FIBONACCI'S ANSWER TO PRIMALITY TESTING?

JULIAN BEAUCHAMP

Abstract. In this paper, we consider various approaches to primality testing and then ask whether an effective deterministic test for prime numbers can be found in the Fibonacci numbers.

Introduction

Prime numbers play an important role in encryption and cyber-security. They protect digitial data, including personal data and bank details. Testing whether a number is prime or not, is therefore becoming increasingly important for mathematicians and for those working in digitial security.

The study of prime numbers and their properties go back to the ancient Greek mathematician Pythagoras (570-495 BC) who understood the idea of primality. But a primality test is a test to determine whether or not a number is prime. This is different from finding a number's constituent prime factors (also known as prime factorization). A number is said to be prime if it is divisible only by 1 and itself. Otherwise it is composite. When the numbers are small, it is relatively easy to determine whether a number is prime. But as they get exponentially larger they get harder to determine.

So the pressing question is what makes an efficient algorithm? The following characteristics make an efficient algorithm - *general*, *deterministic*, *unconditional*, and *polynomial*:[1]

General. An algorithm that is *general* works for all numbers. Algorithms that are not general only work on certain numbers (e.g. the Lucas-Lehmer test for Mersenne numbers).

Deterministic. A *deterministic* test (e.g. the Lucas-Lehmer Test) will tell us with absolute certainty whether a number is prime or not every time it is run. The most basic form of deterministic test was discovered by Greek mathematician Eratosthenes (276-195 BC), who devised an algorithm now called the 'Sieve of Eratosthenes'. However, such tests usually involve complex and time-consuming alogorithms. By contrast, *probabilistic* tests (e.g. the Miller-Rabin test), tend to be much faster but only give us probable results. The reason for this is that certain rogue composite numbers falsely pass the test. These composites, called pseudoprimes, thus mask their true composite nature, and make the test unreliable. For this reason, probabilistic tests are often adapted to make them more accurate, but

Date: Sept 2019.
2010 *Mathematics Subject Classification.* Primary 11B39, 11A41.
Key words and phrases. Fibonnaci, Primes.

[1]adapted from https://www.whitman.edu/Documents/Academics/Mathematics/2018/Worthington.pdf

the changes themselves end up slowing the test down.
Unconditional. An *unconditional* algorithm is one whose correctness does not depend on any unproven hypotheses. For example, there are conditional primality tests that are correct only if the Extended Riemann Hypothesis is true.
Polynomial Time. A polynomial time algorithm is one with computational complexity that is a polynomial function of the input size, with a polynomial function of $\log_2 n$. Polynomial time is preferable to exponential time. In this paper, we do not discuss this aspect of testing.

The only test to possess all four characteristics is the AKS primality test, as we shall see.

Early Primality Testing
But first, let us briefly consider some of the familiar tests, and then we will see if any can be improved on. In early mathematics, it was thought that Mersenne Primes of the form $2^n - 1$ were prime when n is also prime. This certainly holds true for the first few values of n:

$$2^2 - 1 = 3$$
$$2^3 - 1 = 7$$
$$2^5 - 1 = 31$$
$$2^7 - 1 = 127.$$

However, it does not hold for all primes n. For example,

$$2^{11} - 1 = 2047 = 23.89.$$

In 1640, Fermat showed that it also does not hold for $n = 23$ and $n = 37$:

$$2^{23} - 1 = 8388607 = 47.178481$$
$$2^{37} - 1 = 137438953471 = 223.616318177.$$

Thus, this test is probabilistic.

Fermat's Little Theorem
But in the same year Fermat proved that for the number $a^n - a$, if n is prime, then for any co-prime integer a, the number $a^n - a$ is divisible by n. This is known as Fermat's Little Theorem. So for the first few, where $a = 2$, we get:

$$2^2 - 2 = 2.1$$
$$2^3 - 2 = 3.2$$
$$2^5 - 2 = 5.6$$
$$2^7 - 2 = 7.18$$
$$2^{11} - 2 = 11.186$$
$$2^{13} - 2 = 13.630$$
$$2^{17} - 2 = 17.7710$$

So for all prime exponents to infinity, $a^n - a$ is divisible by the prime exponent that produced them. And it is true for any value of a when coprime with n. It is thus the basis for the Fermat primality test and is one of the fundamental results of elementary number theory. However, the converse is not true. This test also produces pseudoprimes. For example, the number $2^{341} - 2$ is divisible by 341 (=11.31). Thus 341 is the smallest base-2 Fermat pseudoprime, i.e. $a = 2$. And yet, in other bases (e.g. a=5) it shows up to be composite.

