THE EUCLIDEAN ALGORITHM AND A GENERALIZATION OF THE FIBONACCI SEQUENCE

IAN COOPER

Abstract. This paper will explore the relationship between the Fibonacci numbers and the Euclidean Algorithm in addition to generating a generalization of the Fibonacci Numbers. It will also look at the ratio of adjacent Fibonacci numbers and adjacent generalized Fibonacci numbers. Finally it will explore some fun applications and properties of the Fibonacci numbers.

1. Introduction

This paper will explore the relationship between Fibonacci numbers and the Euclidean algorithm. If one applies the Euclidean algorithm to an adjacent pair of Fibonacci numbers, the algorithm will march through each preceding Fibonacci number before reaching its end. Once we have explored the implications of this relationship, we will expand it to a *Fibonacci-triple* sequences by the application of the Euclidean Algorithm to sets of three integers. The parallel between Fibonacci and Fibonacci-triple sequences extends to properties such the ratios between successive terms, and to applications, such as tiling. We will formally define a Fibonacci-triple sequence at a later time.

First, we make sure everyone is on the same page when it comes to Fibonacci numbers. The *Fibonacci numbers* are the sequence 0, 1, 1, 2, 3, 5, 8,... etc. In other words, they are generated by the following recursion formula:

$$f_n = f_{n-1} + f_{n-2} \text{ where } f_0 = 0, f_1 = 1$$

Recall that the *division algorithm* asserts that given any integers a, b with $a > 0$, there is a unique pair of integers r and q such that $0 \leq r < a$ and $b = a(q) + r$. We call q the *quotient* and r the *remainder* of b upon division by a.

Date: 1/30/2010.

Next, if a and b are integers, not both of which are 0, then we denote their greatest common divisor by $GCD(a,b)$. Note that if a is positive and $b = q(a) + r$, then d is a common divisor of a and b iff it is a common divisor of a and r. In other words, $GCD(a,b) = GCD(r,a)$. If we iterate this process, we have the *Euclidean algorithm.* In other words,

$$GCD(a,b) = GCD(r_1, a) \text{ where } b = q_1(a) + r_1$$
$$GCD(r_1, a) = GCD(r_2, r_1) \text{ where } a = q_2(r_1) + r_2$$

And so on...

One continues the process until the r term equals 0. The last non-zero remainder is our GCD. Here is a quick example which computes $GCD(12, 8)$:

$$12 = 1(8) + 4$$

$$8 = 2(4) + 0$$

So $GCD(12, 8) = 4$.

Having defined the Euclidean algorithm and Fibonacci numbers, we develop some tools to explore the relationship between the two.

2. Conventions

In this paper, we will have several conventions that will pop up repeatedly. This section is here to tell what these conventions mean.

When we are talking about pairs, $\mathbb{P}$ will be the set of all pairs. Any pair named "X" or "Y" will be some arbitrary pair. Pairs of adjacent Fibonacci numbers will be called F_n where the smaller of the two is the nth Fibonacci number. Finally, individual Fibonacci numbers will be referred to as f_n.

For triples, $\mathbb{T}$ will be the set of all triples. Again, any triple named "X" or "Y" will be some arbitrary triple. Triples of adjacent Fibonacci-like numbers will be called G_n, and nth Fibonacci-like number will be called g_n.

3. Pairs

This section will focus on how long the Euclidean algorithm takes to come to a conclusion; i.e., the number of iterations required to complete the Euclidean algorithm. This will lead us to trying to generate the slowest pairs of certain magnitudes and will eventually lead us to considering the Fibonacci numbers. What it means to be the "slowest pairs of a certain magnitude" will become clear shortly.

To begin, let $\mathbb{N} = 0, 1, 2, 3...$ be the natural numbers.

Definition 3.1. *We define $\mathbb{P}$ to be the collection of all ordered pairs $X = (a, b)$, where $a \leq b$, $b \neq 0$. If $X = (a_1, b_1)$ and $Y = (a_2, b_2)$ are in $\mathbb{P}$, then $X \leq Y$ means $a_1 \leq a_2$ and $b_1 \leq b_2$.*

Proposition 3.2. *The binary relation $\leq$ is a partial ordering of $\mathbb{P}$, but not a total ordering.*

Proof. We must show that for all X, Y and Z in $\mathbb{P}$:

(1) $X \leq X$.

(2) If $X \leq Y$ and $Y \leq Z$, then $X \leq Z$.

(3) If $X \leq Y$ and $Y \leq X$, then $X = Y$.

Let $X = (a_1, b_1)$, $Y = (a_2, b_2)$, and $Z = (a_3, b_3)$. Note that the first condition is clear. As for the second,

$$X \leq Y \Rightarrow a_1 \leq a_2 \text{ and } b_1 \leq b_2$$
$$Y \leq Z \Rightarrow a_2 \leq a_3 \text{ and } b_2 \leq b_3$$

So, $a_1 \leq a_2 \leq a_3$ and $b_1 \leq b_2 \leq b_3$. This can be written as $a_1 \leq a_3$ and $b_1 \leq b_3$ which, by definition, means $X \leq Z$.

The third condition is proved in a similar manner.

$$X \leq Y \Rightarrow a_1 \leq a_2 \text{ and } b_1 \leq b_2$$
$$Y \leq X \Rightarrow a_2 \leq a_1 \text{ and } b_2 \leq b_1$$

So, $a_1 \leq a_2 \leq a_1$ and $b_1 \leq b_2 \leq b_1$. This obviously implies $a_1 = a_2$ and $b_1 = b_2$. Thus, $X = Y$.

