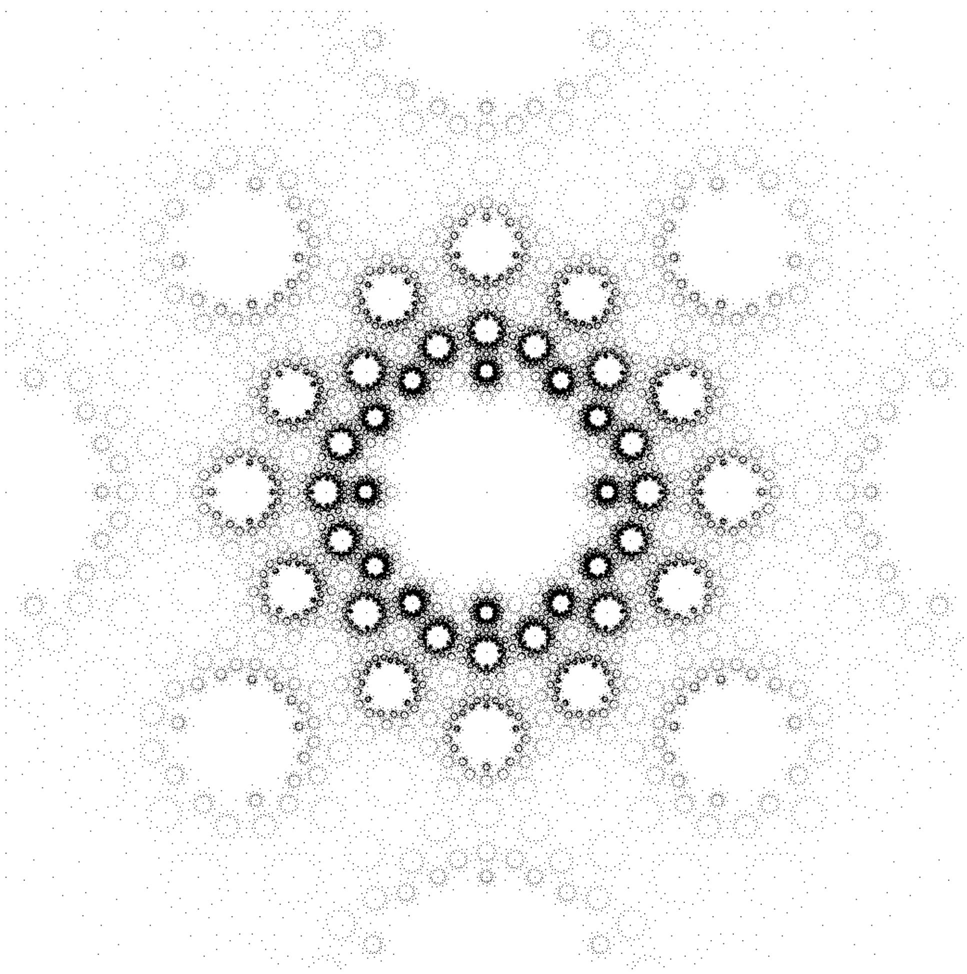
Radboud Universiteit Nijmegen

Faculteit der natuurwetenschappen, Wiskunde en Informatica

Algorithms on Continued Fractions

Noud Aldenhoven
Supervisor: Dr. W. Bosma
Second reader: Dr. C. Kraaikamp

January 30, 2012



Contents

Preface

This thesis is about continued fractions and in particular algorithms on continued fractions.

What are Continued Fractions?

Continued fractions are finite or infinite expressions obtained through an iterative process. An infinite continued fraction can be seen as a limit of finite continued fractions

$$a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\ddots + \cfrac{1}{a_{i-1} + \cfrac{1}{x_i}}}}}.$$

Continued fractions can be seen as one of the most mathematically natural representations of real numbers. Also truncating the continued fraction representation of a real number x yields a rational approximation with is in a certain sense the best possible rational approximation. This is a huge motivation to study continued fractions and to develop certain algorithms on continued fractions.

Thesis Format

Chapter 1 - Continued Fractions

As this thesis is about continued fractions a quick introduction to continued fractions is given in this chapter. This introduction is sufficient for our purposes and references are provided to more rigorous approaches.

Chapter 2 - Measure and Cantor Sets

One very important element in this thesis is the understanding of Cantor Sets. In this chapter some theory of these Cantor Sets is developed. The first part gives a technical, but general, definition of Cantor Sets in such a way that these sets become easy to handle. In the second part of this chapter some general theorems of sums of Cantor Sets are proved. This chapter is by no means a profound introduction to Cantor Sets, it only develops the mathematical tools for proving theorems in the next chapters.

Chapter 3 - Hall's Theorem

In this chapter Hall's Theorem is proved, i.e.t̃hat every real number can be written as a sum of two regular continued fractions with partial quotients less than or equal to 4. In order to prove this one must observe that the set of all regular continued fractions with quotients less than or equal to 4 is actually the sum of infinitely many General Cantor Sets. Then it is possible to use the machinery, developed in chapter 2, to tackle this problem. After that a couple of other problems of the same kind are solved, such as the one for the Nearest Integer Continued Fractions, using the idea of singularisation, and one for all Complex Numbers with bounded quotients.

Chapter 4 - Hall's Algorithm

In this chapter a special algorithm, named after Marshall Hall, is given, to do the following. Given a (complex) continued fraction

$$x = [a_0; e_1/a_1, e_2/a_2, e_3/a_3, ...]$$

and a Möbius transformation

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

with $a, b, c, d \in \mathbb{Z}$ (or $\mathbb{Z}[i]$) such that $|ad - bc| > 0$, this algorithm gives an efficient way for calculating

$$y = Mx = \frac{ax + b}{cx + d} = [b_0; f_1/b_1, f_2/b_2, f_3/b_3, ...].$$

The chapter closes with calculating an explicit example for Regular Continued Fractions.

Chapter 5 - Applications

In this chapter two applications of the theory developed in Chapter 3 and Chapter 4 are discussed. First an explicit algorithm is given to calculate, given a real number x, two elements a, b with quotients between 1 and 4 such that $x = a + b$. Second, a connection is given between Hall's theorem, described in Chapter 3, and Hall's Algorithm described in Chapter 4.

Chapter 6 - Drawing Nice Pictures

As will be seen throughout this thesis a lot of fractals appear. Drawing these Fractals cost a lot of computation time. In this chapter an algorithm is given for drawing these Fractals. This chapter does not contain a lot of mathematical content, however it might be still interesting to read. The author thinks that GiNaC is a very interesting tool to work with and someone interested in Computer Algebra would at least have to read this section of GiNaC.

Notations & Basic definitions

This thesis starts with some useful notation.

$\mathbb{Z}_{>0}, \mathbb{Z}, \mathbb{Q}$, $\mathbb{R}$ and $\mathbb{C}$ denote the sets of all positive integers, integers, rational numbers, real numbers and complex numbers respectively. $\mathbb{Z}[i]$ denotes the set of the Gaussian integers, hence

$$\mathbb{Z}[i] = \{a + bi \in \mathbb{C} : a, b \in \mathbb{Z}\}. \tag{1}$$

Inclusion of sets is denoted by $\subseteq$. We reserve $\subset$ for strict inclusion. So $A \subset B$ means that $A \subseteq B$ and that $A \neq B$.

Let n be a natural number. The set of all invertible n by n matrices over $\mathbb{Z}$ is denoted by $\mathrm{GL}_n(\mathbb{Z})$.

If $a, b \in \mathbb{Z}$, then (a, b) is used for the greatest common divisor of a and b. If $n \geq 3$ and $a_0, a_1, ..., a_n \in \mathbb{Z}$, then $(a_0, a_1, ..., a_n)$ is defined recursively as

$$(a_0, a_1, ..., a_n) = (a_0, (a_1, a_2, ..., a_n)).$$

Let $x, y, z \in \mathbb{R}$, then $x, y \leq z$ means $x \leq z$ and $y \leq z$. $x \geq y, z$, $x, y, < z$, $x > y, z$, $x < y < z$ and $x \leq y \leq z$ are defined similar.

If $a, b, c \in \mathbb{C}$, where $a \neq 0$, then the solution of

$$ax^2 + bx + c = 0$$

is

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

This formula is called the *abc-formula* .

The map μ is reserved for the Lebesgue-measure on the real number line.

Chapter 1

Continued Fractions

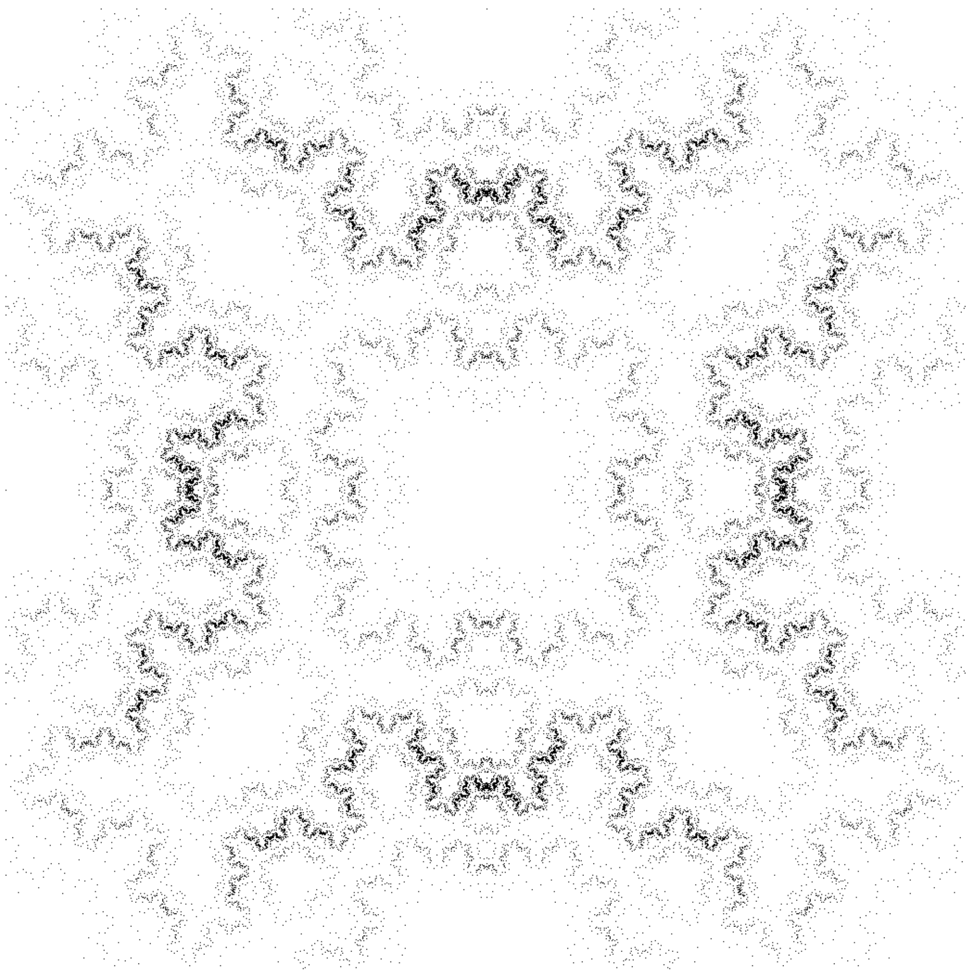


Figure 1.1: $\mathrm{CCF}_1(0)$ in the complex square $[-1,1] \times [-i,i]$.

This is a quick introduction to continued fractions. For a more rigorous approach see [RS92] and [Sch80].

1.1 Regular Continued Fractions

Definition 1.1.1. Every $x \in \mathbb{R}\backslash\mathbb{Q}$ has a unique *regular continued fraction expansion* of the form

$$x = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cfrac{1}{\ddots}}}} = [a_0; a_1, a_2, a_3, ...]$$

where $a_0 \in \mathbb{Z}$ is the *integer part* of x and where a_n for $n > 0$ is a positive integer. These a_n are called the *partial quotients* or simpler *quotients*.

If $x \in \mathbb{Q}$, then a regular continued fraction expansion of x is finite. There are exactly two finite expansions of x, one of them ending with a 1. Writing $x = \frac{p}{q}$, the expansion is obtained from Euclid's algorithm to find the greatest common divisor of p and q. A *finite expansion* of x is

$$x = [a_0; a_1, ..., a_{i-1}, x_i] = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{\ddots + \cfrac{1}{a_{i-1} + \cfrac{1}{x_i}}}}}$$

hence $\frac{p_n}{q_n}$ converges to x. where a_0 is the integer part of x, a_j where $j < i$ are positive integers and x_i is the i-th fractional part of x.

Definition 1.1.2. The regular continued fraction operator T is defined by

$$T : [0, 1) \to [0, 1) : x \mapsto \frac{1}{x} - \left\lfloor \frac{1}{x} \right\rfloor,$$

where $\lfloor y \rfloor$ is the integer part of y. To find the continued fraction of x one puts

$$\begin{aligned} T_0 &= x, \\ T_1 &= T(T_0), \\ T_2 &= T(T_1) = T^2(T_0), \\ &\vdots \end{aligned}$$

and then defines the partial quotients of x by

$$a_0 = \lfloor x \rfloor, \qquad a_n = \left\lfloor \frac{1}{T_{n-1}} \right\rfloor, \qquad n \geq 1.$$

Then $[a_0; a_1, a_2, ...]$ converges to x.

Definition 1.1.3. Define the following recurrence relation for p_n and q_n, where $n \geq 1$:

$$\begin{aligned} p_{-1} &= 1, & p_0 &= 0, & p_n &= a_n p_{n-1} + p_{n-2} \\ q_{-1} &= 0, & q_0 &= 1, & q_n &= a_n q_{n-1} + q_{n-2}. \end{aligned}$$

These p_n and q_n have the property that

$$\frac{p_n}{q_n} = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cdots + \cfrac{1}{a_n}}}}$$

So that $\frac{p_n}{q_n}$ converges to x as $n \to \infty$. The following two facts are known for $n \geq 1$:

$$\left| x - \frac{p_n}{q_n} \right| < \left| x - \frac{p_{n-1}}{q_{n-1}} \right| \qquad \text{and} \qquad \left| x - \frac{p_n}{q_n} \right| < \frac{1}{q_n^2}$$

for every $n \geq 0$.

In 1798 Legendre proved the following result.

Theorem 1.1.4. For every $x \in \mathbb{R}$, If $p, q \in \mathbb{Z}$, $q > 0$, and $\gcd(p, q) = 1$, if

$$\left| x - \frac{p}{q} \right| < \frac{1}{2q^2}$$

then $p = p_n(x)$ and $q = q_n(x)$, for some $n \geq 0$.

Legendre's Theorem is one of the main reasons for studying continued fractions, because it tells us that good approximations of irrational numbers by rational numbers are given by continued fraction convergents.

1.2 Nearest Integer Continued Fractions

The *nearest integer continued fraction* (NICF) is a real continued fraction which allows negative integers as partial quotients. Instead of rounding down, the algorithm to calculate the NICF of a real number x rounds to the nearest integer. In the case of a tie, it rounds to the smallest integer. For this the notation $\lfloor x \rceil$ is used. Thus $\lfloor 2.5 \rceil = 2$ and $\lfloor -2.5 \rceil = -3$. So, the algorithm to find the nearest integer continued fraction of x is equal to the one in Definition 1.1.2 and uses $a_0 = \lfloor x \rceil$ and $a_{n+1} = \lfloor x_{n+1} \rceil$.

Because negative quotients can occur in the NICF expansion the following notation is used. Every irrational number x can be expanded in a NICF as

$$x = [a_0; e_1/a_1, e_2/a_2, e_3/a_3, ...] = a_0 + \cfrac{e_1}{b_1 + \cfrac{e_2}{b_2 + \cfrac{e_3}{b_3 + ...}}}$$

where $b_0 \in \mathbb{Z}$, $e_i = \pm 1$ and $b_i \in \mathbb{Z}_{>1}$.

Theorem 1.2.1. For every $n \geq 1$ one has $e_{n+1} + b_n \geq 2$.

For a good reference on this see **??**. If $e_i = 1$ then it will be omitted usually. For example $[1; 2, -1/3, 4, 2, -1/4, ...]$ is written instead of $[1; 1/2, -1/3, 1/4, 1/2, -1/4, ...]$. In Chapter 4 the e_i are ommited for simplicity, therefore the example above would become $[1; 2, -3, -4, -2, 4, ...]$.

1.3 Complex Continued Fractions

The complex continued fraction can be defined in various ways. One of the ways known at this moment is due to Asmus Schmidt [Sch75]. It gives the best approximations by ratios of Gaussian integers [Hen06, p.67], but the link with real continued fractions is not immediately clear. An algorithm which still gives good approximations, and is a direct extension of the nearest integer continued fraction is the *Hurwitz Continued Fraction* [Hur]. To get the Hurwitz continued fraction (HCF) of a complex number x, take the nearest integer, only now, it is a Gaussian integer. In case of a tie the same rules as for the NICF-expansion are extended to the complex plane, hence $\lfloor -2.5 \rceil = -3$, $\lfloor 2.5 - 3.5i \rceil = 2 - 4i$. The complex plane can then be divided into squares that show which complex numbers round to a specific Gaussian integer.

1.4 Some General Theorems

The following theorems hold for the regular, nearest integer and Hurwitz continued fraction.

Theorem 1.4.1. For every $x \in \mathbb{Q}$ there is a finite continued fraction such that $x = [a_0; \ldots, a_n]$.

Theorem 1.4.2. Let $x = [a_0; a_1, a_2, ...]$ and p_n, q_n defined as in Definition 1.1.3. Then for every $k \geq 0$

$$p_k q_{k-1} - p_{k-1} q_k = (-1)^{k-1}. \tag{1.1}$$

In addition, for all $k \geq 0$, let

$$\frac{p_k}{q_k} = [a_0; a_1, a_2, ..., a_k], \qquad \text{and} \qquad \zeta = \zeta_{k+1} = [a_{k+1}; a_{k+2}, a_{k+3}, ...], \tag{1.2}$$

then

$$x = \frac{p_k \zeta + p_{k-1}}{q_k \zeta + q_{k-1}}. \tag{1.3}$$

A Möbius transformation of a complex number z is a function of the form $y = \frac{az+b}{cz+d}$ with a, b, c, d (complex) integers, and $ad - bc \neq 0$. A few simple Möbius transformations on some expansions can easily be deducted by hand, for example multiplication by -1 or i:

$$-1 \cdot x = -a_0 + \frac{-1}{x_1} = -a_0 + \frac{1}{-x_1}$$

$$i \cdot x = ia_0 + \cfrac{i}{a_1 + \cfrac{1}{x_2}} = ia_0 + \cfrac{1}{-ia_1 + \cfrac{-i}{x_2}} = ia_0 + \cfrac{1}{-ia_1 + \cfrac{1}{ix_2}}$$

Obviously, multiplication by -1 is different for the regular continued fraction, because only the first partial quotient can be negative.

Another simple one is addition by an integer. If k is an integer, then $[a_0; a_0, a_2, \ldots] + k = [a_0 + k, a_1, a_2, \ldots]$.

Chapter 2

Cantor Sets

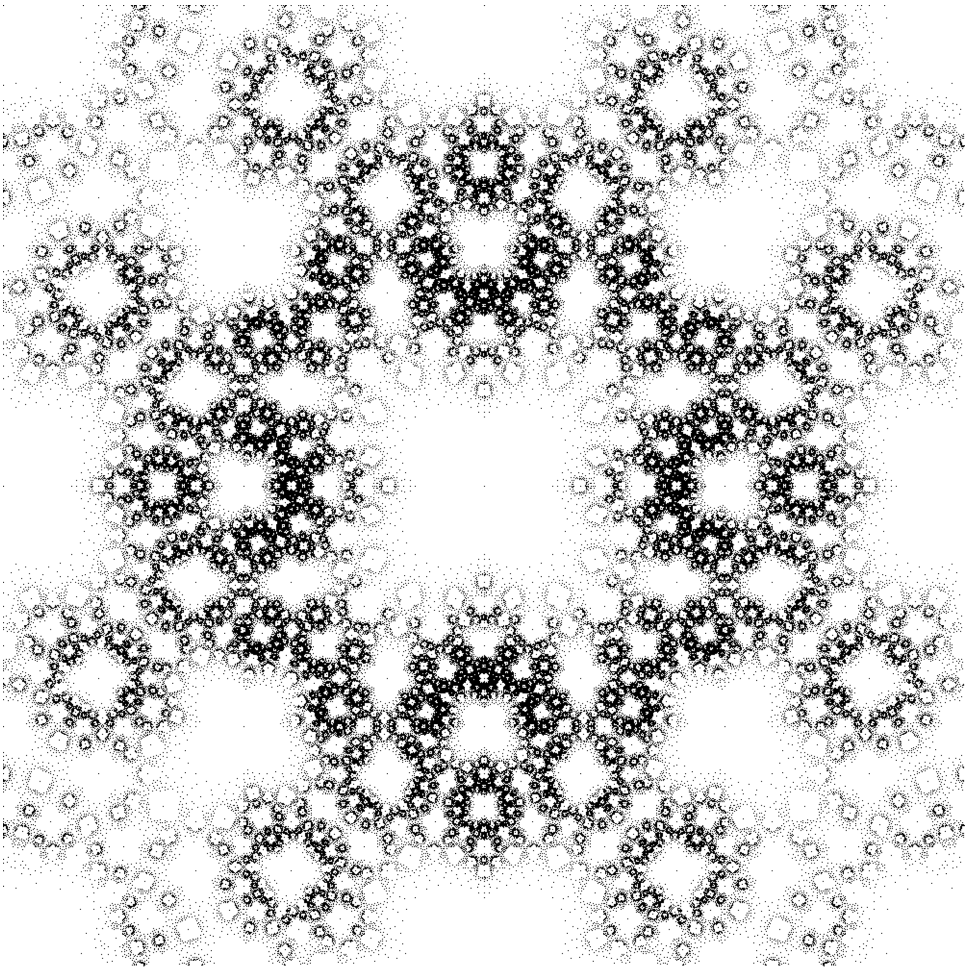


Figure 2.1: $\mathrm{CCF}_2(0)$ in the complex interval $[-1, 1] \times [-i, i]$.

