Tiling the Half-Plane with Squares of Integral Sides

by

Emma Hartman

A Study

Presented to the Faculty

of

Wheaton College

in Partial Fulfillment of the Requirements

for

Graduation with Departmental Honors

in Mathematics

Norton, Massachusetts

May 2014

Acknowledgements

I would like to thank my advisor, Tommy Ratliff, for his endless support, encouragement, and patience, not just throughout this year but all four years. He has always been there to give advice or to listen while I vent, even when I was on another continent. I would also like to thank the committee members, Shelly Leibowitz and James Freeman, for their time, feedback, and enthusiasm. Finally, I would like to thank the Mathematics and Computer Science faculty as a whole for instilling in me the skills, knowledge, and passion necessary to complete this project.

Contents

1 Background

1.1 Tiling the Plane with Squares

Tiling the plane with squares can be done most simply and trivially with infinitely many squares of precisely the same size, lined up corner to corner. These lines could also be shifted horizontally or vertically, as in Figure 1. Additionally, we could consider infinitely many parallel rows of squares, each of a different size, lined up extending infinitely within their rows (see Figure 2). Using this approach, we can easily see that it is possible to tile the plane such that every positive integer occurs as the length of a square that appears in the tiling. However, notice that we use an infinite number of tiles of each size. Now consider a tiling of the plane which uses squares of a variety of sizes, not all adjacently placed to similar squares. One such tiling is found by using the Fibonacci sequence: 1,1,2,3,5,8,13,. . . . Beginning with the first two numbers in the sequence, one square with a side-length of each number in the Fibonacci sequence is placed on the plane, in order, in a somewhat "circular" manner, as in Figure 3. Since the length of each side is the next Fibonacci number, it is equal to the sum of the side-lengths of the two squares placed before it, so this arrangement tiles the full plane. This tiling has the advantage that the only repeated square has side-length 1, but not every positive integer appears as the side-length of a square in the tiling. For example, there is no square of side-length 4 since 4 does not appear in the Fibonacci sequence. [1]

1.2 Tiling the full plane with every unique integer square

A very natural question is whether a tiling of the plane exists that uses exactly one square with side-length of each positive integer. As can often be the case in mathematics, a seemingly simple question has a surprisingly subtle solution. In their

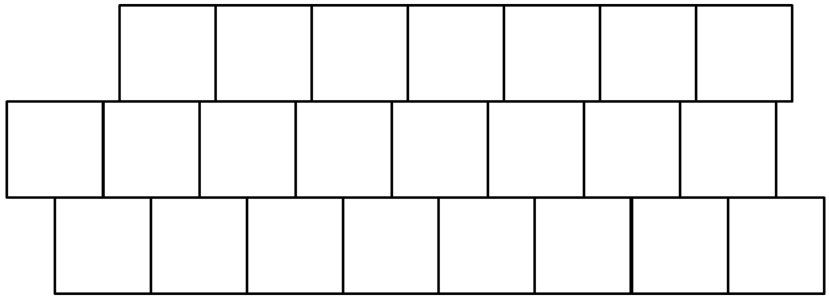

Figure 1: Horizontal shift of a basic tiling of the plane with squares.

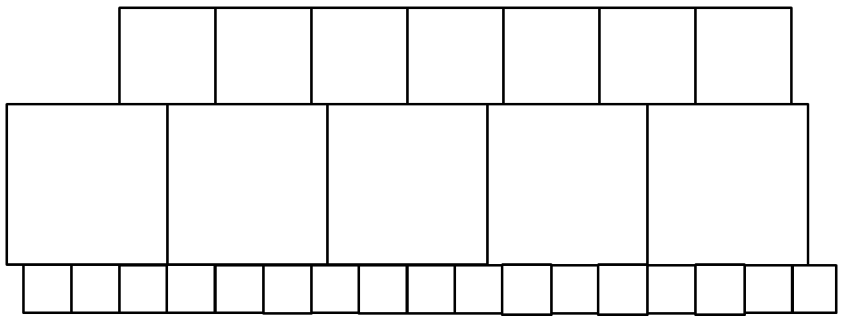

Figure 2: Squares of different sizes tiling the plane.

2008 paper *Squaring the Plane*, which won the prestigious David P. Robbins Prize, Frederick and James Henle describe a new tiling of the plane which uses exactly one square with side-length of each integer [1]. A brief outline of their tiling is described here, as it served as the basis for my own research. The paper starts by describing an "ell", from which the tiling of the plane is built.

Definition 1.1 (Ell) *An ell is a shape that can be described as a rectangle with a smaller rectangle removed from the upper right corner such that it has a concave right angle, or as "any six-sided figure whose sides are parallel to the coordinate axes" [1].*

The lengths of the sides of an ell are labeled with lower case letters as shown in Figure 4. Squares are always added to a certain side of an ell with a side-length equal to the whole length of that side. Adding a square is notated by a letter corresponding to the side of the ell to which it is added: $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}, \mathcal{E}$, or $\mathcal{F}$. A sequence of squares

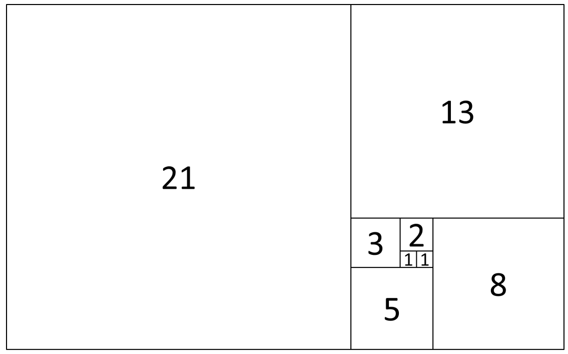

Figure 3: A tiling of the plane using the Fibonacci sequence.

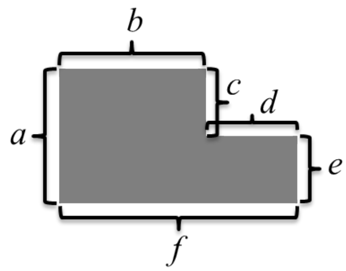

Figure 4: Notation for the lengths of the sides of an ell.

to be added to the ell is read from left to right, as in $\mathcal{AB}$ would add a square to the side of length a first, then the side of length b. See Figures 5 and 6.

(a) (b)

Figure 5: This figure shows the move $\mathcal{B}$ (a) and the move $\mathcal{C}$ (b) performed on an ell.

From Henle's paper, any ell can be "squared up" (actually, rectangled up), meaning squares can be added to the ell to make the shape a rectangle. Starting with an ell consisting of just two squares and then with a clever combination of the moves $\mathcal{FABA}$, $\mathcal{BFA}$, and $\mathcal{ED}$, the authors prove by induction that d will continually decrease. Since d is a nonnegative integer, it will eventually reach 0, meaning the ell

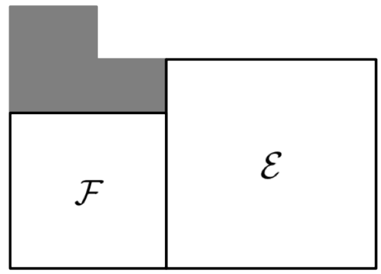

Figure 6: The move $\mathcal{FE}$ adds these squares from left to right, so $\mathcal{F}$ first, then $\mathcal{E}$.

would become a rectangle, as shown in Figure 7.

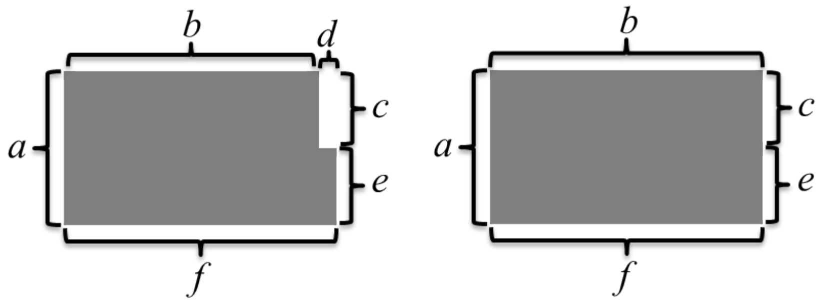

Figure 7: An ell with a very small d (left), and then one in which $d = 0$ (right). When $d = 0$, an ell is just a rectangle.

At this point, the authors claim that the smallest integer-sided square not already used in this construction can be added to this rectangle at the lower right corner, creating a new ell, and the process can be restarted to eventually create a rectangle again. The authors prove that in this construction, no square is ever repeated, and by adding in the smallest square not used every time the ell is "squared up", this construction tiles the plane with exactly one of every integer-sided square. Henle's construction builds the ell out on all sides, expanding in every direction to tile the whole plane, similarly to the Fibonacci tiling in Figure 3. A natural extension to this work would be to see if there exists a similar tiling for the half- or quarter-plane.

For my work, I examined tiling the half-plane with integer-sided squares in a variety of ways. I started by attempting to adapt the Henle construction, but unfortunately without the ability to add squares to all 6 sides of the ell, it is no longer possible to perform this construction, and expanding the ell in any way without repeating squares becomes a more difficult problem. [1]

2 Tiling the half-plane with unique squares

2.1 Refining the Definitions

We use the same Definition 1.1 of an ell used by Henle [1], but in our case, the ell is always positioned on the line of the right half-plane, with the side of length a touching the line dividing the plane, as shown in Figure 8.

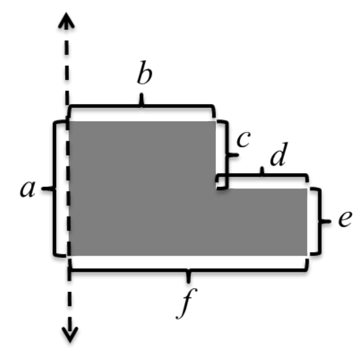

Figure 8: Notation for the lengths of the sides of an ell.

Note that the side of length a is equal to the sides of lengths c and e, $a = c + e$, and that the side of length f is equal to the sides of lengths b and d, $f = b + d$. Therefore only b, c, d, and e are needed to uniquely define an ell.

We use the same notation of $\mathcal{B}, \mathcal{C}, \mathcal{D}, \mathcal{E}$, or $\mathcal{F}$ for adding squares as used in the Henle paper, but note that the move $\mathcal{A}$ will never be performed, as the side of length a is always against the line dividing the plane in half.

We refer to a square either by the move which added it to the ell or by the length of its side. The name of a move written with a subscript denotes both a square added by that move at a certain stage of an algorithm and the length of the side of the square added at that move. For example, $\mathcal{E}_k$ would denote the k^{th} square added by move $\mathcal{E}$ within the execution of a certain algorithm iteratively. Note that these moves are not necessarily consecutive – squares can be added to other sides in between $\mathcal{E}_{k-1}$ and $\mathcal{E}_k$ being added to the ell.

Definition 2.1 (Composite) *The side of an ell is composite if it does not consist of exactly one square.*

This definition does not imply that the side must consist of an exact number of squares or even that it must consist of more than one square; Figure 9 shows different types of sides that are considered composite.