Finally, we show that $\leq$ is not a total ordering of $\mathbb{P}$. To do this, we must show that there are $X, Y \in \mathbb{P}$ such that neither $X \leq Y$ nor $Y \leq X$ is true.

So, if $X = (2, 4)$ and $Y = (1, 5)$, then $X \not\leq Y$ because $2 > 1$, and $Y \not\leq X$ because $5 > 4$. □

Now that we have defined $\mathbb{P}$ and $\leq$, it is time for us to take a look at the "order" of pairs. To begin, we define a couple of functions. Let

$$\mathbb{P}^* = \{(a, b) \in \mathbb{P} : a \neq 0\}.$$

Definition 3.3. *If $S = (a, b) \in \mathbb{P}^*$, let $E(S) = (r, a) \in \mathbb{P}$, where $b = q(a) + r$ in the division algorithm. And if $S = (a, b) \in \mathbb{P}$, let $D(S) = (b, a + b) \in \mathbb{P}^*$.*

Proposition 3.4. *If $S = (a, b) \in \mathbb{P}$ and $a < b$, then $E(D(S)) = S$.*

Proof. Since $D(S) = (b, a + b)$, $b + a = 1(b) + a$ and $a < b$, it follows that $E((b, a + b)) = (a, b)$, giving the result. □

On the other hand, note that $E(D((1, 1))) = E((1, 2)) = (0, 1) \neq (1, 1)$. We do have the following properties that are closely related to the above:

Proposition 3.5. *The function $D : \mathbb{P} \to \mathbb{P}^*$ is injective and order preserving.*

Proof. Suppose $S = (a, b)$ and $S' = (a', b')$. If $D(S) = D(S')$, then $b = b'$ and $a+b = a'+b'$, which implies that $a = a'$, and hence $S = S'$. Therefore, D is injective. If $S \leq S'$, then $a \leq a'$ and $b \leq b'$, so that $D(S) = (b, a + b) \leq (b', a' + b') = D(S')$. □

The following is central to our investigation:

Definition 3.6. *If $S = (a, b) \in \mathbb{P}$, we say S has order 0 if $a = 0$, and we write $O(S) = 0$. Assume next that $S = (a, b)$, where $a > 0$, and that we have defined the order for all elements of $(a', b') \in S$ for which $a' < a$. In this case, we then let*

$O(S) = O((E(S))) + 1.$

In other words, $O(S) = 0$ iff $S \notin \mathbb{P}^*$, and in general, $O(S) = k$ iff k is the smallest integer such that $E^k(S) \notin \mathbb{P}^*$. We mention in passing the following:

Proposition 3.7. *If $S = (a, b) \in \mathbb{P}$, then $O(S) \leq a$.*

Proof. Suppose $E^k(S) = (a_k, b_k)$, so that $a_0 = a$, $b_0 = b$. Since a_k is the remainder upon dividing a_{k-1} into b_{k-1}, we have $a = a_0 > a_1 > a_2 > \cdots$. This process terminates at the first integer $k = O(S)$ such that $a_k = 0$, showing that $a \geq k$, as required. □

Proposition 3.8. *For any $S = (a, b) \in \mathbb{P}$, $O(S) = 1$ iff $a|b$.*

Proof. Note S has strictly positive order iff $a \neq 0$. In this case, let $b = q(a) + r$ be as in the division algorithm. Therefore, $O(S) = 1$ iff $E(S) = (r, b)$ has order 0 iff $r = 0$ iff $a|b$. □

The next definition is also central to our discussion.

Definition 3.9. *Let $F_n = (f_n, f_{n+1}) \in S$ where f_n is the n^{th} Fibonacci Number.*

Our overall aim is to show that these F_n's are the smallest pairs of their order. We first take a look at the orders of the first few Fibonacci pairs or the first five F_n's.

(1) $O(F_0) = 0$ by definition because $F_0 = (0, 1)$.

(2) $O(F_1) = O(F_2) = 1$

Since $F_1 = (1, 1)$, $F_2 = (1, 2)$ and 1 divides any integer, this follows from Proposition 3.8.

(3) $O(F_3) = 2$

$F_3 = (2, 3)$ and $3 = 1(2) + 1$ so $E(F_3) = (1, 2) = F_2$ which is of order 1. We know $O(F_3) = O(E(F_3)) + 1$ so $O(F_3) = 2$.

In fact, using Definition 3.9, we claim that if S has order 0, then $F_0 \leq S$: We know $F_0 = (f_0, f_1) = (0, 1)$ and that for S to have order 0, its first entry, a, must be zero. Since $b > a$, we have $b \geq 1$. So $S \geq F_0$.

We next show that if $S = (a, b)$ has order 1, then $F_1 \leq S$: We know $F_1 = (1, 1)$. Since S does not have order 0, we must have $a \geq 1$. And since $b \geq a$, we must have $b \geq 1$. Therefore, $S \geq (1, 1) = F_1$.

Before continuing these computations, we make the following observation:

Proposition 3.10. *If $n \geq 0$, the following hold:*

(1) $D(F_n) = F_{n+1}$;

(2) If $n \neq 1$, then $E(F_{n+1}) = F_n$.

