let $P$ be a plane, and let $A$ be an affine plane which a translate of $P$ by a non-zero vector $v$ not in $P$. Then, in a Lemma environment, $\mathbb{RP}^2 \cong A \sqcup \mathbb{P}(P)$.
- Add a short proof of Lemma.

An example of such a 2-manifold (which will be discussed throughout this work) is the real projective plane, denoted $\mathbb{RP}^2$.

$\mathbb{RP}^2$ is a topological manifold that is constructed by taking a plane $P$ through the origin in $\mathbb{R}^3$ and taking the disjoint union of the set of lines through the origin that are are co-planar with $P$ and the translate of $P$ along a vector through the origin in $\mathbb{R}^3$ that is not in $P$.

NEVER start a sentence with a math symbol

An *affine chart* in $\mathbb{R}^d$ is a translate of a vector subspace $V$ in $\mathbb{R}^d$. So, we can rephrase our definition of $\mathbb{RP}^2$ as the disjoint union of the projection of $P$, $\pi(P)$ and an affine chart $A$.

I think this could be worded better, but I will also add an illustration later to help clarify it.

Why do we say the projection of $P$ to $\mathbb{RP}^2$?

Remove this paragraph for now.

In the space $\mathbb{RP}^2$, the points correspond to lines through the origin in $\mathbb{R}^3$ and any diverging sequence of points in $A$ are said to converge to a point in $P$.

11.1. **Orientable Surfaces.** In topology, surfaces can be classified into two categories: orientable and non-orientable.

Giuseppe: find easiest definition of orientable

The idea behind this classification is the idea that if you were to place an object with a certain chirality on the surface, then every continuous loop that object takes around the surface will result in the object returning to its original position with the same chirality that it started with. If this can be done, the surface is classified as orientable. If there is a continuous loop that this object can take that results in the object returning to its original position, but is now the mirror image of how it started, then the surface is classified as non-orientable.

11.2. **Fundamental Group $S_{0,3}$.** [William] I think we talked about maybe restricting the broadness of our definitions here to specifically focus on the traits that are relevant to our discussion of $S_{0,3}$.

General idea of fundamental group and then specialize to $S_{0,3}$.

- Let $X$ be a topological space, and $p \in X$. Then, the fundamental group of $X$ based at $p$ is. . . .
- Specialize to $S_{0,3}$
- Since $S_{0,3}$ is path-connected, for any $p, q \in S_{0,3}$, there is an isomorphism of groups between the fundamental groups based at $p$ and $q$. Quick reminder: how. Therefore, we can just talk about the fundamental group of $S_{0,3}$ without specifying the base point.
- Lemma: The fundamental group of $S_{0,3}$ is the free group on two generators.
- Proof: Seifert-Van Kampen.

In a path connected topological space, a *loop* is a continuous path inside the topological space where the starting point and the ending point are the same. There are potentially infinitely many loops that start and end at a point $p$ in the topological space, but we can partition the set of loops based at $p$ by grouping together the loops that are homotopy equivalent. We can operate on these equivalence classes of loops by concatenation. Essentially, you can "glue" the end of the first loop to the start of the second loop. This set of equivalence classes paired with the concatenation operation forms a group with the identity being the "constant" loop (i.e. the loop that never leaves the starting point).

Notice that this group of loops depends on the base point $p$, but as it turns out, if you have two points $p$ and $q$ in a path connected topological space $X$, then the groups consisting of the homotopy equivalent loops starting at $p$ and the homotopy equivalent loops starting at $q$ are isomorphic to each other. In other words, the choice of base point isn't really important.

- What's a 2-dimensional manifold.
- Examples: $\mathbb{RP}^2$, affine chart
- Classification of orientable surfaces.
- Fundamental group of $S_{0,3}$.

## 12. The hyperbolic structure on $S_{0,3}$

- This thesis is motivated by the study of geometric structures on surfaces. In this section we discuss the classical example of hyperbolic structures.
- Define $\mathbb{H}^2$ and its metric (using definition environments)
- Define $\mathsf{PSL}_2(\mathbb{R})$ as the group as a group of matrices.
- Show that an element of $\mathsf{PSL}_2(\mathbb{R})$ defines an isometry via Mobius transformations.
- Define hyperbolic structure $=(\mathsf{PSL}_2(\mathbb{R}), \mathbb{H}^2)$-structure on a surface $S$.
- Theorem: whenever the Euler characteristic of $S$ is negative, then it admits at least one hyperbolic structure.
- Recall: $\chi(S) = 2 - 2g - n$ where $g$ is the genus and $n$ is the number of punctures.
- Example: Explain on Friday.

