Solution to the Collatz Conjecture

By Samuel Ferrer Colás

Abstract

The Collatz or 3x + 1 conjecture is perhaps the simplest stated yet unsolved problem in mathematics in the last 70 years. It was circulated orally by Lothar Collatz at the International Congress of Mathematicians in Cambridge, Mass, in 1950 (Lagarias, 2010).

The problem is known as the Thwaites conjecture (after Sir Bryan Thwaites), Hasse's algorithm (after Helmut Hasse), or the Syracuse problem.

In this concise paper I provide a proof of this conjecture, by finding an upper bound to the Collatz sequence and, as a consequence, a contradiction.

Introduction

The Collatz or 3x + 1 conjecture is perhaps the simplest stated yet unsolved problem in mathematics for the last 70 years. It was circulated orally by Lothar Collatz at the International Congress of Mathematicians in Cambridge, Mass, in 1950.

The problem is known as the Thwaites conjecture (after Sir Bryan Thwaites), Hasse's algorithm (after Helmut Hasse), or the Syracuse problem.

The 3x + 1 problem is concerned with the converge to 1 of the following sequence

$$f(n) = \begin{cases} n/2 & \text{if } n \equiv 0 \bmod 2 \\ 3n + 1 & \text{if } n \equiv 1 \bmod 2 \end{cases}$$

In this concise paper I provide a proof of this conjecture, by finding an upper bound to the Collatz sequence and, as a consequence, a contradiction.

Proof

Let's consider the following sequence, which converges to 1 as n goes to infinity. The proof is found in the appendix.

$$g(n) = \begin{cases} n/2 & \text{if } n \equiv 0 \bmod 2 \\ n+1 & \text{if } n \equiv 1 \bmod 2 \end{cases}$$

Now let's assume that $f(n) \rightarrow \infty$. This implies that

$f(n) < 3 * g(f(n))$

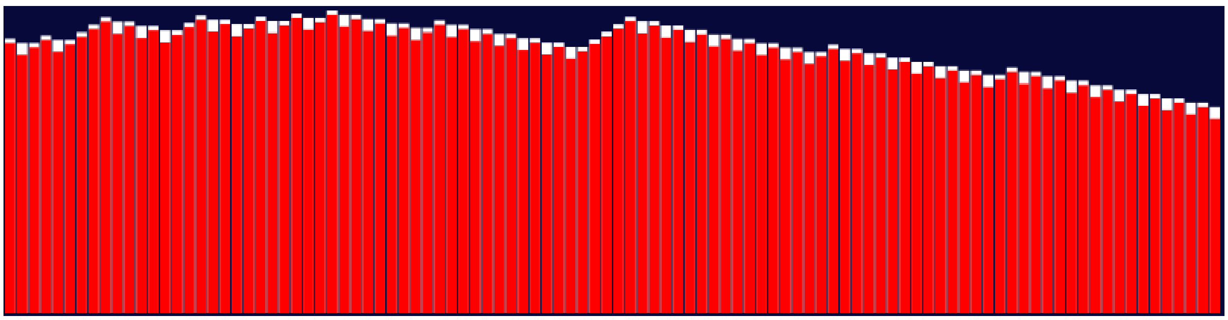
which contradicts the assumption that f(n) converges to infinity, since, regarding the inequality, it has $3 * g(f(n))$ as upper bound, which in turn converges to 3.

One could argue that $f(n)$, after some large index i, stays on a constant K larger than 1.
In this case we can substitute the inequality by *$f(n_i) < K < K + i < 3*g(K + i)$*. Eventually,
$3 * g(K + i)$ will also converge to 3 as the index i goes to infinity.

We conclude that $f(n)$ has 3 as upper bound, for which it takes a few steps to verify that it converges to 1.

Observe that the result can be generalized if 3 is replaced by any prime number greater than 2.
If we wantto generalize even further to the case of the prime number 2, then we just replace
$< by \leq$ in the inequality.

The following chart gives a hint of what is going on for 63728127 as the initial number



The red bars show $f(n)$ and the white bars show $3 * g(f(n))$, using a logarithmic scale. The code used to generate the chart is shown in the appendix.

Conclusion

The proof obtained here is the effort of two years of work. I first came across this problem during the pandemic, while attending a master in embedded systems at Uppsala University, Sweden in 2021.

I believe the same approach could be used to solve problems of similar nature.

Appendix

1. The following sequence of natural numbers converges to 1

$$g(n) = \begin{cases} n/2 & \text{if } n \equiv 0 \bmod 2 \\ n+1 & \text{if } n \equiv 1 \bmod 2 \end{cases}$$

Proof by induction in n.

Observe that this is true for k = 1 and k = 2. Let's assume we have the proof up to k = n-1, and prove it for k = n. If n is even, then we apply the division n/2 < n, which is the hypothesis. So, let's assume that n is odd. In this case there exist m such as n = 2m + 1. Applying the rule for odd numbers we get that the next term is 2m + 1 + 1 = 2(m + 1), which becomes m + 1 in the next iteration.
Now we have m + 1 < 2m + 1 = n. But this is the hypothesis, namely all the numbers strictly less than n, which proves the statement.

2. Javascript code used to generate the sample in the chart.

```
const startNumber = 63728127n

let nextColas = n => 3n*(n % 2n ?  n + 1n : n >> 1n)
let nextColaz = n => n % 2n ? 3n*n + 1n : n >> 1n

let bigmax = (a,b) => a >= b ? a : b

let biglog  = bigint => {
  if (bigint < 0) return NaN
  const s = bigint.toString(10)

  return s.length + Math.log10("0." + s.substring(0, 15))
}

let initialize = n => {
	let data = {
		colaz : [n],
		colas : [3n*n],
		max : -1n
	}
```

```
    for (let count=0; count < 101; count++ ){
        let clz = nextColaz(data.colaz[0])
        let cls = nextColas(clz)

        data.colaz.unshift(clz)
        data.colas.unshift(cls)
        data.max = bigmax(bigmax(data.max, clz), cls)
    }

    return data

}
```

Reference

Lagarias, 2010. Jeffrey C. Lagarias. 2010. The Ultimate Challenge: The 3x + 1 Problem.