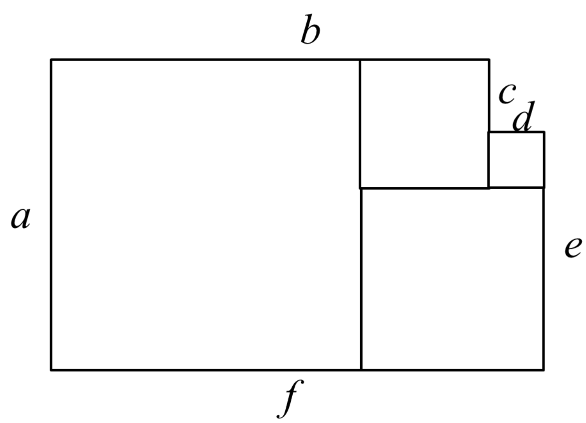

Figure 9: In this ell, the sides of length b, c, e, and f are composite, while the sides of length a and d are not composite.

Now using this notation, we want to find ways to add squares to an ell in order to tile the half-plane. The next two sections will describe two different tilings.

2.2 A Naïve Tiling

Here we describe one tiling of the half-plane using unique squares. Start with a rectangle composed of unique integer-sided squares and with at least one composite side. We know such a rectangle exists from Henle's paper, Figure 10 shows such an example.

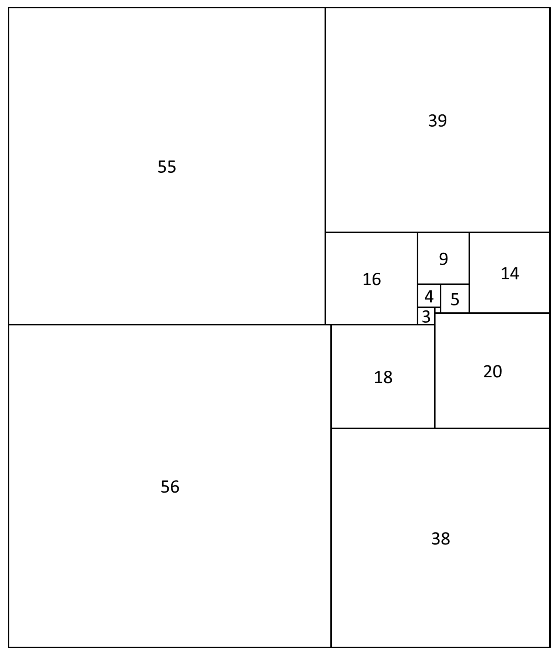

Figure 10: A rectangle composed of unique integer squares, as shown in Henle.

Letting this rectangle be flush to the straight edge of the half-plane, add squares around it as shown in Figure 11. First, a square is added to the right of the figure, then above it, then to the right again, then below. This algorithm is repeated infinitely. This creates a Fibonacci-like sequence of squares, similar to the tiling of the whole plane discussed by Henle [1],p.3.

It is clear that every square added will be larger than the last, so each square in the tiling is unique. This is a fairly simplistic algorithm, but also one that tiles the half-plane with unique integer-sided squares. However, notice that the size of the squares grows large very quickly.

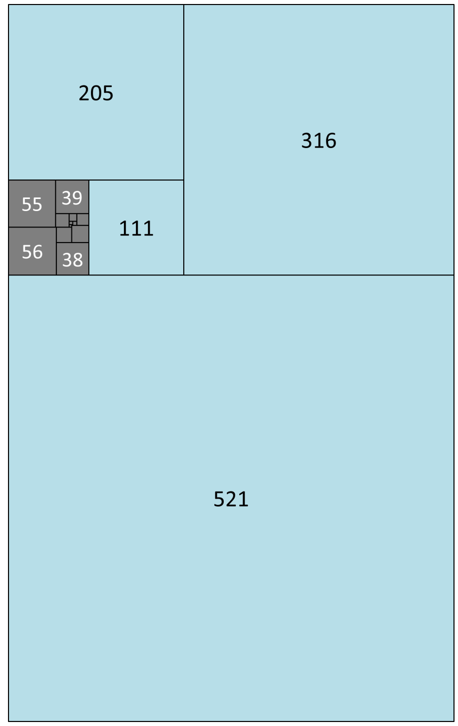

Figure 11: Squares are added to the original rectangle in the manner shown here.

2.3 $\mathcal{FECB}$ algorithm

Consider starting with an ell consisting of two squares of different integer side-lengths – the larger will have side-length α and the smaller side-length β, as shown in Figure 12.

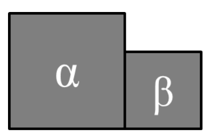

Figure 12: An (α, β) ell.

Now we follow the pattern $\mathcal{FECB}$, as shown in Figure 13, to tile the half-plane with integer-sided squares. We will see that this algorithm uses only integer-sided squares but does not use all integer-sided squares, and that the squares used are unique under most circumstances. The particular circumstances under which the squares used are not unique are given in Theorem 2.4.

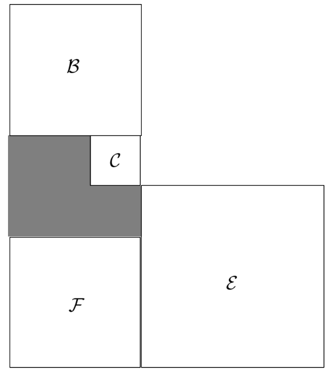

Figure 13: The moves $\mathcal{FECB}$ performed on an ell.

Every square added by a particular move is larger than the square added by that move in the previous round of the pattern. Additionally, the squares added by moves $\mathcal{F}, \mathcal{E},$ and $\mathcal{B}$ always increase the overall size of the ell, so it is clear that at each round of the algorithm, the pattern will create a larger ell that expands up, down, and to the right so that as the number of rounds of the pattern approaches infinity, the figure will tile the right half-plane. Thus we have proven the following theorem.

Theorem 2.2 An ell can be expanded infinitely by adding squares in the pattern $\mathcal{FECB}$ iteratively.

We now have another tiling of the half-plane that seems to use smaller squares than our Fibonacci tiling, but it is not clear that each square used in the tiling is of a unique size. The following theorem allows us to express the lengths of the sides of these squares as a linear combination of α and β, which will give us the tools to determine when the $\mathcal{FECB}$ tiling consists of unique squares.

Theorem 2.3 Starting with an (α, β) ell and performing the moves $\mathcal{FECB}$ (as shown in Figure 13) iteratively will result in squares of the following side-lengths added in the i^{th} round of the algorithm, with respect to α and β, where F_i is the i^{th} Fibonacci number and $F_0 = F_1 = 1$.

i	$\mathcal{F}_i$	$\mathcal{E}_i$	$\mathcal{C}_i$	$\mathcal{B}_i$
0	$\alpha + \beta$	$\alpha + 2\beta$	$\alpha - \beta$	$2\alpha - \beta$
1	$2\alpha + 3\beta$	$3\alpha + 5\beta$	$3\alpha - 2\beta$	$5\alpha - 3\beta$
2	$5\alpha + 8\beta$	$8\alpha + 13\beta$	$8\alpha - 5\beta$	$13\alpha - 8\beta$
3	$13\alpha + 21\beta$	$21\alpha + 34\beta$	$21\alpha - 13\beta$	$34\alpha - 21\beta$
			$\ldots$	
n	$F_{2n}\alpha + F_{2n+1}\beta$	$F_{2n+1}\alpha + F_{2n+2}\beta$	$F_{2n+1}\alpha - F_{2n}\beta$	$F_{2n+2}\alpha - F_{2n+1}\beta$

Proof: *By induction.*

Note from Figure 13 that:

$$\mathcal{F}_n = \mathcal{F}_{n-1} + \mathcal{E}_{n-1}$$

$$\mathcal{E}_n = \mathcal{F}_n + \mathcal{E}_{n-1}$$

$$\mathcal{C}_n = \mathcal{B}_{n-1} + \mathcal{C}_{n-1}$$

$$\mathcal{B}_n = \mathcal{C}_n + \mathcal{B}_{n-1}$$

Base case: For $n = 0$,

$$\mathcal{F}_n = \mathcal{F}_0 = \alpha + \beta = F_0\alpha + F_1\beta$$

$$\mathcal{E}_n = \mathcal{E}_0 = \alpha + 2\beta = F_1\alpha + 2F_2\beta$$

$$\mathcal{C}_n = \mathcal{C}_0 = \alpha - \beta = F_1\alpha - F_0\beta$$

$$\mathcal{B}_n = \mathcal{B}_0 = 2\alpha - \beta = 2F_2\alpha - F_1\beta$$

Inductive step:

Assume that

$$\mathcal{F}_n = F_{2n}\alpha + F_{2n+1}\beta$$

$$\mathcal{E}_n = F_{2n+1}\alpha + F_{2n+2}\beta$$

$$\mathcal{C}_n = F_{2n+1}\alpha - F_{2n}\beta$$

$$\mathcal{B}_n = F_{2n+2}\alpha - F_{2n+1}\beta$$

Show that

$$\mathcal{F}_{n+1} = F_{2(n+1)}\alpha + F_{2(n+1)+1}\beta$$

$$\mathcal{E}_{n+1} = F_{2(n+1)+1}\alpha + F_{2(n+1)+2}\beta$$

$$\mathcal{C}_{n+1} = F_{2(n+1)+1}\alpha - F_{2(n+1)}\beta$$

$$\mathcal{B}_{n+1} = F_{2(n+1)+2}\alpha - F_{2(n+1)+1}\beta$$

Inductive Step for $\mathcal{F}_n$:

$\mathcal{F}_{n+1} = \mathcal{F}_n + \mathcal{E}_n$

$= (F_{2n}\alpha + F_{2n+1}\beta) + (F_{2n+1}\alpha + F_{2n+2}\beta)$, by the inductive hypothesis

$= F_{2n+2}\alpha + F_{2n+3}\beta$

$= F_{2(n+1)}\alpha + F_{2(n+1)+1}\beta$

Inductive Step for $\mathcal{E}_n$:

$\mathcal{E}_{n+1} = \mathcal{F}_{n+1} + \mathcal{E}_n$

$= (F_{2(n+1)}\alpha + F_{2(n+1)+1}\beta) + (F_{2n+1}\alpha + F_{2n+2}\beta)$, by the inductive hypothesis

$= F_{2n+3}\alpha + F_{2n+4}\beta$

$= F_{2(n+1)+1}\alpha + F_{2(n+1)+2}\beta$

Inductive Step for $\mathcal{C}_n$:

$$
\begin{aligned}
\mathcal{C}_{n+1} &= \mathcal{B}_n + \mathcal{C}_n \\
&= (F_{2n+2}\alpha - F_{2n+1}\beta) + (F_{2n+1}\alpha - F_{2n}\beta), \text{ by the inductive hypothesis} \\
&= F_{2n+3}\alpha - F_{2n+2}\beta \\
&= F_{2(n+1)+1}\alpha - F_{2(n+1)}\beta
\end{aligned}
$$

Inductive Step for $\mathcal{B}_n$:

$$
\begin{aligned}
\mathcal{B}_{n+1} &= \mathcal{C}_{n+1} + \mathcal{B}_n \\
&= (F_{2(n+1)+1}\alpha - F_{2(n+1)}\beta) + (F_{2n+2}\alpha - F_{2n+1}\beta), \text{ by the inductive hypothesis} \\
&= F_{2n+4}\alpha - F_{2n+3}\beta \\
&= F_{2(n+1)+2}\alpha - F_{2(n+1)+1}\beta
\end{aligned}
$$

Thus, we have proved the theorem. □

2.3.1 Restrictions on α and β pairs

Since the goal is to prove that the algorithm $\mathcal{FECB}$ tiles the half-plane with *unique* squares, it is important to show that no square added is the same as a square previously added, since these sets of linear combinations, described in Theorem 2.3, are not necessarily mutually exclusive. Here we consider each possible pairing of squares and for what values of α and β they might possibly be equal. First, to demonstrate this, we present several examples.

