Asymptotic Counting in Dynamical Systems

Alan Yan
alanyan2019@gmail.com

3/4/2019

Abstract

We consider several dynamically generated sets with certain measurable properties such as the diameters or angles. We define various counting functions on these geometric objects which quantify these properties and explore the asymptotics of these functions. We conjecture that these functions grow like power functions with exponent the dimension of the residual set. The main objects that we examine are Fatou components of the quadratic family and limit sets of Schottky groups. Finally, we provide heuristic algorithms to compute the counting functions in these examples in an attempt to confirm this conjecture.

Contents

1 Introduction

The limit sets of dynamical systems are varied and generated in unique ways. But, they are almost always associated with clearly recognizable geometric characteristics. For example, in the case of Julia sets of complex valued functions, in particular for those functions whose Julia set is connected, there are clearly defined connected components which can be measured by their areas of diameters. In the case of Cantor sets, the complement consists of a countable union of disjoint intervals which can be measured by their lengths. For arbitrary tilings of the Poincare disk, we can measure each tile by its area or diameter. Along with length, diameters, and areas, we can also look at angles and circumference. In order to quantify the geometry of limit sets, we can define a counting function (usually denoted $N(x)$) which counts the number of components whose "measure" (length, area, angle, etc.) is at least $1/x$. Depending on the definition of $N(x)$, the counting function is usually a monotonically increasing step function as more components are counted as $x \to \infty$. We are interested in the asymptotics of specific counting functions. Our work is motivated by both [1] and [2]. In the former, Alex Kontorovich and Hee Oh consider a counting function on Apollonius circle packings. In particular, for a bounded Apollonius circle packing $\mathcal{P}$, they let $N^{\mathcal{P}}(T)$ denote the number of circles in $\mathcal{P}$ whose curvature is at most T (i.e. whose radius is at least $1/T$). They also proved the following theorem.

Theorem 1.1 (Kontorovich, Oh)**.** *Given a bounded Apollonian circle packing $\mathcal{P}$, there exists $c > 0$ such that as $T \to \infty$,*

$$N^{\mathcal{P}}(T) \sim c \cdot T^{\alpha}$$

where α is the Hausdorff dimension of $\mathcal{P}$.

So, $N^{\mathcal{P}}(T)$ exhibits power-law asymptotics, where the power is the dimension of the residual set. In [2], Mark Pollicott and Mariusz Urbanski consider counting functions in the more general setting of conformal graph directed Markov systems and prove asymptotic

results which include Theorem 1.1, Apollonian triangles, parabolic rational functions, and many more.

In Section 2, we review the definition of the Hausdorff and box-counting dimension. In Section 3, we consider discrete examples of self-similar attractors of iterated function systems and asymptotics on the attractors and we prove that the natural counting functions on these limit sets are asymptotic to $\Theta(x^d)$ where d is the Hausdorff dimension of the residual set. We also prove that under some assumptions about the similarity of a system and the asymptotics of its canonical counting function, the attractor of an iterated function system has Hausdorff dimension equal to the power of its asymptotics. In Section 4 and Section 5, we consider two examples of dynamical systems: Fatou components of the quadratic family and the limit sets of Schottky groups. We define two counting functions for the former: one that counts the number of components with large diameter and one that counts the number of components with large area. For the latter, we define the counting function counting the number of large intervals in the complement. We describe algorithms for generating these sets and approximating these counting functions. We then use linear regression to find an approximation of the power in the functions power-law behavior to verify Theorem 4.2 and Conjecture 5.1.

2 Definitions of Dimension

In this section, we define both the Hausdorff dimension and the box-counting dimension. The former is used for theoretical purposes while the later is used for computational purposes.

2.1 Hausdorff Measure and Hausdorff Dimension

Our definition of Hausdorff measure and Hausdorff dimension depend on the diameters of sets in a metric space. If X is a non-empty subset of a metric space (M, d), then we define the *diameter* of X as

$$\operatorname{diam} X = \sup\{d(x, y) : x, y \in X\}$$

i.e., the greatest distance between two points in our set X. If $\{U_i\}_{i\geq 1}$ is a countable collection of sets of diameter at most δ that cover a set S, that is, $S \subset \bigcup_{n\geq 1} U_n$, then we call $\{U_i\}_{i\geq 1}$ a δ-cover of S. Then, we can define the *d-dimensional Hausdorff measure* of a set S as

$$\mathcal{H}^d(S) = \lim_{\delta \to 0} \mathcal{H}^d_\delta(S)$$

where

$$\mathcal{H}^d_\delta(S) = \inf\left\{\sum_{i=1}^{\infty} (\operatorname{diam} U_i)^d : \{U_i\} \text{ is a } \delta\text{-cover of } S\right\}.$$

In other words, we try to "pack" X most efficiently using sets of diameter at most δ and take the sum of $(\operatorname{diam} U_i)^d$ as our d-dimensional measure. As δ decreases, the number of valid coverings of S reduces hence the infimum increases, approaching a limit as $\delta \to 0$. Observe that if $a > b$, and $\{U_i\}_{i\geq 1}$ is a δ-cover of our set, then

$$\sum_i (\operatorname{diam} U_i)^a = \sum_i (\operatorname{diam} U_i)^{a-b} (\operatorname{diam} U_i)^b \leq \delta^{a-b} \sum_i (\operatorname{diam} U_i)^b.$$

Taking the infimum, we get that $\mathcal{H}^a_\delta(S) \leq \delta^{a-b}\mathcal{H}^b_\delta(S)$. Taking $\delta \to 0$, we see that if $\mathcal{H}^b(F)$ is finite, then $\mathcal{H}^a(F) = 0$. So, there is a value d at which $\mathcal{H}^d(F)$ jumps from 0 to ∞. We define this value to be the *Hausdorff dimension* of F and we denote it by $\dim_H F$.

2.2 Box-Counting Dimension

Most of the time, the Hausdorff dimension, as it is defined in the previous section, is too difficult to compute by hand, especially if the ground set is difficult to describe. For computational purposes, it suffices to use the *box-counting dimension*, also known as the Minkowski-Bouligand dimension, of sets in Euclidean space.

Definition 2.1 (Box-counting Dimension)**.** Let $S \subset \mathbb{R}^n$ be fixed. Suppose that $N(\varepsilon)$ is the number of balls of radius ε required to cover a set S. Then the **box-counting dimension** of S is defined as

$$\lim_{\varepsilon \to 0} \frac{\log N(\varepsilon)}{\log(1/\varepsilon)}$$

provided that this limit exists. We denote it by $\dim_B S$.

Clearly the box-counting dimension is an upper bound for the Hausdorff dimension because it simply specifies a way to cover our set. For many regularly shaped sets, the box-counting dimension and the Hausdorff dimension is the same. However, the box-counting dimension may not exist for some sets while the Hausdorff dimension always exists. Notice that the box-counting dimension is independent of the norm imposed on the space since all norms are equivalent. So for computational purposes, it is best to take $\|\cdot\|_\infty$ and cover our set with boxes since our set will be represented with square pixels. Thus, the box-counting dimension can be approximated by taking several values of ϵ and computing $N(\epsilon)$. Then, the dimension can be found by a simple linear regression. An equivalent definition of the box-counting dimension is the following.

Definition 2.2. The box-counting dimension is the unique number d such that $N(r)$ grows as $1/r^d$ as r approaches zero.

3 Some Discrete Examples

The underlying conjecture that counting functions on dynamically generated sets are asymptotic to power functions with respect to the Hausdorff dimension of the residual set can be directly proven for more discrete fractals. In this section, we consider examples of iterated function systems and prove the counting function conjecture on the attractors of these systems. All of our fractals in this section are generated by similarity maps.

Definition 3.1. Let (X, d) be a metric space. A function $f : X \to X$ is called a **similarity** if for all $x, y \in X$, we have that $d(f(x), f(y)) = cd(x, y)$ for some non-negative constant $c \in \mathbb{R}$.

The following theorem shows that it is easy to compute the Hausdorff dimension for similarity maps whose images do not overlap too much. The proof can be found in [3].

Theorem 3.1. *Let $\mathcal{S} = \{S_1, ..., S_m\}$ be a set of similarities on $\mathbb{R}^n$ with ratios $0 < c_i < 1$ for $1 \leq i \leq m$. Suppose that there is a non-empty bounded open set $V \subset \mathbb{R}^n$ such that*

$$V \supset \bigcup_{i=1}^{m} S_i(V),$$

where the union is disjoint. If F is the attractor of $\mathcal{S}$, then

$$\dim_H F = \dim_B F = s,$$

where s is uniquely determined by

$$\sum_{i=1}^{m} c_i^s = 1.$$

All dimension calculations for the following examples follow from Theorem 3.1.