One very important element in this thesis is the understanding of Cantor Sets. In this chapter some theory for these Cantor Sets is developed. The first part gives a technical, but general, definition of Cantor Sets in such a way that these sets become easy to handle. The second part of this chapter proves some general theorems of sums of Cantor Sets. This chapter is by no means a profound introduction to Cantor Sets, it only develops the mathematical tools for proving theorems in the next chapters. A curious reader is referred to literature such as [Can83], [Hau14] or [Eng89].

2.1 Cantor Sets

The Cantor sets where introduced by the German mathematician Georg Cantor in 1883 [Can83]. The most simple Cantor set is the *Cantor ternary set*. This set is created by recursively removing the open middle thirds of a set of line segments. One starts by deleting the open middle $(\frac{1}{3}, \frac{2}{3})$ from the interval $[0, 1]$, leaving two line segments $[0, \frac{1}{3}]$ and $[\frac{2}{3}, 1]$. Next, the open third of each of these remaining segments is deleted. This process is continued indefinitely. The Cantor ternary set contains all points in the interval [0,1] that are not deleted at any step in this infinite process.

Figure 2.2: The first six steps of this process.

Of course this process can be done a bit more generally. Let $A_0 = [a, b]$ be a closed interval. Then for $a_{00}, a_{01} \in \mathbb{R}$ such that $a < a_{00} < a_{01} < b$ one can split A_0 in

$$A_1^0 = [a, a_{00}], \qquad \text{and} \qquad A_1^1 = [a_{01}, b]$$

removing the middle open interval (a_{00}, a_{01}). Then again $a_{10}, a_{11}, a_{12}, a_{13} \in \mathbb{R}$ can be chosen such that $a < a_{10} < a_{11} < a_{00}$ and $a_{01} < a_{12} < a_{23} < b$. Now let A_1^0 split in

$$A_2^0 = [a, a_{10}], \qquad \text{and} \qquad A_2^1 = [a_{11}, a_{00}],$$

removing the middle open interval (a_{10}, a_{11}). Also let A_1^1 split in

$$A_2^3 = [a_{01}, a_{12}], \qquad \text{and} \qquad A_2^4 = [a_{23}, b],$$

removing the middle open interval (a_{12}, a_{23}). This process is continued infinitely many times.

As a formal technical, but useful, definition.

Definition 2.1.1. Let $A \subseteq \mathbb{R}$ be a closed bounded interval. Let $\{C_k^n \subset A : n \in \mathbb{Z}_{\geq 0}, k < 2^{n-1}\}$ be open subintervals of A. Define subsets A_k^n recursively as follows:

i. $A_0^0 = A$, called the *root*,

ii. $A_{2k}^n = \{x \in A_k^{n-1} \backslash C_k^n : \forall c \in C_k^n, x < c\}$, called the *left interval* of A_k^n,

iii. $A_{2k+1}^n = \{x \in A_k^{n-1} \backslash C_k^n : \forall c \in C_k^n, x > c\}$, called the *right interval* of A_k^n.

In addition suppose that for all $n \in \mathbb{Z}_{n \geq 0}$ and $k < 2^n$ also $C_k^n \subset A_k^{n-1}$. Then the family of sets $\{C_k^n : n \in \mathbb{Z}_{\geq 0}, k < 2^n\}$ is called *Cantor gaps for A*. Now a *General Cantor Point Set* is defined as

$$L(A, \{C_k^n : n \in \mathbb{Z}_{\geq 0}, k < 2^n\}) = A \backslash \bigcup_{\substack{n \in \mathbb{Z}_{\geq 0} \\ k < 2^n}} C_k^n.$$

If no confusion is possible $L(A, \{C_k^n : n \in \mathbb{Z}_{\geq 0}, k < 2^n\})$ is written for simplicity as $L(A)$.

It is convenient to see this General Cantor Point Set as a tree as follows.

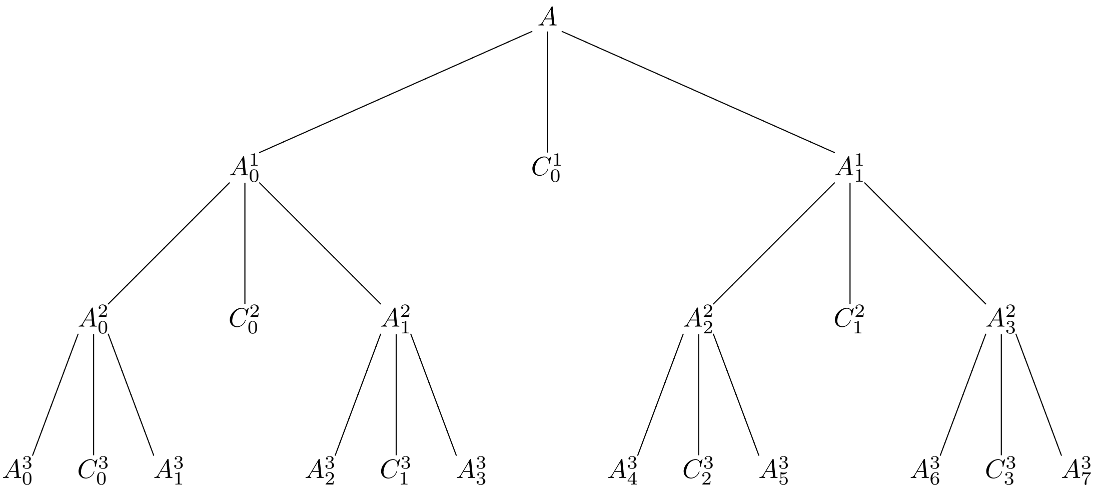


Figure 2.3: The first 3 steps in this process.

Example 2.1.2. The Cantor Gaps of the Cantor Set in Figure 2.2 are now defined as follows. For every $m \geq 0$ and $k < 2^m - 1$

$$C_k^m = \left(\frac{3k+1}{3^m}, \frac{3k+2}{3^m} \right).$$

From which the Cantor Set becomes

$$[0,1] \backslash \bigcup_{m \geq 0} \bigcup_{k < 2^m - 1} C_k^m = [0,1] \backslash \bigcup_{m \geq 0} \bigcup_{k < 2^m - 1} \left(\frac{3k+1}{3^m}, \frac{3k+2}{3^m} \right).$$

Theorem 2.1.3. Let $L(A)$ be a General Cantor Point Set.

i. $L(A)$ is a perfect set in $\mathbb{R}$, i.e.~the set of all limit points of $L(A)$ is again $L(A)$.

ii. All end points of any one of the subdividing interval A_k^i belong to $L(A)$.

iii. $L(A)$ contains an interval or $L(A)$ is nowhere dense.

iv. The measure $\mu(L(A))$ may be zero or any positive quantity less than the length of A.

Proof. See [Hau14], pp.~129-138. □

Here a couple of simple definitions are introduced.

Definition 2.1.4. If $A, B \subseteq \mathbb{R}$, then $A + B = \{a + b : a \in A, b \in B\}$. Hence, if $A = (x_1, x_2)$ and $B = (y_1, y_2)$, then $A + B = (x_1 + y_1, x_2 + y_2)$. Also $A \cdot B = \{ab : a \in A, b \in B\}$.

Definition 2.1.5. If $A \subseteq \mathbb{R}$ then

$$l(A) = \begin{cases} |\sup(A) - \inf(A)| & \text{if } \sup(A) \text{ and } \inf(A) \text{ exists,} \\ \infty & \text{otherwise.} \end{cases}$$

Definition 2.1.6. Let $L(A, \{C_k^n \subset A : n \in \mathbb{Z}_{\geq 0}, k < 2^n\})$ be a General Cantor Point Set. Define A_k^n as in Definition 2.1.1. Then

$$a_k^n = l(A_k^n) \qquad \text{and} \qquad c_k^n = l(C_k^n).$$

2.2 Sums of Cantor Sets

The proofs given in this section are based on the proofs given in [Hal47]. They are however completely rewritten and (all) errors in [Hal47] are corrected.

An interesting question is; given two Cantor sets $L(A)$ and $L(B)$, what is the measure of $L(A)+L(B)$ ($\mu(L(A)+L(B))$)? In this section this question is answered.

Theorem 2.2.1. Let $A, B \subset \mathbb{R}$ be closed bounded subsets. Suppose $\{C_k^n\}$ and $\{D_k^n\}$ are Cantor gaps for respectively A and B. If there exists a map $r : \mathbb{Z}_{\geq 0} \to \mathbb{R}$ such that for all $n \in \mathbb{Z}_{\geq 0}$ and $k < 2^n$

$$c_k^n \geq r(n) a_k^{n-1}, \qquad d_k^n \geq r(n) b_k^{n-1}$$

with

$$\prod_{n=0}^{\infty}(2-2r(n)) = 0,$$

then $\mu(L(A)+L(B)) = 0$.

Proof. Let $i, k, l \geq 0$. Now A_k^i splits up in A_{2k}^{i+1} and A_{2k+1}^{i+1} and similar B_l^i splits up in B_{2l}^{i+1} and B_{2l+1}^{i+1}. Remark that if $A_k^i + B_l^i$ contains the whole set $L(A_k^i) + L(B_l^i)$, then

$$A_{2k}^{i+1} + B_{2l}^{i+1}, A_{2k}^{i+1} + B_{2l+1}^{i+1}, A_{2k+1}^{i+1} + B_{2l}^{i+1} \text{ and } A_{2k+1}^{i+1} + B_{2l+1}^{i+1}$$

cover all of $L(A_k^i) + L(B_l^i)$. The length of the new intervals is

$$\begin{aligned}(a_{2k}^{i+1} + b_{2l}^{i+1}) + (a_{2k}^{i+1} + b_{2l+1}^{i+1}) + (a_{2k+1}^{i+1} + b_{2l}^{i+1}) + (a_{2k+1}^{i+1} + b_{2l+1}^{i+1})) &= 2a_k^i - 2c_k^i + 2b_l^i - 2d_l^i \\ &\leq (2-2r(i))(a_k^i + b_l^i).\end{aligned}$$

Define

$$T_i = \sum_{k=0}^{2^i-1}(l(A_k^i) + l(B_k^i))$$

Then by the previous remark

$$T_{i+1} \leq \prod_{n=0}^{i}(2-2r(n))T_0.$$

Since $\lim \prod_{n=0}^{i}(2-2r(n)) = 0$, it follows that

$$\mu(L(A)+L(B)) = 0.$$

□

Remark 2.2.2. Theorem 2.2.1 can be generalized such that if $k > 0$ and $L(A_1), L(A_2), ..., L(A_k)$ are General Cantor sets with Cantor gaps $\{{}^1C_l^n\}, ..., \{{}^kC_l^n\}$ satisfying

$${}^k c_l^n \geq r(n)^k a_l^{n-1}$$

for all n and l with

$$\prod_{n=0}^{\infty}(k - kr(n)) = 0,$$

then $\mu(L(A_1) + L(A_2) + ... + L(A_k)) = 0$.

Corollary 2.2.3. Let A, B be closed bounded subsets of $\mathbb{R}$. Suppose $\{C_k^n\}$ and $\{D_k^n\}$ are Cantor Gaps for respectively A and B. If for all $n > 0$ and $k < 2^n$ one has $c_k^n > \frac{1}{2}a_k^{n-1}$, then

$$\mu(L(A)+L(B)) = 0.$$

Proof. Take $r(n) = \frac{1}{2} + \epsilon$, where $\epsilon > 0$ and $c_k^n > r(n)a_k^{n-1}$ for all $n > 0$ and $k < 2^n$. Then for all $n > 0$

$$|2 - 2r(n)| = |1 - 2\epsilon| < 1.$$

Therefore

$$0 \leq \left| \prod_{n=0}^{\infty} (2 - 2r(n)) \right| \leq \prod_{n=0}^{\infty} |2 - 2r(n)| = 0,$$

hence

$$\prod_{n=0}^{\infty} (2 - 2r(n)) = 0.$$

Then by Theorem 2.2.1 the measure of $L(A) + L(B)$ is 0. □

So if $L(A) + L(B)$ should not be zero, at least the condition

(C1) for every $n > 0$ and $k < 2^n$ one has $c_k^n \leq a_{2k}^n, a_{2k+1}^n$ and $d_k^n \leq b_{2k}^n, b_{2k+1}^n$

should be true. The next part of this section will show that condition (C1) is also sufficient.

Definition 2.2.4. Let $A = [a_1, a_2]$, $B = [b_1, b_2]$ be two closed intervals and $e = \min(a_2 - a_1, b_2 - b_1)$. Then

$$\underline{[A, B]} = [a_1 + b_1, a_1 + b_1 + 2e],$$

$$\overline{[A, B]} = [a_2 + b_2 - 2e, a_2 + b_2].$$

Lemma 2.2.5. Let $A, B \subset \mathbb{R}$ be closed bounded subsets, $a = l(A)$, $b = l(B)$. Suppose C and D are open subsets of A and B. Let

$$A_1 = \{a \in A : a < c \text{ for all } c \in C\}, \qquad A_2 = \{a \in A : a > c \text{ for all } c \in C\},$$
$$B_1 = \{b \in B : b < c \text{ for all } c \in C\}, \qquad B_2 = \{b \in B : b > c \text{ for all } c \in C\}.$$

and suppose that $l(C) \leq l(A_1), l(A_2)$ and $l(D) \leq l(B_1), l(B_2)$. Let $\gamma \in \underline{[A, B]} \cup \overline{[A, B]}$, then

1. $\gamma \in \underline{[A, B_1]} \cup \overline{[A, B_1]}$ or,
2. $\gamma \in \underline{[A, B_2]} \cup \overline{[A, B_2]}$ or,
3. $\gamma \in \underline{[A_1, B]} \cup \overline{[A_1, B]}$ or,
4. $\gamma \in \underline{[A_2, B]} \cup \overline{[A_2, B]}$.

Proof. By symmetry of cases $a \leq b$ and $b \leq a$ it suffices to treat the case $a \leq b$. Let $A = [a_1, a_2]$, $B = [b_1, b_2]$, then $e = a_2 - a_1$. Therefore

$$\overline{[A, B]} = [2a_1 - a_2 + b_2, a_2 + b_2], \qquad \underline{[A, B]} = [a_1 + b_1, 2a_2 - a_1 + b_1].$$

Suppose that $B_1 = [b_1, x]$, $B_2 = [y, b_2]$. There are four cases depending on the lengths of the intervals A, B_1 and B_2.

i. $l(A) \leq l(B_1), l(B_2)$. Then $e = a_2 - a_1$ and

$$\overline{[A, B_2]} = [2a_1 - a_2 + b_2, a_2 + b_2] = \overline{[A, B]},$$
$$\underline{[A, B_1]} = [a_1 + b_1, 2a_2 - a_1 + b_1] = \underline{[A, B]}.$$

This gives case 1 or 2.

ii. $l(A) \le l(B_1)$ and $l(B_2) < l(A)$. In this case

$$\underline{[A, B_1]} = \underline{[A, B]}.$$

But $e = a_2 - a_1$ in $\overline{[A, B_1]}$ and $e = b_2 - y$ in $\overline{[A, B_2]}$. This gives

$$\overline{[A, B_1]} = [2a_1 - a_2 + x, a_2 + x],$$
$$\overline{[A, B_2]} = [a_2 - b_2 + 2y, a_2 + b_2].$$

Because of $l(A_1), l(A_2) \ge l(C)$ and $l(B_1), l(B_2) \ge l(D)$,

$$y - x = l(D) \le l(B_2) = b_2 - y, \quad \text{so} \quad 2y \le x + b_2 \quad \text{and therefore} \quad a_2 - b_2 + 2y \le a_2 + x,$$

hence the two intervals $\overline{[A, B_1]}$ and $\overline{[A, B_2]}$ overlap. A second observation is that $x \le b_2$, hence

$$2a_1 - a_2 + x \le 2a_1 - a_2 + b_2.$$

But then

$$\overline{[A, B]} = [2a_1 - a_2 + b_2, a_2 + b_2] \subseteq [2a_1 - a_2 + x, a_2 + b_2] = \overline{[A, B_1]} \cup \overline{[A, B_2]}.$$

Which gives case 1 or 2.

iii. $l(B_1) < l(A)$ and $l(A) \le l(B_2)$. This problem is dual to case ii. That is

$$\overline{[A, B]} = \overline{[A, B_2]}, \qquad \text{and} \qquad \underline{[A, B]} \subseteq \underline{[A, B_1]} \cup \underline{[A, B_2]}$$

and again this is of the form 1 or 2.

iv. $l(B_1) < l(A)$, $l(B_2) < l(A)$. Since $x - b_1 \ge y - x$ in this case $l(A) > y - x$ and

$$3l(A) > (x - b_1) + (y - x) + (b_2 - y) = b_2 - b_1 = l(B).$$

Therefore $3a_2 - 3a_1 \ge b_2 - b_1$, yielding

$$2a_1 - a_2 + b_2 \le 2a_2 - a_1 + b_1.$$

But that means that $A + B = \underline{[A, B]} \cup \overline{[A, B]}$. It is useful to look at the four intervals

$$\underline{[A, B_1]} = [a_1 + b_1, a_1 - b_1 + 2x],$$
$$\underline{[A, B_2]} = [a_1 + y, a_1 + 2b_2 - y],$$
$$\overline{[A, B_1]} = [a_2 + 2b_1 - x, a_2 + x],$$
$$\overline{[A, B_2]} = [a_2 - b_2 + 2y, a_2 + b_2].$$

Now note the following.

a. $\underline{[A, B_1]}$ and $\underline{[A, B_2]}$ overlap, because

$$x - b_1 \ge y - x, \quad \text{so} \quad 2x - b_1 \ge y \quad \text{and thus} \quad a_1 - b_1 + 2x \ge a_1 + y.$$

b. $\overline{[A, B_1]}$ and $\overline{[A, B_2]}$ overlap, because

$$b_2 - y \ge y - x, \quad \text{so} \quad x \ge 2y - b_2, \quad \text{and thus} \quad a_2 + x \ge a_2 - b_2 + 2y.$$

c. Because $x - b_1 \ge y - x$, $b_2 - y \ge y - x$ and $l(B) = (x - b_1) + (y - x) + (b_2 - y)$ one has

$$3l(B_1) \ge l(B) \qquad \text{or} \qquad 3l(B_2) \ge l(B).$$

If $3l(B_1) \ge l(B)$, then $\underline{[A, B_1]}$ and $\overline{[A, B_1]}$ overlap, because $l(A) \le l(B)$ by assumption:

$$a_2 - a_1 \le b_2 - b_1 \text{and so} \quad \le 3(b_1 - x), \quad a_2 + 2b_1 - x \le a_1 - b_1 + 2x.$$

If $3l(B_2) \ge l(B)$, then $\underline{[A, B_2]}$ and $\overline{[A, B_2]}$ overlap, because $l(A) \ge l(B)$ by assumption:

$$a_2 - a_1 \le b_2 - b_1 \text{and so} \quad \le 3(b_2 - y), \quad a_2 - b_2 + 2y \le a_1 + 2b_2 - y.$$

Hence γ lies in one of those four intervals.

This proves the lemma. $\square$

Theorem 2.2.6. Let $A = [a_1, a_2], B = [b_1, b_2]$ be closed bounded subsets or $\mathbb{R}$, where $a = a_2 - a_1$ and $b = b_2 - b_1$. Suppose $\{C_k^n\}$ and $\{D_k^n\}$ are Cantor Gaps for A and B, respectively. In addition suppose that condition

(C1) for every $n > 0$ and $k < 2^n$ one has $c_k^n \leq a_{2k}^n, a_{2k+1}^n$ and $d_k^n \leq b_{2k}^n, b_{2k+1}^n$,

is satisfied.
Define $e = \min(a, b)$, then

$$L(A) + L(B) = \underline{[A, B]} \cup \overline{[A, B]} = (a_1 + b_1, a_1 + b_1 + 2e) \cup (a_2 + b_2 - 2e, a_2 + b_2).$$

Proof. Obviously $L(A) + L(B) \subseteq \underline{[A, B]} \cup \overline{[A, B]}$.