Example 1 $\alpha = 12, \beta = 3$

Starting with the ell in Figure 14, we follow the pattern $\mathcal{FECB}$ for one round (blue) and then a second round (orange) as shown in Figure 15.

Figure 14: Starting ell with $\alpha = 12, \beta = 3$

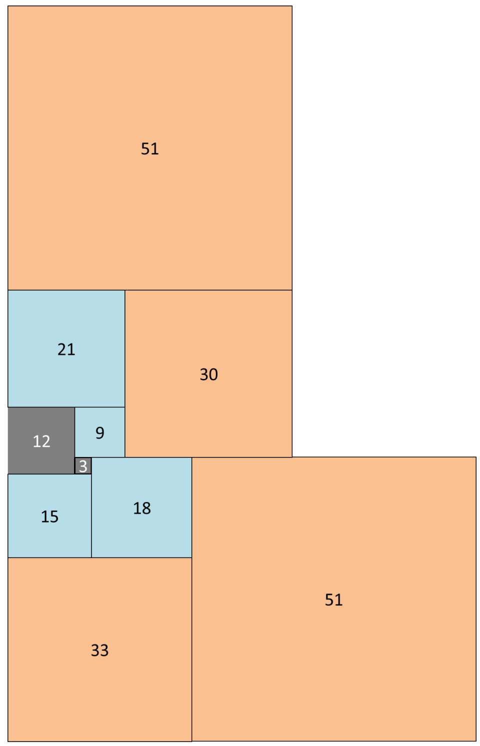

Figure 15: First round of $\mathcal{FECB}$ (blue) and second round (orange).

We see here that the square of size 51 is added twice, so this algorithm starting with $\alpha = 12, \beta = 3$ does not use squares of a unique size.

Example 2 $\alpha = 21, \beta = 16$

i	$\mathcal{F}_i$	$\mathcal{E}_i$	$\mathcal{C}_i$	$\mathcal{B}_i$
0	37	53	5	26
1	90	143	31	57
2	233	376	88	145
3	609	985	233	378

From this table, we can see that the square added by $\mathcal{F}$ in the 2nd round and the square added by $\mathcal{C}$ in the 3rd round are equal, in other words $\mathcal{F}_2 = 233 = \mathcal{C}_3$, so starting with an ell consisting of $\alpha = 21, \beta = 16$ will not tile the half-plane with *unique* squares in this $\mathcal{FECB}$ algorithm. It might seem logical to begin our ell with larger (α, β) pairs to avoid the likelihood of repeating squares. Next, in Examples 3 and 4, we show two (α, β) pairs that seem to produce unique squares from the $\mathcal{FECB}$ algorithm.

Example 3 $\alpha = 416020, \beta = 317811$

i	$\mathcal{F}_i$	$\mathcal{E}_i$	$\mathcal{C}_i$	$\mathcal{B}_i$
0	733831	1051642	98209	514229
1	1785473	2837115	612438	1126667
2	4622588	7459703	1739105	2865772
3	12082291	19541994	4604877	7470649
4	31624285	51166279	12075526	19546175
5	82790564	133956843	31621701	51167876
6	216747407	350704250	82789577	133957453
7	567451657	918155907	216747030	350704483
8	1485607564	2403763471	567451513	918155996
9	3889371035	6293134506	1485607509	2403763505
10	10182505541	16475640047	3889371014	6293134519
		...		

Example 4, like Examples 1 and 2, starts with fairly small α and β values as well, so we might expect it to create a repeated square as well. But after several rounds, it would appear not to repeat.

Example 4 $\alpha = 5, \beta = 3$

i	$\mathcal{F}_i$	$\mathcal{E}_i$	$\mathcal{C}_i$	$\mathcal{B}_i$
0	8	11	2	7
1	19	30	9	16
2	49	79	25	41
3	128	207	66	107
4	335	542	173	280
5	877	1419	453	733
6	2296	3715	1186	1919
7	6011	9726	3105	5024
8	15737	25463	8129	13153
		...		

It appears that choosing either $\alpha = 416020, \beta = 317811$ or $\alpha = 5, \beta = 3$ and repeating the $\mathcal{FECB}$ pattern is not going to create any duplicate squares, but how do we know there is not a square in the n^{th} round of the algorithm that repeats some earlier square? We prove in Theorem 2.4 when a certain (α, β) pair truly tiles the half-plane with an entirely unique set of squares.

Notice that in our examples so far, it always seems that if an (α, β) pair does not produce unique squares, then there is exactly one square that is repeated. Example 5 shows a case in which many squares are repeated, and the $\mathcal{FECB}$ algorithm does not appear to work at all. It seems in this example that it is always true that $\mathcal{C}_n = \mathcal{E}_{n-1}$ and that $\mathcal{F}_n = \mathcal{B}_n$ for all n. This example can also be explained by Theorem 2.4.

Example 5 $\alpha = 6, \beta = 3$

i	$\mathcal{F}_i$	$\mathcal{E}_i$	$\mathcal{C}_i$	$\mathcal{B}_i$
0	9	12	3	9
1	21	33	12	21
2	54	87	33	54
		...		

In Theorem 2.4, we describe the cases in which a chosen α and β pair succeed or fail at tiling with unique squares.

Theorem 2.4 The algorithm $\mathcal{FECB}$, as described in Theorem 2.3, performed iteratively on a starting (α, β) ell, will produce a tiling of the half-plane consisting of unique integer-sided squares if and only if $\frac{\alpha}{\beta} \neq 2, \frac{F_{k+3}}{F_k}$, or $\frac{F_{k+2}}{2F_k}$ $\forall k \geq 0$, where F_i represents the i^{th} number in the Fibonacci sequence.

The proof of this theorem consists of six lemmas that will be proved individually. Each lemma compares squares added by a different pair of moves from the set $\{\mathcal{F}, \mathcal{E}, \mathcal{C}, \mathcal{B}\}$. From these pairs of moves, we can describe ratios between α and β that would cause squares added by these two moves to be equal. By finding all of these exceptions we can concretely define which (α, β) pairs will tile the half-plane with unique squares.

Lemma 2.5 For all $i, j \geq 0$, $\mathcal{B}_i \neq \mathcal{C}_j$.

Proof:

Case 1 *Suppose* $\mathcal{C}_n = \mathcal{B}_{n-k}$*, for some* $n, k \geq 1$*.*

$$\text{Then } \mathcal{C}_n = F_{2n+1}\alpha - F_{2n}\beta$$

$$\text{and } \mathcal{B}_{n-k} = F_{2n-2k+2}\alpha - F_{2n-2k+1}\beta$$

$$\begin{aligned}\text{imply } \frac{\alpha}{\beta} &= \frac{F_{2n} - F_{2n-2k+1}}{F_{2n+1} - F_{2n-2k+2}} \\ &= \frac{F_{2n} - F_{2n-2k+1}}{(F_{2n} + F_{2n-1}) - (F_{2n-2k+1} + F_{2n-2k})} \\ &= \frac{F_{2n} - F_{2n-2k+1}}{(F_{2n} - F_{2n-2k+1}) + F_{2n-1} - F_{2n-2k}}\end{aligned}$$

$$\text{but } k \geq 1 \text{ so } 2n - 2k \leq 2n - 2 < 2n - 1$$

$$\text{so } F_{2n-1} - F_{2n-2k} > 0$$

$$\therefore \frac{F_{2n} - F_{2n-2k+1}}{(F_{2n} - F_{2n-2k+1}) + F_{2n-1} - F_{2n-2k}} < 1 \text{ implying that } \frac{\alpha}{\beta} < 1$$

This is a contradiction, since by definition $\alpha > \beta$ so $\frac{\alpha}{\beta} > 1$. Therefore, $\mathcal{C}_n \neq \mathcal{B}_{n-k}$ for all $n, k \geq 1$.

Case 2 *Suppose* $\mathcal{C}_0 = \mathcal{B}_0$*.*

$$\text{Then } \mathcal{C}_0 = \mathcal{B}_0$$

$$\text{implies } F_{2(0)+1}\alpha - F_{2(0)}\beta = F_{2(0)+2}\alpha - F_{2(0)+1}\beta$$

$$\text{implies } \alpha - \beta = 2\alpha - \beta$$

$$\alpha = 0$$

This is a contradiction, since by definition $\alpha > 0$, therefore $\mathcal{C}_0 \neq \mathcal{B}_0$.

Case 3 *Suppose* $\mathcal{C}_n = \mathcal{B}_n$ *for some* $n \geq 1$*.*

$$\text{Then } \mathcal{C}_n = \mathcal{B}_n$$

$$\begin{aligned}\text{implies } F_{2n+1}\alpha - F_{2n}\beta &= F_{2n+2}\alpha - F_{2n+1}\beta \\ \frac{\alpha}{\beta} &= \frac{F_{2n} - F_{2n+1}}{F_{2n+1} - F_{2n+2}} \\ &= \frac{-F_{2n-1}}{-F_{2n}}\end{aligned}$$

$$\therefore \frac{\alpha}{\beta} < 1$$

This is a contradiction, therefore $\mathcal{C}_n \neq \mathcal{B}_n$ for all $n \geq 1$.

Case 4 *Suppose $\mathcal{C}_n = \mathcal{B}_{n+k}$ for some $n \geq 0, k \geq 1$.*

$$\text{Then } \mathcal{C}_n = \mathcal{B}_{n+k}, k \geq 1$$

$$\begin{aligned}
\text{implies } F_{2n+1}\alpha - F_{2n}\beta &= F_{2(n+k)+2}\alpha - F_{2(n+k)+1}\beta \\
\frac{\alpha}{\beta} &= \frac{F_{2n} - F_{2(n+k)+1}}{F_{2n+1} - F_{2(n+k)+2}} \\
&= \frac{F_{2n} - F_{2(n+k)+1}}{(F_{2n-1} + F_{2n}) - (F_{2(n+k)} + F_{2(n+k)+1})} \\
&= \frac{-1}{-1} \cdot \frac{F_{2n} - F_{2(n+k)+1}}{(F_{2n} - F_{2(n+k)+1}) + F_{2n-1} - F_{2(n+k)}} \\
&= \frac{F_{2(n+k)+1} - F_{2n}}{(F_{2(n+k)+1} - F_{2n}) + F_{2(n+k)} - F_{2n-1}}
\end{aligned}$$

$$\text{but } k \geq 1 \text{ so } 2(n+k) \geq 2n+2 > 2n-1$$

$$\therefore \frac{F_{2(n+k)+1} - F_{2n}}{(F_{2(n+k)+1} - F_{2n}) + F_{2(n+k)} - F_{2n-1}} < 1$$

$$\frac{\alpha}{\beta} < 1$$

This is a contradiction, therefore $\mathcal{C}_n \neq \mathcal{B}_{n+k}$ for all $n \geq 0, k \geq 1$.