Example 3.1 (Sierpinski Triangle)**.** *Consider the Sierpinski triangle $S \subset \mathbb{R}^2$, the attractor for the iterated function systems defined by the contraction mappings*

$$\begin{cases} f_1(x,y) &= \left(\frac{x}{2}, \frac{y}{2}\right) \\ f_2(x,y) &= \left(\frac{x}{2}, \frac{y}{2}\right) + \left(\frac{1}{2}, 0\right) \\ f_3(x,y) &= \left(\frac{x}{2}, \frac{y}{2}\right) + \left(\frac{1}{4}, \frac{\sqrt{3}}{4}\right). \end{cases}$$

For simplicity, we can let the ground set for our IFS to be the equilateral triangle with vertices at $(0,0), (1,0), (1/2, \frac{\sqrt{3}}{2})$. The set S can also be defined recursively by taking an equilateral triangle, removing the center triangle, and then repeating the process for the other three triangles. For this case, we can easily construct counting functions for the triangles leftover on the limit set. Let $N(a)$ be the number of triangles removed in the construction of S whose area is at least $1/a^2$. Then, $N(a) = \Theta(a^d)$ where $d = \dim_H(S) = \ln 3 / \ln 2$.

Proof. At the kth level of the recursion, 3^{k-1} equilateral triangles of side length 2^{-k} (area $c^2(2^{-k})^2$ for $c^2 = \sqrt{3}/4$) are removed. So, if k is the largest number such that $c^2(2^{-k})^2 \geq \frac{1}{a^2}$, then

$$N(a) = \sum_{i=1}^{k} 3^{i-1} = \frac{1}{2}(3^k - 1).$$

In this case, $k = \log_3(a)/\log_3 2 + O(1)$, which gives us the desired asymptotics. □

Example 3.2 (Sierpinski Carpet)**.** *Consider a unit square. Divide the square into 9 squares in the natural way and remove the middle square. Now repeat this procedure on the rest of the squares. The limit set is called the Sierpinski carpet, which we denote by C. Let $N(x)$ denote the number of removed squares whose side length is at least $1/x$. Then, $N(x) = \Theta(x^d)$ where $d = \dim_H C = \ln 8 / \ln 3$.*

Proof. At the kth level of the recursion, 8^{k-1} squares of side length 3^{-k} are removed. So, following the previous example, $N(x) = \frac{1}{7}(8^k - 1)$ where $k = \log_8(x)/\log_8(3) + O(1)$ which gives the desired asymptotics. □

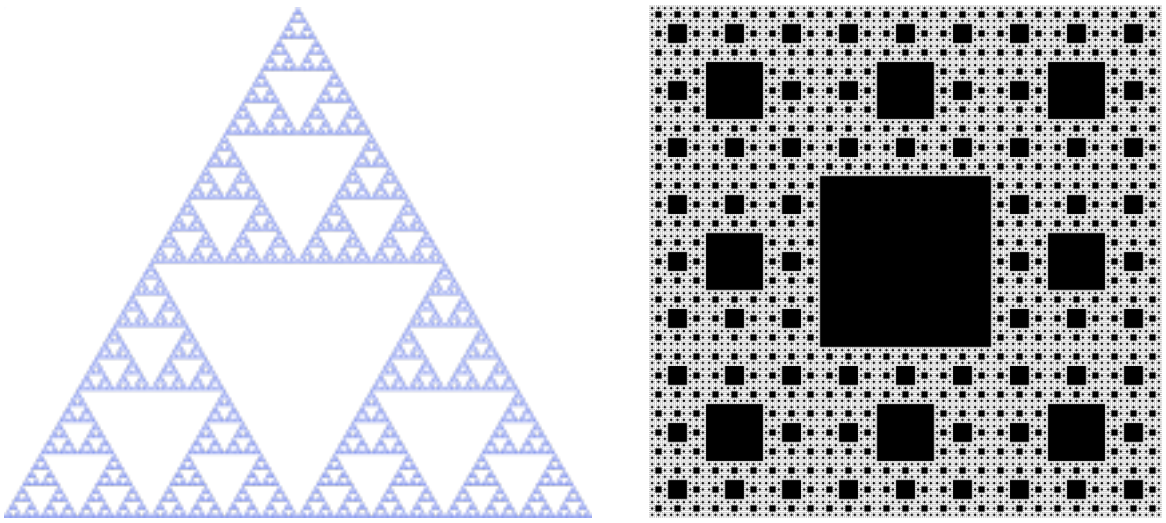

Figure 1: Sierpinski Triangle and Sierpinski Carpet (Wikipedia)

Example 3.3 (Generalized Cantor Sets)**.** *Consider a unit interval. Fix* $0 < a \leq 1/3$*. Construct the set* C_a *by removed the middle* a *from each segment, and then repeating this process on the remaining segments. This is also the limit set of the IFS defined by*

$$\begin{cases} f_1(x) &= \frac{(1-a)}{2}x \\ f_2(x) &= f_1(x) + \frac{1+a}{2}. \end{cases}$$

Let $N(l)$ *be the number of intervals removed with length at least* $1/l$*. Then* $N(l) = \Theta(l^d)$ *where* $d = \ln 2/\ln(1/a)$*.*

Proof. At the kth level of the recursion, 2^{k-1} intervals of length a^k are removed. So, the largest interval greater than $1/l$ will have length a^k where $k = \ln l/\ln(1/a) + O(1)$. So, we have that $N(l) = \sum_{t=1}^{k} 2^{t-1} = 2^k - 1 = \Theta(x^d)$ which suffices for the proof. □

Observe that in Example 3.3, the value of d is not equal to the Hausdorff dimension of the set. This could have been caused by the counting function being primarily defined on the complement of the set, rather than the set itself.

Example 3.4 (von Koch curve)**.** *Consider a unit interval. Fix* $0 < a \leq \frac{1}{3}$ *and construct a curve* F *by repeatedly replacing the middle proportion* a *of each segment with two sides of an equilateral triangle such that the whole set moves outwards. In other words,* F *is the attractor of the iterated function system defined by*

$$\begin{cases} f_1(x,y) &= \left(\frac{(1-a)}{2}x, \frac{(1-a)}{2}y\right) \\ f_2(x,y) &= \left(\frac{(1-a)}{2}x + \frac{1+a}{2}, \frac{(1-a)}{2}y\right) \\ f_3(x,y) &= a\left(\frac{x}{2} - \frac{\sqrt{3}}{2}y, \frac{\sqrt{3}}{2}x + \frac{y}{2}\right) + \left(\frac{1-a}{2}, 0\right) \\ f_4(x,y) &= a\left(-\frac{x}{2} - \frac{\sqrt{3}}{2}y, \frac{\sqrt{3}}{2}x - \frac{y}{2}\right) + \left(\frac{1+a}{2}, 0\right). \end{cases}$$

Let $a = 1/3$ and let $N(x)$ be the number of equilateral triangles whose side length is at least $1/x$. We can similarly show that $N(x) = \Theta(x^d)$ where $d = \ln 4/\ln 3$, the Hausdorff dimension of the curve.

Under some assumptions, we can prove something more general. Let F be the attractor of the iterated function system of similarities $\mathcal{S} = \{S_1, S_2, ..., S_n\}$ with constants $0 < c_i < 1$ for all $1 \leq i \leq n$ and suppose that this system satisfies the conditions of Theorem 3.1. Let $N(F, x)$ be a counting function which takes in a set F and a positive real number x. Suppose that $N(S_i(F), x) = N(F, c_i x)$ for all sets F and $1 \leq i \leq n$ and $N(A \cup B, x) = N(A, x) + N(B, x)$ for disjoint sets A and B which are equivalent to F through rigid motion. Moreover, suppose that $N(x) = N(F, x) \sim cx^d$ for some constant c independent of F. Then, we have the following theorem.

