Generalizations of Conway's Subprime Fibonacci Sequences

Abrar Almahmeed

Abstract

In this experiment, we explore Conway's subprime Fibonacci sequences with different initial values $a_0, a_1 \in \mathbb{Z}$, as well as different parameters $c_1, c_2 \in \mathbb{Z}$. We aim to detect patterns and understand when sequences grow or stabilize.

First, we implemented a procedure to generate these sequences using different parameters and initial values. From our results, we observed different types of behavior, including cycles, increasing sequences, and sometimes chaotic behavior without a clear pattern. This gives some insight into how small modifications can change the behavior of recursive sequences.

1 Introduction

Conway's subprime Fibonacci sequences are variations of classical recursive sequences that combine ideas from recurrence relations and Collatz-type transformations.

The classical Fibonacci sequence is defined by the recurrence relation $F_n = F_{n-1} + F_{n-2}$, starting from given initial values. Despite its simple definition, it produces rich and well-understood behavior.

The Collatz problem (also known as the $3n + 1$ problem [2]). Starting from a positive integer, if the number is even it is divided by 2, and if it is odd it is replaced by $3n + 1$. Repeating this process generates a sequence that is conjectured to eventually reach 1.

Conway's subprime Fibonacci sequences, introduced in [1], are defined as follows

$$a_n = \frac{c_1 a_{n-1} + c_2 a_{n-2}}{B(c_1 a_{n-1} + c_2 a_{n-2})}$$

where $c_1, c_2 \in \mathbb{Z}$ are the parameters of the recurrence, $a_0, a_1 \in \mathbb{Z}$ are the initial values, and B define as

$$B(x) = \begin{cases} 1 & \text{if } x \text{ prime or } |x| \leq 1 \\ \text{lpf}(x) & \text{otherwise} \end{cases}$$

where lpf(x) is the least positive prime factor of $x \in \mathbb{Z}$. Note that lpf(x)=1, for $|x| \leq 1$.

We explore how these sequences behave under different initial values and parameters, focusing on patterns such as cycles, increasing sequences, and irregular or chaotic behavior through computational experiments.

We implement these sequences in Maple and perform computational experiments to generate large amounts of data. By analyzing this data, we aim to identify patterns, classify different types of behavior, and formulate observations about the long-term dynamics of these sequences. Our approach is experimental; rather than proving general theorems, we focus on discovering patterns through computation and supporting them with examples.

2 Maple Implementation

In this section, we describe the procedures implemented in Maple to generate and analyze subprime Fibonacci sequences. These procedures allow us to experiment with different initial values and parameters and to study the behavior of the resulting sequences.

- `SubpFseq(a0,a1,N,c1,c2)` generates the subprime Fibonacci sequence corresponding to the initial values a_0, a_1 and parameters c_1, c_2, up to N terms.

  Input: initial values $a_0, a_1 \in \mathbb{Z}$, parameters $c_1, c_2 \in \mathbb{Z}$, and number of terms N.

  Output: the generated sequence as a list.

- `SubpFCycles(M,N,c1,c2)` computes the cycle information for all initial values in the range $-M \leq a_0, a_1 \leq M$.

  Input: bound M, number of terms N, and parameters c_1, c_2.

  Output: a table containing cycle information for each pair of initial values.

- `FindCycle(S)` detects cycles in a given sequence S.

  Input: a sequence S.

  Output: a pair $[L, p]$, where L is the cycle length and p is the starting position of the cycle, or `[FAIL,FAIL]` if no cycle is detected.

- `CycleLengthSeq(M,N,c1,c2)` computes the set of all possible cycle lengths for initial values in the range $-M \leq a_0, a_1 \leq M$.

  Input: bound M, number of terms N, and parameters c_1, c_2.

  Output: a set of all observed cycle lengths.

3 Experimental Results and Observations

We performed computational experiments on generalized subprime Fibonacci sequences for various initial values and parameters. All observations reported in this section are based on computational experiments within a finite range.

In these experiments, cycle detection depends on the number of computed terms N. In particular, for some initial values, a cycle may not be detected within the tested range.

Case 1: $c_1 = c_2 = 1$

We begin with the original subprime Fibonacci sequence. Using the procedures `SubpFCycles` and `CycleLengthSeq`, we computed for initial values in the range $-50 \leq a_0, a_1 \leq 50$. The results show that the possible cycle lengths are

`CycleLengthSeq(50,1000,1,1)`

$$\{1, 18, 19, 56, 136\}.$$

We also observed that when $a_0 = a_1$, the sequence often converges to a cycle of length 1. Similarly, for initial values of the form $(a_0, a_1) = (a, -a)$ the sequence produces a cycle of length 1 in most cases. Exceptions occur for $a = \pm 1, \pm 8, \pm 16, \pm 24, \pm 32, \pm 48$, where the corresponding sequences instead form cycles of length 18. In addition, for initial values with $(0, a_1)$, the sequence generally has a cycle of length 1, except when $a_1 = \pm 1$ and multiples of 4. No cycle of length 1 detected when $a_0 = \pm 1$.