Suppose that $\gamma \in \underline{[A, B]}$ or $\gamma \in \overline{[A, B]}$. By Lemma 2.2.5 there exists a shrinking sequence

$$(A_1, B_1), (A_2, B_2), (A_3, B_3), ...$$

where for all $i, j > 0$, $A_{i+1} \subseteq A_i$, $B_{j+1} \subseteq B_j$ and there are $m, n, k, l > 0$ such that $A_i = A_k^m$, $B_j = A_l^n$. In addition, for every $i > 0$

$$\gamma \in \overline{[A_i, B_i]}, \qquad \text{or} \qquad \gamma \in \underline{[A_i, B_i]}.$$

There are two cases.

i. $\lim_{i\to\infty} l(A_i) = 0$ and $\lim_{j\to\infty} l(B_j) = 0$. Take in this case

$$\alpha = \lim_{i\to\infty} \max(A_i), \qquad \beta = \lim_{j\to\infty} \max(B_j).$$

Then $\alpha \in L(A)$, $\beta \in L(B)$ and $\gamma = \alpha + \beta$.

ii. $\lim_{i\to\infty} l(A_i) = s \neq 0$ or $\lim_{j\to\infty} l(B_i) = t \neq 0$. Suppose first that $t > s$. There there are $i, j > 0$ such that $A_i = [a_1, a_2]$, $l(A_i) = s$ and $B_j = [b_1, b_2]$, $l(B_j) = t$. By Lemma 4.1.10 the sequence $(A_1, B_1), (A_2, B_2), (A_3, B_3), ...$ could have been taken such that B_j cannot be split up anymore, i.e. $B_j \subseteq L(B)$. Hence $\gamma \in A_i + B_j = [a_1 + b_1, a_2 + b_2]$.

1. If $a_1 + b_1 \leq \gamma \leq a_1 + b_1 + t$, take $\alpha = a_1$ and $\beta = \gamma - a_1$.
2. If $a_1 + b_1 + t \leq \gamma \leq a_2 + b_2$, take $\alpha = a_2$ and $\beta = \gamma - a_2$.

In both cases $\alpha \in L(A)$, because α is an endpoint of interval A_i, and $\beta \in L(B)$. But then also

$$\gamma = \alpha + \beta \in L(A) + L(B).$$

Next suppose that $t = s$. Then there are $i, j > 0$ such that $l(A_i) = t = l(B_j)$. By Lemma 2.2.5 A_i and B_j could have been taken such that they cannot be split up anymore, i.e. $A_i \subseteq L(A)$ and $B_j \subseteq L(B)$. But then

$$\gamma \in A_i + B_j \subseteq L(A) + L(B).$$

$\square$

Corollary 2.2.7. Assume the same setup as in Theorem 2.2.6. Also suppose that the following condition holds:

(C2) $\frac{1}{3} \leq \frac{a}{b} \leq 3$.

Then

$$L(A) + L(B) = A + B.$$

Proof. Because $\frac{1}{3} \leq \frac{a}{b}$ also $b \leq 3a$ and $\frac{a}{b} \leq 3$ implies that $a \leq 3b$. Therefore

$$(a_2 - a_1) + (b_2 - b_1) \leq a + b \leq 4a,$$
$$(a_2 - a_1) + (b_2 - b_1) \leq a + b \leq 4b.$$

Hence

$$(a_2 - a_1) + (b_2 - b_1) \leq 4e.$$

Rearranging the symbols gives

$$a_2 + b_2 - 2e \leq a_1 + b_1 + 2e.$$

Now applying Theorem 2.2.6 gives

$$L(A) + L(B) = (a_1 + b_1, a_1 + b_1 + 2e) \cup (a_2 + b_2 - 2e, a_2 + b_2) = (a_1 + b_1, a_2 + b_2) = A + B.$$

□

Chapter 3

Hall's Theorem

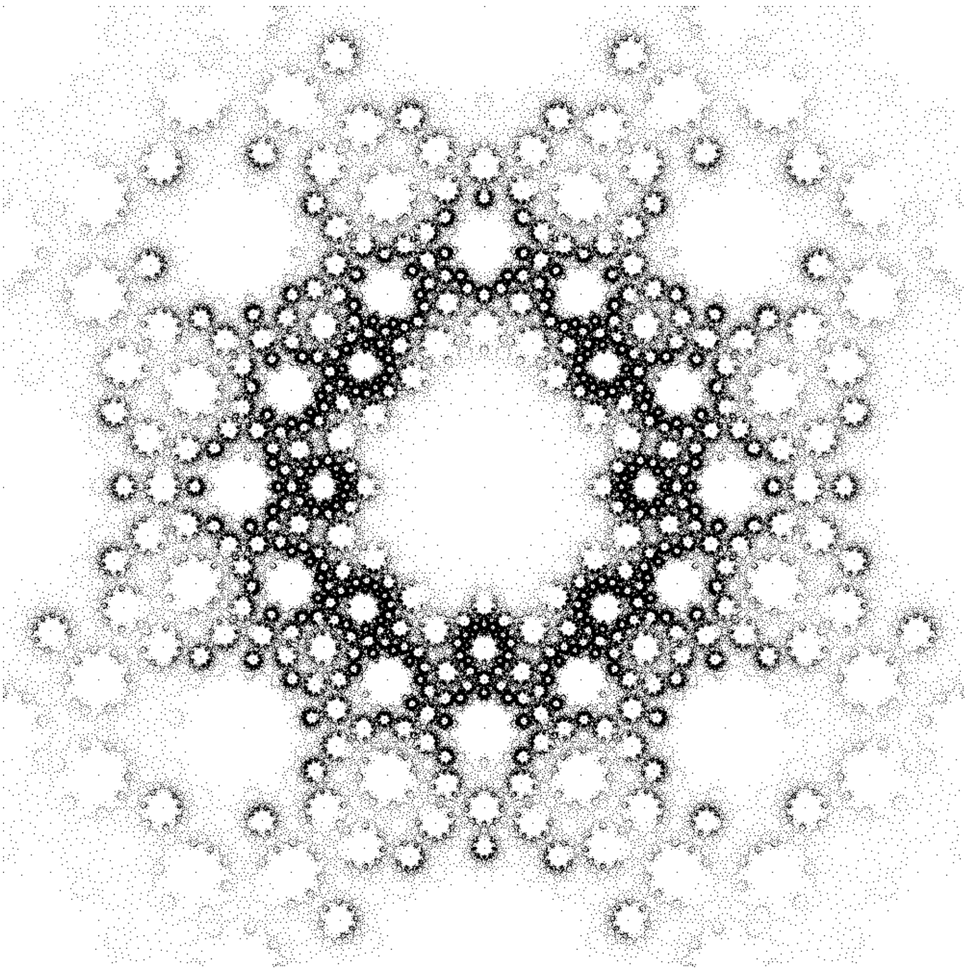


Figure 3.1: $\mathrm{CCF}_3(0)$ in the complex interval $[-1, 1] \times [-i, i]$.

In this chapter Hall's Theorem is proved, i.e.̃that every real number can be written as a sum of two regular continued fractions with quotients less then or equal to 4. In order to prove this one observes that the set of all regular continued fractions with quotients less then or equal to 4 is actually the sum of infinitely many General Cantor Sets. Then it is possible to use the machinery, developed in chapter 2, to tackle this problem. After that a couple of other problems of the same kind are solved, such as the one for the Nearest Integer Continued Fractions, using the idea of singularisation, and one for all Complex Numbers with bounded quotients.

3.1 Basic Definitions

Definition 3.1.1. Let $n \in \mathbb{Z}$ and $k \in \mathbb{Z}_{>0}$, then the *k-bounded Regular Continued Fractions with integral part n* are defined as

$$\mathrm{RCF}_k(n) = \{[a_0; a_1, a_2, a_3, ...] \in \mathbb{R} : a_0 = n, \forall i \geq 1 : 1 \leq a_i \leq k\},$$

and the *k-bounded Regular Continued Fractions* are

$$\mathrm{RCF}_k = \bigcup_{n\in\mathbb{Z}} \mathrm{RCF}_k(n).$$

Definition 3.1.2. Let $n \in \mathbb{Z}$ and $k \in \mathbb{Z}_{>0}$, then the *k-bounded Nearest Integer Continued Fractions with integral part n* are defined as

$$\mathrm{NICF}_k(n) = \{[a_0; e_1/a_1, e_2/a_2, e_3/a_3, ...] \in \mathbb{R} : a_0 = n, \forall i \geq 1 : a_i \leq k,$$
$$[a_0; e_1/a_1, e_2/a_2, e_3/a_3, ...] \text{ is a Nearest Integer Continued Fraction } \},$$

and the *k-bounded Nearest Integer Continued Fractions* are

$$\mathrm{NICF}_k = \bigcup_{n\in\mathbb{Z}} \mathrm{NICF}_k(n).$$

Definition 3.1.3. Let $m + ni \in \mathbb{Z}[i]$ and $k \in \mathbb{Z}_{>0}$, then the *k-bounded Complex Continued Fractions with integral part $m + ni$* are defined as

$$\mathrm{CCF}_k(m + ni) = \{[a_0; e_1/a_1, e_2/a_2, e_3/a_3, ...] \in \mathbb{R} : a_0 = m + ni, \forall i \geq 1 : a_i \in \mathbb{Z}[i], |a_i| \leq k\},$$

and the *k-bounded Complex Continued Fractions* are

$$\mathrm{CCF}_k = \bigcup_{m+ni\in\mathbb{Z}[i]} \mathrm{CCF}_k(m + ni).$$

3.2 $\mathrm{RCF}_4 + \mathrm{RCF}_4$

First there are some calculations.

Lemma 3.2.1. Let $a, b \in \mathbb{Z}_{>0}$, then

$$[0; a, b, a, b, a, b, ...] = -\frac{1}{2}b + \sqrt{\frac{1}{4}b^2 + \frac{b}{a}}.$$

Proof. Let $x \in \mathbb{R}$ such that $x = [0; a, b, a, b, ...]$. Then

$$x = [0; a, b, a, b, ...] = 0 + \cfrac{1}{a + \cfrac{1}{b + x}} = \frac{b + x}{ax + ab + 1}.$$

Hence $ax^2 + abx - b = 0$. Using the ABC-formula gives

$$x = \frac{-ab \pm \sqrt{(ab)^2 + 4ab}}{2a} = -\frac{1}{2}b \pm \sqrt{\frac{1}{4}b^2 + \frac{b}{a}}.$$

x cannot be negative, hence the result. □

Lemma 3.2.2. Let $a, b \in \mathbb{Z}_{>0}$, then

$$[b; a, b, a, b, a, ...] = \frac{1}{2}b + \sqrt{\frac{1}{4}b^2 + \frac{b}{a}}.$$

Proof. This can be easily deduced from Lemma 3.2.1, left as exercise for the reader. □

Take a better look at $\mathrm{RCF}_N(0)$. Using Lemma 3.2.1 one concludes that the maximum and minimum of $\mathrm{RCF}_N(0)$ are both attained and equal to

$$\max(\mathrm{RCF}_N(0)) = [0; 1, N, 1, N, ...] = \frac{\sqrt{N^2+4N} - N}{2},$$
$$\min(\mathrm{RCF}_N(0)) = [0; N, 1, N, 1, ...] = \frac{\sqrt{N^2+4N} - N}{2N}.$$

In particular

$$\max(\mathrm{RCF}_4(0)) = 2\sqrt{2} - 2,$$
$$\min(\mathrm{RCF}_4(0)) = \frac{\sqrt{2}-1}{2}.$$

Hence $\mathrm{RCF}_4(0)$ is contained in the closed interval $A = [\frac{1}{2}(\sqrt{2}-1), 2\sqrt{2}-2]$.

One can obtain the set $\mathrm{RCF}_4(0)$ as a Cantor set of this interval A.

Definition 3.2.3. Let $S \subset \mathbb{R}$ be a bounded subset of the real numbers. Define $[S]$ to be the smallest closed interval containing S. Hence

$$[S] = \bigcap_{\substack{[x,y]\subset\mathbb{R} \\ S\subseteq[x,y]}} [x, y] = [\mu, \nu],$$

where $\mu = \inf(S)$ and $\nu = \sup(S)$.

Closed subsets of $\mathrm{RCF}_4(0)$ can be divided into three different types.

Definition 3.2.4. Let $b_0, b_1, ..., b_k \in \mathbb{Z}_{>0}$, where $k \geq 0$. Define

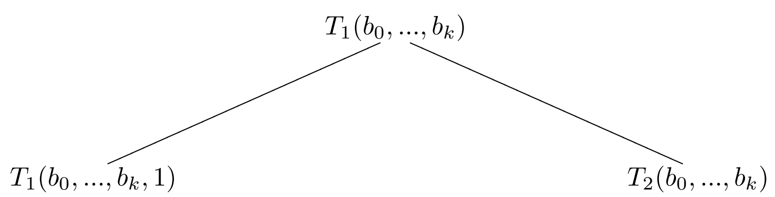
1. $T_1(b_0, b_1, ..., b_k) = [\{[0; b_0, b_1, ..., b_k, a_{k+1}, a_{k+2}, ...] : 1 \leq a_j \leq 4 \text{ for all } j > k\}]$,
2. $T_2(b_0, b_1, ..., b_k) = [\{[0; b_0, b_1, ..., b_k, a_{k+1}, a_{k+2}, ...] : 2 \leq a_{k+1} \leq 4, 1 \leq a_j \leq 4 \text{ for all } j > k+1\}]$,
3. $T_3(b_0, b_1, ..., b_k) = [\{[0; b_0, b_1, ..., b_k, a_{k+1}, a_{k+2}, ...] : 3 \leq a_{k+1} \leq 4, 1 \leq a_j \leq 4 \text{ for all } j > k+1\}]$.

From the definition $A = T_1()$. Now it is possible to obtain $\mathrm{RCF}_4(0)$ as a Cantor subset of A as follows. An infinite binary tree T of nodes $T_i(b_0, ..., b_k)$, where $1 \leq i \leq 3$ and $b_0, ..., b_k \in \mathbb{Z}_{>0}$ where $k \geq 0$; can be defined inductively. The root of T is equal to $A = T_1()$.

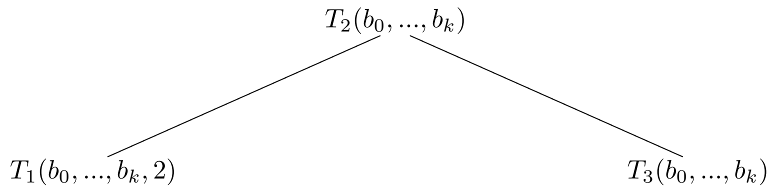
$$T_1()$$

Suppose there is a leaf L of depth n in the binary tree T. Let $b_0, ..., b_k \in \mathbb{Z}_{>0}$ where $k \geq 0$, there are three cases.

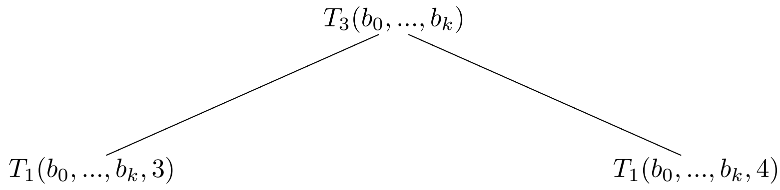
1. $V = T_1(b_0, ..., b_k)$. In this case the two children of V are $T_1(b_0, ..., b_k, 1)$ and $T_2(b_0, ..., b_k)$.



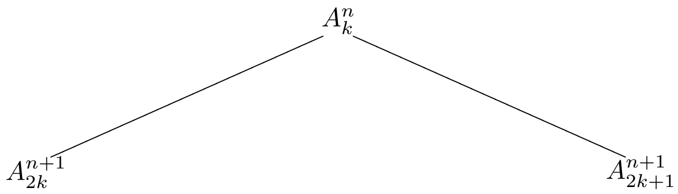
2. $V = T_2(b_0, ..., b_k)$. Then the two children of V are $T_1(b_0, ..., b_k, 2)$ and $T_3(b_0, ..., b_k)$.



3. $V = T_3(b_0, ..., b_k)$. This last case gives children $T_1(b_0, ..., b_k, 3)$ and $T_1(b_0, ..., b_k, 4)$.



This recursive definition defines the complete binary tree T. To extend this tree with Cantor gaps let



be a part of T and define

$$C_k^{n+1} = A_k^n \backslash (A_{2k}^{n+1} \cup A_{2k+1}^{n+1}).$$

Then these intervals $\{A_k^n\}$ and $\{C_k^n\}$ satisfy Definition 2.1.1 and thus they form a General Cantor Point Set $L(A) = L(A, \{C_k^n\})$. By definition $L(A) = \mathrm{RCF}_4(0)$.

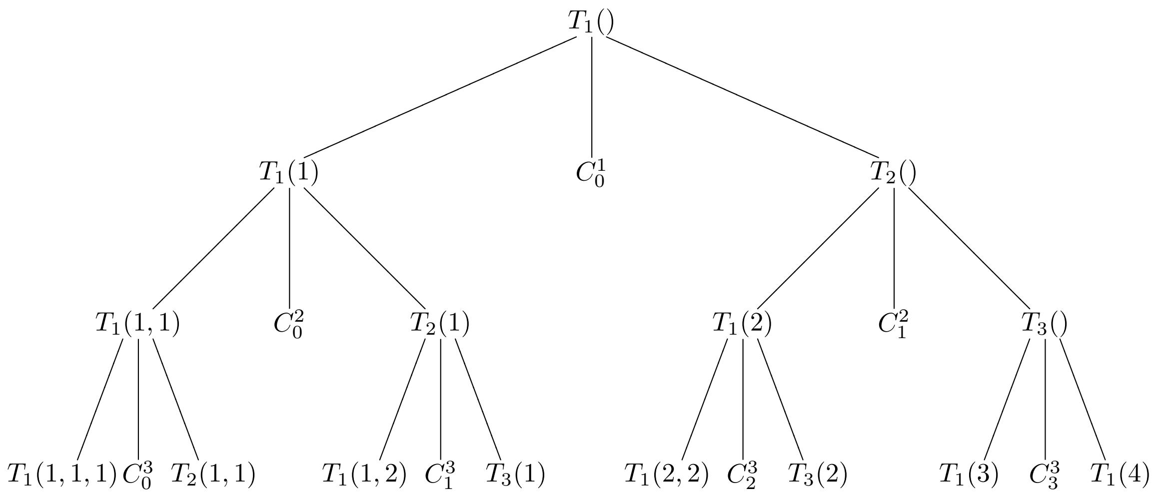


Figure 3.2: The first 4 layers of tree T with Cantor Gaps.

Remark 3.2.5. An important constant in the next calculations is

$$\zeta = [1; 4, 1, 4, 1, 4, ...] = \frac{1}{2}(\sqrt{2} + 1).$$

ζ satisfies the following interesting relations

$$4\zeta = [4; 1, 4, 1, 4, 1, ...] = 2\sqrt{2} + 2,$$
$$\frac{1}{\zeta} = [0; 1, 4, 1, 4, ...] = 4\zeta - 4.$$

Lemma 3.2.6. Let $b_1, ..., b_k \in \mathbb{Z}_{>0}$ and $p_k, q_k, p_{k-1}, q_{k-1} \in \mathbb{Z}$ such that

$$\frac{p_{k-1}}{q_{k-1}} = [0; b_1, b_2, ..., b_{k-1}],$$
$$\frac{p_k}{q_k} = [0; b_1, b_2, ..., b_k].$$

Then for every $\nu \in \mathbb{R}$

$$[0; b_1, b_2, ..., b_k, \nu] = \frac{\nu p_k + p_{k-1}}{\nu q_k + q_{k-1}}.$$

Proof. See Definition 1.1.3 and Theorem 1.4.2. □

Lemma 3.2.7. Let $b_1, ..., b_r \in \mathbb{Z}_{>0}$ and $p_{k-1}, q_{k-1}, p_k, q_k \in \mathbb{Z}$ such that

$$\frac{p_{k-1}}{q_{k-1}} = [0; b_1, b_2, ..., b_{k-1}],$$
$$\frac{p_k}{q_k} = [0; b_1, b_2, ..., b_k].$$

Then for every $\mu, \nu \in \mathbb{R}$ let $x, y \in \mathbb{R}$ such that

$$x = [0; b_1, b_2, ..., b_k, \mu],$$
$$y = [0; b_1, b_2, ..., b_k, \nu].$$

Let $\epsilon = \frac{q_{k-1}}{q_k}$; then

$$|x - y| = \frac{|\mu - \nu|}{q_k(\mu + \epsilon)(\nu + \epsilon)}$$

holds.

Proof. By Theorem 1.4.2, $p_k q_{k-1} - p_{k-1} q_k = (-1)^k$. Knowing this the calculation becomes

$$\begin{aligned}
|x - y| &= |[0; b_1, b_2, ..., b_k, \mu] - [0; b_1, b_2, ..., b_k, \nu]| \\
&= \left| \frac{\mu p_k + p_{k-1}}{\mu q_k + q_{k-1}} - \frac{\nu p_k + p_{k-1}}{\nu q_k + q_{k-1}} \right| \\
&= \left| \frac{(\mu p_k + p_{k-1})(\nu q_k + q_{k-1}) - (\mu q_k + q_{k-1})(\nu p_k + p_{k-1})}{(\mu q_k + q_{k-1})(\nu q_k + q_{k-1})} \right| \\
&= \left| \frac{\mu(p_k q_{k-1} - p_{k-1} q_k) + \nu(p_{k-1} q_k - p_k q_{k-1})}{q_k(\mu + \epsilon)(\nu + \epsilon)} \right| \\
&= \frac{|(-1)^k \mu + (-1)^{k+1} \nu|}{q_k(\mu + \epsilon)(\nu + \epsilon)} \\
&= \frac{|\mu - \nu|}{q_k(\mu + \epsilon)(\nu + \epsilon)}.
\end{aligned}$$

□

Lemma 3.2.8. $L(A) + L(A) = A + A = (\sqrt{2} - 1, 4\sqrt{2} - 4)$.