Theorem 3.2. *$d = \dim_H F$.*

Proof. By the hypothesis of Theorem 3.1, we have that

$$F = \bigcup_{i=1}^{n} S_i(F)$$

So,

$$N(F, x) = \sum_{i=1}^{n} N(S_i(F), x) = \sum_{i=1}^{n} N(F, c_i x).$$

Then,

$$1 = \lim_{x\to\infty} \sum_{i=1}^{n} \frac{N(F, c_i x)}{N(F, x)} = \sum_{i=1}^{n} \lim_{x\to\infty} \frac{N(F, c_i x)}{N(F, x)} = \sum_{i=1}^{\infty} \lim_{x\to n} \frac{c(c_i x)^d}{cx^d} = \sum_{i=1}^{n} c_i^d.$$

From Theorem 3.1, we have that $d = \dim_H F$, as claimed. □

Remark. The asymptotics of the discrete examples considered above were of the form $\epsilon(x)x^d$ where $\epsilon(x)$ oscillates between two constants. Thus, an estimate of the form cx^d does not exist. However, as seen in [2], the counting functions of limit sets which arise through group theoretic/topological means seem to have exact power-law estimates.

4 Dynamics of the Fatou Components of the Quadratic Family

4.1 Background

For the following definitions, $f : \mathbb{C} \to \mathbb{C}$ is a complex-valued function. We only consider the case where $f \in Q$, where Q is the quadratic family, $\{f_c : \mathbb{C} \to \mathbb{C} : f_c(z) = z^2 + c, c \in \mathbb{C}\}$. Although this may seem like a limited set of functions, it can be shown that every quadratic polynomial is conjugate to a unique member of Q. Hence, the dynamics of Q explains the dynamics for every quadratic polynomial. For any complex function f (not necessarily in Q), we can define the following sets:

Definition 4.1. The **filled-in Julia set** of f is defined as

$$K(f) = \{z \in \mathbb{C} : f^k(x) \not\to \infty\}$$

the set of complex numbers which remain bounded under repeated iteration of f.

Definition 4.2. The **Julia set** of f is defined as the boundary of the filled-in Julia set, i.e.

$$J(f) = \partial K(f).$$

Definition 4.3. The **Fatou set** of f is defined as the complement of the Julia set, i.e.

$$F(f) = \mathbb{C} \backslash J(f).$$

Moreover, the connected components of $F(f)$ are called *Fatou components*.

The simplest Julia set that we can consider is that of $f(z) = z^2$. For every $z_0 \in \mathbb{C}$, $|f(z_0)^n| = |z_0^{2^n}| = |z_0|^{2^n}$ which is unbounded whenever $|z_0| > 1$ and bounded whenever $|z_0| \leq 1$. So, $J(f) = S^1$. However, for other constants c, the Julia set is not so easily determined. For example, the filled-Julia sets for $f(z) = z^2 - 4/9$ and $f(z) = z^2 - 0.123 + 0.745i$ are shown in Figure 2 (generated on a 500×500 grid with the algorithm described in Section 4.3) and the boundary cannot be described with simple parametric equations.

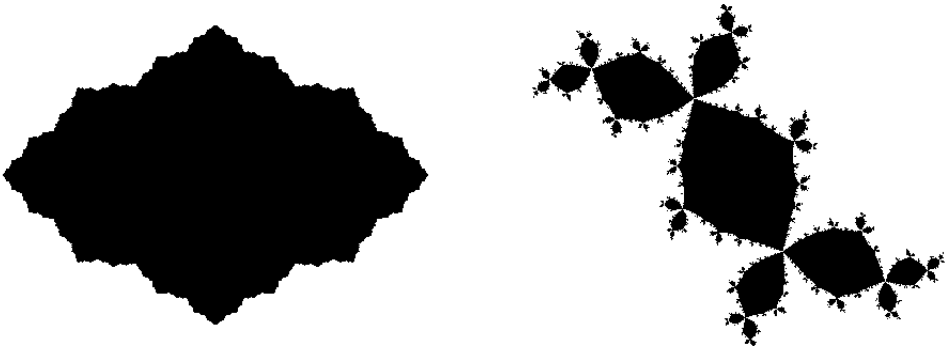

Figure 2: filled-Julia sets for $f(z) = z^2 - 4/9$ and $f(z) = z^2 - 0.123 + 0.745i$

4.2 Counting Function on Fatou Components

For any $c \in \mathbb{C}$, $J(f_c)$ is either connected or totally disconnected by the Fundamental Dichotomy Theorem. The Mandelbrot set $\mathcal{M}$ is defined to be the set of $c \in \mathbb{C}$ for which $f_c^n(0)$ is bounded for all n. The following theorem shows the relationship between connectivity and the Mandelbrot set.

Theorem 4.1 (Mandelbrot's Criterion)**.** *The set $K(f_c)$ is connected if and only if $c \in \mathcal{M}$.*

For $c \notin \mathcal{M}$, $J(f_c)$ is totally disconnected and so it is difficult to discern what counting function can be assigned to the set. However, for $c \in \mathcal{M}$, $K(f_c)$ is connected, so we can define a counting function which counts the number of connected components with diameter at least $1/x$. In particular, we have the following definition.

Definition 4.4. For any function $f : \mathbb{C} \to \mathbb{C}$, we associate the counting function $N(x)$ as the number of Fatou components of f with diameter at least $1/x$.

Motivated by Theorem 1.1, we have the following theorem.

Theorem 4.2. *For $\lambda \in \mathcal{M}$, let $f(z) = z^2 + \lambda$. Then*

$$N(x) \sim cx^d$$

where c is some constant and $d = \dim_H(J(f))$.

Remark. The theorem is a corollary of the theorems proven in [2].

4.3 Algorithm to Compute the Counting Function

In this section, we provide an algorithm to numerically verify the Theorem 4.2 by finding small values of the counting function. Our algorithm does the following:

1. Display the filled-in Julia set of $f(z) = z^2 + c$.
2. Identify the different components.
3. Calculate the diameter of each component.

4.3.1 Step 1: Constructing the Julia Set

There are many well-known algorithms to construct the Julia set of complex-valued functions. In this section, we just present one construction that relies on the following theorem.

Theorem 4.3 (The Escape Criterion)**.** *Let $f(z) = z^2 + c$ be a complex quadratic function. If $|z_0| > 2 \geq |c|$, then $|f^n(z_0)| \to \infty$ as $n \to \infty$. In particular, $z_0 \notin K(f)$.*

Proof. We have that

$$|f(z_0)| = |z_0^2 + c| \geq |z_0|^2 - |c| \geq |z_0|^2 - |z_0| \geq |z_0|(|z_0| - 1) \geq (1 + \lambda)|z_0|$$

for some $\lambda > 0$. Repeating this argument, $|f^n(z_0)| \geq (1 + \lambda)^n |z_0|$, hence $|f^n(z_0)| \to \infty$. □

Using Theorem 4.3, we have the following simple algorithm to constructing the filled-in Julia set.

1. We represent every pixel as a discrete complex number.
2. For every pixel, we iterate its associated complex number N times under the function. (N can be set to be 1000).
3. If the value ever gets above $\max\{|c|, 2\}$, we mark the pixel white, signifying that it is not in the set.
4. If after N iterations, it is still less than $\max\{|c|, 2\}$, then we mark it black, signifying that it is in the filled-in Julia set.

For $c = -1$, we get the filled-Julia set as shown in Figure 3.

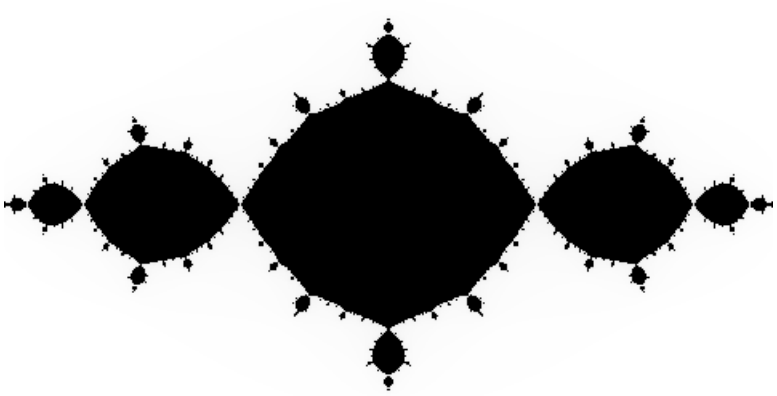

Figure 3: filled-Julia set of $f(z) = z^2 - 1$ on a 500×500 pixel grid

4.3.2 Step 2: Identify Different Components

After displaying the whole filled-in Julia set, we have a grid of black and white pixels representing complex numbers where a pixel is black if it is in the $K(f)$ and white if it is not in $K(f)$. We can consider the set as a subset of $\mathbb{Z}^2$, and we can heuristically model the connectivity of the components by using a graph $G(V, E)$ where V is the subset of $\mathbb{Z}^2$ which is black and $xy \in E$ if and only if $x - y \in \{(\pm 1, 0), (0, \pm 1)\}$. Then the different components can be computed simply as connected components of a graph.