Sadly, using different bases does not solve this problem. There are even more resilient pseudoprimes that resist being exposed which falsely pass the test for every base (while a and n are co-prime). For example, 561 (=3.11.17). Such numbers are called Carmichael numbers and there are infinitely many of them! The Miller-Rabin Test makes some improvements to Fermat's test, but even this test can be fooled. For example, the third Carmichael number 1729 is pseudoprime.

AKS primality test

More recently, M. Agrawal and colleagues made significant advances in primality testing. In August 2002, they announced a deterministic algorithm for determining if a number is prime that runs on polynomial time much faster than the exponential time of Fermat's test (Agrawal et al. 2004). This test is known as the Agrawal-Kayal-Saxena primality test, or AKS primality test. It states, very basically, that given an integer $n \geq 2$ and integer a is coprime with n, then n is prime if and only if the following polynomial congruence holds:

$$(x + a)^n \equiv (x^n + a) \pmod{n}.$$

It is similar to Fermat's Little Theorem, and similarly can be proved using the binomial theorem, but is still considered impractical.

Fibonacci

Here, we come to Fibonacci and primality testing. But how, one may ask, does Fibonacci fit in with all this? The Fibonacci sequence begins as follows:

$$0, 1, 1, 2, 3, 5, 8, 13, 21, 34...$$

In this sequence, starting with 0 and 1, each term is found by adding the previous two numbers. Formally, it is the sequence of numbers ${F_n}_{(n=1)}^{\infty}$ defined by the linear recurrence equation, where $F_0 = 0$ and $F_1 = F_2 = 1$:

$$F_n = F_{(n-1)} + F_{(n-2)}.$$

Existing Fibonacci primality tests

Fibonacci primality tests already exist, but usually test only for Fibonacci primes.[2] A Fibonacci prime is a Fibonacci number F_n that is also a prime number, e.g. 2,3,5,13,89.... It is also known that every Fibonacci prime must also have a prime index (i.e. n is prime), with the exception of $F_4 = 3$. However, the converse is not true. Not every prime index p gives a prime F_p (e.g. $F_{19} = 4181 = 37.113$). So this test is not general, and it is not deterministic.

[2] For further reading, John Brillhart, Note on Fibonacci Primality Testing, https://www.fq.math.ca/Scanned/36-3/brillhart.pdf, https://people.csail.mit.edu/vinodv/COURSES/MAT302-S13/pomerance.pdf and Carl Pomerance, Primality Testing: Variations on a theme of Lucas.

John Selfridge combines two tests conjecturing that if p is an odd number, and $p \equiv \pm \pmod 5$, then p will be prime if both of the following hold:[3]

$$2^p - 1 \equiv 1 \pmod p,$$

$$F_{p+1} \equiv 0 \pmod p,$$

where F_k is the k^{th} Fibonacci number. The first condition is the Fermat primality test using base 2.

However, we wish to go further and find an even simpler general and deterministic test.

A Promising Prime Pattern in the Fibonacci Sequence

Dr. R. Knott of Surrey University has highlighted one promising pattern in the primes, in a sequence of Fibonacci index numbers n where F_n can be divided by n–1 (also http://oeis.org/A100993):

$$2, 3, 4, 8, 14, 18, 24, 38, 44, 48, 54, 68, 74, 84, 98, 104, ...$$ [4]

So for example, $F_{14} = 377$, and 377 is divisible by $n - 1 = 13$. Now, if you subtract 1 from every element in the sequence, you get:

$$1, 2, 3, 7, 13, 17, 23, 37, 43, 47, 53, 67, 73, 83, 97, 103, ...$$

It looks like we have found a way to produce all the primes that end in 3 or 7!