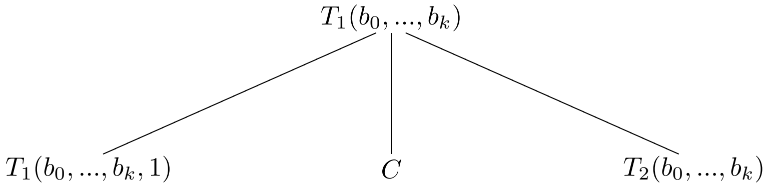
Proof. Because $A = [\frac{1}{2}(\sqrt{2} - 1), 2\sqrt{2} - 2]$, $A + A = [\sqrt{2} - 1, 4\sqrt{2} - 4]$. If the conditions of Corollary 2.2.7 applied to $L(A)$ are satisfied then $A + A = L(A) + L(A)$ and the theorem is proved.

Let us first check condition 1, which states

(C1) for every $n > 0$ and $k \leq 2^{n-1}$ one has $c_k^n \leq a_{2k}^n, a_{2k+1}^n$ and $d_k^n \leq b_{2k}^n, b_{2k+1}^n$.

Let $b_0, b_1, ..., b_k \in \mathbb{Z}_{>0}$, then there are three cases.

i. Case 1 where



Denote by t_1, c, t_2 the lengths of respectively the intervals $T_1(b_0, ..., b_k, 1)$, C and $T_2(b_0, ..., b_k)$. Suppose that k is even.

t_1. The smallest and largest elements of $T_1(b_0, ..., b_k, 1)$ are, respectively,

$$[0; b_1, ..., b_k, 1, 4, 1, 4, 1, ...] \qquad \text{and} \qquad [0; b_1, ..., b_k, 1, 1, 4, 1, 4, ...].$$

Let $\mu = [1; 4, 1, 4, ...] = \zeta$ and $\nu = [1; 1, 4, 1, 4, ...] = 1 + \frac{1}{\zeta}$. Then by Lemma 3.2.7 the length of $T_1(b_0, ..., b_k, 1)$ is

$$t_1 = \frac{1 + \frac{1}{\zeta} - \zeta}{q_k(1 + \frac{1}{\zeta} + \epsilon)(\zeta + \epsilon)}.$$

c. The smallest and largest elements of C are, respectively,

$$[0; b_1, ..., b_k, 1, 1, 4, 1, 4, ...] \qquad \text{and} \qquad [0; b_1, ..., b_k, 2, 4, 1, 4, 1, ...].$$

Let $\mu = [1; 1, 4, 1, 4, ...] = 1 + \frac{1}{\zeta}$ and $\nu = [2; 4, 1, 4, 1, ...] = 2 + \frac{1}{4\zeta}$. Then by Lemma 3.2.7 the length of C is

$$c = \frac{2 + \frac{1}{4\zeta} - 1 - \frac{1}{\zeta}}{q_k(2 + \frac{1}{4\zeta} + \epsilon)(1 + \frac{1}{\zeta} + \epsilon)}.$$

t_2. The smallest and largest elements of $T_2(b_0, ..., b_k)$ are, respectively,

$$[0; b_1, ..., b_k, 2, 4, 1, 4, 1, ...] \qquad \text{and} \qquad [0; b_1, ..., b_k, 4, 1, 4, 1, ...].$$

Let $\mu = [2; 4, 1, 4, 1, ...] = 2 + \frac{1}{4\zeta}$ and $\nu = [4; 1, 4, 1, ...] = 4\zeta$. Then by Lemma 3.2.7 the length of $T_2(b_0, ..., b_k)$ is

$$t_1 = \frac{4\zeta - 2 - \frac{1}{4\zeta}}{q_k(4\zeta + \epsilon)(2 + \frac{1}{4\zeta} + \epsilon)}.$$

Now calculate

$$\frac{c}{t_1} = \frac{2 + \frac{1}{4\zeta} - 1 - \frac{1}{\zeta}}{q_k(2 + \frac{1}{4\zeta} + \epsilon)(1 + \frac{1}{\zeta} + \epsilon)} \frac{q_k(1 + \frac{1}{\zeta} + \epsilon)(\zeta + \epsilon)}{1 + \frac{1}{\zeta} - \zeta} = \frac{(4 - 3\zeta)(\zeta + \epsilon)}{(3\zeta - 3)(1 + \zeta + \epsilon)},$$
$$\frac{c}{t_2} = \frac{(4 - 3\zeta)(4\zeta + \epsilon)}{(3\zeta - 1)(4\zeta - 3 + \epsilon)}.$$

Note that $0 \leq q_{k-1} \leq q_k$, hence $0 \leq \epsilon \leq 1$. $\frac{c}{t_1}$ takes its maximum at $\epsilon = 1$ and $\frac{c}{t_2}$ takes its maximum at $\epsilon = 0$. In both cases

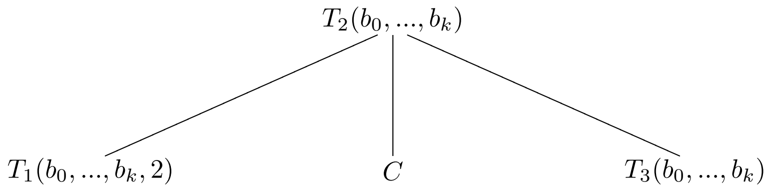
$$\frac{c}{t_1} \leq \frac{(4 - 3\zeta)(\zeta + 1)}{(3\zeta - 3)(2 + \zeta)} < 1$$
$$\frac{c}{t_2} \leq \frac{(4 - 3\zeta)4\zeta}{(4\zeta - 1)(4\zeta - 3)} < 1.$$

Hence $c < t_1$ and $c < t_2$ satisfying the first condition of Corollary 2.2.7.

If k is odd the results are, because of the symmetry in μ and ν in the formula in Lemma 3.2.7, the same.

Case 2 and case 3 are proved similarly to case 1, hence only the results are given. The calculations are left for the reader.

ii. Case 2:

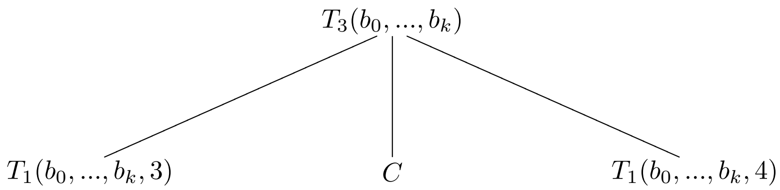


Denote by t_1, c, t_2 the lengths of, respectively, the intervals $T_1(b_0, ..., b_k, 2)$, C and $T_3(b_0, ..., b_k)$.

$$t_1 = \frac{2 + \frac{1}{\zeta} - 2 - \frac{1}{4\zeta}}{q_k(2 + \frac{1}{\zeta} + \epsilon)(2 + \frac{1}{4\zeta} + \epsilon)},$$
$$c = \frac{3 + \frac{1}{4\zeta} - 2 - \frac{1}{\zeta}}{q_k(2 + \frac{1}{\zeta} + \epsilon)(3 + \frac{1}{4\zeta} + \epsilon)},$$
$$t_2 = \frac{4\zeta - 3 - \frac{1}{4\zeta}}{q_k(3 + \frac{1}{4\zeta} + \epsilon)(4\zeta + \epsilon)}.$$

They satisfy $c \leq t_1$, $c \leq t_2$ for all $0 \leq \epsilon \leq 1$.

iii. Case 3:



Denote by t_1, c, t_2 the lengths of, respectively, the intervals $T_1(b_0, ..., b_k, 3)$, C and $T_1(b_0, ..., b_k, 3)$.

$$t_1 = \frac{3 + \frac{1}{\zeta} - 3 - \frac{1}{4\zeta}}{q_k(3 + \frac{1}{\zeta} + \epsilon)(3 + \frac{1}{4\zeta} + \epsilon)},$$
$$c = \frac{4 + \frac{1}{4\zeta} - 3 - \frac{1}{\zeta}}{q_k(3 + \frac{1}{\zeta} + \epsilon)(4 + \frac{1}{4\zeta} + \epsilon)},$$
$$t_2 = \frac{4\zeta - 4 - \frac{1}{4\zeta}}{q_k(4 + \frac{1}{4\zeta} + \epsilon)(4\zeta + \epsilon)}.$$

They satisfy $c \leq t_1$, $c \leq t_2$ for all $0 \leq \epsilon \leq 1$.

Condition 2

(C2) $\frac{1}{3} \leq \frac{a}{b} \leq 3$,

is satisfied trivially, since $\frac{1}{3} \leq \frac{a}{a} \leq 3$.

So Corollary 2.2.7 can be applied, hence $A + A = L(A) + L(A)$. This proves the lemma. □

Theorem 3.2.9. $\text{RCF}_4 + \text{RCF}_4 = \mathbb{R}$.

Proof. This follows from Lemma 3.2.8. Because $L(A) + L(A) = [\sqrt{2} - 1, 4\sqrt{2} - 4]$ every element within $[\frac{1}{2}, \frac{3}{2}]$ can be written as the sum of two elements of $\text{RCF}_4(0)$. Hence

$$\text{RCF}_4 + \text{RCF}_4 \supset \bigcup_{n \in \mathbb{Z}} (\text{RCF}_4(n) + \text{RCF}_4(n)) = \mathbb{R}.$$

□

3.3 Singularization

A large class of continued fraction expansions can be derived from the regular continued fraction expansion by an operation called *singularization* . Here the underlying idea is described. For $a \in \mathbb{Z}$, $b \in \mathbb{Z}_{>0}$ and $x \in [0, 1)$

$$a + \cfrac{1}{1 + \cfrac{1}{b + x}} = a + 1 + \frac{-1}{b + 1 + x}.$$

In this way the digit 1 can be *singularized* in

$$[..., a, 1, b, ...]$$

to

$$[..., a + 1, -1/b + 1, ...]$$

If $x \in [0, 1)$ with RCF expansion

$$x = [0; e_1/a_1, e_2/a_2, e_3/a_3, ...]$$

then any finite or infinite string of consecutive digits

$$a_k = 1, \qquad a_{k+1} = 1, \qquad a_{k_2} = 1, \qquad ... \qquad a_{k+n-1} = 1$$

is called a 1-block if either $k = 1$ and $a_{k+n} \neq 1$ or $k > 1$ and $a_{k-1} \neq 1$, $a_{k+n} \neq 1$.
The following algorithm, [IK02] p257-260, is known to singularize a complete regular continued fraction.

For any $x \in [0, 1)$ singularize the first, third, fifth, etc., components in any 1-block.

Applying this algorithm to a RCF expansion $[0; a_1, a_2, a_3, ...]$ yields a continued fraction of the form

$$b_0 + \cfrac{e_1}{b_1 + \cfrac{e_2}{b_2 + ...}} = [b_0; e_1/b_1, e_2/b_2, ...].$$

where $e_n = \pm 1$, $b_n + e_{n+1} \geq 2$ and $b_n > 1$.
Hence this form of singularization gives a nearest integer continued fraction expansion.

For a better analysis of this algorithm see [IK02].

3.4 $\text{NICF}_6 + \text{NICF}_6$

Lemma 3.4.1. Every 4-bounded Regular Continued Fraction is a 6-bounded Nearest Integer Continued Fraction. i.e. $\text{RCF}_4 \subset \text{NICF}_6$.

Proof. Let $x = [0; a_1, a_2, a_3, ...]$ be a 4-bounded regular continued fraction. Applying the algorithm described above, x can be singularized to a nearest integer continued fraction

$$x = [b_0; e_1/a_1, e_2/a_2, e_3/a_3, ...].$$

It is easy to see that the most extreme case which can occur is the following.

$$\begin{aligned}[..., a, 1, 4, 1, b, ...] &= [..., a+1, -1/5, 1, b, ...] \\ &= [..., a+1, -1/6, -1/b+1, ...]\end{aligned}$$

Hence $x \in \text{NICF}_6$. □

Corollary 3.4.2. $\text{NICF}_6 + \text{NICF}_6 = \mathbb{R}$.

Proof. By Lemma 3.5.1 $\text{RCF}_4 \subset \text{NICF}_6$ and by Theorem 3.2.9 $\text{RCF}_4 + \text{RCF}_4 = \mathbb{R}$. Hence

$$\mathbb{R} = \text{RCF}_4 + \text{RCF}_4 \subseteq \text{NICF}_6 + \text{NICF}_6 \subseteq \mathbb{R}.$$

□

3.5 $\text{CCF}_4 + \text{CCF}_4 + \text{CCF}_4 + \text{CCF}_4$

Lemma 3.5.1. If $a_0 \in \mathbb{Z}$, $a_1, a_2, ... \in \mathbb{Z}\backslash\{0\}$ such that

$$[a_0; a_1, a_2, ...] = x \in \mathbb{R}$$

then

$$[a_0 i; -a_1 i, a_2 i, ..., (-1)^n a_n i, ...] = xi \in \mathbb{R}i.$$

Proof. By induction on n. If $x = a_0 + \frac{1}{x_1}$, then

$$xi = i(a_0 + \frac{1}{x_1}) = a_0 i + \frac{i}{x_1} = a_0 i + \frac{1}{-x_1 i}.$$

Now suppose that $n > 0$ and the induction hypothesis holds, then

$$x_n i = (-1)^n i \left(a_n + \frac{1}{x_{n+1}}\right) = (0)^n a_n i + \frac{(-1)^n i}{x_{n+1}} = (-1)^n a_n i + \frac{1}{(-1)^{n+1} i x_{n+1}}.$$

Hence $xi = [a_0 i; -a_1 i, a_2 i, ..., (-1)^n a_n i, ...]$. □

Theorem 3.5.2. $\text{CCF}_4 + \text{CCF}_4 + \text{CCF}_4 + \text{CCF}_4 = \mathbb{C}$.

Proof. Let $a + bi \in \mathbb{C}$ arbitrary. By Theorem 3.2.9 there exists $x_1, x_2 \in \text{RCF}_4$ such that $x_1 + x_2 = a$ and there exists $y_1, y_2 \in \text{RCF}_4$ such that $y_1 + y_2 = b$. Hence

$$x_1 + x_2 + y_1 i + y_2 i = a + bi.$$

Now suppose that

$$\begin{aligned} y_1 &= [a_0; a_1, a_2, ...], \\ y_2 &= [b_0; b_1, b_2, ...], \end{aligned}$$

where $a_0, b_0 \in \mathbb{Z}$ and $1 \leq a_j, b_k \leq 4$ for all $j, k > 0$. Then by Lemma 3.5.1 also

$$\begin{aligned} y_1 i &= [a_0 i; -a_1 i, a_2 i, ..., (-1)^j a_j i], \\ y_2 i &= [b_0 i; -b_1 i, b_2 i, ..., (-1)^k b_k i]. \end{aligned}$$

Because $|(-1)^j a_j| = |a_j| \leq 4$ and $|(-1)^k b_k i| \leq 4$ it must be that $y_1 i, y_2 i \in \text{CCF}_4$. Since $\text{RCF}_4 \subset \text{CCF}_4$, $x_1, x_2, y_1 i, y_2 i \in \text{CCF}_4$. Because $a + bi$ was arbitrary,

$$\text{CCF}_4 + \text{CCF}_4 + \text{CCF}_4 + \text{CCF}_4 = \mathbb{C}.$$

□

3.6 More results

First some notation. If $n, k > 0$ then

$$n\mathrm{RCF}_k = \overbrace{\mathrm{RCF}_k + \mathrm{RCF}_k + ... + \mathrm{RCF}_k}^{n \text{ times}}.$$

In the same way

$$n\mathrm{CCF}_k = \overbrace{\mathrm{CCF}_k + \mathrm{CCF}_k + ... + \mathrm{CCF}_k}^{n \text{ times}}.$$

There has been done quite some research on the sums of RCF_k. Hall [Hal47] was the first to prove, more or less, that $\mathrm{RCF}_4 + \mathrm{RCF}_4 = \mathbb{R}$. Diviš and Cusick proved independently [Div73, Cus73] that

$$\mathrm{RCF}_3 + \mathrm{RCF}_3 \neq \mathbb{R} \qquad \text{and} \qquad 3\mathrm{RCF}_2 \neq \mathbb{R},$$

but

$$3\mathrm{RCF}_3 = \mathbb{R} \qquad \text{and} \qquad 4\mathrm{RCF}_2 = \mathbb{R}.$$

Later, Hlavka showed [Hla75] that

$$\mathrm{RCF}_4 + \mathrm{RCF}_2 \neq \mathbb{R}, \qquad \text{and} \qquad \mathrm{RCF}_3 + \mathrm{RCF}_2 + \mathrm{RCF}_2 \neq \mathbb{R},$$

however

$$\mathrm{RCF}_4 + \mathrm{RCF}_3 = \mathbb{R}, \quad \mathrm{RCF}_4 + \mathrm{RCF}_2 + \mathrm{RCF}_2 = \mathbb{R}, \quad \mathrm{RCF}_3 + \mathrm{RCF}_3 + \mathrm{RCF}_2 = \mathbb{R}, \quad \mathrm{RCF}_7 + \mathrm{RCF}_2 \neq \mathbb{R}.$$

Astels even went further and proved [Ast00, Ast01, Ast02] that

$$\mathrm{RCF}_5 \pm \mathrm{RCF}_2 = \mathbb{R}, \quad \mathrm{RCF}_3 \pm \mathrm{RCF}_4 = \mathbb{R}, \quad \mathrm{RCF}_3 - \mathrm{RCF}_3 = \mathbb{R}, \quad \mathrm{RCF}_3 \pm \mathrm{RCF}_2 \pm \mathrm{RCF}_2 = \mathbb{R}.$$

Combining these results with Lemma 3.5.1 and the proof of Theorem 3.5.2 one gets an infinite number of equalities.

$$\begin{aligned}
\mathbb{C} &= 3\mathrm{CCF}_3 + 3\mathrm{CCF}_3 \\
&= 3\mathrm{CCF}_3 + 4\mathrm{CCF}_2 \\
&= \mathrm{CCF}_4 + \mathrm{CCF}_3 + 4\mathrm{CCF}_2 \\
&= 2\mathrm{CCF}_3 + 2\mathrm{CCF}_4 \\
&= 2\mathrm{CCF}_3 - 2\mathrm{CCF}_3 \\
&= 4\mathrm{CCF}_3 + 2\mathrm{CCF}_2 \\
&\vdots
\end{aligned}$$

Chapter 4

Hall's Algorithm

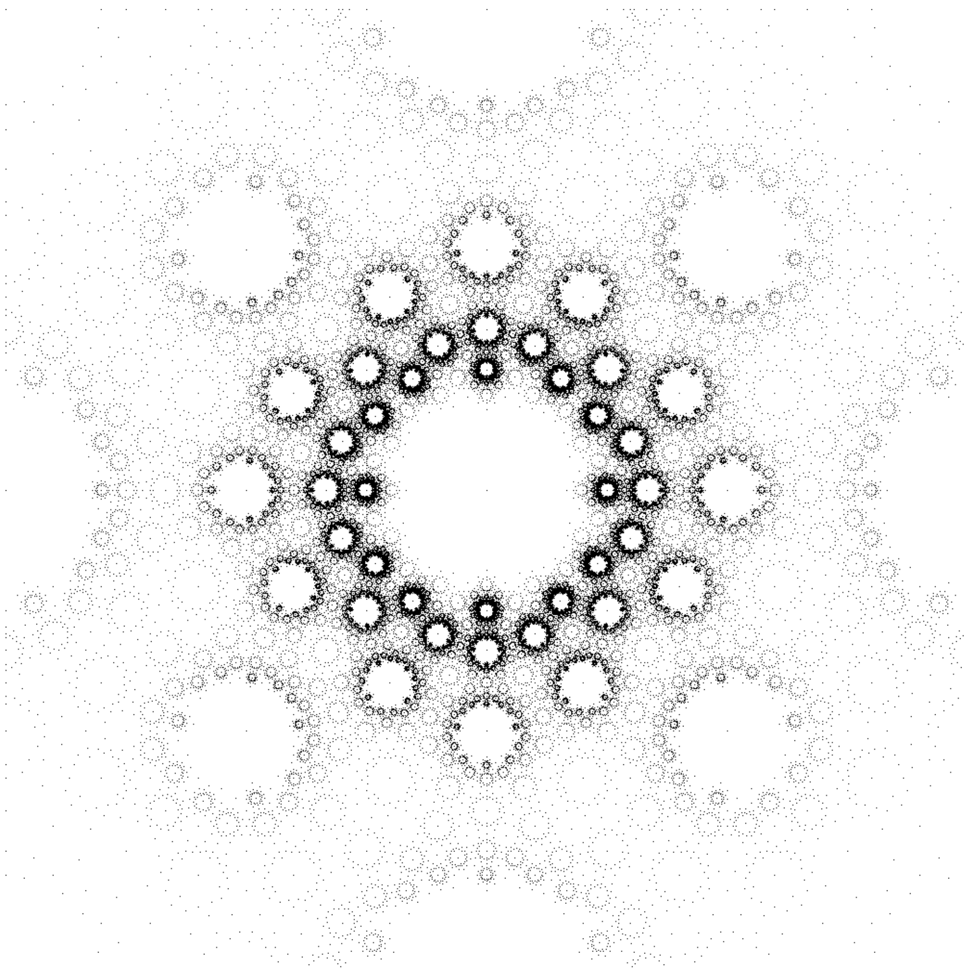


Figure 4.1: $\mathrm{CCF}_4(0)$ in the complex square $[-1,1] \times [-i,i]$.