Proof. As to (1), we have $D(F_n) = D((f_n, f_{n+1})) = (f_{n+1}, f_n + f_{n+1}) = F_{n+1}$. Next (2) follows from (1) and Proposition 3.4, since $f_n = f_{n+1}$ iff $n = 1$. □

Now it is time to prove the following, which is half of the primary aim of this section.

Theorem 3.11. *If $n \geq 2$, then $O(F_{n+1}) = n$.*

Proof. By the above computations, this works for $n = 2$. Next, we assume $n \geq 3$ and the result holds for F_n. Now, let us determine the order of F_{n+1}. We have $E(F_{n+1}) = F_n$. We just assumed $O(F_n) = n - 1$, so $O(F_{n+1}) = O(E(F_n)) + 1 = O(F_n) + 1 = (n - 1) + 1 = n - 1$. Thus $O(F_{n+1}) = n$. □

Finally, we can prove the theorem that was our chief aim in all of this.

Theorem 3.12. *For $n \geq 2$, if S has order n, then $F_{n+1} \leq S$.*

Proof. Suppose $S = (a, b)$ is of order 2. If $a = 0$, then $O(S) = 0$, and if $a = 1$, then by Proposition 3.4, $O(S) = 1$. Therefore, we can conclude $a \geq 2$. If $a = b$, then again by Proposition 3.4, $O(S) = 1$, so that $2 \leq a < b$. However, this gives $(a, b) \geq (2, 3) = F_3$, as required.

Now, we assume this works up to $n-1$ and $S=(a,b)$ has order n. If $a=b$, we could again conclude that $O(S)=1$, which does not hold. Note that $O(E(S))=n-1$, so that by induction we have $E(S)\geq F_n$. Using Propositions 3.4 and 3.5, we have that $S=D(E(S))\geq D(F_n)=F_{n+1}$. Thus, any S of order n satisfies $S\geq F_{n+1}$. □

We note in passing that the converse of Theorem 3.12 does not hold. For example, $F_4=(3,5)$ has order 3 and $(3,5)\leq(3,6)$, but since 3 divides 6, we know that $(3,6)$ has order 1.

In summary, we have just shown that if S has order n, then $S\geq F_{n+1}$. In other words, adjacent pairs of Fibonacci numbers are the slowest pairs to which one can apply the Euclidean Algorithm. Again, this is because the Euclidean Algorithm simply marches back down through the Fibonacci numbers on its quest for their GCD.

4. Estimating the order of F_n

In this section, we prove the following result:

Theorem 4.1. *If $X=(a,b)\in\mathbb{P}$, then $O(X)\leq 5*log_{10}(a)+1$.*

Proof. Recall that F_n denotes the pair of Fibonacci numbers f_n, f_{n+1} and that the order of any pair of numbers is the number of times one can apply the Euclidean Algorithm to it before it terminates. Also recall that if $n\geq 2$, then $O(F_{n+1})=n$, and in fact, F_{n+1} is the smallest element of $\mathbb{P}$ of order n.

First, notice that for any pair $X=(a,b)$ of order n, $X\geq F_{n+1}$ implies $a\geq f_{n+1}$. We would now like to show that $f_{n+1}>\Phi^{n-1}$ for $n\geq 2$, where $\Phi=(1+\sqrt{5})/2$ or the Golden Ratio. We will do this by induction. For our base case, look at $n=2$ and $n=3$.

$$\Phi<2=f_3\ \checkmark$$
$$\Phi^2=(3+\sqrt{5})/2<3=f_4\ \checkmark$$

Now let us assert that $\Phi^{k-1} < f_{k+1}$ for all integers k such that $3 \leq k < n$. We know that Φ is the solution to $x^2 - x - 1 = 0$ so we can say $\Phi^2 = \Phi + 1$. This allows us to do the following,

$$\Phi^{n-1} = \Phi^2 * \Phi^{n-3} = (\Phi + 1) * \Phi^{n-3} = \Phi^{n-2} + \Phi^{n-3}$$

From our previous induction assumption, we know $\Phi^{n-2} < f_n$ and $\Phi^{n-3} < f_{n-1}$. If we add these two inequalities, we get,

$$\Phi^{n-1} = \Phi^{n-2} + \Phi^{n-3} < f_n + f_{n-1} = f_{n+1}.$$

So, our assumption also holds for $k = n$. Thus, $f_{n+1} > \Phi^{n-1}$ for $n \geq 2$.

Now, if we look back, we notice that this means that $a \geq f_{n+1} > \Phi^{n-1}$ for $n \geq 2$. We also know that $log_{10}\Phi > 1/5$. So,

$$log_{10}a > (n-1)log_{10}\Phi > (n-1)/5.$$

Therefore,

$$n - 1 < 5 * log_{10}a.$$

as required. □

Corollary 4.2. *Suppose $X = (a, b) \in \mathbb{P}$ and there are k digits in the decimal representation of a. Then $O(X)$ is no greather than $5k$.*

Proof. We have $a < 10^k$, so that $log_{10}a < k$. We then have

$$O(X) < 5k + 1$$

Since $O(X)$ is an integer, we can conclude that $O(X) \leq 5k$, as required. □

We have just shown that the order of any pair is no more than 5 times the number of digits in the smaller of the two numbers in a pair.