Thus we have proved that $\forall i, j \geq 0, \mathcal{B}_i \neq \mathcal{C}_j$. □

Lemma 2.6 *$\forall i, j \geq 0, \mathcal{B}_i \neq \mathcal{E}_j$ except in the following cases:*

- $\mathcal{B}_n = \mathcal{E}_{n-1}$ *iff* $\frac{\alpha}{\beta} = \frac{F_{2n+2}}{2F_{2n}}$
- $\mathcal{B}_n = \mathcal{E}_n$ *iff* $\frac{\alpha}{\beta} = \frac{F_{2n+3}}{F_{2n}}$

Proof:

Case 1 *Suppose* $\mathcal{B}_n = \mathcal{E}_{n-k}$, *for some* $n, k \geq 2$.

$$
\begin{aligned}
\text{Then } \mathcal{B}_n &= \mathcal{E}_{n-k} \\
\text{implies } F_{2n+2}\alpha - F_{2n+1}\beta &= F_{2(n-k)+1}\alpha + F_{2(n-k)+2}\beta \\
\frac{\alpha}{\beta} &= \frac{F_{2n+1} + F_{2(n-k)+2}}{F_{2n+2} - F_{2(n-k)+1}} \\
&= \frac{F_{2n+1} + F_{2(n-k)+2}}{(F_{2n} + F_{2n+1}) - (F_{2(n-k)+3} - F_{2(n-k)+2})} \\
&= \frac{F_{2n+1} + F_{2(n-k)+2}}{(F_{2n+1} + F_{2(n-k)+2}) + F_{2n} - F_{2(n-k)+3}}
\end{aligned}
$$

$$
\text{but } k \geq 2 \text{ so } 2(n-k) + 3 \leq 2n - 1
$$

$$
\frac{F_{2n+1} + F_{2(n-k)+2}}{(F_{2n} + F_{2n+1}) - (F_{2(n-k)+3} - F_{2(n-k)+2})} < 1
$$

$$
\therefore \frac{\alpha}{\beta} < 1
$$

This is a contradiction, therefore $\mathcal{B}_n \neq \mathcal{E}_{n-k}$ for all $n, k \geq 2$.

Case 2 *Suppose* $\mathcal{B}_n = \mathcal{E}_{n-1}$, *for some* $n \geq 1$.

$$
\begin{aligned}
\text{Then } \mathcal{B}_n &= \mathcal{E}_{n-1} \\
\text{implies } F_{2n+2}\alpha - F_{2n+1}\beta &= F_{2(n-1)+1}\alpha + F_{2(n-1)+2}\beta \\
\frac{\alpha}{\beta} &= \frac{F_{2n+1} + F_{2n}}{F_{2n+2} - F_{2n-1}} \\
&= \frac{F_{2n+2}}{F_{2n+2} - F_{2n-1}} \\
&= \frac{F_{2n+2}}{F_{2n+2} - (-F_{2n} + F_{2n+1})} \\
&= \frac{F_{2n+2}}{(F_{2n+1} + F_{2n}) + F_{2n} - F_{2n+1}} \\
\therefore \frac{\alpha}{\beta} &= \frac{F_{2n+2}}{2F_{2n}} \text{ implying that } \mathcal{B}_n = \mathcal{E}_{n-1}
\end{aligned}
$$

Therefore in the case that for $n \geq 1$ we have $\frac{\alpha}{\beta} = \frac{F_{2n+2}}{2F_{2n}}$, then $\mathcal{B}_n = \mathcal{E}_{n-1}$.

Case 3 *Suppose* $\mathcal{B}_n = \mathcal{E}_n$*, for some* $n \geq 0$*.*

$$\begin{aligned} \text{Then } \mathcal{B}_n &= \mathcal{E}_n \\ \text{implies } F_{2n+2}\alpha - F_{2n+1}\beta &= F_{2n+1}\alpha + F_{2n+2}\beta \\ \frac{\alpha}{\beta} &= \frac{(F_{2n+1} + F_{2n+2})}{F_{2n+2} - F_{2n+1}} \\ &= \frac{F_{2n+3}}{(F_{2n+1} + F_{2n}) - F_{2n+1}} \\ \therefore \frac{\alpha}{\beta} &= \frac{F_{2n+3}}{F_{2n}} \text{ implying that } \mathcal{B}_n = \mathcal{E}_n \end{aligned}$$

Therefore in the case that for $n \geq 0$ we have $\frac{\alpha}{\beta} = \frac{F_{2n+3}}{F_{2n}}$, then $\mathcal{B}_n = \mathcal{E}_n$.

Case 4 *Suppose* $\mathcal{B}_n = \mathcal{E}_{n+1}$*, for some* $n \geq 0$*.*

$$\begin{aligned} \text{Then } \mathcal{B}_n &= \mathcal{E}_{n+1} \\ \text{implies } F_{2n+2}\alpha - F_{2n+1}\beta &= F_{2(n+1)+1}\alpha + F_{2(n+1)+2}\beta \\ \frac{\alpha}{\beta} &= \frac{F_{2n+1} + F_{2n+4}}{F_{2n+2} - F_{2n+3}} < 0 \end{aligned}$$

This is a contradiction, so $\mathcal{B}_n \neq \mathcal{E}_{n+1}$ for all $n \geq 0$.

Case 5 *Suppose* $\mathcal{B}_n = \mathcal{E}_{n+k}$*, for some* $n \geq 0, k > 1$*.*

From Case 4, we have that $\mathcal{E}_{n+1} \neq \mathcal{B}_n$ for all n, and in particular that $\mathcal{E}_{n+1} \neq \mathcal{B}_n$ since $F_{2n+2}\alpha - F_{2n+1}\beta > F_{2(n+1)+1}\alpha + F_{2(n+1)+2}\beta$ for all n. Since it is always true that $\mathcal{E}_{n+k} > \mathcal{E}_n$ for all $k > 0$, this result also implies that $\mathcal{E}_{n+k} > \mathcal{B}_n \forall n, k \geq 1$, and therefore $\mathcal{B}_n \neq \mathcal{E}_{n+k}$ for all $n \geq 0, k \geq 1$.

Thus we have proved the lemma. □

Lemma 2.7 *$\forall i, j \geq 0, \mathcal{B}_i \neq \mathcal{F}_j$ except if $\frac{\alpha}{\beta} = 2$, in which case $\mathcal{B}_n = \mathcal{F}_n$ for all n.*

Proof:

Case 1 *Suppose $\mathcal{B}_n = \mathcal{F}_{n-k}$, for some $n, k \geq 1$.*

$$
\begin{aligned}
\text{Then } \mathcal{B}_n &= \mathcal{F}_{n-k} \\
\text{implies } F_{2n+2}\alpha - F_{2n+1}\beta &= F_{2(n-k)}\alpha + F_{2(n-k)+1}\beta \\
\frac{\alpha}{\beta} &= \frac{F_{2n+1} + F_{2(n-k)+1}}{F_{2n+2} - F_{2(n-k)}} \\
&= \frac{F_{2n+1} + F_{2(n-k)+1}}{(F_{2n} + F_{2n+1}) - (F_{2(n-k)+2} - F_{2(n-k)+1})} \\
&= \frac{F_{2n+1} + F_{2(n-k)+1}}{(F_{2n+1} + F_{2(n-k)+1}) + F_{2n} - F_{2(n-k)+2}} \\
\text{Noting that } 2(n-k) + 2 &\leq 2n \\
\therefore \frac{\alpha}{\beta} &= \frac{F_{2n+1} + F_{2(n-k)+1}}{(F_{2n+1} + F_{2(n-k)+1}) + F_{2n} - F_{2(n-k)+2}} \leq 1
\end{aligned}
$$

This is a contradiction, so $\mathcal{B}_n \neq \mathcal{F}_{n-k}$ for all $n, k \geq 1$.

Case 2 *Suppose $\mathcal{B}_n = \mathcal{F}_n$, for some $n \geq 0$.*

$$
\begin{aligned}
\text{Then } \mathcal{B}_n &= \mathcal{F}_n \\
\text{implies } F_{2n+2}\alpha - F_{2n+1}\beta &= F_{2n}\alpha + F_{2n+1}\beta \\
\frac{\alpha}{\beta} &= \frac{F_{2n+1} + F_{2n+1}}{F_{2n+2} - F_{2n}} \\
&= \frac{2F_{2n+1}}{(F_{2n+1} + F_{2n}) - F_{2n}} \\
\therefore \frac{\alpha}{\beta} &= 2 \text{ implying that } \mathcal{B}_n = \mathcal{F}_n \text{ for all n.}
\end{aligned}
$$

Case 3 *Suppose $\mathcal{B}_n = \mathcal{F}_{n+1}$, for some $n \geq 0$.*

$$\begin{aligned}\text{Then } \mathcal{B}_n &= \mathcal{F}_{n+1}\\ \text{implies } F_{2n+2}\alpha - F_{2n+1}\beta &= F_{2(n+1)}\alpha + F_{2(n+1)+1}\beta\\ F_{2n+2}\alpha - F_{2n+1}\beta &= F_{2n+2}\alpha + F_{2n+3}\beta\\ (F_{2n+1} + F_{2n+3})\beta &= 0\end{aligned}$$

But since $F_{2n+1} + F_{2n+3} > 0$ and $\beta > 0$, this is a contradiction. Therefore it is the case that $\mathcal{B}_n \neq \mathcal{F}_{n+1}$ for all $n \geq 0$.

Case 4 *Suppose* $\mathcal{B}_n = \mathcal{F}_{n+k}$, *for some* $n \geq 0, k \geq 2$.

$$\begin{aligned}\text{Then } \mathcal{B}_n &= \mathcal{F}_{n+k}\\ \text{implies } F_{2n+2}\alpha - F_{2n+1}\beta &= F_{2(n+k)}\alpha + F_{2(n+k)+1}\beta\\ F_{2n+2}\alpha - F_{2n+1}\beta &= F_{2n+2k}\alpha + F_{2n+2k+1}\beta\\ \frac{\alpha}{\beta} &= \frac{F_{2n+1} + F_{2n+2k+1}}{F_{2n+2} - F_{2n+2k}} < 0\end{aligned}$$

This is a contradiction, so $\mathcal{B}_n \neq \mathcal{F}_{n+k}$ for all $n \geq 0, k \geq 2$. Thus we have proved that $\forall i, j \geq 0, \mathcal{B}_i \neq \mathcal{F}_j$, except if $\frac{\alpha}{\beta} = 2$, in which case $\mathcal{B}_n = \mathcal{F}_n$. □

Lemma 2.8 $\forall i, j \geq 0, \mathcal{C}_i \neq \mathcal{E}_j$ *except if* $\frac{\alpha}{\beta} = 2$, *in which case* $\mathcal{C}_n = \mathcal{E}_{n-1}$ *for all* n.

Proof:

Case 1 *Suppose* $\mathcal{C}_n = \mathcal{E}_{n-k}$, *for some* $n \geq 2, k \geq 2$.

Then $\mathcal{C}_n = \mathcal{E}_{n-k}$*implies* $F_{2n+1}\alpha - F_{2n}\beta = F_{2(n-k)+1}\alpha + F_{2(n-k)+2}\beta$

$$\begin{aligned}\frac{\alpha}{\beta} &= \frac{F_{2n} + F_{2(n-k)+2}}{F_{2n+1} - F_{2(n-k)+1}}\\ &= \frac{F_{2n} + F_{2(n-k)+2}}{(F_{2n-1} + F_{2n}) - (-F_{2(n-k)+2} + F_{2(n-k)+3})}\\ &= \frac{F_{2n} + F_{2(n-k)+2}}{(F_{2n} + F_{2(n-k)+2}) + F_{2n-1} - F_{2(n-k)+3})}\end{aligned}$$

$$\text{but } k \geq 2 \text{ implies that } F_{2n-1} \geq F_{2(n-k)+3}$$

$$\therefore \frac{\alpha}{\beta} \leq 1$$

This is a contradiction, so $\mathcal{C}_n \neq \mathcal{E}_{n-k}$ for all $n \geq 0, k \geq 2$.