In this chapter a special algorithm, named after Marshall Hall, is given to do the following. Given a (complex) continued fraction

$$x = [a_0; a_1, a_2, a_3, ...]$$

and a Möbius transformation

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

with $a, b, c, d \in \mathbb{Z}$ (or $\mathbb{Z}[i]$) such that $|ad - bc| > 0$, this algorithm gives an efficient way of calculating

$$y = Mx = \frac{ax + b}{cx + d} = [b_0; b_1, b_2, b_3, ...].$$

The chapter closes with calculating an explicit example for Regular Continued Fractions.

This chapter is a revision of the algorithm described in [Hal47]. There have also been found other ways to solve this solution described above. For Regular Continued Fractions Raney showed in [Ran73] an other algorithm using transducers. Recently, an other student at the Radboud University, tried to generalize this algorithm of Raney to the case of Hurwitz Continued Fractions in this Master Thesis [Lui11].

4.1 Möbius transformations and successors

In this section Möbius transformations and successors are introduced. Also two important finiteness theorems about successors are proved.

Definition 4.1.1. The relation between indeterminate x and y such that

$$y = \frac{ax + b}{cx + d} \tag{4.1}$$

where $a, b, c, d \in \mathbb{Z}[i]$ with $|ad-bc| = N$, $N > 0$, is called a *linear fractional form* or a *Möbius transformation*. In this paper the latter is name is used. N is called the *determinant of the Möbius transformation.* Often throughout this paper the following notation is used for a Möbius transformation

$$y = Mx, \qquad M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \tag{4.2}$$

where $\det(M) = |ad - bc| = N$.

Remark 4.1.2. The definition of det in Definition 4.1.1 is not the usual definition for the determinant. However this notation makes a lot of calculations a bit easier.

Lemma 4.1.3. If $x, x_1, y \in \mathbb{C}$ are such that x and y satisfy (4.2) and there exists $a_0 \in \mathbb{Z}[i]$ with $x = [a_0, x_1]$, then

$$y = \frac{a'x_1 + b'}{c'x_1 + d'} = M'x,$$

with $a', b', c', d' \in \mathbb{Z}[i]$ and $|a'd' - b'c'| = N$. In addition, if $y_1 \in \mathbb{C}$, $b_0 \in \mathbb{Z}[i]$ such that $y = b_0 + \frac{1}{y_1}$, then

$$y_1 = \frac{a''x + b''}{c''x + d''} = M''x$$

with $a'', b'', c'', d'' \in \mathbb{Z}[i]$ and $|a''d'' - b''c''| = N$.

Proof. This is a simple calculation. Let $x = [a_0, x_1]$, then

$$y = \frac{ax + b}{cx + d} = \frac{a(a_0 + \frac{1}{x_1}) + b}{c(a_0 + \frac{1}{x_1}) + d} = \frac{(aa_0 + b)x_1 + a}{(ca_0 + d)x_1 + c}.$$

Hence $|(aa_0 + b)c - a(ca_0 + d)| = |ad - bc| = N$, which proves the first part of the lemma. The second part is done similarly. ☐

Remark 4.1.4. Suppose that $x, y \in \mathbb{C}$ and

$$x = [a_0; a_1, ..., a_{i-1}, x_i], \tag{4.3}$$
$$y = [b_0; b_1, ..., b_{j-1}, y_j], \tag{4.4}$$

are complex continued fraction expansions of x and y such that x_i and y_i are their complete quotients. Also suppose that $y = Mx$, $\det(M) = N$. Then by repeated application of Lemma 4.1.3, $y_i = M'x_j$, where $\det(M') = N$.

Definition 4.1.5. Suppose that

$$x = [a_0; x_1], \qquad\qquad y = [b_0; y_1] \tag{4.5}$$

are expansions of x and y and $y = Mx$, where M is a Möbius transformation, then

i. If $c \neq 0$, then $y = M'x_1$ and $y_1 = M''x$ and the triples (M', x_1, y) and (M'', x, y_1) are called *immediate successors* of the triple (M, x, y).

ii. If $c = 0$, then $y_1 = M'''x_1$ and the triple (M''', x_1, y_1) is called an *immediate successor* of the triple (M, x, y).

Now *successors* of (M, x, y) are defined recursively as follows

1. An immediate successor of (M, x, y) is a *successor* of (M, x, y).
2. An immediate successor of a successor of (M, x, y) is again a *successor*.
3. All successors of (M, x, y) are defined by rules 1 and 2.

Definition 4.1.6. Now let M be a Möbius transformation. The *set of immediate successors* T or *immediate successors* of M is defined as follows

$$T = \bigcup_{x \in \mathbb{C}} \{(M', x', y') : y = Mx, (M', x', y') \text{ is an immediate successor of } (M, x, y)\}.$$

Hence S is the set of all possible immediate successors of M ranging over $x, y \in \mathbb{C}$. An element of T is called an immediate successor of M. A *successor* of M is defined similarly.

a. The set of immediate successors of M are *successors* of M.

b. The set of immediate successors of a successor of M is again a *successor* of M.

c. All successors of M are defined by rules a and b.

Remark 4.1.7. These successors play a very important role in the next theorems, so it is important to understand them well. Given the continued faction expansion of $x \in \mathbb{C}$ and $y = Mx$, then the triple (M, x, y) has at most two immediate successors M' and M''. But the set of all immediate successors of M could be infinite.

Luckily there is a finiteness property for this set of all possible immediate successors. This gives the first main theorem of this chapter.

Theorem 4.1.8. For every Möbius transformation

$$y = \frac{ax + b}{cx + d} = Mx, \qquad\qquad \det(M) = N$$

there exists a *finite* subset S of the immediate successors T of M such that, for every $x, y \in \mathbb{C}$ with $y = Mx$ and

$$x = [a_0; x_1], \qquad\qquad y = [b_0; y_1] \tag{4.6}$$

there is an immediate successor (M', x', y') of (M, x, y) for which $M' \in S$.

Proof. There are two cases.

$c \neq 0$. Take $E_1 = y - \frac{a}{c}$ and $E_2 = x - \frac{d}{c}$. Then

$$|E_1 E_2| = |\frac{ad - bc}{c^2}| = \frac{N}{|c^2|}.$$

Hence $|E_1 E_2| = \frac{N}{|c|^2}$ and therefore

$$|y - \frac{a}{c}| = |E_1| \leq \frac{\sqrt{N}}{|c|} \qquad \text{or} \qquad |x - \frac{d}{c}| = |E_2| \leq \frac{\sqrt{N}}{|c|}.$$

Now let $x = [a_0; x_1] = a_0 + \frac{1}{x_1}$, $y = [b_0; y_1] = b_0 + \frac{1}{y_1}$, this gives

$$|b_0 - \frac{a}{c}| \leq |b_0 - y| + |y - \frac{a}{c}| \leq 1 + \frac{\sqrt{N}}{|c|} \quad \text{or} \quad |a_0 - \frac{d}{c}| \leq |a_0 - x| + |x - \frac{a}{c}| \leq 1 + \frac{\sqrt{N}}{|c|}.$$

Since N is fixed this shows that there are only a finite number of a_0 or b_0. Let T be the set

$$T = \{a_0 : |a_0 - \frac{d}{c}| \leq 1 + \frac{\sqrt{N}}{|c|}\} \cup \{b_0 : |b_0 - \frac{a}{c}| \leq 1 + \frac{\sqrt{N}}{|c|}\}$$

Each of these a_0 and b_0 from T determine immediate successors (M', x_1, y) and (M', x, y_1). Let S be the finite set of these immediate successors.

$c = 0$. Suppose

$$x = [a_0; x_1], \qquad y = [b_0; y_1].$$

A straightforward calculation shows that the immediate successor $x_1 = M'y_1$ is of the form

$$y_1 = \frac{dx_1}{Cx_1 + a}$$

where $C = b + aa_0 - db_0$. Because

$$y = \frac{ax - b}{-d}$$

one has that $b = ax - dy$, hence

$$C = ax - dy - aa_0 + db_0 = \frac{a}{x_1} - \frac{d}{y_1}.$$

But, because $x_1, y_1 \geq 1$, also

$$|C| \leq |a| + |d|,$$

hence the set of immediate successors is finite.

□

Remark 4.1.9. If $c = 0$ in the proof of Theorem 4.1.8 one can even prove a stronger result. Note that in this case $|ad| = N$, hence

$$|a| \leq N \qquad \text{and} \qquad |d| \leq N$$

hence $|C| \leq 2N$. So the immediate successors may be chosen from a finite set depending only on N, rather than M.

Lemma 4.1.10. Given a Mobïus transformation $y = \frac{ax+b}{cx+d} = Mx$, where $c \neq 0$. Suppose $r, s \geq 2$ and

$$x = [a_0; a_1, ..., a_{r-1}, x_r], \qquad y = [b_0; b_1, ..., b_{s-1}, y_s].$$

Let $E_1 = y - \frac{a}{c}$ and $E_2 = x - \frac{d}{c}$. Suppose that $|E_1| < \frac{1}{2}$ or $|E_2| < \frac{1}{2}$. Then there are $i, j \leq 2$ and $a', b', c', d' \in \mathbb{Z}[i]$ with $|c'| < c$ where

$$y_j = \frac{a'x_i + b'}{c'x_i + d'} = M'x_i$$

such that (M', x_i, y_j) is a successor of (M, x, y).

Proof. The proof is given in form of an algorithm.

$|E_1| < \frac{1}{2}$. Let $y = [b_0; y_1]$, now

$$-\frac{1}{2} < |E_1| = |y - \frac{a}{c}| = |b_0 - \frac{a}{c} + \frac{1}{y_1}| < \frac{1}{2}.$$

Let $y_1 = [b_1; y_2]$, there are two cases.

$|b_1| \geq 2$. Then $y_1 \geq 2$, hence

$$-\frac{1}{2} < |b_0 - \frac{a}{c} + \frac{1}{y_1}| < \frac{1}{2}, \qquad -1 < |b_0 - \frac{a}{c}| < 1, \qquad |cb_0 - a| < |c|.$$

Remark now that

$$y_1 = \frac{cx + d}{(a - cb_0)x + (-b - db_0)} = M'x$$

such that (M', x, y_1) is a successor of (M, x, y) which satisfies the properties of the theorem.

$|b_1| = 1$. Then $y_1 = 1 + \frac{1}{y_2}$, hence $y_1 < 2$. Now

$$-\frac{3}{2} < |b_0 - \frac{a}{c}| < 0, \qquad -\frac{1}{2} < |b_0 + 1 - \frac{a}{c}| < 1, \qquad |cb_0 + c - a| < |c|.$$

Now take

$$y_2 = \frac{(a - cb_0)x + (-b - db_0)}{(cb_0 + c - a)x + (db_0 + d + b_0)} = M'x$$

such that (M', x, y_2) as successor (notice here that $b_1 = 1$) of (M, x, y). This successor satisfies the properties of the theorem.

$|E_2| < \frac{1}{2}$. This proof is similar to the case $|E_1| < \frac{1}{2}$.

□

Lemma 4.1.11. Given a Mobïus transformation $y = \frac{ax+b}{cx+d} = Mx$ and $|c| > 2\sqrt{N}$. Suppose $r, s \geq 2$ and

$$x = [a_0; a_1, ..., a_{r-1}, x_r], \qquad y = [b_0; b_1, ..., b_{s-1}, y_s].$$

Then there are $i, j \leq 2$ and $a', b', c', d' \in \mathbb{Z}[i]$ with $|c'| < |c|$ where

$$y_j = \frac{a'x_i + b'}{c'x_i + d'} = M'x_i$$

such that (M', x_i, y_j) is a successor of (M, x, y).

Proof. This is an immediate consequence of the previous lemma. Let $E_1 = y - \frac{a}{c}$ and $E_2 = x - \frac{d}{c}$. Then

$$|E_1 E_2| = \frac{N}{|c|^2} < \frac{1}{2}.$$

Hence $|E_1| < \frac{1}{2}$ or $|E_2| < \frac{1}{2}$. Now apply Lemma 4.1.10. □

Lemma 4.1.12. Given a Möbius transformation $y = \frac{ax+b}{cx+d} = Mx$ with $|ad - bc| = N$. Suppose $r, s \geq 2|c| + 2$ and

$$x = [a_0; a_1, ..., a_{r-1}, x_r], \qquad y = [b_0; b_1, ..., b_{s-1}, y_s].$$

Then there are $i \leq r$, $j \leq s$ and $a', b', c', d' \in \mathbb{Z}$ with

$$y_j = \frac{a'x_i + b'}{c'x_i + d'} = M'x_i$$

such that (M', x_i, y_j) is a successor of (M, x, y) with

$$|a'| \leq 2N + 1, \qquad |b'| \leq 2\sqrt{N}, \qquad |c'| \leq 4N + 2, \qquad |d'| \leq 2N + 1.$$

Proof. By, at most $2|c| - 4\sqrt{N}$, repeated applications of Lemma 4.1.11 one may assume that $|c| \leq 2\sqrt{N}$. If $|E_1| > \frac{1}{2}$ or $|E_2| > \frac{1}{2}$ apply Lemma 4.1.10 until both $|E_1|, |E_2| \leq \frac{1}{2}$. This takes at most $\lfloor 4\sqrt{N} \rfloor$ steps. So we may assume that $|c| \leq 2\sqrt{N}$ and $|E_1|, |E_2| \leq \frac{1}{2}$. There are two cases.

$c = 0$. Then by Theorem 4.1.8 there is an immediate successor (M', x_1, y_1) with

$$y_1 = \frac{a'x_1}{c'x_1 + d'} = M'x_1$$

where, by Remark 4.1.9,

$$|a'| \leq N \leq 2N+1, \qquad |b'| = 0 \leq 2\sqrt{N}, \qquad |c'| \leq 2N \leq 4N+2, \qquad |d'| \leq N \leq 2N+1.$$

This is the successor satisfying the theorem.

$c \neq 0$. Then $1 \leq |c| \leq 2\sqrt{N}$. Let $x = a_0 + \frac{1}{x_1}$ and $y = b_0 + \frac{1}{y_1}$, then

$$y_1 = \frac{a'x_1 + b'}{c'x_1 + d'} = M'x_1$$

where

$$a' = d - a_0 c, \qquad b' = -c, \qquad c' = ca_0b_0 - ab_0 - da_0 + b, \qquad d' = cb_0 - a$$

and (M', x_1, y_1) is a successor of (M, x, y). These four elements satisfy the properties of the theorem.

b': $|b'| = |c| \leq 2\sqrt{N}$.

d': First

$$|b_0 + \frac{1}{y_1} - \frac{a}{c}| = |y - \frac{a}{c}| = |E_1| = |\frac{N}{c^2}E_2| \leq 2\frac{N}{c^2}.$$

Therefore

$$|b_0 - \frac{a}{c}| \leq 2\frac{N}{c^2} + 1$$

and

$$|d'| = |cb_0 - a| \leq 2\frac{N}{|c|} + |c| \leq 2N + 1.$$

a': In a way similar to d'.

c': Because $cxy - dy - ax + b = 0$, also

$$\begin{aligned} c' &= ca_0b_0 - aa_0 - db_0 + b \\ &= \frac{d - ca_0}{y_1} + \frac{a - cb_0}{x_1} \\ &= \frac{a'}{y_1} + \frac{d'}{x_1} \end{aligned}$$

hence

$$|c'| \leq |a'| + |d'| \leq 4N + 2.$$

□

Remark 4.1.13. Notice that every time Lemma 4.1.12 is applied to a Möbius transformation M where $y = Mx$ then the triple (M, x, y) gives a new successor (M', x_i, y_j) where $i, j > 0$. This is very important, because if it is known that

$$x = [a_0; a_1, a_2, ..., a_{i-1}, x_j]$$

then after applying Lemma 4.1.12 it must also be known what the values of $b_0, b_1, ..., b_{j-1}$ in

$$y = [b_0; b_1, b_2, ..., b_{j-1}, y_i].$$

must be. Hence this lemma provides a way to calculate the first $j > 0$ quotients of y given the first i quotients of x.

This is the second main theorem of this chapter. It shows that given a Möbius transformation $y = Mx$ there exists a finite set of successors of $y = Mx$ where it is possible to calculate, given

$$x = [a_0; a_1, a_2, ..., a_{i-1}, x_i]$$

the first j quotients

$$y = [b_0; b_1, b_2, ..., b_{j-1}, y_j].$$

Theorem 4.1.14. Given an integer $N > 0$. There exists a finite set C of forms of determinant N and $m > 0$ such that the following is true.
Let M a Möbius transformation with $\det(M) = N$ with $x \in \mathbb{R}$ and $y = Mx$. Then for all $i > m$ with

$$x = [a_0; a_1, ..., a_{i-1}, x_i]$$

there is $j > 0$ such that

$$y = [b_0; b_1, ..., b_{j-1}, y_j]$$

where $y_j = M'x_i$ and $M' \in C$.

Proof. Define

$$C_0 = \{M \in \mathrm{GL}_2(\mathbb{Z}[i]) : \det(M) = N, |a| \leq 2N + 1, |b| \leq 2\sqrt{N}, |c| \leq 4N + 2, |d| \leq 2N + 1\}.$$

Let $M \in C_0$. Then by Theorem 4.1.8 there exists a finite set S of Möbius transformation of determinant N of M such that for every $x, y \in \mathbb{R}$ with $y = Mx$ there is an immediate successor (M_0, x', y') of (M, x, y) in S. For every $M_0 \in S$ there are, by Lemma 4.1.12, successors

$$(M_1, x_{i_1}, y_{j_1}), (M_2, x_{i_2}, y_{j_2}), (M_3, x_{i_3}, y_{j_3}), ..., (M_k, x_{i_k}, y_{j_k})$$

of determinant N leading from (M, x, y) to (M_k, x_{i_k}, y_{j_k}) with $M_k \in C_0$. Let C_M be the union of all sequences $M_1, M_2, M_3, ..., M_i$ belonging to a $M_0 \in S$. Now define

$$C = C_0 \cup \bigcup_{M \in C_0} C_M.$$

Take $m = 4|c| + 4$. Lemma 4.1.12 explains how to get, given a Möbius transformation M with $y = Mx$, in at most $4|c| + 4$ steps to a Möbius transformation $M' \in C$ with $y_j = M'x_i$. So this m is the desired upper bound and by construction this C is the desired set. ☐

4.2 Hall's Algorithm

Theorem 4.1.14 provides an algorithm to give a finite set of rules to calculate the quotients of y. This set of rules, although it is finite, can become extremely large. In this section the special case is treated when a continued fraction is a regular continued fraction. Some important modifications are made to get this set of rules for regular continued fractions smaller.

Definition 4.2.1. Fix $N > 0$. Any finite set C of forms of determinant N satisfying Theorem 4.1.14 will be called a *canonical set of N-forms.*

Our aim is to find a canonical set of N-forms which is as small as possible. This set can be found if $N = 1$.

Theorem 4.2.2. Suppose $N = 1$, then $C = \{M\}$, where

$$M = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

is a canonical set of 1-forms.

Proof. If $y = x$ and $x = [a_0; x_1], y = [b_0; y_1]$ then also $y_1 = x_1$. So every immediate successor of $(1, x, y)$ is again $(1, x, y)$. Therefore it is sufficient to show that for every Möbius transformation

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \qquad \text{where } ad - bc = \pm 1$$

and $y = Mx$ for $x, y \in \mathbb{R}$ there exists $i, j \geq 0$ such that

$$x = [a_0; a_1, a_2, ..., a_{i-1}, x_i],$$
$$y = [b_0; b_1, b_2, ..., b_{j-1}, y_j]$$

with $y_j = x_i$.

To prove this an algorithm is given. By repeated application of Lemma 4.1.12 one may assume that $|c| < 2\sqrt{N} = 2$. Hence $c = -1, 1$ or 0. The first two cases can be reduced to the last one. Because:

$c = -1$. Multiply the numerator and denominator by -1, this gives the same Mobïus transformation with $c = 1$.