An interesting question that arises from this is what the the ratio of the estimate to the actual order of a pair approaches as the order approaches infinity. We will simplify this question to simply looking at the estimate of the order of a pair F_n against its actual order, $n-1$. It turns out that

$$\lim_{n\to\infty} \frac{5 * log_{10}(F_n)}{n-1} = 5 * log_{10}(\Phi).$$

To show this, we know that $F_n = \frac{\Phi^n - (-1/\Phi)^n}{\sqrt{5}}$ which can be rewritten $\frac{\Phi^n(1-(-1/\Phi)^{2n})}{\sqrt{5}}$. If we plug this into the above and expand we get something like this...

$$\frac{5 * [log(\Phi^n) + log(1-(1/\Phi)^{2n}) - log(\sqrt{5})]}{n-1}$$

Taking the limit of the above as $n \to \infty$ causes $log(1-(1/\Phi)^{2n}$ to go to zero. This leaves us with:

$$lim_{n\to\infty} \frac{5n * log(\Phi)}{n-1} - \frac{log_{10}\sqrt{5}}{n-1}$$

By l'Hospital's rule, the first term goes to $5 * log_{10}(\Phi)$ and the second term goes to zero as n goes to infinity. This leaves us with the desired result of $5log(\Phi)$.

5. Triplets

This section will be very similar to the previous except we will focus on the Euclidean Algorithm applied to triples. The aim here is to identify the slowest triples, develop some way to generate them, and observe similarities between the resulting sequence and the Fibonacci numbers.

First, we clarify what it means to apply the Euclidean algorithm to a triple. Say we have three integers, (a, b, c), where $a \leq b \leq c$. We run two separate instances of the division algorithm using the smallest of the three integers as our divisor. In other words,

$$c = q_c(a) + r_c$$
$$b = q_b(a) + r_b$$

Now, a is the largest of the triple $(\min(r_b, r_c), \max(r_b, r_c), a)$. One then repeats the process. Here is a numerical example:

Given $(8, 10, 12)$,

$$10 = 1(8) + 2$$

$$12 = 1(8) + 4$$

The new triple is $(2, 4, 8)$. One more iteration.

$$8 = 4(2) + 0$$

$$4 = 2(2) + 0$$

So $\mathrm{GCD}(8, 10, 12) = 2$.

We formally begin our investigation of triples with a few Definitions. Again let $\mathbb{N} = \{0, 1, 2, 3, \dots\}$ be the natural numbers.

Definition 5.1. *Let $\mathbb{T}$ be the collection of all ordered triples $S = (a, b, c)$, where $a, b, c \in \mathbb{N}$, $a \leq b \leq c$ and $c \neq 0$. If $X = (a_1, b_1, c_1)$ and $Y = (a_2, b_2, c_2)$ are in $\mathbb{T}$, we again let $X \leq Y$ mean $a_1 \leq a_2$, $b_1 \leq b_2$ and $c_1 \leq c_2$.*

Almost identically to pairs, $\mathbb{T}$ is partially ordered, but not totally ordered, by $\leq$. Define $\mathbb{T}^* = \{(a, b, c) \in \mathbb{T} : a = 0\}$. Next, if $S = (a, b, c) \in \mathbb{T}^*$, let

$$E(S) = (\min(r_b, r_c), \max(r_b, r_c), a)$$

and if $S = (a, b, c) \in \mathbb{T}$, let

$$D(S) = (c, a + c, b + c) \in \mathbb{T}^*.$$

We note that the analogues of Propositions 3.4, 3.5 and 3.7 carry over in the obvious manner.

Definition 5.2. *If $S = (a, b, c) \in \mathbb{T}$, we say S has order 0 if $a = 0$, and we write $O(S) = 0$. Assume next that $S = (a, b, c)$, where $a > 0$, and that we have defined the order for all elements of $(a', b', c') \in \mathbb{T}$ for which $a' < a$. We then let $O(S) = O((E(S)) + 1$.*

In other words, the order of a triple is how many iterations of this form of the Euclidean Algorithm it takes to reduce from the case of triples of numbers to the case of pairs of numbers (one term terminates with a zero). Again, as in the case of ordered pairs, it is clear that if $S = (a, b, c)$, then $O(S) \leq a$, and that $O(S) = 1$ iff $a|b$ or $a|c$.

The first part of our investigation will be much like what we did for pairs. We begin by finding the smallest triples of order 0, 1, 2 and 3.

(1) Let us begin with $G_0 = (0, 0, 1)$. We would like to show that this is the smallest element of $\mathbb{T}$ of order 0.

To do this, first we must show that it is of order zero. Obviously, it is because its a slot is a zero. Next, we have to show the other two positions are as small as possible. This is the case because the b slot is a zero and the c slot is not allowed to be zero. Thus $U_0 = (0, 0, 1)$ is the smallest element of $\mathbb{T}$ of order 0.

(2) Next, we show that $G_1 = (1, 1, 1)$ is the smallest element of $\mathbb{T}$ of order 1.

Let $S = (a, b, c)$ have order 1. First, we know that to have a non-zero order, $a > 0$; so 1 is the next smallest integer to fill this slot. Since $a \leq b \leq c$, the smallest b and c can possibly is 1. Now we just have to show that G_1 is of order 1. Since $E(G_1) = (0, 0, 1)$, which is of order zero and $O(S) = O((E(S)) + 1$, we have $O(G_1) = 1$.