We use a disjoint union set data structure in the standard way in order to distinguish the different components. First, we define the functions $\texttt{parent} : V \to V$ and $\texttt{rank} : V \to \mathbb{Z}_{\geq 0}$ by first initializing $\texttt{parent}(v) = v$ and $\texttt{rank}(v) = 0$ for all $v \in V$. This can be visualized as a digraph $D(V, E)$ where the only edges are ones which are directed from v to $\texttt{parent}(v)$ for all $v \in V$. Define the function $\texttt{findSet} : V \to V$ as

$$\texttt{findSet}(v) = \begin{cases} \texttt{findSet}(\texttt{parent}(v)) & \text{if } \texttt{parent}(v) \neq v \\ v & \text{otherwise .} \end{cases}$$

So, to compute the $\texttt{findSet}(v)$ of a $v \in V$, start at v and keep walking along the directed edges (there is a unique way to do this) until a loop is reached. That vertex is the value of $\texttt{findSet}(v)$. We now define the function `merge` which takes values on $V \times V$. Suppose we call $\texttt{merge}(v, w)$. Then we first compute $v_p = \texttt{parent}(v), w_p = \texttt{parent}(w)$. If $v_p = w_p$, then we do nothing. Otherwise, v_p and w_p are distinct. If $\texttt{rank}(v_p) = \texttt{rank}(w_p)$, then we set $\texttt{parent}(w_p) = v_p$ and increase $\texttt{rank}(v_p)$ by 1. Otherwise, without loss of generality, $\texttt{rank}(v_p) > \texttt{rank}(w_p)$. Then, set $\texttt{parent}(w_p) = v_p$. So, after any sequence of `merge`s, we have a forest of directed binary trees. Each tree can be considered a connected component. So, if for all $xy \in E$, we call $\texttt{merge}(x, y)$, then the forest we are left with are exactly the connected components. To store the components for further use, we simply need to call `findSet` on every single vertex and store the vertices with the same `findSet` in its own list.

Each list contains one component of our graph and we can do computations on each list for the next step.

In this algorithm, we used two heuristics. The first is our use of rank when merging. This ensures that the trees do not get too tall and `findSet` does not get too slow. The second heuristic we use (which is not explained above) is called *path compression*. Using this heuristic, we can improve the time complexities of our functions to $O(\alpha(n))$ where $\alpha(n)$ is the inverse-Ackermann function (in practice it is $O(1)$). A detailed treatment of path compression can be found in [4].

We still need another heuristic. Due to imprecisions in the generation of the filled-in Julia set, certain pixels may act as a "bridge" between two adjacent components. Hence, running the above algorithm immediately after generation results in one single connected component. To fix this, we can first remove all pixels which have a pair of opposite white pixels which are adjacent to it. After removing these pixels, we can proceed with our algorithm.

Figure 4: Example of a Bridge

4.3.3 Step 3: Finding the Diameter of a Component

To compute the diameter of a Fatou component, it suffices to write an function which takes in any (finite) subset of $\mathbb{R}^2$ and outputs the diameter of the set. Since our set is finite, we can complete this task by iterating over all pairs of points in our set, compute the distance, and simply pick the largest length distance that we compute. However, this runs the risk of being too slow. The brute force algorithm described runs in $O(n^2)$ time. To have an accurate representation of the Julia set, it is necessary to represent the set with larger and larger grids. As the grid increases linearly, the number of points in a specific component increases quadratically. So, the algorithm becomes very slow quickly. As a point of reference, this algorithm takes approximately n^2 operations when the size of our set is n. If $n = 10^8$, then it would take approximately $1/3$ of a year for a fast computer to complete the algorithm.

We now present an improvement from $O(n^2)$ to $O(n \log n)$ as proved in [5]. It relies on the following lemma about the convex hull of our set.

Lemma 4.4. *Given a finite set of points S, the diameter of S is equal to the diameter of its convex hull.*

Proof. There is a set $S' \subset S$ such that S' is convex and S is contained in the convex hull of S'. Clearly, $\operatorname{diam} S' \leq \operatorname{diam} S$. To prove the reverse inequality, consider two arbitrary points $a, b \in S$ such that $d(a, b) = \operatorname{diam} S$. Let the line through a and b intersect the convex hull at points a', b'. Then, a' and b' can be either a point on a segment between two members of S' or a member of S'. If one of the points is on a segment, we can always make the distance between the points bigger by moving the point to one of the adjacent elements of S'. Indeed, when the distance between a' and b' is considered as a real-valued function on points on the segment, it is a convex function. Hence, it achieves its maximum on the boundary. Thus, we can increase the distance by moving both points to elements of S'. We have

$$\operatorname{diam} S = d(a, b) \leq d(a', b') \leq \operatorname{diam} S'.$$

So, $\operatorname{diam} S = \operatorname{diam} S'$, as claimed. □

So, it suffices to first compute the convex hull of a set of points, and then compute the diameter of the convex hull. There are many algorithms to quickly compute the convex hull. In particular, the Graham Scan can compute the convex hull in $O(n \log n)$ time. Moreover, the diameter of the convex hull can be computed in linear time. Theoretically, for our purposes of calculating the diameter of components, this should greatly improve the speed of our algorithm. Components are "dense", and so the convex hull should have significantly less points than the actual set itself. In fact, if we approximate each component as a circle, the amount of points we consider is about $\sqrt{n}$ where n is the amount of points in our original set. Lemma 4.4 allows us to just find the diameter of the convex hull. We can find the diameter of a convex polygon in linear time as shown in [5]. Furthermore, we can find the convex hull in $O(n \log n)$ time by using Graham scan algorithm. Hence, the whole process can be done in $O(n \log n)$ time for a single component. We now show that the diameter of the convex hull can be computed in linear time. The correctness of the algorithm is proven in [5].

Proposition 4.1. *Let $S \subset \mathbb{R}^2$ be a finite set of points. Then,* $\operatorname{diam}(S)$ *can be computed in $O(|S| \log |S|)$ time.*

Proof. Let $n = |S|$. The convex hull can be computed in $O(n \log n)$ time. Let the convex hull be $\mathcal{C} = \{p_1, p_2, ..., p_m\}$ where the vertices are in counter-clockwise order. If e is an edge of $\mathcal{C}$, then we denote l_e the line through e. Let L be the set of all pairs $(p, q) \in \mathcal{C}^2$ for which there is an edge e having p as an endpoint such that $d(q, l_e) = \max\{d(r, l_e) : r \in \mathcal{C}\}$. The following algorithm computes the diameter of a convex polygon in linear time:

(i) (Construction of L) For every edge, there are at most 2 points whose distance to the edge is maximum since the list of distances of the vertices (taken in counter-clockwise order) from a fixed edge is uni-modal. Moreover, since no three vertices are collinear,

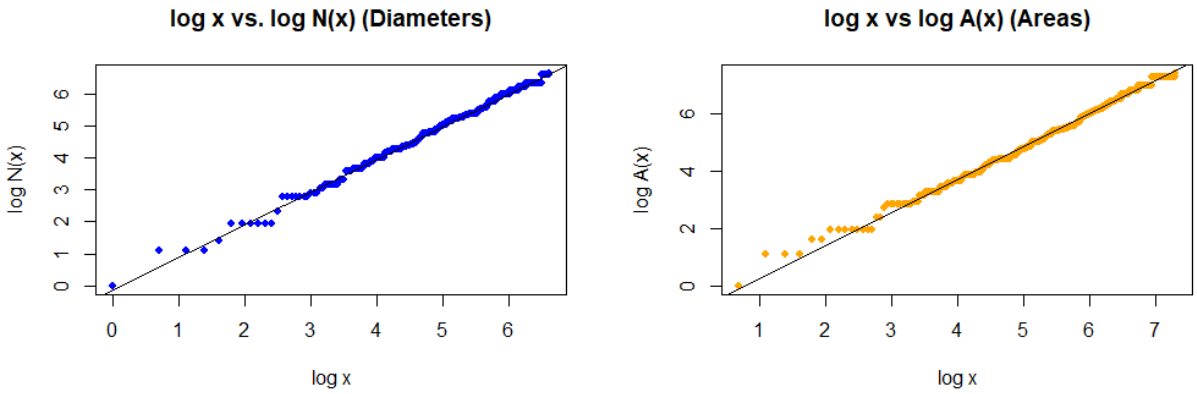

Figure 5: $\log x$ vs $\log N(x)$ and $\log A(x)$

this ensures that there is at most 2 vertices which achieve the maximum. So every edge contributes at most 4 tuples to L. To construct L, we iterate through the edges, find the two points at a maximal distance from this edge, and then put the corresponding four tuples into L. First, let e be the edge between p_1 and p_2. Then, we can compute the distances starting from p_3 and going counter-clockwise until the distance stops increasing. When we stop, if the distance decreases, then the vertex which achieves the greater distance achieves the maximum distance from the edge. If the distance stays the same, the two vertices achieve the maximum distance from the edge. In either case, we append the corresponding tuples to L. We could start the algorithm over with $e = p_2p_3$, but instead we just move our edge over, and continue computing distances from where we left off. We continue the process until we have iterated through all of the edges. Since we walked around the polygon at most 2 times, constructing L takes $O(n)$ time.