Knott then gives a second list, a sequence of Fibonacci index numbers n where F_n can be divided by $n + 1$ (also http://oeis.org/A100992):

$$10, 18, 28, 30, 40, 58, 60, 70, 78, 88, 100, 108, 130, 138, ...$$

For example, $F_{30} = 832040$, and 832040 is divisible by $n + 1 = 31$. This time, we add 1 to each element to get:

$$11, 19, 29, 31, 41, 59, 61, 71, 79, 89, 101, 109, 131, 139, ...$$

Indeed, this time we find all the primes that end in 1 or 9! This is remarkable. If we combine the two algorithms we appear to have an unconditonal algorithm that produces all primes (apart from 5, the only prime that does not end in 1,3,7,9). Unfortunately, it is rarely that simple! Although all the numbers in the first list are one more than a prime (i.e. where $n - 1$ is prime), this is not true in general. Once again, a pseudoprime snags the system. The smallest such pseudoprime is F_{324}, which has a composite factor $323 = 17.19$. Nevertheless, the algorithm is so simple, it could still be faster than exponential time, and perhaps even than polynomial time (depending on the speed of generating Fibonacci numbers). It is general, but not deterministic.

A General and Deterministic Fibonacci Test?

But what if we are dividing by the wrong numbers? What if, instead of dividing F_n by $n \pm 1$ (i.e. where n is composite), we divided $F_{n\pm1}$ by n?

---

[3]https://en.wikipedia.org/wiki/Primality-testHeuristic-tests.

[4]http://www.maths.surrey.ac.uk/hosted-sites/R.Knott/Fibonacci/fibmaths.htmlsection2

The table below gives the results up to $n = 75$. In the first column, n, the prime values of n are highlighted in bold; the second column is the Fibonacci sequence; the third and fourth rows are the results for $\frac{F_n+1}{n}$ and $\frac{F_n-1}{n}$ respectively to 2 decimal places (integer results for $n = p$ are marked in bold, and for $n = 2p$ are marked with [ ]*); the last column shows whether 1 was added or subtracted. Note that the only 2 cases for which this does not work is $n = p = 5$, $n = 2p = 10$.

n	F_n	$\frac{F_n+1}{n}$	$\frac{F_n-1}{n}$	$\equiv \pm 1 \pmod p$
1	1	2.00 (trivial)	0.00	-
2	1	**1.00**	0.00	+1
3	2	**1.00**	0.33	+1
4	3	[1.00]*	0.50	+1
5	5	1.20	0.80	-
6	8	1.50	1.17	-
7	13	**2.00**	1.71	+1
8	21	2.75	2.50	-
9	34	3.89	3.67	-
10	55	5.60	5.40	-
11	89	8.18	**8.00**	-1
12	144	12.08	11.92	-
13	233	**18.00**	17.85	+1
14	377	[27.00]*	26.86	-
15	610	40.73	40.60	-
16	987	61.75	61.63	-
17	1597	**94.00**	93.88	+1
18	2584	143.61	143.50	-
19	4181	220.11	**220.00**	-1
20	6765	338.30	338.20	-
21	10946	521.29	521.19	-
22	17711	805.09	[805.00]*	-
23	28657	**1246.00**	1245.91	+1
24	46368	1932.04	1931.96	-
25	75025	3001.04	3000.96	-
26	121393	[4669.00]*	4668.92	-
27	196418	7274.78	7274.70	-
28	317811	11350.43	11350.36	-
29	514229	17732.07	**17732.00**	-1
30	832040	27734.70	27734.63	-
31	1346269	43428.06	**43428.00**	-1
32	2178309	68072.19	68072.13	-
33	3524578	106805.42	106805.36	-
34	5702887	[167732.00]*	167731.94	-
35	9227465	263641.89	263641.83	-
36	14930352	414732.03	414731.97	-
37	24157817	**652914.00**	652913.95	+1