(3) We now want to show $G_2 = (2, 3, 3)$ is the smallest element of $\mathbb{T}$ of order 2.

Again, suppose $S = (a, b, c)$ has order 2. First, let us examine the a slot. We have $2 = O(S) \leq a$. Next, we know $a \leq b \leq c$, so the smallest b and c can be is 2, as well. But if a divided either b or c, then S would again be of order 1. Therefore, the smallest possible value for b and c is 3. Now, we make sure $(2, 3, 3)$ is of order 2. But this follows from $E((2, 3, 3)) = (1, 1, 2)$ and $E((1, 1, 2)) = (0, 0, 1)$. Thus, G_2 is of order 2 because the process took two steps.

(4) Now, we would like to show that $G_3 = (5, 7, 8)$ is the smallest element of $\mathbb{T}$ of order 3. This is a somewhat scary argument.

First, let $S = (a, b, c)$ have order 3. Let $b = q_b(a) + r_a$ and $c = q_c(a) + r_c$. The following must therefore hold:

(1) $r_b \neq 0, 1$ and $r_c \neq 0, 1$: If either of these terms were 0, then $O(E(S)) = 0$ so that $O(E) = 1$; and if either were equal to 1 then $O(E(S)) \leq 1$ so that $O(E(S)) \leq 2$.

(2) $r_b \neq r_c$: If this were the case, then $O(E(S)) \leq 1$, so that $O(S) \leq 2$.

Now, if $a = 2$, then we must have $r_b, r_c \leq 1$, which violates (1). Therefore, $a \geq 3$.

Next, if $a = 3$, then by (1), we would have $r_b = r_c = 2$, which would violate (2). Therefore, $a \geq 4$.

If $a = 4$, then by (1), neither r_b nor r_c can be 0 or 1. And by (2), we cannot have $r_b = r_c$. It follows that $E(S) = (2, 3, 4)$. However, this would imply that $E^2(S) = (0, 1, 2)$, so that $O(S) = 2 < 3$.

It follows that $a \geq 5$. Since we must have $r_b, r_c \geq 2$, and $b < c$, we can conclude that $b = q_b(a) + r_b \geq 5 + 2 = 7$ and $c \geq b + 1 \geq 8$. This shows that $S \geq G_3$.

Finally, we verify that $O(G_3) = 3$:

$$E(G_3) = (2, 3, 5)$$

$$E((2, 3, 5)) = (1, 2, 2)$$

$$E((1, 2, 2)) = (0, 0, 1).$$

So, $(5, 7, 8)$ is of order 3, and in fact, it is the smallest element of $\mathbb{T}$ of order 3.

Now that we have built the first few elements of $\mathbb{T}$ that are the smallest of their order, it is time for another definition.

Definition 5.3. *If $k \geq 4$ and we have defined $G_{k-1} = (a_{k-1}, b_{k-1}, c_{k-1})$, let $G_k = D(G_{k-1}) = (a_k, b_k, c_k)$ where*

$$a_k = c_{k-1}$$
$$b_k = c_{k-1} + a_{k-1}$$
$$c_k = c_{k-1} + b_{k-1}$$

that is, G_k is the kth row of the following table:

	0	0	1
	1	1	2
	2	3	5
G_3	5	7	8
G_4	8	13	15
G_5	15	23	28

Theorem 5.4. *For all $n \geq 0$, G_n is the least element of $\mathbb{T}$ of order n.*

Proof. All of the above work has gone to verify this for $n = 0, 1, 2, 3$, so assume $n \geq 4$ and the result has been verified for $n-1$; we prove that it also holds for n.

Since $E(G_n) = E(D(G_{n-1})) = G_{n-1}$ has order $n-1$, we can conclude that $O(G_n) = (n-1) + 1 = n$.

Suppose next that $S = (a, b, c)$ is any other element of $\mathbb{T}$ of order n. We need to show $S \geq G_n$. Let $E(S) = (r, s, a)$ (so that $r \leq s < a$) and $Y = D(E(S)) = (a, a+r, a+s)$.

We claim that $Y \leq S$: To verify this we must show $a+r \leq b$ and $a+s \leq c$ (since both triples have a in their first coordinate). From $Y = E(S)$, we know that either

$$(1)\ b = q_1a + r \text{ and } c = q_2a + s;\text{ or } (2)\ b = q_1a + s \text{ and } c = q_2a + r,$$

where $q_1, q_2 \geq 1$.

First let us look at case 1. Observe that $b \geq q_1a+r \geq a+r$ and $a \geq q_2a+s \geq a+s$, as required.

Now we look at case 2. Since $s \geq r$, we have $b = q_1a + s \geq a + r$. And since $c \geq b$, we have $c \geq q_1a + s \geq a + s$.