Case 2 *Suppose $\mathcal{C}_n = \mathcal{E}_{n-1}$, for some $n \geq 1$.*

$$\text{Then } \mathcal{C}_n = \mathcal{E}_{n-1}$$

$$\text{implies } F_{2n+1}\alpha - F_{2n}\beta = F_{2(n-1)+1}\alpha + F_{2(n-1)+2}\beta$$

$$\begin{aligned}
F_{2n+1}\alpha - F_{2n}\beta &= F_{2n-1}\alpha + F_{2n}\beta \\
\frac{\alpha}{\beta} &= \frac{F_{2n} + F_{2n}}{F_{2n+1} - F_{2n-1}} \\
&= \frac{2F_{2n}}{(F_{2n} + F_{2n-1}) - F_{2n-1}} \\
&= \frac{2F_{2n}}{2F_{2n}} = 2 \\
\therefore \frac{\alpha}{\beta} &= 2 \text{ implying that } \mathcal{C}_n = \mathcal{E}_{n-1}
\end{aligned}$$

So in the case that $\frac{\alpha}{\beta} = 2$, then $\mathcal{C}_n = \mathcal{E}_{n-1}$ for all $n \geq 1$.

Case 3 *Suppose $\mathcal{C}_n = \mathcal{E}_n$, for some $n \geq 0$.*

$$\text{Then } \mathcal{C}_n = \mathcal{E}_n$$

$$\text{implies } F_{2n+1}\alpha - F_{2n}\beta = F_{2n+1}\alpha + F_{2n+2}\beta$$

$$(F_{2n} + F_{2n+2})\beta = 0$$

But since $F_{2n} + F_{2n+2} > 0$ and $\beta > 0$, this is a contradiction. Therefore it is the case that $\mathcal{C}_n \neq \mathcal{E}_n$ for all $n \geq 0$.

Case 4 *Suppose $\mathcal{C}_n = \mathcal{E}_{n+k}$, for some $n \geq 0, k \geq 1$.*

$$\text{Then } \mathcal{C}_n = \mathcal{E}_{n+k}$$

$$\text{implies } F_{2n+1}\alpha - F_{2n}\beta = F_{2(n+k)+1}\alpha + F_{2(n+k)+2}\beta$$

$$\frac{\alpha}{\beta} = \frac{F_{2n} + F_{2(n+k)+2}}{F_{2n+1} - F_{2(n+k)+1}}$$

$$\text{but } k \geq 1 \text{ implies that } 2(n+k)+1 \geq 2(n+1)+1 = 2n+3$$

$$\therefore F_{2n+1} - F_{2(n+k)+1} < 0 \text{ implying that } \frac{\alpha}{\beta} < 0$$

This is a contradiction, so $\mathcal{C}_n \neq \mathcal{E}_{n+k}$ for all $n \geq 0, k \geq 1$. Thus we have proved that $\forall i, j \geq 0, \mathcal{C}_i \neq \mathcal{E}_j$, except if $\frac{\alpha}{\beta} = 2$, in which case $\mathcal{C}_n = \mathcal{E}_{n-1}$ for all $n \geq 1$. □

Lemma 2.9 $\forall i, j \geq 0, \mathcal{C}_i \neq \mathcal{F}_j$ *except in the following cases:*

- $\mathcal{C}_n = \mathcal{F}_{n-1}$ *iff* $\frac{\alpha}{\beta} = \frac{F_{2n+1}}{2F_{2n-1}}$
- $\mathcal{C}_n = \mathcal{F}_n$ *iff* $\frac{\alpha}{\beta} = \frac{F_{2n+2}}{F_{2n-1}}$

Proof:

Case 1 *Suppose* $\mathcal{C}_n = \mathcal{F}_{n-k}$*, for some* $n, k \geq 2$*.*

$$\text{Then } \mathcal{C}_n = \mathcal{F}_{n-k}$$

$$\text{implies } F_{2n+1}\alpha - F_{2n}\beta = F_{2(n-k)}\alpha + F_{2(n-k)+1}\beta$$

$$\frac{\alpha}{\beta} = \frac{F_{2n} + F_{2(n-k)+1}}{F_{2n+1} - F_{2(n-k)}}$$

$$= \frac{F_{2n} + F_{2(n-k)+1}}{(F_{2n-1} + F_{2n}) - (-F_{2(n-k)+1} + F_{2(n-k)+2})}$$

$$\text{which can be written as } \frac{F_{2n} + F_{2(n-k)+1}}{(F_{2n} + F_{2(n-k)+1}) + F_{2n-1} - F_{2(n-k)+2}}$$

Note that since $2n-1 > 2(n-k)+2$ for all $k \geq 2$

we know that $F_{2n} > F_{2(n-k)+2}$ and therefore $F_{2n} - F_{2(n-k)+2} > 0$

This implies that $F_{2n} + F_{2(n-k)+1} < (F_{2n} + F_{2(n-k)+1}) + F_{2n-1} - F_{2(n-k)+2}$

So now we have that $\dfrac{\alpha}{\beta} = \dfrac{F_{2n} + F_{2(n-k)+1}}{(F_{2n-1} + F_{2n}) - (-F_{2(n-k)+1} + F_{2(n-k)+2})} < 1$

$\therefore \dfrac{\alpha}{\beta} < 1$

This is a contradiction, therefore $\mathcal{C}_n \neq \mathcal{F}_{n-k}$ for all $n, k \geq 2$.

Case 2 *Suppose* $\mathcal{C}_n = \mathcal{F}_{n-1}$ *for some* $n \geq 1$.

$$
\begin{aligned}
\text{Then } \mathcal{C}_n &= \mathcal{F}_{n-1} \\
\text{implies } F_{2n+1}\alpha - F_{2n}\beta &= F_{2n-2}\alpha + F_{2n-1}\beta \\
\frac{\alpha}{\beta} &= \frac{(F_{2n} + F_{2n-1})}{F_{2n+1} - F_{2n-2}} \\
&= \frac{F_{2n+1}}{(F_{2n} + F_{2n-1}) - F_{2n-2}} \\
&= \frac{F_{2n+1}}{F_{2n} + F_{2n-1} - (-F_{2n-1} + F_{2n})} \\
&= \frac{F_{2n+1}}{2F_{2n-1}} \\
\therefore \frac{\alpha}{\beta} &= \frac{F_{2n+1}}{2F_{2n-1}} \text{ implying that } \mathcal{C}_n = \mathcal{F}_{n-1}
\end{aligned}
$$

Therefore in the case that for $n \geq 1$ we have $\frac{\alpha}{\beta} = \frac{F_{2n+1}}{2F_{2n-1}}$, then $\mathcal{C}_n = \mathcal{F}_{n-1}$.

Case 3 *Suppose* $\mathcal{C}_0 = \mathcal{F}_0$.

$$\text{Then } \mathcal{C}_0 = \mathcal{F}_0$$

$$\text{implies } F_{2(0)+1}\alpha - F_{2(0)}\beta = F_{2(0)}\alpha + F_{2(0)+1}\beta$$

$$\text{implies } \alpha - \beta = \alpha + \beta$$

$$\beta = 0$$

This is a contradiction, since by definition $\beta > 0$, therefore $\mathcal{C}_0 \neq \mathcal{F}_0$.

Case 4 *Suppose* $\mathcal{C}_n = \mathcal{F}_n$*, for some* $n \geq 1$*.*

$$\begin{aligned} \text{Then } \mathcal{C}_n &= \mathcal{F}_n \\ \text{implies } F_{2n+1}\alpha - F_{2n}\beta &= F_{2n}\alpha + F_{2n+1}\beta \\ \frac{\alpha}{\beta} &= \frac{(F_{2n} + F_{2n+1})}{(F_{2n+1} - F_{2n})} \\ &= \frac{F_{2n+2}}{F_{2n-1}} \\ \therefore \frac{\alpha}{\beta} &= \frac{F_{2n+2}}{F_{2n-1}} \text{ implying that } \mathcal{C}_n = \mathcal{F}_n \end{aligned}$$

Therefore in the case that for $n \geq 1$ we have $\frac{\alpha}{\beta} = \frac{F_{2n+2}}{F_{2n-1}}$, then $\mathcal{C}_n = \mathcal{F}_n$.

Case 5 *Suppose* $\mathcal{C}_n = \mathcal{F}_{n+k}$*, for some* $n \geq 0, k \geq 1$*.*

$$\begin{aligned} \text{Then } \mathcal{C}_n &= \mathcal{F}_{n+k} \\ \text{implies } F_{2n+1}\alpha - F_{2n}\beta &= F_{2(n+k)}\alpha + F_{2(n+k)+1}\beta \\ \frac{\alpha}{\beta} &= \frac{F_{2n} + F_{2(n+k)+1}}{F_{2n+1} - F_{2(n+k)}} \\ &= \frac{F_{2n} + F_{2(n+k)+l}}{F_{2n+1} - F_{2(n+k)}} < 0 \end{aligned}$$

$$\text{but } k \geq 1 \text{ implies that } 2(n+k) \geq 2n+2 > 2n+1$$

$$\text{which implies that } F_{2n+1} - F_{2(n+k)} < 0$$

$$\therefore \frac{\alpha}{\beta} < 1$$

This is a contradiction, so $\mathcal{C}_n \neq \mathcal{F}_{n+k}$ for all $n \geq 0, k \geq 1$. Thus we have proved the lemma. □

Lemma 2.10 For all $i, j \geq 0$, $\mathcal{E}_i \neq \mathcal{F}_j$.

Proof:

Case 1 *Suppose* $\mathcal{E}_n = \mathcal{F}_{n-k}$*, for some* $n, k \geq 1$*.*

$\mathcal{F}_{n-k} < \mathcal{F}_n \forall k$, and in turn $\mathcal{F}_n < \mathcal{E}_n \forall n$, so $\mathcal{E}_n \neq \mathcal{F}_{n-k}$ for all $n, k \geq 1$.

Case 2 *Suppose* $\mathcal{E}_0 = \mathcal{F}_0$*.*

Then $\mathcal{E}_0 = \mathcal{F}_0$ implying that $\alpha + \beta = \alpha + 2\beta$. But since this would imply that $\beta = 0$ which by definition is not true, we have that $\mathcal{E}_0 \neq \mathcal{F}_0$.

Case 3 *Suppose* $\mathcal{E}_n = \mathcal{F}_n$ *for some* $n \geq 1$*.*

$\mathcal{E}_n = \mathcal{E}_{n-1} + \mathcal{F}_n$ implying that $\mathcal{E}_n > \mathcal{F}_n$, so $\mathcal{E}_n \neq \mathcal{F}_n$ for all $n \geq 1$.