Case 2: $c_1 = c_2 = c$, with $c = 2, 3, 4$

For these parameters, the sequences do not exhibit periodic behavior within the tested range. For example,

$$\texttt{CycleLengthSeq(20,100,2,2)} = \{1, \mathrm{FAIL}\},$$

where $-20 \leq a_0, a_1 \leq 20$, and similar results are observed for $c = 3, 4$. The only initial value that produces a cycle of length 1 in this range is $(0, 0)$.

We also observe that the behavior depends on the signs of the initial values. For instance,

$$\texttt{SubpFseq(1,1,10,2,2)} = 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89$$

is an increasing sequence, while

$$\texttt{SubpFseq(-1,-1,10,2,2)} = -1, -1, -2, -3, -5, -8, -13, -21, -34, -55, -89$$

is a decreasing sequence.

More generally, for initial values in the tested range, when both a_0 and a_1 are positive, the sequence tends to be increasing, while when both are negative, it tends to be decreasing.

When the signs of a_0 and a_1 differ, no consistent pattern was observed: in some cases the sequence appears to increase, while in others it decreases, depending on the chosen initial values.

Case 3: $c_1 = c_2 = c < 0$, with c up to -17

For negative parameters, we tested values up to $c = -17$. When $c = -1, -2, -3$, we observed different behavior using the procedures `FindCycle` and `CycleLenTable`. The possible cycle lengths are

$$\{1, 3\}.$$

In this case, all the tested initial values excluding $(0, 0)$ have a cycle of length 3.

For $c = p$, where $p = -5, -7, -11, -13, -17$ we observed the possible cycle lengths

$$\{1, 3, \text{FAIL}\}$$

In contrast, when $c = q$ where $q = -4, -6, -8, -9, -10, -12, -14, -15, -16$, no cycle is detected within the tested range (denoted by FAIL).

The initial value $(a_0, a_1) = (0, 0)$ is the only initial value in the tested range that leads to a cycle of length 1 (a constant cycle).

Case 4: $c_1 = -c_2$, For $c_1 = 1, 2, \ldots, 13$

For parameters with opposite signs, we tested values for $c_1 = 1, 2, 3$. The possible cycle lengths in the tested parameters using `FindCycle` are

$$\{1, 6\}$$

all tested initial values within the considered range, excluding $(0, 0)$, produce a cycle of length 6. Using the procedures `S:=SubpFCycles(50,1000,1,-1)` and `T:=CycleLenTable(S)`, we get `T[1] = {[0, 0]}`, tested also for $c_1 = 2, 3$.

Using the same procedures when $c_1 = 5, 7, 11, 13$ is prime, we observed the possible cycle lengths are

$$\{1, 6, \text{FAIL}\}$$

In contrast, when $c_1 = 4, 6, 8, 9, 10, 12$ is a composite, no cycle is detected within the tested range (denoted by FAIL), except for $(0, 0)$.

Within the tested range, $(a_0, a_1) = (0, 0)$ is the only initial value that produces a cycle of length 1 (a constant cycle).

Case 5: $c_1 = -c_2$, with $c_1 = -1$

All the results reported in this case are based on computational experiments for initial values in the range $-50 \leq a_0, a_1 \leq 50$. We tested the case $c_1 = -1$ and $c_2 = 1$ for larger values, computations become more time consuming, and cycles may not be detected within the tested bounds.

The possible cycle lengths are

```
SubpFCycles(20,1000,-1,1)
```

$$\{1, 2, 18, 56, 136\}$$

in the tested range, the only initial value $(0, 0)$ producing a constant cycle. For initial values of the form $(a_0, a_1) = (a, a)$, we observed that most sequences fall into cycles of length 2. In a smaller number of cases,a cycle of length 18 appears. Specifically, cycles of length 18 were observed with the following initial values $\{(-48, -48),\ (-32, -32), (-24, -24),\ (-16, -16),\ (-8, -8),\ (-1, -1),\ (1, 1),\ (8, 8),$ $(16, 16),\ (24, 24),\ (32, 32),\ (48, 48)\}$.

