e-mail: mr.leszek.mazurek@gmail.com

COLLATZ CONJECTURE - THE PROOF

LESZEK MAZUREK

ABSTRACT. In this paper, we prove the Collatz conjecture. The proof consists of two parts. The first, shows that if an integer can be iterated through the Collatz conjecture to one, it is the equivalent of the condition that it can be presented as a certain equation. In the second part, we prove that for every initial integer, this equation can be found. To achieve this, we propose a procedure that can be iterated, and we prove that by doing this we arrive at this equation. We also prove that initial integer can be presented in an infinite number of ways in the form of needed equation. All analysis is done using binary representation of numbers.

1. Introduction

The Collatz conjecture is a well known mathematical problem. It claims that for every positive integer I_0 if iterating

$$I_{n+1} = \begin{cases} \frac{1}{2} \cdot I_n & for,\ I_n\ \ even \\ \\ 3 \cdot I_n + 1 & for,\ I_n\ \ odd \end{cases} \tag{1.1}$$

ultimately we get 1.

The purpose of this paper is to prove that the Collatz conjecture is true. The proof consists of two parts:

Theorem 1.1. *If the Collatz conjecture is true for a positive integer I_0, it is the equivalent of the condition that a positive integer n and a sequence of integers $m_n > m_{n-1} > m_{n-2} > ... > m_1 > m_0 \geq 0$ exists, for which*

$$3^n I_0 \ = \ 2^{m_n} - 2^{m_{n-1}} 3^0 - 2^{m_{n-2}} 3^1 - \ \cdots \ - \ 2^{m_1} 3^{n-2} \ - \ 2^{m_0} 3^{n-1}. \tag{1.2}$$

Theorem 1.2. *For every positive integer I_0, such a positive integer n and a sequence of integers $m_n > m_{n-1} > m_{n-2} > ... > m_1 > m_0 \geq 0$ can be found. Therefore, (by Theorem 1.1) the Collatz conjecture is true.*

2. Remarks and Definitions

To understand how the Collatz conjecture works and make it more accessible, we have to iterate integers in their binary representations. This paper explains when binary numbers are even or odd, how they are affected by different operations

---

2020 *Mathematics Subject Classification.* Primary 11Y55, 11B83, 60G40.

and examines how they iterate through the Collatz formula. The definitions and remarks introduced below are used over the course of this paper.

Remark 2.1. An integer is odd when in binary representation its least significant bit is 1. An integer is even when in binary representation its least significant bit is 0.

Remark 2.2. Every even positive integer can be reduced to the odd positive integer by recursively dividing it by 2 until the result is odd.

When I_{even} is an even positive integer, I_{odd} is an odd positive integer and p is the number of divisions by 2 required for I_{even} to became the odd integer I_{odd}, then

$$\frac{I_{even}}{2^p} = I_{odd}. \tag{2.1}$$

Example 2.3. Reduction of an even integer to an odd integer in binary representation.

Let I_{even} be an even positive integer

$$I_{even} = 20 = 10100_b.$$

Then

$$\begin{aligned}\frac{I_{even}}{2^p} &= \frac{20}{2^2}\\ &= \frac{10\mathbf{100}_b}{\mathbf{100}_b}\\ &= 10\mathbf{1}_b\\ &= 5 = I_{odd}.\end{aligned}$$

We see that an even positive integer I_{even} can be reduced to an odd positive integer I_{odd}. In this case 20 is reduced to 5.

Remark 2.4. By multiplying an odd positive integer by 3 and adding 1, we get a result which is always even

$$3 \cdot I_{odd} + 1 = I_{even}. \tag{2.2}$$

Example 2.5. Example in binary representation.
Let I_{odd} be an odd positive integer

$$I_{odd} = 7 = 111_b.$$

Then

$$\begin{aligned}3I_{odd} + 1 &= 21 + 1\\ &= 10101_b + 1\\ &= 1011\mathbf{0}_b\\ &= 22 = I_{even}.\end{aligned}$$

We see that by multiplying an odd positive integer I_{odd} by 3 and increasing by 1, we get an even positive integer I_{even}.

Definition 2.6. For any positive integer I, let $lsb(I)$ be **the least significant nonzero bit** in the binary representation of I.