$c = 1$. In this case

$$y = \frac{ax - b}{x - d}, \qquad \text{with } ad - b = \pm 1.$$

But then

$$(x - d)(y - a) = (ax - b) - a(x - d) = ad - b = \pm 1.$$

The last equation shows that there is an immediate successor given by one of the four following transformations

$$\begin{aligned} x &= [d; x_1], & y &= [y], \\ x &= [d - 1; x_1], & y &= [y], \\ x &= [x], & y &= [a; y_1], \\ x &= [x], & y &= [a - 1; y_1]. \end{aligned}$$

They lead respectively to the following Möbius transformations

$$y = \pm x_1 + a, \tag{4.7}$$

$$y = \frac{(a \mp 1)x_1 - a}{x_1 - 1}, \tag{4.8}$$

$$y_1 = \pm(x - d), \tag{4.9}$$

$$y_1 = \frac{x - d}{x - (d \mp 1)} \tag{4.10}$$

where (4.7) and (4.9) have $c = 0$. In (4.8)

$$y = \frac{(a \mp 1)x_1 - a}{x_1 - 1}.$$

Remark that the determinant of this transformation $(a \mp 1) + a = \pm 1$. Hence also $(x_1 - 1)(y - (a \mp 1)) = \pm 1$. This gives one of the following three expansions

$$\begin{aligned} x_1 &= [1; x_2], & y &= [y], \\ x_1 &= [x_1], & y &= [a \mp 1; y_1], \\ x_1 &= [x_1], & y &= [a \mp 1 - 1; y_1]. \end{aligned}$$

These lead respectively to the following Möbius transformations

$$y = \mp x_2 - (1 \pm a),$$

$$y_1 = \frac{x_1 - 1}{\pm 1},$$

$$y_1 = \frac{x_1 - 1}{x_1 - 2}.$$

The first two transformations satisfy $c = 0$. For the last one

$$(x_1 - 2)(y_1 - 1) = 1$$

which leads to one of the two expansions

$$x_1 = [2; x_2], \qquad y_1 = [y_1],$$
$$x_1 = [x_1], \qquad y_1 = [1; y_2].$$

These two expansions, respectively, give transformations

$$y_1 = x_2 + 1,$$
$$y_2 = x_1 - 2$$

both having $c = 0$.

$c = 0$. Because now $ad = \pm 1$ we must have $a = \pm 1$ and $d = \pm 1$ so the Möbius transformation is $y = \pm x - b$. Let

$$x = [a_0; x_1], y = [b_0; y_1]$$

then

$$\frac{1}{y_1} = \pm a_0 \pm \frac{1}{x_1} - b - b_0.$$

As $x_1, y_1 > 0$ and $b, a_0, b_0 \in \mathbb{Z}$ there are only two possibilities: $b = 0$ or $b = 1$. For $b = 0$

$$y_1 = x_1.$$

For $b = 1$,

$$\pm x - 1 = \pm a_0 + \frac{1}{x_1} - 1 = \pm a_0 - 1 \pm 1 + \frac{1}{\frac{x_1}{x_1 - 1}}$$

hence

$$y_1 = \frac{x_1}{x_1 - 1}.$$

The first of these is the successor desired to prove the theorem. In the second case $(x_1 - 1)(y_1 - 1) = \pm 1$, so there is one of the expansions

$$x_1 = [1; x_2], y_1 = [y_1],$$
$$x_1 = [x_1], y_1 = [1; y_2]$$

leading respectively to successors

$$y_1 = x_2 + 1,$$
$$y_2 = x_1 - 1.$$

These leading again respectively to successors

$$y_2 = x_3,$$
$$y_3 = x_2.$$

Transformation (4.10) works in the same fashion as transformation (4.8). This proves the theorem.

□

Remark 4.2.3. The proof of Theorem 4.2.2 gives an explicit algorithm to calculate, given a Möbius transformation M with $\det(M) = \pm 1$ and $x = [a_0; a_1, a_2, ...]$,

$$y = Mx = [b_0; b_1, b_2, ...].$$

Since M can be transformed in a finite number of steps to the Möbius transformation $M = 1$, there exist $j, k \geq 0$ and $b_0, b_1, ..., b_j \in \mathbb{Z}$ such that

$$y = [b_0; b_1, ..., b_j, a_k, a_{k+1}, a_{k+2}, ...].$$

This result gives an alternative proof of a well-known theorem in the theory of continued fractions; also see Theorem 2, pp. 6-8 of [RS92].

Remark 4.2.4. Let $y = Nx$, where $N > 0$ is a Möbius transformation. Applying theorem 4.1.14 to this transformation gives a set C which is a canonical set of N-forms for $y = Nx$. Moreover this set is in most cases sufficiently smaller and therefore easy for practice.

Our goal is to create a set of rules for a sets like C. It is showed now that every arbitrary Möbius transformation $y = Mx$ of determinant $\pm N$ can be transformed to this special Möbius transformation $y = Nx$. First an important algorithm, called *Smith Normal Form*, is explained.

Theorem 4.2.5. (Smith Normal Form) Let $a, b, c, d \in \mathbb{Z}$, $(a, b, c, d) = 1$ and $ad - bc = \pm N$. Then there are $e, f, g, h, t, u, v, w \in \mathbb{Z}$ such that $eh - fg = \pm 1$, $tw - uv = \pm 1$ and

$$\begin{pmatrix} N & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} e & f \\ g & h \end{pmatrix} = \begin{pmatrix} t & u \\ v & w \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Example 4.2.6. No proof will be given here. One could find the original proof in Smith's paper [Smi61]. Instead an example will provide all the details how to find these matrices. Take the following Möbius transformation

$$M = \begin{pmatrix} 3 & -4 \\ 1 & -2 \end{pmatrix}.$$

We will determine matrices $P, Q \in \mathrm{GL}_2(\mathbb{Z})$ such that

$$PMQ = D = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}.$$

M can always be brought to diagonal form D with elementary row and column operations. Let P_0 and Q_0 be the identity matrices and $M_0 = M$. Every time we apply a row operation on M we also apply the same row operation on P_0, every time we apply a column operation on M we also apply the same column operation on Q_0. So

$$P_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, M_0 = \begin{pmatrix} 3 & -4 \\ 1 & -2 \end{pmatrix}, Q_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Add the first column of M_0 to the second column

$$P_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, M_1 = \begin{pmatrix} 3 & -1 \\ 1 & -1 \end{pmatrix}, Q_1 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

Now add the second column of M_0 to the first column

$$P_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, M_2 = \begin{pmatrix} 2 & -1 \\ 0 & -1 \end{pmatrix}, Q_2 = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}.$$

Then subtract the second row from the first row

$$P_3 = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}, M_3 = \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}, Q_3 = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}.$$

As last subtract the second column 2 times from itself

$$P = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}, D = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, Q = \begin{pmatrix} 2 & -1 \\ 1 & -1 \end{pmatrix}.$$

Because matrices P and Q consist only of elementary operations their determinant is ± 1. Also $Q^{-1} \in \mathrm{GL}_2(\mathbb{Z})$, so a solution is

$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 1 & -2 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 3 & -4 \\ 1 & -1 \end{pmatrix}.$$

The Algorithm

Given a Möbius transformation $y = Mx$ where

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

and $ad - bc = \pm N$. Determine, with Theorem 4.2.5, $e, f, g, h, t, u, v, w \in \mathbb{Z}$ such that

$$\begin{pmatrix} N & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} e & f \\ g & h \end{pmatrix} = \begin{pmatrix} t & u \\ v & w \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

$eh - fg = \pm 1$ and $tw - uv = \pm 1$.

Now define

$$x' = \frac{ex + f}{gx + h}, \qquad y' = \frac{ty + u}{vy + w},$$

then

$$y' = Nx'.$$

Now, given the continued fraction of

$$x = [a_0; a_1, a_2, ...],$$

then, by the algorithm of Theorem 4.2.2, there is an efficient way to calculate

$$x' = [a'_0; a'_1, a'_2, ...].$$

Since $y' = Nx'$ there is, by Remark 4.2.4, an efficient algorithm to calculate

$$y' = [b'_0; b'_1, b'_2, ...].$$

But then again by Theorem 4.2.2 one can calculate

$$y = [b_0; b_1, b_2, ...].$$

4.3 An example for $N = 2$

Here we calculate the following example for $N = 2$. Let

$$M = \begin{pmatrix} 3 & -4 \\ 1 & -2 \end{pmatrix}$$

such that $y = Mx$. By Example 4.2.6

$$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 1 & -2 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 3 & -4 \\ 1 & -2 \end{pmatrix}.$$

and so putting

$$x' = \frac{x - 1}{x - 2}, \qquad y' = \frac{y - 1}{1}$$

gives

$$y' = 2x'.$$

Now use Theorem 4.2.2 to find explicit calculations. As an example we show one calculation.

Example 4.3.1. Since
$$x' = \frac{x-1}{x-2}$$
$c = 1$, $a = 1$ and $d = 2$. Also $(x-2)(x'-1) = 1$, which gives the following 4 possible continued fractions

$$\begin{aligned} x &= [2; x_1], & x' &= [x'] \\ x &= [1; x_1], & x' &= [x'] \\ x &= [x], & x' &= [1; x_1'] \\ x &= [x], & x' &= [0; x_1']. \end{aligned}$$

Suppose the first continued fraction occurs. Now $x = 2 + \frac{1}{x_1}$, hence

$$x' = \frac{x-1}{x-2} = \frac{2 + \frac{1}{x_1} - 1}{2 + \frac{1}{x_1} - 2} = x_1 + 1.$$

Now let $x_1 = [a_1; x_2]$ and $x' = [b_0; x_1']$. Then

$$b_0 + \frac{1}{x_1'} = x' = x_1 + 1 = a_1 + 1 + \frac{1}{x_2}$$

hence

$$a_1 + 1 = b_0, \qquad \text{and} \qquad x_1' = x_2.$$

This gives the following rule
$$x = [2; a, x_2] \to x' = [a+1; x_2].$$
For example if
$$x = [2; 3, 4, 5, 6, 7, 8, 9, ...]$$
then
$$x' = \frac{x-1}{x-2} = [3; 3, 4, 5, 6, 7, 8, 9, ...].$$
In this same fashion all other possibilities can be calculated.

The next step is to calculate
$$y' = 2x'.$$
By Remark 4.2.4 there is a canonical set $\mathcal{C}$ generated by $y' = Ax'$ where
$$A = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$$

Example 4.3.2. As an example we calculate the first step. Let

$$x' = [a_0; x_1'], \qquad y' = [b_0; y_1'].$$

Hence

$$b_0 + \frac{1}{y_1'} = y' = 2x' = 2a_0 + \frac{2}{x_1'}. \tag{4.11}$$

By Theorem 4.1.8 the relation between y_1' and x_1' is

$$y_1' = \frac{dx_1}{Cx_1 - a}$$

where $C = b - aa_0 - db_0 = -2a_0 - db_0$. Now there are two cases

$\frac{1}{x_1} < \frac{1}{2}$. In this case $0 < \frac{2}{x_1} < 1$. So equation (4.11) implies that $b_0 = 2a_0$. So $C = 0$ and the successor is

$$y_1' = \frac{-x_1}{-2} = \frac{x_1}{2} = A^{-1}x.$$

$\frac{1}{x_1} \geq \frac{1}{2}$. In this case $1 \leq \frac{2}{x_1} < 2$. Hence by equation (4.11) $b_0 = 2a_0 + 1$. So then $C = 1$ and the successor becomes

$$y_1' = \frac{-x_1}{x_1 - 2} = \frac{x_1}{-x_1 + 2} = Bx,$$

where

$$B = \begin{pmatrix} 1 & 0 \\ -1 & 2 \end{pmatrix}.$$

So $A^{-1}, B \in \mathcal{C}$.

Successively applying lemma 4.1.12 finally produces the complete canonical set $\mathcal{C}$. Here the set $\mathcal{C}$ is given

$$\begin{array}{llll}
y' = Ax', & y' = A^{-1}x', & A = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, & A^{-1} = \begin{pmatrix} 0 & 1 \\ 0 & 2 \end{pmatrix}, \\
y' = Bx', & y' = B^{-1}x', & B = \begin{pmatrix} 1 & 0 \\ -1 & 2 \end{pmatrix}, & B^{-1} = \begin{pmatrix} 2 & 0 \\ 1 & 1 \end{pmatrix}, \\
y' = Cx', & y' = C^{-1}x', & C = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, & C^{-1} = C, \\
y' = Dx', & y' = D^{-1}x', & D = \begin{pmatrix} 2 & 1 \\ 0 & 1 \end{pmatrix}, & D^{-1} = \begin{pmatrix} 1 & -1 \\ 0 & 2 \end{pmatrix}, \\
y' = Ex', & y' = E^{-1}x', & E = \begin{pmatrix} 3 & 1 \\ 1 & 1 \end{pmatrix}, & E^{-1} = \begin{pmatrix} -1 & 1 \\ 1 & -3 \end{pmatrix}, \\
y' = Fx', & y' = F^{-1}x', & F = \begin{pmatrix} 1 & 0 \\ 2 & -2 \end{pmatrix}, & F^{-1} = \begin{pmatrix} 2 & 0 \\ 2 & -1 \end{pmatrix}, \\
y' = Gx', & y' = G^{-1}x', & G = \begin{pmatrix} 2 & -1 \\ 1 & 0 \end{pmatrix}, & G^{-1} = \begin{pmatrix} 0 & -1 \\ 1 & -2 \end{pmatrix}.
\end{array}$$

If $y' > 1$ or $x' > 1$ the forms B, E, F and their inverses have only one possible successor. But in these six cases always $y' > 1$ and $x' > 1$. So

B. if $y' > 1$, then $x' = 2\frac{y'}{y'+1}$ and so $1 < x' < 2$. Therefore

$$x' = [1; x_1'], \qquad\qquad y' = [y']$$

and

$$y' = \frac{x'}{-x' + 2} = \frac{1 + \frac{1}{x_1'}}{-1 - \frac{1}{x_1'} + 2} = \frac{x_1' + 1}{x_1' - 1} = Cx_1.$$

Hence if we are in B we can immediate transform to C.

E. This transforms into C with $x' = [x']$ and $y' = [2; y_1']$.

F. This transforms into G^{-1} with $x' = [1; x_1']$ and $y = [y]$.

Now it is sufficient to look at the set $\{A, A^{-1}, C, D, D^{-1}, G, G^{-1}\}$ and find a set of rules transforming one state to an other.

Example 4.3.3. As an example we look (again) at transformation $y' = Ax'$. In this setting

$$b_0 + \frac{1}{y_1'} = 2a_0 + \frac{2}{x_1'}$$

and there are two possible new transformations between x_1' and y_1'.

A^{-1}. In this case $\frac{2}{x_1'} < 1$, so $x_1' > 2$ and $b_0 = 2a_0$. This leads to the rule

$$x' = [a; x_1'], x_1' > 2, \qquad \rightarrow \qquad y' = [2a; y_1'], \qquad A \rightarrow A^{-1}.$$

B. Which happens if $1 \leq \frac{2}{x'_1} < 2$, so $b_0 = 2a_0 + 1$. This leads to the rule

$$x' = [a; x'_1], x'_1 < 2 \qquad \rightarrow \qquad y' = [2a+1; y'_1], \qquad A \rightarrow B.$$

But by previous observations case B can be immediately transformed to case C. Therefore the rule becomes

$$x' = [a; 1, x'_2] \qquad \rightarrow \qquad y' = [2a+1; y'_1], \qquad A \rightarrow C.$$

The complete set of rules is given here:

$A \mapsto A^{-1}$,	$x' = [a; x'_1], x'_1 > 2$	$\rightarrow$	$y' = [2a; y'_1]$
$A \mapsto C$,	$x' = [a, 1; x'_2]$	$\rightarrow$	$y' = [2a+1; y'_1]$
$A^{-1} \mapsto A$,	$x' = [2a; x'_1]$	$\rightarrow$	$y' = [a; y'_1]$
$A^{-1} \mapsto C$,	$x' = [2a+1; x'_1]$	$\rightarrow$	$y' = [a; 1, y'_2]$
$C \mapsto D$,	$x' = [1; x'_1]$	$\rightarrow$	$y' = [y']$
$C \mapsto C$,	$x' = [2; x'_1]$	$\rightarrow$	$y' = [2; y'_1]$
$C \mapsto D^{-1}$,	$x' = [x'], x' > 2$	$\rightarrow$	$y' = [1; y'_1]$
$D \mapsto G^{-1}$,	$x' = [a, 1; x'_2]$	$\rightarrow$	$y' = [2a+2; y'_1]$
$D \mapsto A^{-1}$,	$x' = [a; x'_1], x'_1 > 2$	$\rightarrow$	$y' = [2a+1; y'_1]$
$D^{-1} \mapsto A$,	$x' = [2a+1; x'_1]$	$\rightarrow$	$y' = [a; y'_1]$
$D^{-1} \mapsto G$,	$x' = [2a+2; x'_1]$	$\rightarrow$	$y' = [a; 1, y'_2]$
$G \mapsto C$,	$x' = [a; 1, x'_2]$	$\rightarrow$	$y' = [2a; y'_1]$
$G \mapsto A^{-1}$,	$x' = [a; x'_1], x'_1 > 2$	$\rightarrow$	$y' = [2a-1; y'_1]$
$G^{-1} \mapsto A$,	$x' = [2a-1; x'_1]$	$\rightarrow$	$y' = [a; y'_1]$
$G^{-1} \mapsto C$,	$x' = [2a; x'_1]$	$\rightarrow$	$y' = [a; 1, y'_2]$

Finally a set of rules for the transformation

$$y = y' + 1$$

can be found using again theorem 4.2.2. It is, of course

$$y' = [a; y'_1] \rightarrow y = [a+1; y'_1].$$

Example 4.3.4. Consider the following continued fraction of the Euler constant e

$$x = e = [2; 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 8, ...].$$

Suppose we are interested in the first few quotients of continued fraction

$$y = \frac{3x-4}{1-2x}.$$

Then first

$$x' = \frac{x-1}{x-2} = [2; 2, 1, 1, 4, 1, 1, 6, 1, 1, 8, ...].$$

The next step is to calculate

$$y' = 2x'.$$

This can be done with the rules explained above. Here the important quotients of x'_i which are used to calculate a few quotients of y'_j are underlined.

$$\begin{aligned}
x'_1 &= [\underline{2}; 1, 1, 4, 1, 1, 6, 1, 1, 8, ...], & y' &= [4; y'_1], & A &\mapsto A^{-1}, \\
x'_2 &= [\underline{1; 1}, 4, 1, 1, 6, 1, 1, 8, ...], & y' &= [4; 1, y'_2], & A^{-1} &\mapsto A, \\
x'_4 &= [4; 1, 1, 6, 1, 1, 8, ...], & y' &= [4; 1, 3, y'_3], & A &\mapsto C, \\
x'_4 &= [\underline{4}; 1, 1, 6, 1, 1, 8, ...], & y' &= [4; 1, 3, 1, y'_4], & C &\mapsto D^{-1}, \\
x'_5 &= [1; 1, 6, 1, 1, 8, ...], & y' &= [4; 1, 3, 1, 1, 1, y'_6], & & \\
& & &\vdots & &
\end{aligned}$$

Hence $y' = [4; 1, 3, 1, 1, 1, ...]$. The last step is

$$y = y' + 1 = [5; 1, 3, 1, 1, 1, ...].$$

As one can check, these are indeed the first six quotients of the number

$$\frac{3e - 4}{e - 2}.$$

Chapter 5

Applications

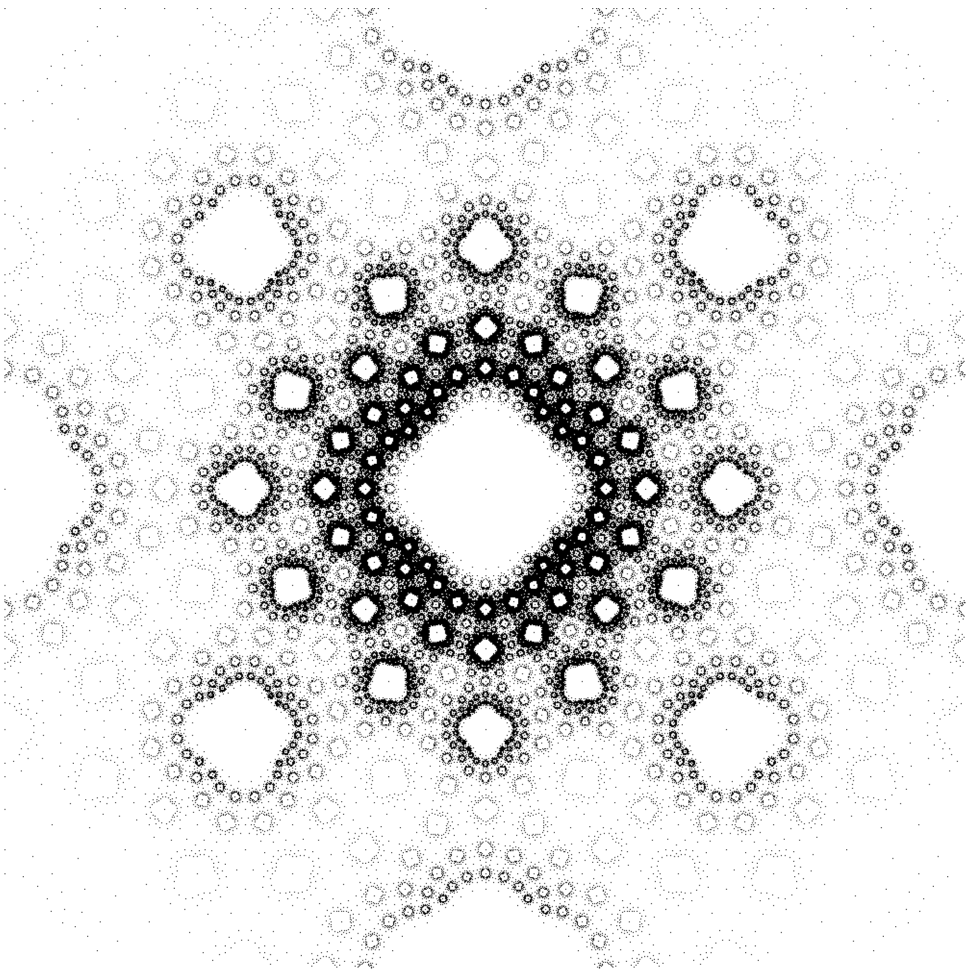


Figure 5.1: $\mathrm{CCF}_5(0)$ in the complex square $[-1, 1] \times [-i, i]$.