Since $O(Y) = O(E(S)) = n-1$, we know $G_{n-1} \leq Y$. It follows that $G_n = D(G_{n-1}) \leq D(Y) = D(E(S)) = S$, and our induction is complete. $\square$

We call a_n the n^{th} *Fibonacci-triple number.* This terminology is due to the following recursion result:

Proposition 5.5. *If $G_k = (a_k, b_k, c_k)$ is as above, then for $k \geq 4$ we have*

$$a_k = a_{k-1} + a_{k-2} + a_{k-3}.$$

Proof. We know

$$\begin{aligned} a_k &= c_{k-1} \\ b_k &= c_{k-1} + a_{k-1} \\ c_k &= c_{k-1} + b_{k-1} \end{aligned}$$

So, $a_k = c_{k-2} + b_{k-2} = a_{k-1} + c_{k-3} + a_{k-3} = a_{k-1} + a_{k-2} + a_{k-3}$. $\square$

In other words, if the G's are arranged in columns as in 5.3(in previous section), then each entry in G_k where $k > 3$ are the sum of the previous three in its column. This is very similar to the way Fibonacci numbers work. If we arrange the first few elements F_n of $\mathbb{P}$ in a similar manner we get,

$$\begin{aligned} F_1 = &\quad 1 \quad 1 \\ F_2 = &\quad 1 \quad 2 \\ F_3 = &\quad 2 \quad 3 \\ F_4 = &\quad 3 \quad 5 \\ F_5 = &\quad 5 \quad 8 \\ F_6 = &\quad 8 \quad 13 \end{aligned}$$

Notice that every number after the second row is the sum of the previous 2 in their column. This is why we refer to these a_n as *Fibonacci-triple numbers.* Again, we have shown that for every $n \in \mathbb{N}$, G_n is the smallest element of $\mathbb{T}$ of order n.

Slightly more generally, if g_1, g_2 and g_3 is a non-decreasing set of positive integers, we can iteratively define Fibonacci-triple numbers using the recursion relation

$$g_k = g_{k-1} + g_{k-2} + g_{k-3}.$$

In fact, it is not really the initial few values that are of most interest, but the way in which the sequence behaves in the long term.

6. Convergence of f_n/f_{n-1}

Begin by defining $\rho_n = f_{n+1}/f_n$. Since the definition of a Fibonacci number is $f_n = f_{n-1} + f_{n-2}$, we can divide through by f_{n-1} and get $\rho_n = 1 + 1/\rho_{n-1}$. This gives us the following definition.

Definition 6.1. $f(x) = 1 + 1/x$

Therefore, $\rho_n = f(\rho_{n-1}) = ... = f^{n-1}(1)$. Our objective is the following well known result:

Theorem 6.2. *We have*

$$\lim_{n\to\infty} \rho_n = \Phi,$$

where $\Phi = (1 + \sqrt{5})/2$ *is the Golden Ratio.*

Proof. We examine the first few ρ's.

$$\rho_1 = 1$$
$$\rho_2 = 2$$
$$\rho_3 = 3/2$$
$$\rho_4 = 5/3$$

Notice here that $\rho_1 < \rho_3 < \rho_4 < \rho_2$. In fact, we can generalize this to $\rho_{2k+1} < \rho_{2k+3} < \rho_{2k+2} < \rho_{2k}$ for some integer k. [This follows by induction: If it holds for k, then applying f to this leads to $\rho_{2k+1} < \rho_{2k+3} < \rho_{2k+4} < \rho_{2k+2}$, and applying f a second time leads to $\rho_{2k+3} < \rho_{2k+5} < \rho_{2k+4} < \rho_{2k+2}$, which is the $k+$ 1st case of the statement.] Thus, since the sequence ρ_{2k+1} is bounded and decreasing, it must converge to some value. Let us call this value Φ. The same can be said for $\lim_{k\to\infty}(\rho_{2k})$. Let us call the value that this approaches Φ'.

Notice that to move one spot up the ρ_{2k+1} or ρ_{2k} lines, one must apply $f(x)$ twice, where x is the current ρ. In other words, $\rho_{2k+2} = f(f(\rho_{2k}))$. Since f is continuous, we must have $f(f(\Phi)) = \Phi$ and $f(f(\Phi')) = \Phi'$. So, if $x = f(f(x))$ has a unique solution, $\Phi = \Phi'$. This begs the question: what does $f(f(x))$ look like? Plugging $1 + 1/x$ in for x yields,

$$1 + 1/(1 + 1/x).$$

Multiply the last term by x/x for,

$$1 + x/(x + 1).$$

We can simplify this further like so,

$$1 + (x + 1 - 1)/(x + 1) = 2 - 1/(x + 1).$$

Thus,

$$f(f(x)) = 2 - 1/(x + 1).$$

Now, let's solve $x = 2 - 1/(x + 1)$ and see if we get the expected value, the Golden Ratio ($(1 + \sqrt{5})/2$).

$$\begin{aligned} x &= 2 - 1/(x + 1) \\ x^2 + x &= 2x + 1 \\ x^2 - x - 1 &= 0 \end{aligned}$$

This is a very simple problem. If we just plug into the quadratic formula, we do indeed get back $x = (1 + \sqrt{5})/2$. We can discard the negative solution because all of these numbers are positive.

So, we have shown that the ratio ρ_n of adjacent Fibonacci numbers approaches some number Φ as $n \to \infty$. In addition, we have shown that $\Phi = (1 + \sqrt{5})/2$. □

7. Convergence of g_{n+1}/g_n

In this section we would like to show that the ratio between adjacent Fibonacci-triple numbers, g_n and g_{n+1}, approaches some number Ψ as $n \to \infty$.

Throughout this section we will let

$$g(x) = x + 1 + 1/x.$$

To begin with we would like to show that $g(x)$ is increasing on $(1, \infty)$: We take the derivative and get $g'(x) = 1 - \frac{1}{x^2}$. It is fairly easy to see that $g'(x) \geq 0$ for all values on the interval $(1, \infty)$, since every larger value for x decreases the value of the x term being subtracted.