Case 4 *Suppose* $\mathcal{E}_n = \mathcal{F}_{n+k}$*, for some* $n \geq 0, k \geq 1$*.*

$\mathcal{F}_{n+1} = \mathcal{F}_n + \mathcal{E}_n$ implying that $\mathcal{F}_{n+1} > \mathcal{E}_n$

It follows that for $k \geq 1, \mathcal{F}_{n+k} \geq \mathcal{F}_{n+1} > \mathcal{E}_n$ and therefore $\mathcal{E}_n \neq \mathcal{F}_{n+k}$ for all $n \geq 0, k \geq 1$. Thus we have proved that for all $i, j \geq 0$, $\mathcal{E}_i \neq \mathcal{F}_j$. □

To summarize, from all this we get the following "bad" (α, β) pairs:

$\frac{\alpha}{\beta}$	Repeated Squares
2	$\mathcal{B}_n = \mathcal{F}_n$
2	$\mathcal{C}_n = \mathcal{E}_{n-1}$
$\frac{F_{2n+3}}{F_{2n}}$	$\mathcal{B}_n = \mathcal{E}_n$
$\frac{F_{2n+2}}{F_{2n-1}}$	$\mathcal{C}_n = \mathcal{F}_n$
$\frac{F_{2n+2}}{2F_{2n}}$	$\mathcal{B}_n = \mathcal{E}_{n-1}$
$\frac{F_{2n+1}}{2F_{2n-1}}$	$\mathcal{C}_n = \mathcal{F}_{n-1}$

Table 2.1: $\frac{\alpha}{\beta}$ ratios that will result in specific repeated squares.

It is important to note here that the four sequences in the last four lines of the table are all disjoint. Additionally, notice that we can combine $\frac{\alpha}{\beta} = \frac{F_{2n+3}}{F_{2n}}$ and $\frac{\alpha}{\beta} = \frac{F_{2n+2}}{F_{2n-1}}$ into the single case $\frac{\alpha}{\beta} = \frac{F_{k+3}}{F_k}$. Similarly, we can combine $\frac{\alpha}{\beta} = \frac{F_{2n+2}}{2F_{2n}}$ and $\frac{\alpha}{\beta} = \frac{F_{2n+1}}{2F_{2n-1}}$ into the single case $\frac{F_{k+2}}{2F_k}$. So now we know that for $\frac{\alpha}{\beta} \neq 2, \frac{F_{k+3}}{F_k}$, or $\frac{F_{k+2}}{2F_k}$, all squares added from the pattern $\mathcal{FECB}$ starting with an ell of the α and β squares will be unique, and we have proved Theorem 2.4. □

As it turns out, the sequences given by Theorem 2.4, $\frac{F_{k+3}}{F_k}$ and $\frac{F_{k+2}}{2F_k}$, are distinct from one another. To show this, we must describe the sequences explicitly. We know from number theory that there is an explicit formula for terms in the Fibonacci sequence: [2]

$$\text{Let } \varphi = \frac{1+\sqrt{5}}{2}, \text{ the golden ratio.}$$

$$F_n = \frac{\varphi^{n+1} - (-\varphi)^{-(n+1)}}{\sqrt{5}}$$

Therefore we can rewrite the $\frac{\alpha}{\beta}$ ratios described above explicitly, in terms of φ and k:

$$\frac{\alpha}{\beta} = \frac{F_{k+3}}{F_k} = \frac{\frac{\varphi^{(k+3)+1} - (-\varphi)^{-((k+3)+1)}}{\sqrt{5}}}{\frac{\varphi^{k+1} - (-\varphi)^{-(k+1)}}{\sqrt{5}}} = \frac{\varphi^{(k+3)+1} - (-\varphi)^{-((k+3)+1)}}{\varphi^{k+1} - (-\varphi)^{-(k+1)}}$$

$$\frac{\alpha}{\beta} = \frac{F_{k+2}}{2F_k} = \frac{\frac{\varphi^{(k+2)+1} - (-\varphi)^{-((k+2)+1)}}{\sqrt{5}}}{2(\frac{\varphi^{k+1} - (-\varphi)^{-(k+1)}}{\sqrt{5}})} = \frac{\varphi^{(k+2)+1} - (-\varphi)^{-((k+2)+1)}}{2(\varphi^{k+1} - (-\varphi)^{-(k+1)})}$$

Note that for $k \geq 0, k \in \mathbb{Z}$, these sequences are both convergent. Figure 16 shows the sequences.

$$\lim_{k\to\infty} \frac{F_{k+3}}{F_k} = \lim_{k\to\infty} \frac{\varphi^{(k+3)+1} - (-\varphi)^{-((k+3)+1)}}{\varphi^{k+1} - (-\varphi)^{-(k+1)}} = \lim_{k\to\infty} \frac{\varphi^{(k+3)+1} - 0}{\varphi^{k+1} - 0} = \varphi^3 = 2 + \sqrt{5}$$

$$\lim_{k\to\infty} \frac{F_{k+2}}{2F_k} = \lim_{k\to\infty} \frac{\varphi^{(k+2)+1} - (-\varphi)^{-((k+2)+1)}}{2(\varphi^{k+1} - (-\varphi)^{-(k+1)})} = \lim_{k\to\infty} \frac{\varphi^{(k+2)+1} - 0}{2(\varphi^{k+1} - 0)} = \frac{\varphi^2}{2} = \frac{3+\sqrt{5}}{4}$$

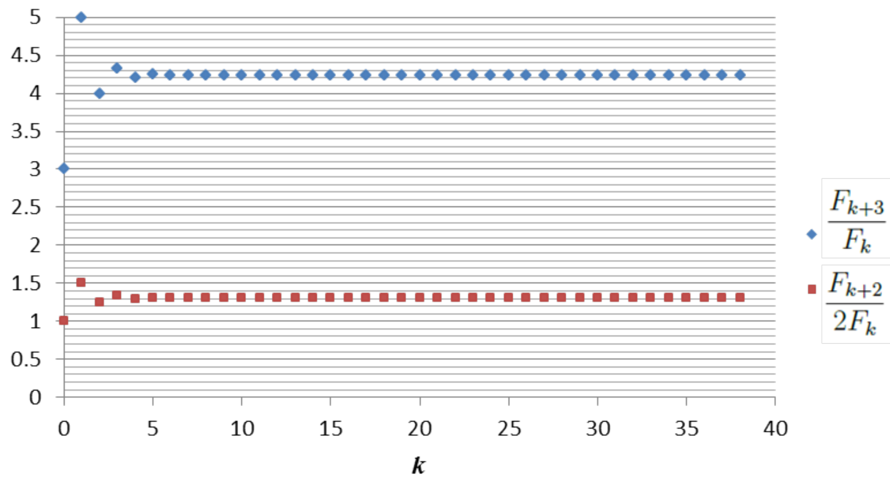

Figure 16: Graphing the two series.

Recall the examples given before the proof of Theorem 2.4. Examples 1 and 2 showed repeated squares in the first few rounds of the $\mathcal{FECB}$ algorithm, and now we have a mathematical argument for why specific squares were repeated. In the case of Example 1, we had $\alpha = 12, \beta = 3$, which means $\frac{\alpha}{\beta} = 4 = \frac{8}{2} = \frac{F_5}{F_2} = \frac{F_{2n+3}}{F_{2n}}$ when $n = 1$. This means that according to Theorem 2.4, $\mathcal{B}_1 = \mathcal{E}_1$, which we can see was true from Figure 15 (recall that counting the squares added starts at $n = 0$). Similarly, in Example 2, we had $\alpha = 21, \beta = 16$, which means $\frac{\alpha}{\beta} = \frac{21}{2 \cdot 8} = \frac{F_7}{2F_5} = \frac{F_{2n+1}}{2F_{2n-1}}$ when $n = 3$. We can then see that in this case, according to Theorem 2.4, it should be the case that $\mathcal{C}_3 = \mathcal{F}_2$, which we found to be true in the statement of the example before.

Now let us consider Examples 4 and 3. These examples both appeared to be free of repeating squares, but looking at the $\frac{\alpha}{\beta}$ ratio, we see otherwise. Example 4 has $\alpha = 5, \beta = 3$, so $\frac{\alpha}{\beta} = \frac{5}{3} \approx 1.667$. Notice that $\forall k$, $3 \leq \frac{F_{k+3}}{F_k} \leq 5$ and $1 \leq \frac{F_{k+2}}{2F_k} \leq 1.5$, so $\frac{5}{3}$ is clearly not an element of either of these sequences. So by Theorem 2.4, this (α, β) pair tiles the half-plane with entirely unique squares. Example 3, however, is not so neat. With $\alpha = 416020, \beta = 317811$, we have $\frac{\alpha}{\beta} = \frac{832040}{2 \cdot 317811} = \frac{F_{29}}{2F_{27}} = \frac{F_{2n+1}}{2F_{2n-1}}$ when $n = 14$. It will then be the case that $\mathcal{C}_{14} = \mathcal{F}_{13}$, so a square will repeat. In fact, we only had to calculate a few more rounds of the algorithm to see this. Continuing from the table at Example 3:

i	$\mathcal{F}_i$	$\mathcal{E}_i$	$\mathcal{C}_i$	$\mathcal{B}_i$
		...		
11	26658145588	43133785635	10182505533	16475640052
12	69791931223	112925716858	26658145585	43133785637
13	182717648081	295643364939	69791931222	112925716859
14	478361013020	774004377959	182717648081	295643364940

It is clear now from this table that using the results of Theorem 2.4, we were able to determine the exact location of a repeated square at $\mathcal{C}_{14} = \mathcal{F}_{13}$.

For our last example, we had $\alpha = 6, \beta = 3$. In Example 5, we found that many of the squares were repeated, and this is supported by the result of Theorem 2.4. Because $\frac{\alpha}{\beta} = \frac{6}{3} = 2$, we have the result that $\mathcal{B}_n = \mathcal{F}_n$ and that $\mathcal{C}_n = \mathcal{E}_{n-1} \forall n$. This result is empirically supported by the table at Example 5.

From these examples and the discussion of the convergent sequences that result from Theorem 2.4, it should be clear to the reader how Theorem 2.4 can be used effectively.

2.3.2 Less-than-perfect (α, β) Pairs

In the previous section, we describe "good" and "bad" pairings of values for α and β – "good" pairings meaning those that create a starting ell that, when the pattern $\mathcal{FECB}$ is followed, will tile the half-plane with unique integers. A "bad" pairing is one that does not accomplish this goal, but instead repeats some of the squares used to tile the half-plane.

But how "bad" are our bad tilings, exactly? The equations in lines 1 and 2 of Table 2.1 describe cases in which $\forall n$, squares are repeated (see Example 5). In contrast, notice that the equations in lines 3 through 6 of Table 2.1 are all if and only if statements, meaning that there is precisely one scenario in which a repeated square is produced for a certain $\frac{\alpha}{\beta}$ ratio. Additionally, note that each element of the sequences $\frac{F_{k+3}}{F_k}$ and $\frac{F_{k+2}}{2F_k}$ occurs only once within the sequence, as stated in Theorems 2.11 and 2.12.