(ii) (Compute the distances between each tuple in L) It can be shown that the maximum distance of a tuple in L is equal the diameter of the polygon. It can also be shown that $|L| = O(n)$, so computing the maximum distance takes $O(n)$ time.

□

4.4 Approximating the Exponent

After computing the first few values of $N(x)$, we can run a linear regression on $\log N(x)$ and $\log x$ in order to estimate the exponent. Figure 5 shows scatter-plots of $\log x \sim \log N(x)$ and $\log x \sim \log A(x)$ where $N(x)$ is the aforementioned counting function and $A(x)$ is the modified counting function which counts the number of components whose area is at least $1/x^2$.

In the $\log x \sim \log N(x)$ plot, the line of best fit was $y = 1.022x - 0.122$ and in the $\log x \sim \log A(x)$, the line of best fit was $y = 1.1499x - 0.8959$. From [6], $\dim_H(J(f_{-1})) \approx 1.2683$.

5 Limit Sets of Schottky Groups

5.1 Background

We now digress from the dynamics of Julia sets and now consider dynamics on groups actions on the hyperbolic plane. In particular, we consider Schottky groups, subgroups of the isometry group of hyperbolic space, and their limit sets. Before giving the definition of a Schottky group, we present a quick review of Hyperbolic geometry and its models. In Section 5.1.1, we consider the geometry of the Poincare disc model and the Poincare half-plane model. In Section 5.1.2, we classify isometries which help us simplify calculations. In Section 5.1.3, we provide a short introduction to the notion of a limit set.

5.1.1 Poincare half plane/disc, geodesics, angles, and metrics

Definition 5.1. The **Poincare half-plane** is defined as $\mathbb{H} = \{z \in \mathbb{C} : \mathrm{Re}(z) > 0\}$ and the unit disc is defined as $\mathbb{D} = \{z \in \mathbb{C} : |z| < 1\}$. These two spaces along with their closures are the ambient spaces for the Poincare half-plane model and the Poincare disk model, respectively.

Definition 5.2. Let (X, d) be a metric space. $\varphi : X \to X$ is called an **isometry** if and only if $d(\varphi(x), \varphi(y)) = d(x, y)$ for all $x, y \in X$.

In hyperbolic space, we are interested in the various isometry groups. For a set U, we let $\psi \in \mathrm{Conf}(U)$ if and only if $\psi : U \to U$ is a conformal diffeomorphism. Then,

$$G = \left\{ h_{\alpha,\beta}(z) = \frac{\alpha z + \beta}{\overline{\beta} z + \overline{\alpha}} : |\alpha|^2 - |\beta|^2 = 1 \right\} \subseteq \mathrm{Conf}(\mathbb{D}).$$

In fact, using a bit of complex analysis we can prove the following proposition.

Proposition 5.1. $G = \mathrm{Conf}(\mathbb{D})$.

Proof. See Proposition 1.1 in [7] □

Since $\Psi : \mathbb{H} \to \mathbb{D}$ defined by $\Psi(z) = i(z-i)/(z+i)$ is a holomorphic diffeomorphism, we have that $\mathrm{Conf}(\mathbb{H}) = \Psi^{-1} \mathrm{Conf}(\mathbb{D}) \Psi$ and

$$\mathrm{Conf}(\mathbb{H}) = \Psi^{-1} \mathrm{Conf}(\mathbb{D}) \Psi = \Psi^{-1} G \Psi = \left\{ h_{a,b,c,d}(z) = \frac{az+b}{cz+d} : \begin{bmatrix} a & b \\ c & d \end{bmatrix} \in \mathrm{SL}_2(\mathbb{R}) \right\} \cong \mathrm{PSL}_2(\mathbb{R}).$$

The the dynamics of $\mathbb{H}$ and $\mathbb{D}$ under respective groups of transformations are the same and we will switch between the two whenever convenient. Now consider $h \in \mathrm{Conf}(\mathbb{H})$, $h(z) = (az+b)/(cz+d)$. Then for any two vectors u, v, we have that

$$\frac{1}{\mathrm{Im}\, h(z)^2} \langle T_z h(u), T_z h(v) \rangle = \frac{1}{\mathrm{Im}\, z^2} \langle u, v \rangle.$$

So, by constructing the metric $g_z(u,v) = (\operatorname{Im} z)^{-2}\langle u, v\rangle$, we have turned $\mathrm{Conf}(\mathbb{D})$ and $\mathrm{Conf}(\mathbb{H})$ into isometries of their respective ambient spaces. Then, $\mathbb{H}$ turns into a metric space where for any $x, y \in \mathbb{H}$, we have that

$$d(x,y) = \inf_{\gamma} \int_{\gamma} \frac{|dz|}{y}$$

where the infimum is over all continuous paths from x to y and the integral is the length of γ. In $\mathbb{D}$, we calculate distances with the formula

$$d(x,y) = \inf_{\gamma} \int_{\gamma} \frac{2|dz|}{1-|z|^2}.$$

By construction, if $\phi \in \mathrm{Conf}(S)$ where $S = \mathbb{H}$ or $\mathbb{D}$, we have that $d_S(\phi(x), \phi(y)) = d_S(x,y)$. Moreover, we have the following proposition about the geodesics and the angles between them in $\mathbb{H}$ and $\mathbb{D}$.

Proposition 5.2. *The geodesic between $z, w \in \mathbb{H}$ is the arc between z and w of the unique circle which passes through z, w and whose diameter lies on the real line. If z, w have the same real value, then the geodesic is the segment between them. Moreover, because Ψ maps $\mathbb{H}$ into $\mathbb{D}$ and preserves angles, if $z, w \in \mathbb{D}$, then the geodesic is the arc of the unique circle containing z, w which is orthogonal to $\partial\mathbb{D}$. The angle between two geodesics is the Euclidean angle formed by the tangent vectors to the geodesics at the intersection point.*

More practically, we have the following formulas

Theorem 5.1. *If $z, w \in \mathbb{H}$ then,*

$$d(z,w) = \log \frac{|z-\overline{w}| + |z-w|}{|z-\overline{w}| - |z-w|}.$$

When $z, w \in \mathbb{D}$, then

$$d(z,w) = \log \frac{|1-z\overline{w}| + |z-w|}{|1-z\overline{w}| - |z-w|}.$$

Proof. See Theorem 7.2.1 in [8] □

Remark. Observe that in the topology induced by our metric, $\mathbb{H}$ is not compact. To make it compact, we can add in the real line and also ∞. Thus, $\mathbb{H}(\infty) = \mathbb{R} \cup \{\infty\}$ is the boundary at infinity. Analogously, the boundary at infinity for $\mathbb{D}$ would be the unit circle.

5.1.2 Classification of Positive Isometries

The group G consists of orientation-preserving isometries. Moreover, G is exactly the group of positive isometries. We can decompose G into three subgroups:

$$\begin{aligned} K &= \left\{ r(z) = \frac{z\cos\theta - \sin\theta}{z\sin\theta + \cos\theta} : \theta \in \mathbb{R} \right\} \\ A &= \{h(z) = az : a > 0\} \\ N &= \{t(z) = z + b : b \in \mathbb{R}\}. \end{aligned}$$

It is easy to show the following for $g \in G$:

(i) $g \in K$ if and only if $g(i) = i$.

(ii) $g \in A$ if and only if $g(0) = 0$ and $g(\infty) = \infty$.

(iii) $g \in N$ if and only if $g(\infty) = \infty$ and there are not other fixed points.