In this chapter two applications of the theory developed in Chapter 3 and Chapter 4 are discussed. First, an explicit algorithm is given to calculate, given a real number x, two elements a, b with partial quotients between 1 and 4 such that $x = a + b$. Second, a connection is given between Hall's theorem, described in Chapter 3, and Hall's Algorithm described in Chapter 4.

5.1 Explicit calculation of Hall Sums

Theorem 2.2.6 does not provide an algorithm to calculate, given a real number x, two elements a, b of the General Cantor sets such that $x = a + b$. However, there are special cases where it is possible to find these two elements. Here the case of Theorem 3.2.9 is discussed.

Suppose that $x \in [\sqrt{2}-1, 4\sqrt{2}-4]$, then

$$x \in [\sqrt{2}-1, 4\sqrt{2}-4] = \underline{[T_1(), T_1()]} \cup \overline{[T_1(), T_1()]}. \tag{5.1}$$

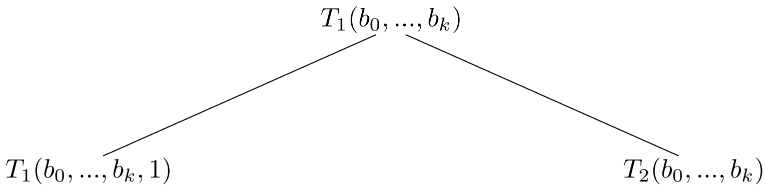
Now by Lemma 4.1.10 this x must be in one of the sets

$$\overline{[T_1(), T_2()]}, \underline{[T_1(), T_2()]}, \overline{[T_1(1), T_1()]}, \underline{[T_1(1), T_2()]}. \tag{5.2}$$

Hence

$$x \in \underline{[T_1(), T_2()]} \cup \overline{[T_1(), T_2()]}, \qquad \text{or} \qquad x \in \underline{[T_1(1), T_1()]} \cup \overline{[T_1(1), T_2()]}, \tag{5.3}$$

because the beginning of the tree for the Cantor Set is



In this way one can continue finding elements $\underline{[A_i, B_j]} \cup \overline{[A_i, B_j]}$ containing x. This process should be seen as follows. Take two trees A and B and point at the roots A_0 and B_0. Suppose the pointer points to A_i and B_j. The next step should move the pointer of A_i to one of his leaves A_k or it should move the pointer of B_j to one of his leaves B_l. This is done is such a way that $x \in \underline{[A_k, B_j]} \cup \overline{[A_k, B_j]}$ (resp. $x \in \underline{[A_i, B_l]} \cup \overline{[A_i, B_l]}$).

Suppose $b_0, ..., b_k \in \mathbb{Z}$ such that $1 \leq b_i \leq 4$ for all b_i. Let us take a look at the structure of the trees in Theorem 3.2.9.

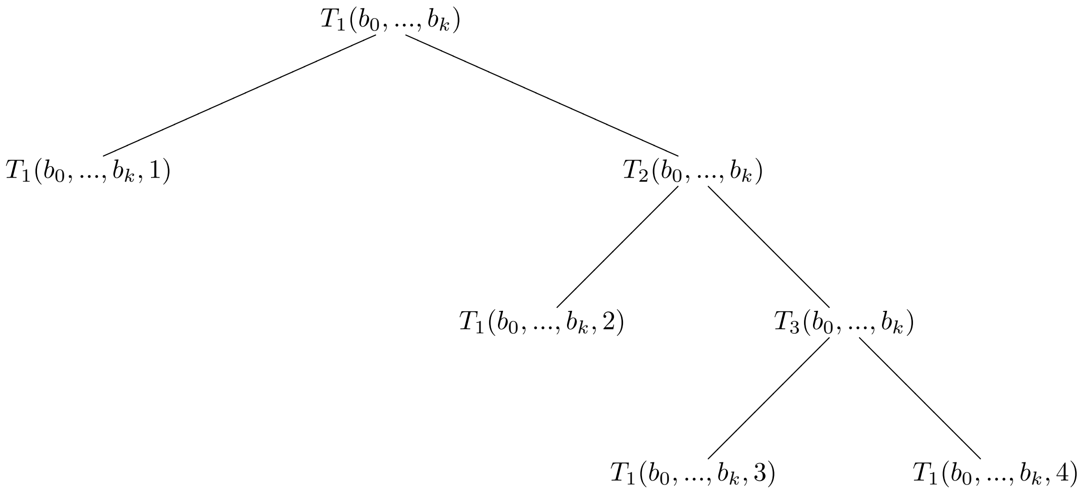


Figure 5.2: A part of the Cantor Tree of Theorem 3.2.9.

Suppose $x \in \underline{[T_i(a_0, ..., a_k), T_j(b_0, ..., b_l)]} \cup \overline{[T_i(a_0, ..., a_k), T_j(b_0, ..., b_l)]}$. As seen in this tree there are two cases.

1.
$$x \in \underline{[T_{i+1}(a_0,...,a_k),T_j(b_0,...,b_l)]} \cup [T_{i+1}(a_0,...,a_k),T_j(b_0,...,b_l)]$$
or
$$x \in \underline{[T_i(a_0,...,a_k),T_{j+1}[b_0,...,b_l]} \cup \overline{[T_i(a_0,...,a_k),T_{j+1}[b_0,...,b_l]}.$$

2.
$$x \in \underline{[T_1(a_0,...,a_k,a),T_j(b_0,...,b_l)]} \cup \overline{[T_1(a_0,...,a_k,a),T_j(b_0,...,b_l)]}$$
or
$$x \in \underline{[T_i(a_0,...,a_k),T_1(b_0,...,b_l,b)]} \cup \overline{[T_i(a_0,...,a_k),T_1(b_0,...,b_l,b)]},$$
where $1 \leq a, b \leq 4$.

Now remark that, looking at the tree in Figure 5.2, case 2 will happen arbitrarily often. This is because after at most 5 steps Case 2 always occures once.
Also remark that the quotients $a_0,...,a_k$ and $b_0,...,b_l$ give an approximation for the numbers a and b. I.e, $x \approx [0; a_0,...,a_k] + [0; b_0,...,b_l]$. Hence if

$$\underline{[A_0,B_0]} \cup \overline{[A_0,B_0]}, \underline{[A_1,B_1]} \cup \overline{[A_1,B_1]}, \underline{[A_2,B_2]} \cup \overline{[A_2,B_2]}, ... \tag{5.4}$$

is the result of the process described above and if in addition $\lim l(A_i) = 0$ and $\lim l(B_j) = 0$ then A_i converges to a point a and B_j convergence to a point b such that $x = a + b$. Also all coefficients are known, because $\lim A_i = T_1(a_0, a_1, a_2, ...)$ and $\lim B_j = T_1(b_0, b_1, b_2, ...)$.

Hence it is sufficient to prove that $\lim l(A_i) = 0$ and $\lim l(B_j) = 0$. Because Case 2 occurs arbitrary often, one may assume that one of A_i or B_j convergences to one point. Without loss of generality suppose $\lim l(B_j) = 0$.
Suppose $A_i = [a_1^i, a_2^i]$, $B_i = [b_1^i, b_2^i]$ and $e_i = \min(l(A_i), l(B_i))$; because $\lim l(B_j) = 0$ it must be that $\lim e_i = 0$. Therefore

$$\lim l(\underline{[A_i,B_i]}) = \lim l([a_1^i + b_1^i, a_1^i + b_1^i + 2e_i]) = \lim 2e_i = 0, \tag{5.5}$$
$$\lim l(\overline{[A_i,B_i]}) = \lim l([a_2^i + b_2^i - 2e_i, a_2^i + b_2^i]) = \lim 2e_i = 0. \tag{5.6}$$

In addition by Theorem 3.2.9

$$A_i + B_i = \underline{[A_i,B_i]} \cup \overline{[A_i,B_i]} = [a_1^i + b_1^i, a_2^i + b_2^i]. \tag{5.7}$$

But that means that $\lim l(A_i + B_i) = 0$ from where one concludes that $\lim B_i = 0$.
This shows that there is an algorithm for, given x a real number between $\sqrt{2} - 1$ and $4\sqrt{2} - 4$, to find an arbitrary number $a_0,...,a_k$ and $b_0,...,b_l$ such that

$$x \approx [0; a_0,...,a_k] + [0; b_0,...,b_l]. \tag{5.8}$$

5.2 Sage source

Using the findings of the previous section one can make a program to calculate these quotients. The source presented here is written in Sage, a free open source mathematics software.

The first function is used to check the results of the calculations of Hall Sums.

```
def cf2elm(continued_fraction):
    '''cf2elm([..]) : input finite continued fraction, output element of the
                      ring with unity.'''

    if len(continued_fraction) == 0:
        return 0
    elif len(continued_fraction) == 1:
        return continued_fraction[0]
    else:
        rest = cf2elm(continued_fraction[1:])
        if rest == 0:
            return continued_fraction[0]
        else:
            return continued_fraction[0] + 1/rest
```

As seen in theorem 3.2.9 the ζ constant is very important to find the boundries of the Hall intervals.

```
# The constant zeta = [1,4,1,4,1,4,1,4,...]
zeta = 1/2 * (sqrt(2) + 1)
```

These three functions calculate the Hall intervals.

```
def T1(l=[]):
    '''T1([b1,b2,...,bk]), returns first Hall set with begin [0;b1,...,bk].'''

    e1 = cf2elm([0] + l + [zeta])
    e2 = cf2elm([0] + l + [4*zeta])

    return min(e1, e2), max(e1, e2)

def T2(l=[]):
    '''T2([b1,b2,...,bk]), returns second Hall set with begin [0;b1,...,bk].'''

    e1 = cf2elm([0] + l + [4*zeta])
    e2 = cf2elm([0] + l + [2, 4*zeta])

    return min(e1, e2), max(e1, e2)

def T3(l=[]):
    '''T3([b1,b2,...,bk]), returns third Hall set with begin [0;b1,...,bk].'''

    e1 = cf2elm([0] + l + [3, 4*zeta])
    e2 = cf2elm([0] + l + [4*zeta])

    return min(e1, e2), max(e1, e2)
```

Then put the three functions together. This programming style is just a matter of taste.

```
# Groups all three Hall gaps as one function list.
T = [T1, T2, T3]
```

Here $\underline{[[a_1, a_2], [b_1, b_2]]}$ and $\overline{[[a_1, a_2], [b_1, b_2]]}$ are implemented.

```
def under((a1, a2), (b1, b2)):
    '''under((a1,a2),(b1,b2)), returns the underline interval of [a1,a2] + [b1,b2].'''

    e = min(a2 - a1, b2 - b1)

    return a1 + b1, a1 + b1 + 2*e

def over((a1, a2), (b1, b2)):
    '''over((a1,a2),(b1,b2)), returns the overline interval of [a1,a2] + [b1,b2].'''

    e = min(a2 - a1, b2 - b1)

    return a2 + b2 - 2*e, a2 + b2
```

Check if elm is in the interval $[a_1, a_2]$. This function can probably be optimized in a lot of ways.

```
def ein(elm, (a1, a2)):
    '''ein(elm, (a1, a2)), check of elm is in closed interval [a1,a2].'''

    return bool(a1 <= elm and elm <= a2)
```

This is the implementation of the tree seen in chapter 3 for $\mathrm{RCF}_4(0)$.

```
def getchildren(case, prox):
    '''getchildren(case, prox), walks through the General Cantor set of RCF4(0).
       case gives the place in the tree and prox the quotients. Hence it calculates
       the children of T_case(prox).'''

    if case == 1:
        return (1, prox + [1]), (2, prox)
    elif case == 2:
        return (1, prox + [2]), (3, prox)
    elif case == 3:
```

```
        return (1, prox + [3]), (1, prox + [4])
    else:
        return None, None
```

This is where it all happens. This function checks in which interval elm is and returns the first of these intervals.

```
def next(elm, ((acase, aprox), (bcase, bprox))):
    '''next(elm, ((acase, aprox), (bcase, bprox))), returns the next two Hall sets
       (a1case, a1prox) and (bcase, bprox) such that elm is an element of the sum
       of two elements of these Hall sets.'''

    (a1case, a1prox), (a2case, a2prox) = getchildren(acase, aprox)
    (b1case, b1prox), (b2case, b2prox) = getchildren(bcase, bprox)

    if ein(elm, under(T[acase-1](aprox), T[b1case-1](b1prox))) \
            or ein(elm, over(T[acase-1](aprox), T[b1case-1](b1prox))):
        return (acase, aprox), (b1case, b1prox)

    elif ein(elm, under(T[acase-1](aprox), T[b2case-1](b2prox))) \
            or ein(elm, over(T[acase-1](aprox), T[b2case-1](b2prox))):
        return (acase, aprox), (b2case, b2prox)

    elif ein(elm, under(T[a1case-1](a1prox), T[bcase-1](bprox))) \
            or ein(elm, over(T[a1case-1](a1prox), T[bcase-1](bprox))):
        return (a1case, a1prox), (bcase, bprox)

    else:
        return (a2case, a2prox), (bcase, bprox)
```

Calculate the first couple of steps and return the result.

```
def findsum(elm, steps):
    '''findsum(elm, steps), calculates at least the first steps/6 quotients of
    x = [0;x1,x2,x3,...] and y = [0;y1,y2,y3,...] such that elm = x + y.
    elm must be in T1([])+T1([]).'''

    state = ((1,[]), (1,[]))

    for i in xrange(steps):
        state = next(elm, state)

    return state[0][1], state[1][1]
```

A couple of examples.

```
if __name__ == '__main__':
        x, y = findsum(sqrt(2), 50)
        print [0]+x, '=', cf2elm([0]+x)
        print [0]+y, '=', cf2elm([0]+y)
        print cf2elm([0]+x), '+', cf2elm([0]+y), '=', float(cf2elm([0]+x) + cf2elm([0]+y))
        print float(sqrt(2))

        x, y = findsum(sqrt(3)-1,50)
        print [0]+x, '=', cf2elm([0]+x)
        print [0]+y, '=', cf2elm([0]+y)
        print cf2elm([0]+x), '+', cf2elm([0]+y), '=', float(cf2elm([0]+x) + cf2elm([0]+y))
        print float(sqrt(3))

        x, y = findsum((1 + sqrt(5))/2, 100)
        print [0]+x, '=', cf2elm([0]+x)
        print [0]+y, '=', cf2elm([0]+y)
        print cf2elm([0]+x), '+', cf2elm([0]+y), '=', float(cf2elm([0]+x) + cf2elm([0]+y))
        print float((1 + sqrt(5))/2)

        x, y = findsum(euler_gamma,50)
        print [0]+x, '=', cf2elm([0]+x)
        print [0]+y, '=', cf2elm([0]+y)
        print cf2elm([0]+x), '+', cf2elm([0]+y), '=', float(cf2elm([0]+x) + cf2elm([0]+y))
        print float(euler_gamma)
```

```
        x, y = findsum(e-2,50)
        print [1]+x, '=', cf2elm([1]+x)
        print [1]+y, '=', cf2elm([1]+y)
        print cf2elm([1]+x), '+', cf2elm([1]+y), '=', float(cf2elm([1]+x) + cf2elm([1]+y))
        print float(e)

        x, y = findsum(pi-2,50)
        print [1]+x, '=', cf2elm([1]+x)
        print [1]+y, '=', cf2elm([1]+y)
        print cf2elm([1]+x), '+', cf2elm([1]+y), '=', float(cf2elm([1]+x) + cf2elm([1]+y))
        print float(pi)
```

Running the program gives the following nice results.

$$\begin{aligned}
\sqrt{2} &= [0; 1, 3, 2, 1, 1, 4, 4, 1, 1, 4, 1, 3, ...] + [0; 1, 1, 1, 3, 1, 4, 1, 1, 1, 2, 1, 2, 1, 4, 1, 1, 3, ...], \\
\sqrt{3} &= [1; 2, 1, 4, 1, 2, 4, 1, 1, 1, 1, 4, 1, 4, ...] + [0; 2, 1, 1, 1, 4, 1, 1, 1, 2, 1, 1, 1, 3, 1, 2, 3, 1, 1, ...], \\
\frac{1+\sqrt{5}}{2} &= [0; 1, 4, 1, 4, 4, 2, 1, 3, 1, 3, 1, 4, 1, 4, ...] + [0; 1, 3, 1, 3, 3, 1, 1, 1, 1, 1, 1, 1, 3, 1, 1, 2, 1, ...], \\
\gamma &= [0; 4, 1, 2, 1, 4, 3, 1, 1, 3, 1, 4, 1, 1, ...] + [0; 2, 1, 2, 1, 2, 1, 1, 2, 1, 2, 1, 3, 1, 1, 1, 3], \\
e &= [1; 2, 1, 3, 1, 1, 1, 4, 2, 1, 4, 1, 2, 3, ...] + [1; 2, 1, 3, 1, 1, 1, 2, 1, 1, 1, 1, 1, 4, 1, 1, 2, 2, ...], \\
\pi &= [1; 2, 4, 1, 4, 2, 1, 1, 1, 4, 1, 1, 1, 1, 4, ...] + [1; 1, 2, 4, 1, 2, 1, 1, 1, 1, 1, 3, 3, 1, 2, 4, ...].
\end{aligned}$$

5.3 An Application of Hall's Algorithm

In [Hal47] a couple of applications of Hall's Algorithm are given. It is nice to mention one of them. This one combines Hall's Theorem and Hall's Algorithm.

Definition 5.3.1. A continued fraction is called *periodic* if the quotients repeat after a while. I.e, if

$$x = [a_0; a_1, a_2, a_3, ...]$$

there are $k > 0$ such that for all $m > 0$ and $n > 0$

$$a_{k+n} = a_{mk+n}.$$

Suppose that x is pericodic and k minimal, then the continued fraction of x is then denoted by

$$x = [a_0; a_1, a_2, ..., \overline{a_m, a_{m+1}, ..., a_{m+k}}].$$

Theorem 5.3.2. Every rational number is representable as the sum of two periodic (irreducible) continued fractions of RCF_4.

Proof. Let p/q be the rational number such that $(p, q) = 1$. By Theorem 3.2.9 there exists $x, y \in \mathbb{R}$ such that

$$x + y = \frac{p}{q}.$$

This equation may be rewritten

$$y = \frac{-qx + p}{p}$$

which is a Möbius transformation with determinant $N = q^2$. Then by Theorem 4.1.14 there exists a canonical set C such that

$$\begin{aligned}
x &= [a_0; a_1, a_2, ..., a_{m-1}, x_m], \\
y &= [b_0; b_1, b_2, ..., b_{n-1}, y_n],
\end{aligned}$$

$x = My$ for some $M \in \mathrm{GL}_2(\mathbb{Z})$ and $M \in C$. Since C is finite, m and n can be chosen large enough such that

$$x_m = [a_m; a_{m+1}, ..., a_{m+s-1}, x_{m+s}],$$
$$y_n = [a_n; a_{n+1}, ..., a_{n+t-1}, y_{n+t}],$$

with $y_{n+t} = Mx_{m+s}$ for the same Möbius transformation M. But then x_{m+s} and y_{n+t} have the same quotients as x_m and y_n. Hence x_m and y_n are periodic. But then x and y are periodic too, with

$$x = [a_0; a_1, a_2, ..., a_{m-1}, \overline{a_m, a_{m+1}, ..., a_{m+s-1}}],$$
$$y = [b_0; b_1, b_2, ..., b_{n-1}, \overline{b_n, b_{n+1}, ..., b_{n+t-1}}].$$

This proves the theorem and shows a connection between Hall's theorem and Hall's algorithm. $\square$

Chapter 6

Drawing Nice Pictures

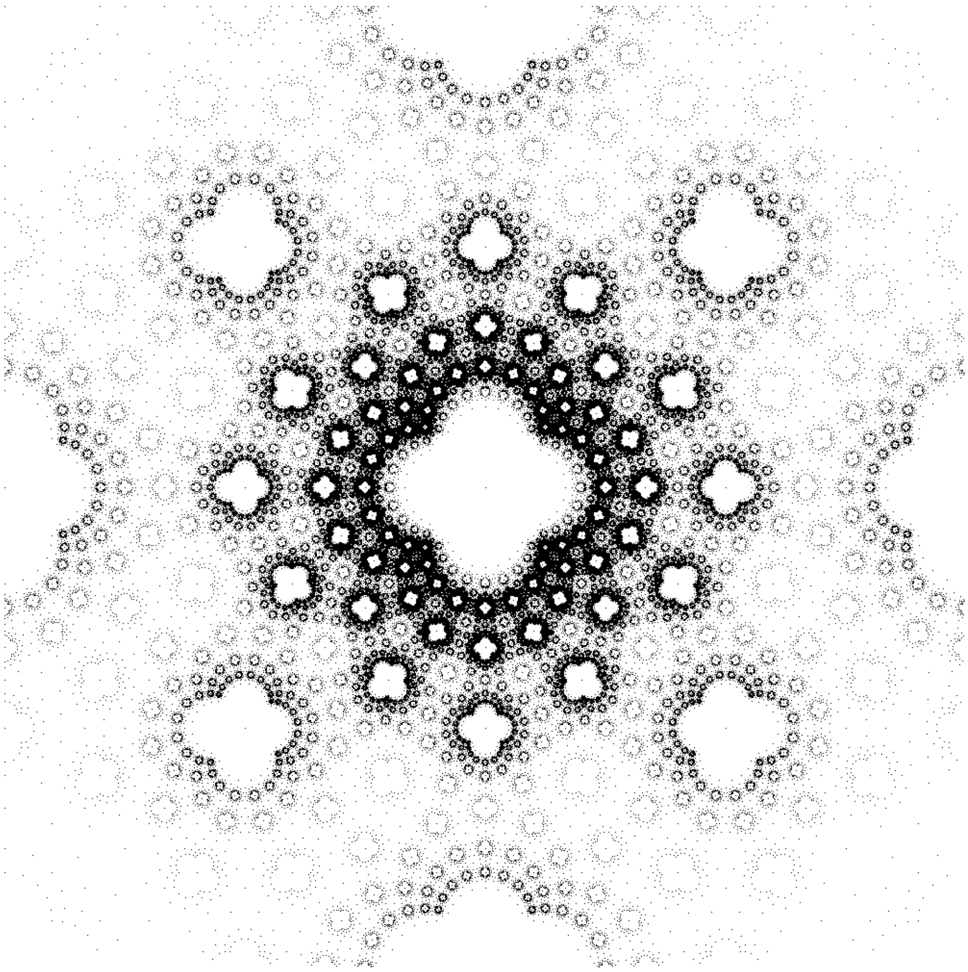


Figure 6.1: $\mathrm{CCF}_6(0)$ in the complex square $[-1, 1] \times [-i, i]$.