Continued on next page

Continued from previous page

n	F_n	$\frac{F_n+1}{n}$	$\frac{F_n-1}{n}$	$\equiv \pm 1 \pmod{p}$
38	39088169	1028636.05	[1028636.00]*	-
39	63245986	1621691.97	1621691.92	-
40	102334155	2558353.90	2558353.85	-
41	165580141	4038540.05	**4038540.00**	-1
42	267914296	6378911.83	6378911.79	-
43	433494437	**10081266.00**	10081265.95	$+1$
44	701408733	15941107.59	15941107.55	-
45	1134903170	25220070.47	25220070.42	-
46	1836311903	[39919824.00]*	39919823.96	-
47	2971215073	**63217342.00**	63217341.96	$+1$
48	4807526976	100156812.02	100156811.98	-
49	7778742049	158749837.76	158749837.71	-
50	12586269025	251725380.52	251725380.48	-
51	20365011074	399313942.65	399313942.61	-
52	32951280099	633678463.46	633678463.42	-
53	53316291173	**1005967758.00**	1005967757.96	$+1$
54	86267571272	1597547616.17	1597547616.13	-
55	139583862445	2537888408.11	2537888408.07	-
56	225851433717	4033061316.39	4033061316.36	-
57	365435296162	6411145546.72	6411145546.68	-
58	591286729879	10194598791.03	[10194598791.00]*	-
59	956722026041	16215627560.03	**16215627560.00**	-1
60	1548008755920	25800145932.02	25800145931.98	-
61	2504730781961	41061160360.03	**41061160360.00**	-1
62	4052739537881	65366766740.03	[65366766740.00]*	-
63	6557470319842	104086830473.70	104086830473.67	-
64	10610209857723	165784529026.94	165784529026.91	-
65	17167680177565	264118156577.94	264118156577.91	-
66	27777890035288	420877121746.80	420877121746.77	-
67	44945570212853	**670829406162.00**	670829406161.97	$+1$
68	72723460248141	1069462650707.97	1069462650707.94	-
69	117669030460994	1705348267550.65	1705348267550.62	-
70	190392490709135	2719892724416.23	2719892724416.20	-
71	308061521170129	4338894664368.03	**4338894664368.00**	-1
72	498454011879264	6922972387212.01	6922972387211.99	-
73	806515533049393	**11048157986978.00**	11048157986977.97	$+1$
74	1304969544928657	[17634723580117.00]*	17634723580116.97	-
75	2111485077978050	28153134373040.68	28153134373040.65	-

This has been tested up to 5,000 primes by R. Knott using Mathematica. Thus we wish to conjecture that (except for $p = 5$) if p is prime, then p will always divide $F_p + 1$ (if F_p terminates in digits 3 or 7) or will divide $F_p - 1$ (if F_p terminates in digits 1 or 9). It also holds true for all $2p$, where $2p$ divides $F_{2p} \pm 1$ (under equivalent conditions). In other words, for all p, $F_p \equiv \pm 1 \pmod p$, $F_{2p} \equiv \pm 1 \pmod{2p}$.

It would significantly improve primality testing if we could state the converse, i.e. that if $F_n + 1$ is divisible by n (when F_n terminates in digits 3 or 7) then n is prime, and if $F_n - 1$ is divisible by n (when F_n terminates in digits 1 or 9) then n is prime.

Pseudoprimes

However, Knott has also found 11 pseudoprimes (below 10,000) which pass the test but are in fact composite. They are: 572, 646, 754, 3782, 4181, 5777, 6479, 6721, 7654, 7743, and 8362. This is half as many as the Fermat pseudoprimes under 10,000.

This led us to investigate the property of Fibonacci numbers further. We also found that when F_n ends in 3,7, if n is prime, then n also appears to divide $F_{(n+k)} + F_{(k-1)}$ exactly (for all integers $k > 0$). And when F_n ends in 1,9, n also divides $F_{(n+k)} - F_{(k+1)}$ exactly (for all integers $k > 0$).