Theorem 2.11 $\forall k, m \geq 0, \frac{F_{k+3}}{F_k} = \frac{F_{m+3}}{F_m} \Rightarrow k = m.$

Proof: *By contradiction.*

Suppose $\frac{F_{k+3}}{F_k} = \frac{F_{m+3}}{F_m}$ where $k \neq m$ WLOG assume $k < m$.

$$\text{Then } \frac{\varphi^{(k+3)+1} - (-\varphi)^{-((k+3)+1)}}{\varphi^{k+1} - (-\varphi)^{-(k+1)}} = \frac{\varphi^{(m+3)+1} - (-\varphi)^{-((m+3)+1)}}{\varphi^{m+1} - (-\varphi)^{-(m+1)}}$$

$$(\varphi^{k+4} - (-\varphi)^{-(k+4)}) \cdot (\varphi^{m+1} - (-\varphi)^{-(m+1)}) \\ = (\varphi^{(m+4} - (-\varphi)^{-(m+4)}) \cdot (\varphi^{k+1} - (-\varphi)^{-(k+1)})$$

$$\begin{gathered} -\varphi^{k+4}(-\varphi)^{-(m+1)} - \varphi^{m+1}(-\varphi)^{-(k+4)} = -\varphi^{k+1}(-\varphi)^{-(m+4)} - \varphi^{m+4}(-\varphi)^{-(k+1)} \\ \frac{\varphi^k\varphi^3}{(-\varphi)^m} - \frac{\varphi^m}{(-\varphi)^k\varphi^3} = \frac{\varphi^m\varphi^3}{(-\varphi)^k} - \frac{\varphi^k}{(-\varphi)^m\varphi^3} \\ \frac{\varphi^k\varphi^3(-\varphi)^k\varphi^3}{(-\varphi)^k(-\varphi)^m\varphi^3} - \frac{\varphi^m(-\varphi)^m}{(-\varphi)^k(-\varphi)^m\varphi^3} = \frac{\varphi^m\varphi^3(-\varphi)^m\varphi^3}{(-\varphi)^k(-\varphi)^m\varphi^3} - \frac{\varphi^k(-\varphi)^k}{(-\varphi)^k(-\varphi)^m\varphi^3} \\ (-1)^k\varphi^6\varphi^{2k} + (-1)^{m+1}\varphi^{2m} + (-1)^{m+1}\varphi^6\varphi^{2m} + (-1)^k\varphi^{2k} = 0 \end{gathered}$$

Case 1 *m is odd*

$$\begin{gathered} \text{Then } (-1)^k\varphi^6\varphi^{2k} + (-1)^{m+1}\varphi^6\varphi^{2m} > 0 \\ \text{and } (-1)^{m+1}\varphi^{2m} + (-1)^k\varphi^{2k} > 0 \\ \text{so } (-1)^k\varphi^6\varphi^{2k} + (-1)^{m+1}\varphi^{2m} + (-1)^{m+1}\varphi^6\varphi^{2m} + (-1)^k\varphi^{2k} \neq 0 \end{gathered}$$

so it is not possible that $\frac{F_{k+3}}{F_k} = \frac{F_{m+3}}{F_m}$ for $k < m$ where m is odd.

Case 2 *m is even*

$$\text{Then } (-1)^k\varphi^6\varphi^{2k} + (-1)^{m+1}\varphi^6\varphi^{2m} < 0$$
$$\text{and } (-1)^{m+1}\varphi^{2m} + (-1)^k\varphi^{2k} < 0$$
$$\text{so } (-1)^k\varphi^6\varphi^{2k} + (-1)^{m+1}\varphi^{2m} + (-1)^{m+1}\varphi^6\varphi^{2m} + (-1)^k\varphi^{2k} \neq 0$$

so it is not possible that $\frac{F_{k+3}}{F_k} = \frac{F_{m+3}}{F_m}$ for $k < m$ where m is even.

So $\forall k \neq m$ where WLOG $k < m$, it is the case $\frac{F_{k+3}}{F_k} \neq \frac{F_{m+3}}{F_m}$. This is a contradiction since we assumed $\frac{F_{k+3}}{F_k} = \frac{F_{m+3}}{F_m}$, so it must be true that $m = k$. □

Theorem 2.12 $\forall k, m \geq 0$, $\frac{F_{k+2}}{2F_k} = \frac{F_{m+2}}{2F_m} \Rightarrow k = m$.

Proof: *By contradiction.*

Suppose $\frac{F_{k+2}}{2F_k} = \frac{F_{m+2}}{2F_m}$ where $k \neq m$ WLOG assume $k < m$.

$$\text{Then } \frac{\varphi^{(k+2)+1} - (-\varphi)^{-((k+2)+1)}}{2(\varphi^{k+1} - (-\varphi)^{-(k+1)})} = \frac{\varphi^{(m+2)+1} - (-\varphi)^{-((m+2)+1)}}{2(\varphi^{m+1} - (-\varphi)^{-(m+1)})}$$
$$(\varphi^{k+3} - (-\varphi)^{-(k+3)}) \cdot 2(\varphi^{m+1} - (-\varphi)^{-(m+1)})$$
$$= (\varphi^{m+3} - (-\varphi)^{-(m+3)}) \cdot 2(\varphi^{k+1} - (-\varphi)^{-(k+1)})$$
$$(\varphi^{k+3} - (-\varphi)^{-(k+3)}) \cdot (\varphi^{m+1} - (-\varphi)^{-(m+1)}) = (\varphi^{m+3} - (-\varphi)^{-(m+3)}) \cdot (\varphi^{k+1} - (-\varphi)^{-(k+1)})$$
$$-\frac{\varphi^{m+1}}{(-\varphi)^k(-\varphi)^3} - \frac{\varphi^{k+3}}{(-\varphi)^m(-\varphi)^1} = -\frac{\varphi^{k+1}}{(-\varphi)^m(-\varphi)^3} - \frac{\varphi^{m+3}}{(-\varphi)^k(-\varphi)^1}$$
$$\frac{\varphi^{m-2}(-\varphi)^m}{(-\varphi)^{k+m}} + \frac{\varphi^{k+2}(-\varphi)^k}{(-\varphi)^{k+m}} = \frac{\varphi^{k-2}(-\varphi)^k}{(-\varphi)^{k+m}} + \frac{\varphi^{m+2}(-\varphi)^m}{(-\varphi)^{k+m}}$$
$$(-1)^m\varphi^{2m-2} + (-1)^k\varphi^{2k+2} + (-1)^{k+1}\varphi^{2k-2} + (-1)^{m+1}\varphi^{2m+2} = 0$$

Case 1 *m odd, k even*

$$\text{Then } -\varphi^{2m-2} + \varphi^{2m+2} > 0$$
$$\text{and } \varphi^{2k+2} - \varphi^{2k-2} > 0$$
$$\text{so } (-1)^m\varphi^{2m-2} + (-1)^k\varphi^{2k+2} + (-1)^{k+1}\varphi^{2k-2} + (-1)^{m+1}\varphi^{2m+2} \neq 0$$

so it is not possible that $\frac{F_{k+2}}{2F_k} = \frac{F_{m+2}}{2F_m}$ for $k < m$ where m is odd and k is even.

Case 2 *m even, k odd*

$$\text{Then } \varphi^{2m-2} - \varphi^{2m+2} < 0$$
$$\text{and } -\varphi^{2k+2} + \varphi^{2k-2} < 0$$
$$\text{so } (-1)^m \varphi^{2m-2} + (-1)^k \varphi^{2k+2} + (-1)^{k+1} \varphi^{2k-2} + (-1)^{m+1} \varphi^{2m+2} \neq 0$$

so it is not possible that $\frac{F_{k+2}}{2F_k} = \frac{F_{m+2}}{2F_m}$ for $k < m$ where m is even and k is odd.

Case 3 *m, k both odd*

$$\text{Then } \varphi^{2m-2} - \varphi^{2m+2} < 0$$
$$\text{and } \varphi^{2k+2} - \varphi^{2k-2} < 0$$

Notice that for $x > 1$ and $a > b > 1$ and $c > 0$, $x^a > x^b$ implies $x^a(1-x^c) > x^b(1-x^c)$ and therefore $x^a - x^{a-c} > x^b - x^{b-c}$. So because $m > k$, then we have

$$(\varphi^{2m-2} - \varphi^{2m+2}) + (\varphi^{2k+2} - \varphi^{2k-2}) < 0$$
$$\text{which means that } (-1)^m \varphi^{2m-2} + (-1)^k \varphi^{2k+2} + (-1)^{k+1} \varphi^{2k-2} + (-1)^{m+1} \varphi^{2m+2} \neq 0$$

so it is not possible that $\frac{F_{k+2}}{2F_k} = \frac{F_{m+2}}{2F_m}$ for $k < m$ where m and k are both odd.

Case 4 *m, k both even*

$$\text{Then } -\varphi^{2m-2} + \varphi^{2m+2} > 0$$
$$\text{and } -\varphi^{2k+2} + \varphi^{2k-2} > 0$$

Again using the fact from the third case that for $x > 1$ and $a > b > 1$ and $c > 0$, $x^a - x^{a-c} > x^b - x^{b-c}$, and because $m > k$, then we have

$$(\varphi^{2m-2} - \varphi^{2m+2}) + (\varphi^{2k+2} - \varphi^{2k-2}) < 0$$
$$\text{which means that } (-1)^m \varphi^{2m-2} + (-1)^k \varphi^{2k+2} + (-1)^{k+1} \varphi^{2k-2} + (-1)^{m+1} \varphi^{2m+2} \neq 0$$

so it is not possible that $\frac{F_{k+2}}{2F_k} = \frac{F_{m+2}}{2F_m}$ for $k < m$ where m and k are both even. So $\forall k \neq m$ where WLOG $k < m$, it is the case $\frac{F_{k+2}}{2F_k} \neq \frac{F_{m+2}}{2F_m}$. This is a contradiction since we assumed $\frac{F_{k+2}}{2F_k} = \frac{F_{m+2}}{2F_m}$, so it must be true that $m = k$. □

An interesting consequence of the result that some $\frac{\alpha}{\beta}$ ratios produce only one repeated square is that the repeated square can be precisely identified given only α and β. This means that given a specific "bad" (α, β) pair matching one of the equations in lines 3-6 of Table 2.1, we know exactly which squares will be contained in the ell before the repeated square is added, the order in which the squares were added, and therefore the dimensions of the ell. This allows us to create tilings of the plane where the only repeated square occurs arbitrarily far out in the tiling. That is, given a g, we can find a tiling where the first repeated square occurs after g squares have been added. In addition to the $\frac{\alpha}{\beta}$ ratios that produce no repeated squares in a complete tiling, we also now have infinitely many "bad" $\frac{\alpha}{\beta}$ ratios that produce no repeated squares in the first g squares added. This could provide a tool for a new type of tiling composed of ells based on both "good" and "bad" $\frac{\alpha}{\beta}$ ratios that contain no repeated squares.

2.4 "Compactness" of the $\mathcal{FECB}$ Algorithm

Though the $\mathcal{FECB}$ algorithm creates a tiling that does not contain every square, we can develop a measure of how densely the integer-sided squares are used in this tiling of the half-plane. We compare the $\mathcal{FECB}$ tiling to the naïve tiling described in Section 2.2.