In addition, for initial values of the form $(a_0, a_1) = (a, -a)$, we observed that the sequence enters a cycle of length 2 for all tested values of a.

Furthermore, we tested initial values of the form $(a_0, a_1) = (ka, a)$ for $k = 2, 3, 4$ lead to cycles of length 2, with a few cases producing cycles of length 18.

4 Generalizations

Based on our computational experiments, we propose the following conjectures.

- **Proposition 0.** The initial value $(a_0, a_1) = (0, 0)$ always has a constant cycle for any values of $c_1, c_2 \in \mathbb{Z}$ (see [1]).

  Proof. Consider $a_n = \dfrac{c_1 a_{n-1} + c_2 a_{n-2}}{B(c_1 a_{n-1} + c_2 a_{n-2})}$ for $c_1, c_2 \in \mathbb{Z}$, since $a_0 = a_1 = 0$, we compute

  $$a_2 = \frac{c_1 a_1 + c_2 a_0}{B(c_1 a_1 + c_2 a_0)} = \frac{0}{B(0)} = \frac{0}{1} = 0$$

  Now assume for some $n \geq 2$ we have $a_{n-1} = a_n = 0$, then

  $$a_{n+1} = \frac{c_1 a_n + c_2 a_{n-1}}{B(c_1 a_n + c_2 a_{n-1})} = \frac{0}{B(0)} = 0.$$

  By induction, $a_n = 0$ for all n. Hence, the sequence is constant and forms a cycle of length 1.

- **Proposition 1.** When $c_1 = c_2 = 1$, if $a_0 = a_1$, the sequence converges to a constant cycle (cycle of length 1).

  Proof. For $c_1 = c_2 = 1$ and initial values $a_0 = a_1 = a$, the we have

  $$a_2 = \frac{a_1 + a_0}{B(a_1 + a_0)} = \frac{2a}{B(2a)}$$

  $B(2a) = 2$, since the least prime factor of $2a$ is 2, then we get $a_2 = a$.

  Now assume for some $n \geq 2$, we have $a_{n-1} = a_n = a$, then

  $$a_{n+1} = \frac{a_n + a_{n-1}}{B(a_n + a_{n-1})} = \frac{2a}{B(2a)} = a.$$

  By Induction, $a_n = a$ for all n. Hence, the sequence converges to a constant cycle.

- **Conjecture 2.** When $c_1 = c_2 > 1$, sequences are increasing when $a_0, a_1 > 0$, and decreasing when $a_0, a_1 < 0$ and do not form cycles within the tested range.

- **Conjecture 3.** When $c_1 = c_2 = c < 0$ and c is prime or 1, cycles of length 3 appear in the corresponding sequence.

- **Conjecture 4.** When $c_1 = -c_2$, if c_1 is prime or 1, cycles of length 6 appear.

- **Conjecture 5.** When $c_1 = -1, c_2 = 1$, initial values of the form $(a_0, a_1) = (ka, a)$, with $k = 1, 2, 3, 4$ leads to cycles of lengths 2 and 18.

All conjectures are based on computational experiments within a finite range of initial values.

5 Conclusion

In this project, we implemented and explored generalized subprime Fibonacci sequences using computational methods in Maple. By testing various initial values and parameters, we investigated the behavior of these sequences and identified several patterns related to cycle formation and growth.

Our experiments show that the behavior of the sequences depends strongly on the choice of parameters and initial values $c_1, c_2, a_0, a_1 \in \mathbb{Z}$. In particular, certain structures in the initial values, such as (a, a) or $(a, 2a)$, tend to produce cycles or constant behavior, while other choices may lead to increasing or apparently chaotic sequences.

However, it is important to note that all results are based on calculations within a limited range of initial values and parameters. Therefore, these observations do not constitute formal proofs, but rather suggest patterns and conjectures, some of which can be formally proven.

Further work could include extending the computational range, improving efficiency for larger values, and investigating whether these observed patterns can be proven.

Acknowledgments

We would like to express our sincere thanks to Dr. Doron Zeilberger for his guidance, valuable feedback, and continuous support throughout this project.

We also sincerely thank Pablo Blanco for his collaboration, helpful discussions, and insightful comments.

References

[1] V. Ponomarenko, S. Rhinehart, L. Schwarz, and J. Tong, *On Subprime Recurrences*, The American Mathematical Monthly, 133(3), 275–280, 2026.

`https://doi.org/10.1080/00029890.2025.2594388`

[2] R. K. Guy, T. Khovanova, and J. Salazar, *Conway's subprime Fibonacci sequences*, arXiv:1207.5099, 2014.

`https://arxiv.org/abs/1207.5099`