As seen throughout this thesis a lot of Fractals appear. Drawing these Fractals costs a lot of computation time. In this chapter an algorithm is given for drawing these Fractals. This chapter does not contain a lot of mathematical content, however it might be still interesting to read. The writer found GiNaC in particular a very interesting tool to work with, someone interested in Computer Algebra would at least have to read this section of GiNaC.

6.1 GiNaC

GiNaC is a free computer algebra system released under the GNU General Public License. It encourages its users to write symbolic algorithms directly in C++. Algebraic syntax is achieved in C++ through the use of operator overloading. Therefore it is not as fast as bignum, but still much faster than other computer algebra systems. Symbolically it can do

- multivariable polynomial arithmetic,
- factor polynomials,
- compute GCDs,
- expand series,
- compute matrices,
- symbolic integration for polynomials.

It can also work with non-commutative algebras, Clifford algebras, SU(3) Lie algebras and Lorentz tensors. This makes it also an interesting tool for high energy physics.

For our purposes it is only important that GiNaC supports a very fast and clear way to work with complex integers.

6.2 EasyBMP

EasyBMP is a simple, cross-platform, open source (revised BSD) C++ library designed for easily reading, writing, and modifying Windows bitmap (BMP) image files. The library is oriented towards the novice programmer with little formal experience, but it is sufficiently capable for anybody who desires to do I/O and pixel operations on uncompressed 1, 4, 8, 16, 24, and 32 bpp (bits per pixel) BMP files.

It is sufficient for our purposes to draw fractals.

6.3 Source

The first part of the code is a header including all import constants.

```
/* constants.h
 */

#ifndef __CONSTANTS_H
#define __CONSTANTS_H

/* Constants */
const int N = 2000;                    /* Height and width of image */
const unsigned long ULMAX = 100;       /* Max size for matrix elm */
const int DEPTH = 4;                   /* Depth of iteration */

#endif /* __CONSTANTS_H */
```

Now it is important to have a good class which supports Continued Fractions. Using templates it is possible to have the quotients be anything.

```
/* cf.h */

#ifndef __CF_H
#define __CF_H


#include <stdexcept>
using namespace std;

```

```

/* CF : Continued Fraction class */
template<class Num>
class CF
{
        public:
                CF(int length);

                ~CF();

                Num & operator [] (int index);

                void setQuotient(int index, Num z);

                Num getQuotient(int index);

                int getLength();

                Num foldUp();

        private:
                Num * quotients;

                int length;

                bool calculated;

                Num result;
};

#endif /* __CF_H */
```

And the cpp file.

```
/* cf.cpp
 *
 * This class is used to work with continued fractions of arbitrary length.
 * It supports a FoldUp function which calculates the element associated
 * by the quotients of the continued fraction.
 * It works well with the class numeric from the GiNaC library.
 *
 * Example:
 *      CF<numeric> cf(5);
 *
 *      for(i = 0; i < 5; i++)
 *              cf[i] = i + i*I;
 *
 *      cout << "Continued Fraction: " << cf.foldUp() << endl;
 */

#include "cf.h"

/* Initializer, length gives the numer of quotients for the continued fraction
 */
template<class Num>
CF<Num>::CF(int length)
{
        if (length < 0)
                throw range_error("CF(length)␣:␣negative␣length␣value");

        this->quotients = new Num[length];
        this->length = length;
        this->calculated = false;
}

/* Destructor, makes sure that the list of quotients is deleted.
 */
template<class Num>
CF<Num>::~CF()
{
```

```
        delete [] this->quotients;
}

/* A simple overload of the [] function. It returns quotient[index]
 */
template<class Num>
Num & CF<Num>::operator [] (int index)
{
        if (index < 0)
                throw range_error("CF[index]␣:␣negative␣index␣value");
        if (index >= this->length)
                throw out_of_range("CF[index]␣:␣index␣larger␣than␣range");

        return this->quotients[index];
}

/* setQuotient(int index, Num z), sets quotient[index] to z.
 */
template<class Num>
void CF<Num>::setQuotient(int index, Num z)
{
        if (index < 0)
                throw range_error("CF[index]␣:␣negative␣index␣value");
        if (index >= this->length)
                throw out_of_range("CF[index]␣:␣index␣larger␣than␣range");

        this->quotients[index] = z;
        this->calculated = false;
}

/* getLength(), returns the maximum number of quotients. */
template<class Num>
int CF<Num>::getLength()
{
        return this->length;
}

/* foldUp(), folds up all the quotients and returns a Num element. It uses
 * a modified version of Euclides Extend Algorithm
 */
template<class Num>
Num CF<Num>::foldUp()
{
        if (calculated == true)
                return this->result;

        int i;

        Num p0, p1, pn;
        Num q0, q1, qn;

        p0 = 1;
        p1 = this->quotients[0];
        pn = this->quotients[0];

        q0 = 0;
        q1 = 1;
        qn = 1;

        for (i = 1; i < this->length; i++){
                pn = p1*this->quotients[i] + p0;
                qn = q1*this->quotients[i] + q0;

                p0 = p1;
                p1 = pn;

                q0 = q1;
                q1 = qn;
        }

        if (qn != 0)
```

```
                this->result = pn/qn;
        else
                this->result = 0;

        this->calculated = true;

        return this->result;
}
```

Using EasyBMP it is possible to draw very fast single pixels into a file. This script makes the correct preperations to do this.

```
/* drawBMP.cpp
 * This is an easy tool using EasyBMP <http://easybmp.sourceforge.net/> to turn
 * a matrix into a BMP picture. It only draws black or white points, but it can
 * be modified easily to support more colors.
 */

#include "EasyBMP/EasyBMP.cpp"
#include "constants.h"
using namespace std;

/* matrix2bmp, turns nmatrix[N][N] with width x height into a bmp image filename
 *  of width x height.
 */
void matrix2bmp(char *filename, unsigned long nmatrix[N][N],
        int width, int height)
{
        BMP output;
        output.SetSize(width, height);
        output.SetBitDepth(24);

        int i, j;
        unsigned long value;

        for (i = 0; i < width; i++)
                for (j = 0; j < height; j++) {
                        value = nmatrix[i][j];

                        if(value > 0) {
                                output(i, j)->Blue  = 0;
                                output(i, j)->Red   = 0;
                                output(i, j)->Green = 0;
                        } else {
                                output(i, j)->Blue  = 255;
                                output(i, j)->Red   = 255;
                                output(i, j)->Green = 255;
                        }
                }

        output.WriteToFile(filename);
}
```

The main part. This part calculates all continued fractions with DEPTH quotients. It plots them all in a matrix which is then written to the file *fractal.bmp*.

```
/* ksubsets.cpp
 *
 * This file draws the fractal of Complex Continued Fractions with
 * bounded (absolute) quotients of length 4 or lower.
 * The ksub_iter and print_ccf functions draw to points to a matrix
 * pmatrix when matrix2bmp draws the pmatrix to a bmp file.
 */


/* Includes */

#include <iostream>
using namespace std;
```

```
#include "ginac/ginac.h"
using namespace GiNaC;

#include "cf.cpp"
#include "drawBMP.cpp"
#include "constants.h"


/* Global variables */

unsigned long pmatrix[N][N];     /* Raw matrix representation of the image */


/* Iteration functions */

/* ksub_iter, Depth first iter for all combinations from list with length
 * depth.
 * (list, len_list) is the list to iterate over,
 * depth gives the iteration depth,
 * (elm, len_elm) is a buffer in which the iteration is stored,
 * (*f)(numeric *, in) is a function which is called if (elm, len_elm) is of
 *       length depth.
 */
void ksub_iter(numeric *list, int len_list, int depth, numeric *elm,
               int len_elm, void (*f)(numeric *, int))
{
        int i;

        if(depth == 0)
                f(elm, len_elm);
        else
                for(i = 0; i < len_list; i++) {
                        elm[len_elm] = list[i];
                        ksub_iter(list, len_list, depth-1, elm, len_elm+1, f);
                }
}

/* print_list, prints a list of length len_list. Debug function!
 */
void print_list(numeric * list, int len_list)
{
        int i;

        for(i = 0; i < len_list -1; i++)
                cout << list[i] << ", ";
        cout << list[len_list -1] << endl;
}

/* plot_ccf, given a list of quotients it draws a point in pmatrix.
 */
void plot_ccf(numeric *list, int len_list)
{
        int i;
        int x, y;

        numeric res;

        CF<numeric> ccf(len_list+1);

        ccf[0] = 0;
        for(i = 0; i < len_list; i++)
                ccf[i+1] = list[i];

        res = ccf.foldUp();

        /* Draws one forth of the picture. */
        x = (int)N*real(res).to_double();
        y = (int)N*imag(res).to_double();

        if (0 <= x && x < N
                       && 0 <= y && y < N
```

```
                        && pmatrix[x][y] < ULMAX)
                pmatrix[x][y]++;
}
```

```
int main()
{
        int i;

        /* clean matrix */
        for (i = 0; i < N; i++)
                pmatrix[i] = {0};

        try {
                /* All complex Gausians with absolute value less or equal
                 * to 1.
                 */
                numeric list1[] = {
1, -1, I, -I
                };

                /* All complex Gausians with absolute value less or equal
                 * to 2.
                 */
                numeric list2[] = {
1, -1, I, -I, 1 + I, 1 - I, -1 + I, -1 - I, 2, 2*I, -2, -2*I
                };

                /* All complex Gausians with absolute value less or equal
                 * to 3.
                 */
                numeric list3[] = {
1, -1, I, -I, 1 + I, 1 - I, -1 + I, -1 - I,  2, 2*I, -2, -2*I,
3, 2 + I, 2 + 2*I, 1 + 2*I, 3*I, -1 + 2*I, -2 + 2*I, -2 + I,
-3*I, -2 - I, -2 - 2*I, -1 - 2*I, -3*I, 1 - 2*I, 2 - 2*I, 2 - I
                };


                /* All complex Gausians with absolute value lesser or equal
                 * to 4.
                 */
                numeric list4[] = {
-4, -2*I - 3, -I - 3, -3, I - 3, 2*I - 3, -3*I - 2, -2*I - 2, -I - 2,
-2, I - 2, 2*I - 2, 3*I - 2, -3*I - 1, -2*I - 1, -I - 1, -1, I - 1,
2*I - 1, 3*I - 1, -4*I, -3*I, -2*I, -I, I, 2*I, 3*I, 4*I, -3*I + 1,
-2*I + 1, -I + 1, 1, I + 1, 2*I + 1, 3*I + 1, -3*I + 2, -2*I + 2,
-I + 2, 2, I + 2, 2*I + 2, 3*I + 2, -2*I + 3, -I + 3, 3, I + 3, 2*I + 3, 4
                };

                /* All complex Gausians with absolute value lesser or equal
                 * to 5.
                 */
                numeric list5[] = {
-3*I - 4, -2*I - 4, -I - 4, -4, I - 4, 2*I - 4, 3*I - 4, -4*I - 3,
-3*I - 3, -2*I - 3, -I - 3, -3, I - 3, 2*I - 3, 3*I - 3, 4*I - 3,
-4*I - 2, -3*I - 2, -2*I - 2, -I - 2, -2, I - 2, 2*I - 2, 3*I - 2,
4*I - 2, -4*I - 1, -3*I - 1, -2*I - 1, -I - 1, -1, I - 1, 2*I - 1,
3*I - 1, 4*I - 1, -4*I, -3*I, -2*I, -I, I, 2*I, 3*I, 4*I, -4*I + 1,
-3*I + 1, -2*I + 1, -I + 1, 1, I + 1, 2*I + 1, 3*I + 1, 4*I + 1,
-4*I + 2, -3*I + 2, -2*I + 2, -I + 2, 2, I + 2, 2*I + 2, 3*I + 2,
4*I + 2, -4*I + 3, -3*I + 3, -2*I + 3, -I + 3, 3, I + 3, 2*I + 3,
3*I + 3, 4*I + 3, -3*I + 4, -2*I + 4, -I + 4, 4, I + 4, 2*I + 4, 3*I + 4
                };

                /* All complex Gausians with absolute value lesser or equal
                 * to 6.
                 */
                numeric list6[] = {
-4*I - 4, -3*I - 4, -2*I - 4, -I - 4, -4, I - 4, 2*I - 4, 3*I - 4,
4*I - 4, -4*I - 3, -3*I - 3, -2*I - 3, -I - 3, -3, I - 3, 2*I - 3,
3*I - 3, 4*I - 3, -4*I - 2, -3*I - 2, -2*I - 2, -I - 2, -2, I - 2,
```

```
2*I - 2, 3*I - 2, 4*I - 2, -4*I - 1, -3*I - 1, -2*I - 1, -I - 1, -1,
I - 1, 2*I - 1, 3*I - 1, 4*I - 1, -4*I, -3*I, -2*I, -I, I, 2*I, 3*I,
4*I, -4*I + 1, -3*I + 1, -2*I + 1, -I + 1, 1, I + 1, 2*I + 1, 3*I + 1,
4*I + 1, -4*I + 2, -3*I + 2, -2*I + 2, -I + 2, 2, I + 2, 2*I + 2,
3*I + 2, 4*I + 2, -4*I + 3, -3*I + 3, -2*I + 3, -I + 3, 3, I + 3,
2*I + 3, 3*I + 3, 4*I + 3, -4*I + 4, -3*I + 4, -2*I + 4, -I + 4,
4, I + 4, 2*I + 4, 3*I + 4, 4*I + 4
                };

                numeric elm[10];

                /* N = 1 */
                cout << "Fractal N = 1" << endl;

                for (i = 0; i < N; i++)
                        pmatrix[i] = {0};

                ksub_iter(list1, 4, 13, elm, 0, plot_ccf);
                matrix2bmp("fractal1.bmp", pmatrix, N, N);


                /* N = 2 */
                cout << "Fractal N = 2" << endl;

                for (i = 0; i < N; i++)
                        pmatrix[i] = {0};

                ksub_iter(list2, 12, 8, elm, 0, plot_ccf);
                matrix2bmp("fractal2.bmp", pmatrix, N, N);

                /* N = 3 */
                cout << "Fractal N = 3" << endl;

                for (i = 0; i < N; i++)
                        pmatrix[i] = {0};

                ksub_iter(list3, 28, 5, elm, 0, plot_ccf);
                matrix2bmp("fractal3.bmp", pmatrix, N, N);

                /* N = 4 */
                cout << "Fractal N = 4" << endl;

                for (i = 0; i < N; i++)
                        pmatrix[i] = {0};

                ksub_iter(list4, 48, 5, elm, 0, plot_ccf);
                matrix2bmp("fractal4.bmp", pmatrix, N, N);

                /* N = 5 */
                cout << "Fractal N = 5" << endl;
                for (i = 0; i < N; i++)
                        pmatrix[i] = {0};

                ksub_iter(list5, 76, 5, elm, 0, plot_ccf);
                matrix2bmp("fractal5.bmp", pmatrix, N, N);


                /* N = 6 */
                cout << "Fractal N = 6" << endl;
                for (i = 0; i < N; i++)
                        pmatrix[i] = {0};

                ksub_iter(list6, 80, 5, elm, 0, plot_ccf);
                matrix2bmp("fractal6.bmp", pmatrix, N, N);


        } catch (exception &p) {
                cerr << p.what() << endl;

                return 1;
        }
```

```
        matrix2bmp("fractal.bmp", pmatrix, N, N);

        return 0;
}
```

Conclusion and Further Research

In this master thesis we have seen the proof of Hall's Theorem. This proof is then used to extend Hall's Theorem to NICF_6 using singularisation, to CCF_4 by a simple observation. Hence if x is a real number, there are two real numbers a and b with quotients less than or equal to 4 such that $x = a + b$. It is even possible to calculate the a and b explicitly. Also, Hall's algorithm is developed for the complex numbers, showing that Hall's Algorithm is really an algorithm. And it ends with a theorem which combines Hall's Algorithm and Hall's Theorem.

There are also some questions which arise from this research.

- In $k\text{CCF}_4 = \mathbb{C}$, is $k = 4$ the least number possible. Can it be shown that $\text{CCF}_4+\text{CCF}_4+\text{CCF}_4 \neq \mathbb{C}$? It is not clear if this is true or not. If however $\text{CCF}_4 + \text{CCF}_4 + \text{CCF}_4 \neq \mathbb{C}$, then this could maybe be proved similar to [Div73], where it is proved that $\text{RCF}_3 + \text{RCF}_3 \neq \mathbb{R}$.
- In Chapter 4 Canonical Sets are introduced for Regular Continued Fractions. Canonical Sets also exist for Complex Continued Fractions. Is there an easy way to compute them?
- In Chapter 5 some explicit sums are given for $e, \sqrt{2}, \sqrt{3}, \tau, ...$, but do they contain some sort of regularity? Because the regular continued fractions of these numbers do.

Bibliography

[Ast00] Stephen Astels. Cantor sets and numbers with restricted partial quotients. *Trans. Amer. Math. Soc.*, 352:133–170, 2000.

[Ast01] Stephen Astels. Sums of numbers with small partial quotients ii. *J. Number Theory*, 91:187–205, 2001.

[Ast02] Stephen Astels. Sums of numbers with small partial quotients. *Proc. Amer. Math. Soc.*, 130:637–642, 2002.

[Can83] Georg Cantor. Über unendliche, lineare punktmännichfaltigkeiten. *Math. Ann*, pages 545–591, 1883.

[Cus73] Thomas Cusick. On m. hall's continued fraction theorem. *Trans. Am. Math. Soc.*, 38:253–254, 1973.

[Div73] Bohuslav Diviš. On the sums of continued fractions. *Acta. Arith.*, 22:157–173, 1973.

[Eng89] Ryszard Engelking. *General Topology.* Heldermann Verlag, 1989.

[Hal47] Marshall Hall. On the sum and product of continued fractions. *The Annals of Mathematics*, 48(4):966–993, October 1947.

[Hau14] F. Hausdorff. *Grundzüge der Mengenlehre.* Chelsea Publishing Company New York, 1914.

[Hen06] Doug Hensley. *Continued Fractions.* World Scientific, 2006.

[Hla75] James Hlavka. *Trans. Amer. Math. Soc.*, 211:123–134, 1975.

[Hur] Adolf Hurwitz. Über die entwicklung complexer grössen in kettenbruüche. *Acta Math.*, 11:187–200.

[IK02] Marius Iosifescu and Cor Kraaikamp. *Metrical Theory of Continued Fractions.* Kluwer Academic Publishers, 2002.

[Lui11] Joris Luijsterburg. Möbius transformations on complex continued fractions. Master's thesis, Radboud University Nijmegen, 2011.

[Ran73] George Raney. On continued fractions and finite automata. *Math. Ann*, 206:265–283, 1973.

[RS92] Andrew Rockett and Peter Szusz. *Continued Fractions.* World Scientific Publishing Co. Pte. Ltd., 1992.

[Sch75] Asmus Schmidt. Diophantine approximation of complex numbers. *Acta. Math.*, 134:1–85, 1975.

[Sch80] Wolfgang Schmidt. *Diophantine Approximation*, volume 785 of *Lecture Notes in Mathematics.* Springer-Verlag, 1980.

[Smi61] Henry J. Stephen Smith. On systems of linear indeterminate equations and congruences. *Philosophical Transactions of the Royal Society of London*, 151:293–326, 1861.

Index