This means we can run a very simple secondary test by adding 0,1,1,2,3,5,8... (or subtracting 1,2,3,5,8...) to each subsequent Fibonacci number respectively and seeing if it still divides by n. So let us take $F_7 = 13$:

$F_7 + 1 = 14$ (=7.2)
$F_8 + 0 = 21$ (=7.3)
$F_9 + 1 = 35$ (=7.5)
$F_{10} + 1 = 56$ (=7.8)
$F_{11} + 2 = 91$ (=7.13)
$F_{12} + 3 = 147$ (=7.21)
$F_{13} + 5 = 238$ (=7.34) and so on...

Not only does it divide exactly, you can even see (in this example) the Fibonacci sequence reappearing as a factor! Sometimes Lucas numbers (L_n), or multiples of Lucas numbers, appear as a factor. For example, take $F_{13} = 233$:

$F_{13} + 1 = 234$ (=13.18 = 13.L_6)
$F_{14} + 0 = 377$ (=13.29 = 13.L_7)
$F_{15} + 1 = 611$ (=13.47 = 13.L_8)
$F_{16} + 1 = 988$ (=13.76 = 13.L_9)
$F_{17} + 2 = 1599$ (=13.123 = 13.L_{10})
$F_{18} + 3 = 2587$ (=13.199 = 13.L_{11})
$F_{19} + 5 = 4186$ (=13.322 = 13.L_{12}) ... and so on.

Interestingly, if we apply the logic backwards below each prime number, the pattern continues backwards, although rather strangely ($\pm$) polarity switches:

$F_{12} - 1 = 143$ (=13.11 = 13.L_5)
$F_{11} + 2 = 91$ (=13.7 = 13.L_4)
$F_{10} - 3 = 52$ (=13.4 = 13.L_3)
$F_9 + 5 = 39$ (=13.3 = 13.L_2)
$F_8 - 8 = 13$ (=13.1 = 13.L_1)

$F_7 + 13 = 26$ ($=13.2 = 13.L_0$) ... and so on.

So, just as the Fermat Primality Test makes a second test to find pseudoprimes, so does this Fibonacci Test. But it raises the further question: are there any *strong pseudoprimes* using the Fibonacci test (like Fermat's Carmichael numbers)? Answer: It appears so. Using this secondary test, the first four pseudoprimes we found (572, 646, 754, 3782) all correctly show themselves to be composite. However, 4181, seems to present prime properties (at least up to $k = 9$) even though it is composite. 4181 is itself a Fibonacci number, and the first composite Fibonacci number that has a prime index (19). But 5777 also passes the test (up to $k = 2$) and 6479 (up to $k = 4$). We could test not further with the software available, so we simply speculate that odd pseudoprimes are strong pseudoprimes.

Conclusion

We have established that this test is general (not limited to certain numbers), probabilistic (not deterministic), and conditional (needing to be proved to hold for all prime numbers to infinity). At first glance, it seems to offer a potential alternative to the Fermat Test. But it raises several questions. First, are there any primes for which our new Fibonacci Test fails? Do the known pseudoprimes have any common properties that mean we can predict them, thereby making the test more efficient? And what we would really like to know is what are the deeper underlying properties of the Fibonacci sequence that allow us to test like this?

References

1. John Brillhart, Peter L. Montgomery, and Robert D. Silverman, Tables of Fibonacci and Lucas Factorizations, MATHEMATICS OF COMPUTATION, VOLUME 50, NUMBER 181, JANUARY 1988, PAGES 251-260
2. John Brillhart, NOTE ON FIBONACCI PRIMALITY TESTING. https://www.fq.math.ca/Scanned/36-3/brillhart.pdf
3. http://mathworld.wolfram.com/FibonacciNumber.html. Accessed 21.09.19.
4. http://www.maths.surrey.ac.uk/hosted-sites/R.Knott/Fibonacci/fibtable301.html
5. http://www.maths.surrey.ac.uk/hosted-sites/R.Knott/contactron.htmlsection1.1
6. https://en.wikipedia.org/wiki/Primality-testHeuristic-tests
7. Carl Pomerance, Primality Testing: Variations on a theme of Lucas, https://people.csail.mit.edu/vinodv/COURSES/MAT302-S13/pomerance.pdf

The Rectory, Village Road, Waverton, Chester Ch3 7QN, UK
Email address: julianbeauchamp47@gmail.com