We define a measure of "compactness" of a tiling, in other words a way to measure how many size squares have been added under a certain size when a given number of squares have been added to the ell. We use the average area of a square in the ell. Letting S be the set of all squares used and $S[r]$ be the partial tiling of the first r

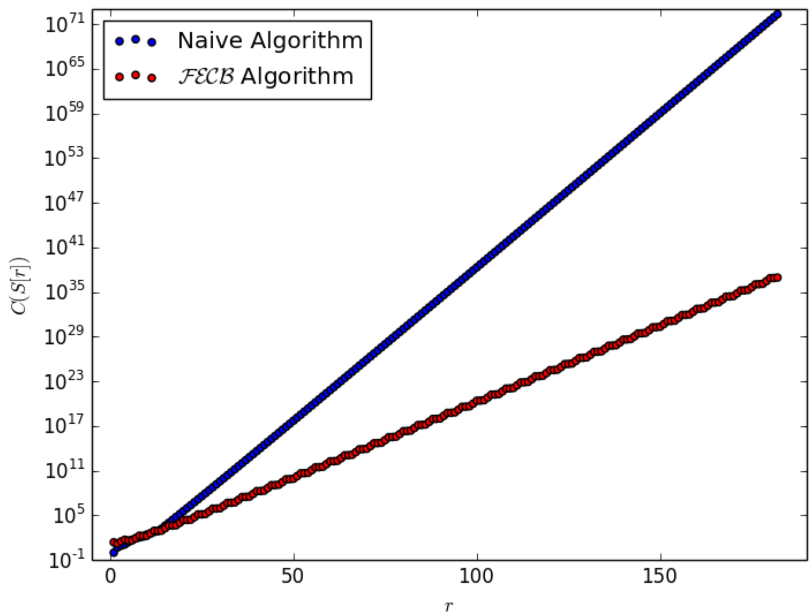

Figure 17: Plot of $C(S[r])$ for the $\mathcal{FECB}$ algorithm (red) vs. the naïve algorithm (blue). The horizontal axis is r. Shown on a log scale, produced using Matplotlib for Python.

squares in S and letting S_i be the side-length of the i^{th} square added:

$$\text{Define } C(S[r]) = \frac{\sum_{i=1}^{r} (S_i)^2}{r}$$

This measure is useful specifically for comparing tilings that use unique sized squares; notice that a tiling entirely of 1×1 square would have a compactness measure of 1. Now compare two different algorithms for tiling the plane. Let T be the set of all squares used in a different algorithm, starting with the same α and β squares. If $C(S[r]) < C(T[r])$ then the average area of one square in the S algorithm is less than that of T, and so the S algorithm is more compact.

The graph in Figure 17 shows a comparison between the compactness of the $\mathcal{FECB}$ algorithm (red) and the Fibonacci-like algorithm (blue) for $r = 1..182$ graphed on a log scale. It is clear from this graph that the $\mathcal{FECB}$ algorithm is significantly more compact.

3 Conclusion

In this thesis I describe an algorithm which tiles the half-plane using unique integral-sided squares and a compactness equation with which the algorithm can be compared to other methods of tiling the half-plane using unique squares, such as the naïve tiling described in Section 2.2. An ideal extension to Henle's work would have been to find a tiling which uses precisely one square of each integral side-length. This proved difficult to extend to the half-plane because the limited directions for expanding the ell made the ways in which squares could be added very limiting.

A Computational tools for empirical data

In order to better understand and interpret the tilings I was creating, I wrote several Python programs to keep track of the size of the ell and the squares added to the ell for different algorithms. The code in Appendix A.1 performs the $\mathcal{FECB}$ algorithm a given number of times for a given (α, β) pair. It checks for repeated squares as it adds new squares. The code in Appendix A.2 performs the naïve tiling described in Section 2.2 a given number of times for a given composite rectangle, composed of unique integer-sided squares. The code in Appendix A.3 uses the data received from these first two programs, or any two programs that produce tilings of the half-plane, to plot the data using the Matplotlib package for Python and compare the compactness of the two input tilings.

A.1 Python program for $\mathcal{FECB}$ Algorithm

```
from __future__ import division
import math

# This class keeps track of all of the squares within an ell and
#   contains the methods to add new squares. It also contains a
#   method to check if a newly added square is repeating an old
#   square.
class Ell:
    def __init__(self,sqAlpha,sqBeta):
        self.sqAlpha = sqAlpha
        self.sqBeta = sqBeta

        self.btuple = Comb(1,0)
        self.ctuple = Comb(1,-1)
        self.dtuple = Comb(0,1)
        self.etuple = Comb(0,1)

        self.squares = [self.sqAlpha,self.sqBeta]
        self.totalarea = pow(self.sqAlpha,2)+pow(self.sqBeta,2)
        self.logarea = [pow(self.sqAlpha,2),pow(self.sqAlpha,2)+pow(
            self.sqBeta,2)]
        self.bsquares = []
        self.csquares = []
```

```
        self.dsquares = []
        self.esquares = []
        self.fsquares = []

    def B(self):
        self.ctuple += self.btuple
        sq = Comb(self.btuple.partA,self.btuple.partB)
        self.checkadd(sq,self.bsquares)

    def C(self):
        self.btuple += self.ctuple
        self.dtuple -= self.ctuple
        sq = Comb(self.ctuple.partA,self.ctuple.partB)
        self.checkadd(sq,self.csquares)

    def D(self):
        self.etuple += self.dtuple
        self.ctuple -= self.dtuple
        sq = Comb(self.dtuple.partA,self.dtuple.partB)
        self.checkadd(sq,self.dsquares)

    def E(self):
        self.dtuple += self.etuple
        sq = Comb(self.etuple.partA,self.etuple.partB)
        self.checkadd(sq,self.esquares)

    def F(self):
        self.etuple += self.btuple
        self.etuple += self.dtuple
        sq = Comb(self.btuple.partA,self.btuple.partB)
        sq += self.dtuple
        self.checkadd(sq,self.fsquares)

    def checkadd(self,newsq,sqlist):
        roundcount = 0
        for sq in self.bsquares:
            if self.val(sq) == self.val(newsq):
                print "Error: This square has side-length equal to",
                    "the square added by move B in round", roundcount
                print sq, "=", newsq
                return()
            roundcount +=1
        roundcount = 0
        for sq in self.csquares:
            if self.val(sq) == self.val(newsq):
```

```
                print "Error: This square has side-length equal to",
                    "the square added by move C in round", roundcount
                print sq, "=", newsq
                return()
            roundcount +=1
        roundcount = 0
        for sq in self.dsquares:
            if self.val(sq) == self.val(newsq):
                print "Error: This square has side-length equal to",
                    "the square added by move D in round", roundcount
                print sq, "=", newsq
                return()
            roundcount +=1
        roundcount = 0
        for sq in self.esquares:
            if self.val(sq) == self.val(newsq):
                print "Error: This square has side-length equal to",
                    "the square added by move E in round", roundcount
                print sq, "=", newsq
                return()
            roundcount +=1
        roundcount = 0
        for sq in self.fsquares:
            if self.val(sq) == self.val(newsq):
                print "Error: This square has side-length equal to",
                    "the square added by move F in round", roundcount
                print sq, "=", newsq
                return()
            roundcount +=1
        sqlist.append(newsq)
        self.squares.append(self.val(newsq))
        self.totalarea += pow(self.val(newsq),2)
        self.logarea.append(self.totalarea)

    def val(self, comb):
        return (self.sqAlpha*comb.partA)+(self.sqBeta*comb.partB)

    def __str__(self):
        return "<"+str(self.val(self.btuple))+","+str(self.val(self.
            ctuple))+","+str(self.val(self.dtuple))+","+str(self.val(
            self.etuple))+">"

# This class is built to describe a particular square added to the
#   ell. The square is stored as partA and partB, where partA is the
#   coefficient of alpha and part B is the coefficient of beta,
```

```
#   since every square added is represented as a linear combination
#   of alpha and beta. There's no need to store alpha and beta in
#   these objects, though, because alpha and beta are stored in the
#   Ell class, and they don't change from square to square within an
#   ell.
class Comb:
    def __init__(self,initA,initB):
        self.partA = initA
        self.partB = initB

    def __iadd__(self,otherComb):
        self.partA += otherComb.partA
        self.partB += otherComb.partB
        return self

    def __isub__(self,otherComb):
        self.partA -= otherComb.partA
        self.partB -= otherComb.partB
        return self

    def __str__(self):
        return "alpha*"+str(self.partA)+" + "+"beta*"+str(self.partB)

def main():
    inAlpha = 5
    inBeta = 3
    firstEll = Ell(inAlpha,inBeta)
    for i in range(0,45):
        firstEll.F()
        firstEll.E()
        firstEll.C()
        firstEll.B()

    for sq in firstEll.fsquares:
        print firstEll.val(sq),
    print ""
    for sq in firstEll.esquares:
        print firstEll.val(sq),
    print ""
    for sq in firstEll.csquares:
        print firstEll.val(sq),
    print ""
    for sq in firstEll.bsquares:
        print firstEll.val(sq),
    print ""
```

```
    print "Number of Squares: ", len(firstEll.squares)
    numsq  = []
    for i in range(len(firstEll.squares)):
        numsq.append(i+1)
    logratio = []
    for i in range(len(firstEll.logarea)):
        logratio.append(firstEll.logarea[i]/numsq[i])

    # passing data to the output file
    out = open("algout.csv",'w')
    for j in logratio:
        out.write(str(j)+'\n')
    out.close()

main()
```

A.2 Python program for the Naïve Tiling

```
from __future__ import division
import math

# This class keeps track of all of the squares within an ell and
#   contains the methods to add new squares.
class Ell:

    def __init__(self,x,y,presquares):
        self.sqx = x
        self.sqy = y

        self.squares = presquares
        self.totalarea = 0
        self.logarea = []
        for i in range(len(self.squares)):
            self.totalarea += pow(self.squares[i],2)
            self.logarea.append(self.totalarea)

    def right(self):
        self.squares.append(self.sqy)
        self.totalarea += pow(self.sqy,2)
        self.logarea.append(self.totalarea)
        self.sqx += self.sqy

    def top(self):
        self.squares.append(self.sqx)
```

```
        self.totalarea += pow(self.sqx,2)
        self.logarea.append(self.totalarea)
        self.sqy += self.sqx

    def bottom(self):
        self.squares.append(self.sqx)
        self.totalarea += pow(self.sqx,2)
        self.logarea.append(self.totalarea)
        self.sqy += self.sqx

def main():
    presquares = [1,3,4,5,9,14,16,18,20,38,39,55,56]
    firstEll = Ell(94,111,presquares)
    for j in range(0,42):
        firstEll.right()
        firstEll.top()
        firstEll.right()
        firstEll.bottom()
    firstEll.right()

    print (firstEll.squares)
    print (len(firstEll.squares))

    numsq  = []
    for i in range(len(firstEll.squares)):
        numsq.append(i+1)
    print numsq
    logratio = []
    for i in range(len(firstEll.logarea)):
        logratio.append(firstEll.logarea[i]/numsq[i])

    # passing data to the output file
    out = open("BCout.csv",'w')
    for j in logratio:
        out.write(str(j)+'\n')
    out.close()

main()
```

A.3 Python program Comparing the Naïve and $\mathcal{FECB}$ Tilings

```
import matplotlib.pyplot as plt
import math
# Imports the data from the other two programs and graphs it using
#   the Matplotlib package.
```

```python
def main():
    inf = open('BCout.csv','r')
    BCratios = []
    for line in inf:
        BCratios.append(float(line))
    print BCratios
    inf.close()

    inf = open('algout.csv','r')
    algratios = []
    for line in inf:
        algratios.append(float(line))
    print algratios
    inf.close()

    numsq  = []
    for i in range(len(algratios)):
        numsq.append(i+1)

    fig = plt.figure()
    ax = fig.add_subplot(111)
    ax.set_yscale('log')
    ax.set_xlim(-5,190)
    ax.scatter(numsq,BCratios,label = 'Naive Algorithm')
    ax.scatter(numsq,algratios,c='r',label =
        '$\mathcal{F}\mathcal{E}\mathcal{C}\mathcal{B}$ Algorithm')
    ax.set_xlabel("$r$")
    ax.set_ylabel("$C(S[r])$")
    ax.legend(loc='upper left')
    plt.show()
main()
```

References

[1] Henle, F. V. and Henle, J. M (2008). *Squaring the Plane*. The American Mathematical Monthly, 115(1), 3-12.

[2] Pommersheim, J., Marks, T., and Flapan E. (2010). *Number Theory: A Lively Introduction with Proofs, Applications, and Stories* New York: Wiley, John & Sons.