## 13. (Geometry) $(G, X)$ structures and convex real projective structures

Let $X$ be a $n$-manifold that is equipped with a metric d. Since $X$ is a manifold, we know that it is locally Euclidean. This can be thought of as picking any point in our manifold and, if we examine a very local patch around that point, this patch should resemble $\mathbb{R}^n$. We can express idea more explicitly by specifying an atlas $\{(\mathcal{U}_\alpha, \varphi_\alpha)\}_{\alpha \in A}$ where $\mathcal{U}_\alpha$ is one of these local patches (or neighborhoods) in $X$ and $\varphi_\alpha$ is the homeomorphism that maps $\mathcal{U}_\alpha$ to a set $\mathcal{V}_\alpha$ in $R^n$. We will let $G$ be be the set of isometries on $X$. Then, if $\alpha, \beta \in A$, where $A$ is the index set of our atlas, then we have the homeomorphisms

I don't really like how this section is worded, but edits can be made later

$$\varphi_\alpha : \mathcal{U}_\alpha \to \mathcal{V}_\alpha$$
$$\varphi_\beta : \mathcal{U}_\beta \to \mathcal{V}_\beta.$$

We can then create a map $\varphi_\beta \circ \varphi_\alpha^{-1} \colon \mathcal{V}_\alpha \to \mathcal{V}_\beta$. If we restrict this map to $G$, then this structure allows us to determine geometric relationships between different areas of our manifold.

One example of a $(G, X)$ structure is $(\mathsf{SL}_3(\mathbb{R}), \mathbb{RP}^2)$. $\mathsf{SL}_3(\mathbb{R})$ is the set of $3 \times 3$ matrices with real entries and whose determinant is 1. Geometrically, $\mathsf{SL}_3(\mathbb{R})$ can be thought of as the group of linear transformations that preserve volume and orientation of vectors in $\mathbb{RP}^2$. In other words, $\mathsf{SL}_3(\mathbb{R})$ is a group of isometries on $\mathbb{RP}^2$.

- What is a $(G, X)$ structure?
- The example of $(\mathsf{SL}_3(\mathbb{R}), \mathbb{RP}^2)$-structures
- Properly convex domains

If we consider an open set $\Omega \subseteq \mathbb{RP}^2$, $\Omega$ is *proper* if there exists a plane $P \subseteq \mathbb{R}^3$ passing thorugh the origin such that $\overline{\Omega} \cap \pi(P) = \varnothing$. In other words, the closure of $\Omega$, $\overline{\Omega}$ is entirely contained in the affine chart $A$.

A proper set $\Omega \subseteq \mathbb{RP}^2$ is *convex* if for any two points $x, y \in \Omega$, the line $l_{xy}$ passing through $x$ and $y$ intersects $\Omega$ is a connected segment.

Finish adding strictly convex, etc. Example: Klein-Beltrami model of $\mathbb{H}^2$.
The example of hyperbolic structure on $S_{0,3}$ can be seen as a convex real projective structure using the matrices $R_{1,1}$, $R_{2,1}$ and $R_{3,1}$.

## 14. Hilbert length and topological entropy

### 14.1. **Hilbert Length in $\mathbb{H}^2$.**

In a geometric structure, we will need a way of measuring distance. However, since there are many different paths that could be taken from one point to another, it is useful to have a concept of a "shortest" path. We will call the shortest path between any two points a *geodesic*. For example, in a Euclidean space, the geodesic between any two points is a straight line.

Since we are not working in Euclidean space though, we will need a different way of measuring geodesics. In the standard model of a 2-dimensional hyperbolic space, $\mathbb{H}^2$, the "hyperbolic length" is measured by taking a path between two points, represented by the parametric expression $\gamma(t) = (x(t), y(t))$ where $t \in [0, 1]$. The length of this path is measured using

$$l(\gamma) = \int_0^1 \frac{\sqrt{\dot{x}(t)^2 + \dot{y}(t)^2}}{y(t)} \mathsf{d}t$$

From this, the geodesic, also known as the Hilbert length in $\mathbb{H}^2$, is given by

$$d_{\mathbb{H}^2}(z, w) = \inf \{ l(\gamma) \colon \gamma \text{ is a path from } z \text{ to } w \}.$$

Geometrically, this shortest curve is represented as the arc length between two points $z, q \in \mathbb{H}^2$ created by a semi-circle who's endpoints are perpendicular with the $x$ axis. .

if this ends up staying, we can add a picture

**14.2. Hilbert Length in $\mathbb{RP}^2$.** If we have a strictly convex set $\Omega$ in $\mathbb{RP}^2$, we can construct a similar notion of a geodesic by drawing a line through any two points $x, y \in \Omega$ and label the places where the line intersects the boundary of $\Omega$ $a$ and $b$, respectively. The cross ratio $\mathrm{CR}\,[a, x, y, b] \geq 1$ and or geodesic is given by

$$\mathrm{d}(x, y) = \frac{1}{2} \log \mathrm{CR}\,[a, x, y, b]\,.$$

As it turns out, there is exists an isometry between $\mathbb{H}^2$ with the Hilbert length and $(\mathbb{RP}^2, \mathrm{d})$, so it makes sense to also refer to d as the Hilbert length in $\mathbb{RP}^2$.

The isometry is between $\mathbb{H}^2$ and a circle or an ellipse in $\mathbb{RP}^2$ with its Hilbert metric.

I think this is correct, but it sounds a bit weird regardless, in my opinion

## 15. The case of ideal pants groups

Examples: Triangle reflections groups that give you a map from the fundamental group of $S_{0,3}$ to the space of convex real projective structures.

## 16. Manhattan curve

## 17. Results

### 17.1. Main idea of the code.

## 18. Supporting materials: code