K, A, and N partition G in the sense that every map in G is conjugate in G to a unique map in either K, A, or N. That is, it exhibits similar dynamics to maps in one of these subgroups. Any map in G can have at most 2 fixed points. Moreover, K, A, N are characterized by the type of fixed points that they have which is explained by the following proposition.

Proposition 5.3. *Let $g \in G - \{\mathrm{id}\}$.*

(i) If g fixes exactly two points in $\mathbb{H}(\infty)$, both of which are in $\mathbb{H}(\infty)$ then g is conjugate in G to an element of A. In this case, g is said to be hyperbolic.

(ii) If g fixes exactly one point of $\overline{\mathbb{H}}$ which is in $\mathbb{H}$, then g is conjugate in G to an element of K. In this case, g is said to be elliptic.

(iii) If g fixes exactly one point of $\overline{\mathbb{H}}$ which is in $\mathbb{H}(\infty)$, then g is conjugate in G to an element of N. In this case, g is said to be parabolic.

Proof. (i) Suppose $g(f_1) = f_1, g(f_2) = f_2$ for $f_1, f_2 \in \mathbb{H}(\infty)$. Without loss of generality, $f_1 \neq 0$. We can impose this condition because f_1 and f_2 are distinct. Define

$$\varphi_1(z) = \frac{z/f_1}{-z + f_1}$$
$$\varphi_2(z) = z - \frac{f_2}{f_1(f_1 - f_2)}$$

Now observe that $h = (\varphi_2 \circ \varphi_1) \circ g \circ (\varphi_2 \circ \varphi_1)^{-1}$ satisfies $h(0) = 0$ and $h(\infty) = \infty$. So $h \in A$.

(ii) Suppose that $g(f_1) = f_1$ and $f_1 \in \mathbb{H}$. Let $\varphi(z) = (\operatorname{Im} f)z + (\operatorname{Re} f)$. Then $\varphi^{-1} \circ g \circ \varphi$ fixes i. So, $g \in K$.

(iii) Define

$$\varphi(z) = \frac{z/f}{-z + f}$$
$$\varphi_0(z) = \frac{z/i}{-z + i}$$
$$\psi = \varphi_0^{-1} \circ \varphi$$

Then let $h = \psi \circ g \circ \psi^{-1}$. If z_0 is a fixed point of h, then $h(z) = z \implies g(\psi^{-1}(z)) = \psi^{-1}(z)$. Since g has a unique fixed point, we must have $\psi^{-1}(z) = f \implies z = i$. So, $h \in N$. $\square$

5.1.3 Fuchsian Groups and the Limit Set

We now restrict our attention to discrete subgroups of G, called Fuchsian groups, which are finitely generated. We are concerned with the dynamics of their limit sets.

Definition 5.3. Let Γ be a Fuchsian group. The **limit set** $L(\Gamma)$ of Γ is the closed subset of $\mathbb{H}(\infty)$ defined by

$$L(\Gamma) = \overline{\Gamma z} \cap \mathbb{H}(\infty)$$

where Γz is the Γ-orbit of z in $\mathbb{H}$.

The limit set does not depend on our choice of z. A proof of this fact is presented in [7]. We are only interested in interesting limit sets. The following proposition guarantees the existence of such limit sets.

Proposition 5.4. *If Γ contains at least two hyperbolic isometries that do not have a fixed point in common, then $L(\Gamma)$ contains infinitely many elements. Otherwise,*

- *if all hyperbolic isometries of Γ have the same axis, then $L(\Gamma)$ is reduced to the endpoints of that axis,*
- *if Γ contains no hyperbolic isometries, then either $L(\Gamma)$ is the empty set or $L(\Gamma)$ is reduced to a single point and Γ is generated by a parabolic isometry fixing this point.*

Proof. See Proposition 3.3 in [7]. □

We are ready to begin our discussion of Schottky groups.

5.2 Schottky Groups and its limit set

We now work in $\mathbb{D}$. Fix $g \in G$ and let $z_0 \in \mathbb{D}$ be a fixed point such that $g(z_0) \neq z_0$. Define the sets

$$D(z_0, g) = \{z \in \mathbb{D} : d(z, z_0) \geq d(z, g(z_0))\}$$
$$M(z_0, g) = \{z \in \mathbb{D} : d(z, z_0) = d(z, g(z_0))\}$$

where $D(z_0, g)$ to be the closed half plane containing $g(z_0)$ defined by the perpendicular bisector of z_0 and $g(z_0)$ and $M(z_0, g)$ is the perpendicular bisector of z_0 and $g(z_0)$. When the point z_0 is clear from the context, we write $M(z_0, g) = M(g)$ and $D(z_0, g) = D(g)$. The function $M(z_0, g)$ has the following properties.

Property 5.2.1. We have that $g(M(z_0, g^{-1})) = M(z_0, g)$ and $g(D(z_0, g^{-1})) = \mathbb{D} \setminus \operatorname{Int}(D(z_0, g))$.

(i) $M(z_0, g) \cap M(z_0, g^{-1}) \cap \mathbb{H} \neq \emptyset$ if and only if g is elliptic.

(ii) The closures of $M(z_0, g)$ and $M(z_0, g^{-1})$ are disjoint in $\overline{\mathbb{H}}$ if and only if g is hyperbolic.

(iii) $M(z_0)$ and $M(z_0, g^{-1})$ have exactly one endpoint in common if and only if g is parabolic. Their common endpoint is the only fixed point of g.

Definition 5.4. A **Schottky group** of rank 2 is a subgroup of G which has 2 non-elliptic, non-trivial generators g_1 and g_2 such that there exists a point $z_0 \in \mathbb{D}$ such that

$$\overline{(D(z_0, g_1) \cup D(z_0, g_1^{-1}))} \cap \overline{D(z_0, g_2) \cup D(z_0, g_2^{-1})} = \emptyset.$$

For a Schottky group, we define $\mathcal{A} = \{g_1^{\pm 1}, g_2^{\pm 1}\}$ as its alphabet and we define a finite sequence $s_1 \circ s_2 \circ ... \circ s_n$ where $s_i \in \mathcal{A}$ as a word. A word is called reduced if $w = \text{id}$ or $s_i \neq s_{i+1}^{\pm 1}$. If a word $w = s_1...s_n$ is reduced, the length of the word is n. Let E_n be the set of reduced words of length n. Then, we have that

$$S(g_1, g_2) = \{\text{id}\} \cup \bigcup_{n \geq 1} E_n.$$

For an element $w = s_1 s_2 ... s_n \in S(g_1, g_2)$ of our group, we define $D(z_0, s_1 s_2 ... s_n) = s_1 s_2 ... s_{n-1} D(z_0, s_n)$. We want to study the dynamics of the limit set of a Schottky group $S(g_1, g_2)$ and relate the construction of the limit set with repeated iteration of D in hopes of having a recursive way of constructing $L(S(g_1, g_2))$. The following proposition gives us this link.

Proposition 5.5.

$$L(S(g_1, g_2)) = \bigcap_{n=1}^{\infty} \bigcup_{\substack{\text{reduced words} \\ \text{of length } n}} \overline{D(s_1, ..., s_n)}.$$

Proof. This is Proposition 1.11 in [7]. □

5.3 Counting Function on the Limit Set

Let $S(g_1, g_2)$ be a Schottky group. For simplicity, let g_1, g_2 be hyperbolic so that for distinct $x, y \in S(g_1, g_2)$, $D(x)$ and $D(y)$ are disjoint. Then, $L(S(g_1, g_2))$ forms a Cantor set on $\mathbb{D}(\infty)$ where $\mathbb{D}(\infty) \backslash L(S(g_1, g_2))$ is a countable union of intervals from Proposition 5.5. Then, we define the following counting function.

Definition 5.5. Fix a Schottky group $S(g_1, g_2)$ and let $p \in \mathbb{D}$ be a point such that by letting $z_0 = p$, it satisfies the non-intersecting condition of Definition 5.4. If I is an interval on $\mathbb{D}(\infty)$ with endpoints s and t, define $a(I)$ as the angle between the geodesics from s to p and from t to p. Let $N(x)$ be the number of intervals in $I \in \mathbb{D}(\infty) \backslash L(S(g_1, g_2))$ such that $a(I) \geq 1/x$.

As usual, we conjecture the following.

Conjecture 5.1. *$N(x) \sim cx^d$ where $d > 0$ is a constant.*

It would nice if $d = \dim_H(L(S(g_1, g_2)))$, but from the similar construction of the counting function in Example 3.3, it likely that this is not the case.