We next let

$$f(x) = 1 + 1/x + 1/x^2.$$

We now observe that $f(x)$ is decreasing on the interval $(1, \infty)$: In fact, any increase in the value of x makes the fraction portions smaller thus decreasing the value of $f(x)$.

This brings us to the following important step in evaluating the ratios of Fibonacci-triple numbers:

Lemma 7.1. *There is a unique $x > 1$ with the property that $f^2(x) = f(f(x)) = x$, which is also the unique solution to the equation*

$$x^3 - x^2 - x - 1 = 0.$$

Proof. Suppose that $x_0 > 1$ is some number such that $f^2(x_0) = x_0$. We would like to show that if $x_1 = f(x_0)$ then $g(x_0) = x_0 x_1 = g(x_1)$. We have

$$g(x_0) = x_0 + 1 + 1/x_0 = x_0(1 + 1/x_0 + 1/x_0^2) = x_0 f(x_0) = x_0 x_1.$$

We also have $x_0 = f(f(x_0)) = f(x_1)$, so that

$$g(x_1) = x_1 + 1 + 1/x_1 = (1 + 1/x_1 + 1/x_1^2)x_1 = f(x_1)x_1 = x_0 x_1.$$

We would now like to verify that there is a unique solution $x > 1$ to $x = f(x)$. We know $f(1) = 3 > 1$ and $f(2) = 7/4 < 2$. Since $f(x) - x$ is strictly decreasing on $[1, \infty)$, there is a unique solution to $f(x) = x$ in this interval.

Next, we need to show that $f^2(x) = x$ also has the same solution. Suppose we have a solution x_0 for $f(f(x_0)) = x_0$. If we let $x_1 = f(x_0)$, we know from the above that $g(x_1) = g(x_0)$. Since g is strictly increasing, we can conclude $f(x_0) = x_1 = x_0$. This establishes our statement.

Finally, note that $f(x) = x$ iff

$$f(x) = 1 + 1/x + 1/x^2 = x$$

iff

$$x^2 + x + 1 = x^3$$

iff

$$0 = x^3 - x^2 - x - 1.$$

□

Let $\Psi > 1$ be the unique solution to the above equation.

Next, let $a_0 = 1$ and $a_n = f(a_{n-1})$ or $a_n = f^n(1)$. Let us quickly pound out the first few of these and look at their relationships to one another. $a_0 = 1$, $a_1 = 3$, $a_2 \approx 1.444444$, and $a_3 \approx 2.1716$. Thus $a_0 < a_2 < a_3 < a_1$.

Lemma 7.2. *For all $k \in \mathbb{N}$ we have $a_0 < a_2 < ... < a_{2k} < a_{2k-1} < ... < a_3 < a_1$.*

Proof. We prove by induction on k that

$$a_{2k} < a_{2k+2} < a_{2k+3} < a_{2k+1}.$$

We have just verified this for $k = 0$. If we assume it is true for k, then since f is strictly decreasing, applying f to this gives

$$a_{2k+2} < a_{2k+4} < a_{2k+3} < a_{2k+1}.$$

If we apply f once again, we obtain

$$a_{2k+2} < a_{2k+4} < a_{2k+5} < a_{2k+3},$$

which is the $k+1$st case of our hypothesis. □

Lemma 7.3. *We have*

$$\lim_{n\to\infty} a_n = \Psi.$$

Proof. By Lemma 7.2, a_{2k} is a bounded increasing sequence, so it converges to some value α. Since $f(f(a_{2k})) = a_{2k+2}$ and f is continuous, we can conclude that $f(f(\alpha)) = \alpha$. However, by Lemma 7.1, this means that $\alpha = \Psi$. Similarly, the subsequence a_{2k+1} must also converge to Ψ. However, this is just what we want to prove. □

Definition 7.4. *With the above notation, let* $I_1 = [a_0, a_1], I_2 = [a_2, a_1]$ *and so on. In other words, for* $n = 1, 2, 3, ...,$ $I_n = [a_n, a_{n+1}]$ *if* n *is odd and* $I_n = [a_{n+1}, a_n]$ *if* n *is even.*

Notice that for any n, $I_n = f^n(1, 3)$ and $I_{n+1} \subseteq I_n$.

Lemma 7.5. *For any* $\rho, \rho' \in I_n$*, we have* $1 + 1/\rho + 1/\rho\rho' \in I_{n+1}$.

Proof. Two cases arise here: $\rho \leq \rho'$ and $\rho' \leq \rho$. Let $I_n = [b, c]$, thus $I_{n+1} = [f(c), f(b)]$. Since $\rho, \rho' \in I_n$, $b \leq \rho, \rho' \leq c$. In the case of $\rho \leq \rho'$, we have

$$f(c) \leq f(\rho') \leq 1 + 1/\rho' + 1/(\rho')^2 \leq 1 + 1/\rho + 1/\rho\rho' \leq 1 + 1/\rho + 1/\rho^2 \leq f(\rho) \leq f(b)$$

This shows that $1 + 1/\rho + 1/\rho\rho' \in I_{n+1}$.