5.4 Algorithm to Construct the Limit Set

From Proposition 5.5, the algorithm to construct the limit set is straightforward. Fix a Schottky group $S(g_1, g_2)$. On the first step, construct $\mathbb{D}(\infty) \cap \bigcup_{a \in \mathcal{A}} \overline{D(a)}$ and let I_1, I_2, I_3, I_4 be the complement with respect to $\mathbb{D}(\infty)$. Remove I_1, I_2, I_3, I_4. On the nth step, remove $\omega(I_i)$ for all $1 \leq i \leq 4$ and $\omega \in E_{n-1}$.

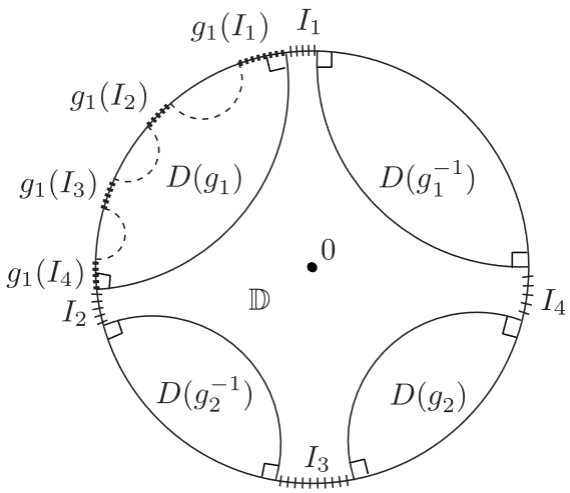

Figure 6: Third Level of the Recursion ([7])

5.5 Notes on Implementation

We store the necessary information about a single interval simply by storing it's endpoints. To implement the algorithm and to verify the conjecture, we need the following components:

5.5.1 Object: elements in G

For any $g \in \mathrm{Conf}(\mathbb{D})$, it suffices to store the two complex numbers α, β satisfying $|\alpha|^2 - |\beta|^2 = 1$ such that $g(z) = (\alpha z + \beta)/(\overline{\beta} z - \overline{\alpha})$. For $g \in \mathrm{Conf}(\mathbb{H})$, it suffices to store a, b, c, d where $g(z) = (az + b)/(cz + d)$ and $ad - bc = 1$. After creating these objects, it is easy to create inverse functions.

```
# isometry of H
class Hmap:

    # initialization
   def __init__(self, a, b, c, d):
      D = math.sqrt(a*d - b*c)
      self.a = a/D
      self.b = b/D
      self.c = c/D
      self.d = d/D

   # evaluation
   def eval(self, z):
      return (self.a*z + self.b) / (self.c * z + self.d)

   #inverse
   def inverse(self):
```

```
        x = Hmap(self.d, -self.b, -self.c, self.a)
        return x

    #compose with another map
    def compose(self, g):
        return Hmap(self.a * g.a + self.b * g.c, self.a * g.b + self.b * g.d,
            self.c * g.a + self.d * g.c, self.c * g.b + self.d * g.d)

# isometry on D
class Dmap(object):

    # initialization
    def __init__(self, a, b):
        D = math.sqrt(abs(a)**2 - abs(b)**2)
        self.a = a/D
        self.b = b/D

    # evaluation
    def eval(self, z):
        return (self.a * z + self.b) / (self.b.conjugate() * z + self.a.conjugate())

    # inverse
    def inverse(self):
        return Dmap(self.a.conjugate(), -1 * self.b)

    #compose with another map
    def compose(self, g):
        return Dmap(self.a * g.a + self.b * g.b.conjugate(), self.a * g.b + self.b
            * g.a.conjugate())
```

5.5.2 Construction: perpendicular bisector of two arbitrary points

Recall that we have the holomorphic diffeomorphism $\Psi : \mathbb{H} \to \mathbb{D}$ defined by

$$\Psi(z) = i\frac{z-i}{z+i}, \quad \mathcal{G}(z) = \Psi^{-1}(z) = \frac{z+i}{iz+1}.$$

Let $z, w \in \mathbb{D}$ be two arbitrary points. Let $z_1 = \mathcal{G}(z), w_1 = \mathcal{G}(w)$ where $z_1, w_1 \in \mathbb{H}$. Define $T_1(z) = z - \mathrm{Re}(z_1)$ and let $z_2 = T_1(z_1)$ and $w_2 = T_1(w_1)$. Let $T_2(z) = iz/z_2$ and let $z_3 = T_2(z_2) = i, w_3 = T_2(w_2)$. We now seek to find

$$T_3(z) = \frac{z\cos\theta - \sin\theta}{z\sin\theta + \cos\theta} \in K$$

such that $\mathrm{Re}(T_3(w_3)) = 0$. So, $T_3(w_3) + \overline{T_3(w_3)} = 0 \implies \tan(2\theta) = (w_3 + \overline{w_3})/(1 - |w_3|^2)$. So, we have that

$$\theta = \begin{cases} \frac{\pi}{4} & \text{if } |w_3| = 1 \\ \frac{1}{2}\tan^{-1}\left(\frac{w_3+\overline{w_3}}{1-|w_3|^2}\right) & \text{otherwise.} \end{cases}$$

So, let $w_4 = T_3(w_3)$. We also have that $i = T_3(z_3)$ because $T_3 \in K$. Let $w_4 = si$ for $r \in \mathbb{R}$. Then, the midpoint of i and w_4 is $\sqrt{s}i$ and it is easy to see that the endpoints are the points $\pm\sqrt{s}$. So, the endpoints in $\mathbb{D}$ are

$$\Psi \circ T_1^{-1} \circ T_2^{-1} \circ T_3^{-1}(\pm\sqrt{s}).$$

All of these computations are valid because all of our maps have been members of G besides the diffeomorphism Ψ which just carries the information between the two models.

```
def psi(z):
   return 1j * (z- 1j) / (z + 1j)

def Gsi(z):
   return (z + 1j) / (1j*z + 1)

# Computes the endpoints of the perpendicular bisector of z, w in the Poincare
    disc
def endpoints(z, w):
   z1 = Gsi(z)
   w1 = Gsi(w)

   T1 = Hmap(1, -1 * z1.real, 0, 1)

   z2 = T1.eval(z1)
   w2 = T1.eval(w1)

   T2 = Hmap(1, 0, 0, z2.imag)

   z3 = T2.eval(z2)
   w3 = T2.eval(w2)

   theta = math.pi / 4
   if(abs(w3) != 1):
      theta = 0.5 * math.atan((2 * w3.real) / (1 - abs(w3)**2))

   T3 = Hmap(math.cos(theta), -1 * math.sin(theta), math.sin(theta),
       math.cos(theta))

   w4 = T3.eval(w3)
   arg = math.sqrt(w4.imag)

   first = psi(T1.inverse().eval(T2.inverse().eval(T3.inverse().eval(arg))))
   second = psi(T1.inverse().eval(T2.inverse().eval(T3.inverse().eval(-1 * arg))))

   return [first/abs(first), second/abs(second)]
```

5.5.3 Function: distance between two arbitrary points

From Theorem 5.1, we can the define the function as

```
def d_dist(z, w):
   a = abs(1 - z * w.conjugate())
   b = abs(z-w)
   arg = (a+b)/(a-b)
   return math.log(arg)
```

5.5.4 Function: $a(I)$

We prove the following proposition.

Proposition 5.6. *Let C_1 and C_2 be two circles centered at O_1 and O_2 with radii r_1 and r_2, respectively. If C_1 and C_2 intersect at an angle of θ, then*

$$\theta = \pi - \cos^{-1}\left(\frac{r_1^2 + r_2^2 - d^2}{2r_1r_2}\right).$$

Proof. Suppose that C_1 and C_2 intersect at X. From the Law of Cosines,

$$\cos \angle O_1XO_2 = \frac{-d^2 + r_1^2 + r_2^2}{2r_1r_2}.$$

The angle between the two circles is $\theta = \pi - \angle O_1XO_2$ which suffices for the proof. □

We want $a([x_1, x_2])$ which is the angle between the circles C_1 and C_2 where C_i is the circle which passes through x_i and p which is orthogonal to $\mathbb{D}(\infty)$. Let $\mathcal{I} : \mathbb{C} \to \mathbb{C}$ be the involution defined by $\mathcal{I}(z) = 1/\overline{z}$. It can be checked that orthogonal circles to $\mathbb{D}(\infty)$ are invariant under $\mathbb{I}$. Moreover, if C is a circle orthogonal to $\mathbb{D}(\infty)$ and $\{m, n\}$ be a pair of points on C such that $0, m, n$ are collinear, then $\mathcal{I}(m) = n$. So, C_1 and C_2 pass through p and $\mathcal{I}(p) = 1/\overline{p}$. To find the circumcenter, we need the next proposition.