However, in the case of $\rho' \leq \rho$, we have

$$f(c) \leq f(\rho) \leq 1 + 1/\rho + 1/\rho^2 \leq 1 + 1/\rho + 1/\rho\rho' \leq 1 + 1/\rho' + 1/(\rho')^2 \leq f(\rho') \leq f(b),$$

which again leads to what we want. □

Definition 7.6. *Let* $\rho_n = \frac{g_{n+1}}{g_n}$ *where* $g_1, g_2, g_3, ...$ *etc. are the Fibonacci-triple numbers* $1, 1, 1, 3, 5, 9...$

Lemma 7.7. *For any $k \geq 2$, we have*

$$\rho_{k+1} = 1 + \frac{1}{\rho_k} + \frac{1}{\rho_k \rho_{k-1}}.$$

Proof. If we take the equation

$$g_{k+2} = g_{k+1} + g_k + g_{k-1}$$

and divide by g_{k+1}, it simplifies to the above. $\square$

This brings us to the main result of this section.

Theorem 7.8. *We have*

$$\lim_{n \to \infty} \rho_n = \Psi.$$

Proof. We show by induction that for all positive integers n, that $\rho_{2n-1}, \rho_{2n} \in I_n$. Since $\rho_1 = \rho_2 = 1$, this clearly holds for $n = 1$. Assume $\rho_{2n-1}, \rho_{2n} \in I_n$. By Lemma 7.7 (with $k = 2n$) and Lemma 7.5 (with $\rho = \rho_{2n-1}$ and $\rho' = \rho_{2n}$), we can conclude that $\rho_{2n+1} \in I_{n+1} \subseteq I_n$. Again by Lemma 7.7 (with $k = 2n+1$) and Lemma 7.5 (with $\rho = \rho_{2n}$ and $\rho' = \rho_{2n+1}$), we can conclude that $\rho_{2n+2} \in I_{n+1}$. This completes our induction.

Therefore, by the squeezing theorem, our result follows from Lemma 7.3. $\square$

The number Ψ which is the limit of the ratios of the Fibonacci-triple numbers, is the unique solution $x > 1$ to the polynomial

$$x^3 - x^2 - x - 1 = 0.$$

Using Maple, we have an approximate solution

$$\Psi = 1.839286755...,$$

and an exact solution of

$$\Psi = \frac{(19 + 3\sqrt{33})^{1/3}}{3} + \frac{4}{3(19 + 3\sqrt{33})^{1/3}} + 1/3.$$

Notice that this is larger than the golden ratio ($(1+\sqrt{5})/2 \approx 1.618$) which is the ratio that adjacent Fibonacci numbers approach.

8. Fun application

Fibonacci numbers crop up in naturally occurring problems all the time. One such application is tiling. First we look at tiling with two types of tiles. Imagine you have square tiles (let us call these one unit) and rectangular tiles that are exactly the size of two square tiles side by side or are two units. Now imagine you have a section of floor one width of a square tile wide and some number of square tile widths long or n units. It would be interesting to see how many ways you can tile the floor, would it not?

We start with a section of floor that is non-existent. There is one way to tile this section: you just don't tile it. Now look at a section of floor that is exactly the one unit in length. There is one way to tile this. You can put a single square tile on it. Finally look at a section that is two units long. There are two ways to tile this: one rectangular tile or two square tiles. Generally, if one has a section of floor n units long, there are f_n ways to tile it, where f_n is the nth Fibonacci number.

This same idea can be used for tiling a section n units long with three kinds of tile, squares, rectangles of length two, and rectangles of length three. The only difference here is that instead of the normal Fibonacci numbers, we are going to see the Fibonacci-triple numbers generated in section 2. Here is the result.

Theorem 8.1. *The number of ways to tile a rectangle that is 1 by n units using tiles that are either 1 by 1, 1 by 2 or 1 by 3, is the nth element of the sequence*

$$1, 2, 4, 7, 13, 24, \ldots.$$

Proof. Let t_n be the nth element of this sequence, so that for $n \geq 3$ we have $t_{n+1} = t_n + t_{n-1} + t_{n-2}$. We proceed by induction, and it is easy to verify the result for $n = 1, 2, 3$. So assume $n \geq 4$ and we have verifies the result for $n - 1 \geq 3$. If we have n squares and only look at the last one, there are 3 ways in which to tile that last square. One could use a single square, a double, or a triple. For the single case, there are $n-1$ remaining squares so there are t_{n-1} ways of tiling them. For the double case there are $n-2$ remaining squares, thus t_{n-2} ways of tiling them. Finally for the triple case, we are left with $n-2$ squares to tile and t_{n-2} ways to do so. Thus the number of ways to tile n squares is $t_{n-1}+t_{n-2}+t_{n-3}=t+n$. $\square$

9. Conclusion

This concludes our investigation of the Fibonacci numbers and the Euclidean Algorithm. We have shown that the adjacent Fibonacci numbers are the smallest pairs of their order and that they are the slowest pairs when the Euclidean Algorithm is applied to them. In addition we showed that the smallest triplets of their order are in fact, very similar to the Fibonacci numbers. We also inspected the ratios of adjacent Fibonacci numbers and Fibonacci Triples and found that both approach some ratio and the Fibonacci numbers approach the Golden Ratio while the triples approach some value slightly larger than the Golden Ratio. Finally, we played around with the fun application of tiling and its expansion to the triples.

10. References

(1) http://www.math.ualberta.ca/ isaac/math324/s05/lame.pdf
(2) http://www.math.hmc.edu/funfacts/ffiles/20004.5.shtml
(3) http://mathworld.wolfram.com/BinetsFibonacciNumberFormula.html