Proposition 5.7. *Let $a, b, c \in \mathbb{C}$ be three distinct non-collinear complex numbers. Then, the complex number representing the circumcenter of the triangle with vertices at a, b, c is*

$$\mathcal{O}(a, b, c) = \frac{\det \begin{bmatrix} a & |a|^2 & 1 \\ b & |b|^2 & 1 \\ c & |c|^2 & 1 \end{bmatrix}}{\det \begin{bmatrix} a & \overline{a} & 1 \\ b & \overline{b} & 1 \\ c & \overline{c} & 1 \end{bmatrix}}.$$

Proof. See [9] □

So, $O_1 = \mathcal{O}(x_1, p, \mathcal{I}(p))$ and $O_2 = \mathcal{O}(x_2, p, \mathcal{I}(p))$, $r_1 = |O_1 - x_1|$, $r_2 = |O_2 - x_2|$. So, we can calculate the angle between the two geodesics. The only case we have to be careful is when $p = 0$.

```
 # A is an array of length 9
def det(A):
   return A[0] * A[4] * A[8] + A[3] * A[7] * A[2] + A[1] * A[5] * A[6] - A[6] *
       A[4] * A[2] - A[1] * A[3] * A[8] - A[0] * A[5] * A[7]

# Circumcenter
def O(a, b, c):
   top = det([a, abs(a)**2, 1, b, abs(b)**2, 1, c, abs(c)**2, 1])
   bot = det([a, a.conjugate(), 1, b, b.conjugate(), 1, c, c.conjugate(), 1])
   return top/bot

# Involution
def I(z):
   return 1/z.conjugate()

# endpoints = [x1, x2]
def a(endpoints, p):
   x1 = endpoints[0]
   x2 = endpoints[1]
   if(p == 0):
      num = x1 / x2
      if(num.real == 0):
         return math.pi/2
      return math.atan(num.imag/num.real)
   o1 = O(x1, p, I(p))
   o2 = O(x2, p, I(p))
   r1 = abs(o1 - x1)
   r2 = abs(o2 - x2)
   d = abs(o1 - o2)
   return math.pi - math.acos((r1**2 + r2**2 - d**2)/(2*r1*r2))
```

5.5.5 Function: composing a word with an interval

This can be done with a simple for loop.

```
 def movement(word, endpoints):
   a1 = endpoints[0]
   a2 = endpoints[1]
   for w in word:
      a1 = w.eval(a1)
      a2 = w.eval(a2)
   return [a1, a2]
```

The recursive part of the algorithm can be done by attaching each interval the last letter that was applied to it, and then to apply the other three letters instead.

5.6 Approximating the Exponent

Consider the Schottky group generated by the hyperbolic isometries

$$\varphi_1(z) = \frac{2z + \sqrt{3}}{\sqrt{3}z + 2}$$
$$\varphi_2(z) = \frac{2z - \sqrt{3}i}{\sqrt{3}iz + 2}.$$

The graphs shown in Figure 7 are the $\log x \sim \log N(x, p)$ graphs for $p = 0, 0.1, -0.2 + 0.3i$. For the first plot, we have a best-fit line of $y = 0.514x + 2.093$. For the second plot, we have a best-fit line of $y = 0.5133x + 2.1255$. For the third plot, we have a best-fit line of $y = 0.5149x + 2.1019$. Since the slopes are close together, this suggests that the conjecture is correct.

6 Further Work

For the future, one may consider proving Conjecture 5.1, but we believe that it can be proved using techniques in [2]. The algorithms used to construct the Julia set and the limit sets were quite elementary, so perhaps more precise and faster algorithms can be made to do the same task. Assuming that Conjecture 5.1 is correct, it could be interesting to consider the constant c of $N(x, p)$ when p is varied.

7 Acknowledgements

I would like to thank my mentor, Prof. Sergiy Merenkov, for both suggesting the project and his helpful guidance throughout. I would like to thank the MIT PRIMES program for giving me the opportunity to conduct this research project. I would like to thank Dr. Tanya Khovanova and Dr. Claude Eicher for providing helpful revisions for this paper.

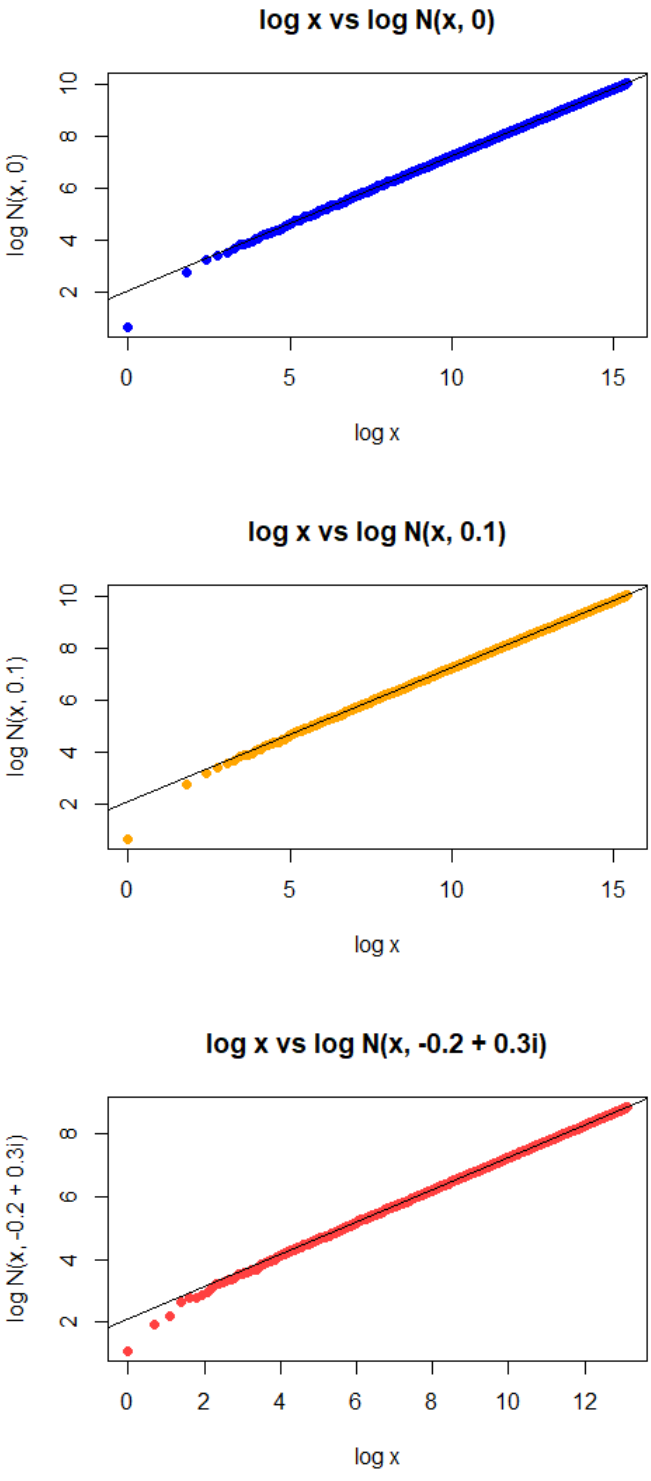

Figure 7: $\log x \sim \log N(x, p)$

References

[1] Alex Kontorovich and Hee Oh. Apollonian circle packings and closed horospheres on hyperbolic 3-manifolds, 2008.

[2] Mark Pollicott and Mariusz Urbanski. Asymptotic counting in conformal dynamical systems, 2017.

[3] K. J. Falconer. *Fractal geometry mathematical foundations and applications*. Wiley-Blackwell, 2014.

[4] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd edition, 2009.

[5] Michiel Smid. Computing the diameter of a point set: sequential and parallel algorithms, 2003.

[6] Taryn Flock. Hausdorff measure calculation for the standard basilica $c = -1$, 2009.

[7] Dalbo Françoise. *Geodesic and Horocyclic Trajectories*. Springer, 2011.

[8] Alan F. Beardon. *The geometry of discrete groups*. World Publishing Corporation, 2011.

[9] Titu Andreescu, Sam Korsky, and Cosmin Pohoata. *Lemmas in olympiad geometry*. XYZ Press, 2016.