Example 2.7. Binary numbers with their least significant nonzero bits in bold:

$$lsb(10110101\mathbf{1}000_b) = \mathbf{1000}_b,$$
$$lsb(100101\mathbf{1}0_b) = \mathbf{10}_b,$$
$$lsb(101101011\mathbf{1}00_b) = \mathbf{100}_b,$$
$$lsb(110011\mathbf{1}_b) = \mathbf{1}_b,$$
$$lsb(110111\mathbf{1}000_b) = \mathbf{1000}_b.$$

Remark 2.8. For every odd positive integer I_{odd}

$$lsb(I_{odd}) = 2^0 = 1. \tag{2.3}$$

Example 2.9. We find $lsb(I_{odd})$ for an odd positive integer I_{odd}.
For $I_{odd} = 25$ we have

$$lsb(25) = lsb(1100\mathbf{1}_b) \ = \ 2^0 = 1.$$

Remark 2.10. For every even positive integer I_{even}

$$lsb(I_{even}) = 2^p, \tag{2.4}$$

where p is a positive integer, and then

$$\frac{I_{even}}{2^p} \ = \ I_{odd}, \tag{2.5}$$

therefore

$$I_{even} = 2^p I_{odd}. \tag{2.6}$$

Example 2.11. We find $lsb(I_{even})$ for an even positive integer I_{even}.
For $I_{even} = 28$ we have

$$lsb(28) = lsb(11\mathbf{1}00_b) \ = 2^2$$

and thus

$$\begin{aligned} \frac{I_{even}}{2^p} &= \frac{28}{lsb(28)} \\ &= \frac{28}{2^2} \\ &= \frac{11\mathbf{100}_b}{\mathbf{100}_b} \\ &= \ 7 = I_{odd}. \end{aligned}$$

When we divide 28 by $lsb(28)$ it gives us an odd positive integer 7.

Definition 2.12. For any positive integer I, let $msb(I)$ be **the most significant bit** in a binary representation of I.

Example 2.13. Binary numbers with their most significant bits in bold:

$$msb(\mathbf{1}01101011000_b) = \mathbf{100000000000}_b,$$
$$msb(\mathbf{1}0010110_b) = \mathbf{10000000}_b,$$
$$msb(\mathbf{1}0110101100_b) = \mathbf{10000000000}_b,$$
$$msb(\mathbf{1}100111_b) = \mathbf{1000000}_b,$$
$$msb(\mathbf{1}101111000_b) = \mathbf{1000000000}_b.$$

Definition 2.14. Let O denote a **base odd integer** of I and be defined as

$$O = \frac{I}{lsb(I)}, \tag{2.7}$$

where I can be an even or odd positive integer.

Example 2.15. Finding a base odd integer.
We check the case for an odd integer

$$I = 9 = 1001_b,$$
$$lsb(I) = lsb(100\mathbf{1}_b) = \mathbf{1}_b,$$

$$\begin{aligned} O &= \frac{I}{lsb(I)} \\ &= \frac{100\mathbf{1}_b}{\mathbf{1}_b} \\ &= 1001_b \\ &= 9. \end{aligned}$$

We conclude that for odd integers

$$O = I. \tag{2.8}$$

Notice that when I is an odd positive integer, its base odd integer O is equal to I.
Now we check the case for an even integer

$$I = 20 = 10100_b,$$
$$lsb(I) = lsb(10\mathbf{100}_b) = \mathbf{100}_b,$$

$$\begin{aligned} O &= \frac{I}{lsb(I)} \\ &= \frac{10\mathbf{100}_b}{\mathbf{100}_b} \\ &= 101_b \\ &= 5. \end{aligned}$$

To find the base odd integer O for an even integer I, we divide integer I by 2 until we get an odd result. We do this by dividing I by its least significant nonzero bit $lsb(I)$.

3. Simplification of The Collatz conjecture

Using the above remarks and definitions, standard form of the Collatz conjecture (1.1) can be substantially simplified. Despite each of the following simplifications iterating integers in slightly different way, all of them are fully aligned with original definition and therefore can be used to prove the Collatz conjecture.

Example 3.1. Iteration of the Collatz conjecture (1.1) starting from $I_0 = 11$.

TABLE 1. Original Collatz iterations starting from $I_0 = 11$.

n	I_n	$(I_n)_b$	$even/odd$	p_n	$(2^{p_n})_b$
0	11	1011	o		
1	34	100010	e	1	10
2	17	10001	o		
3	52	110100	e	2	100
4	26	11010	e		
5	13	1101	o		
6	40	101000	e	3	1000
7	20	10100	e		
8	10	1010	e		
9	5	101	o		
10	16	10000	e	4	10000
11	8	1000	e		
12	4	100	e		
13	2	10	e		
14	1	1	o		

In binary notation, division by 2 is simply a shift of the whole number by one position(bit) to the right. In Table 1, we see it for every even integer. Instead of multiple divisions by 2, it can be shortened to one operation. We divide by 2^{p_n}, where p_n is a positive integer and represents a number of consecutive zeros at the end of a binary number. Notice that 2^{p_n} is the least significant nonzero bit of an even integer, defined earlier in Definition 2.6. Merging all single divisions by 2 into one division by 2^{p_n}, we can simplify iterations of the Collatz conjecture to iterations presented in Table 2.

TABLE 2. Collatz iterations with divisions by 2^{p_n}.

n	I_n	$(I_n)_b$	$even/odd$	p_n	$(2^{p_n})_b$
0	11	1011	o		
1	34	100010	e	1	10
2	17	10001	o		
3	52	110100	e	2	100
4	13	1101	o		
5	40	101000	e	3	1000
6	5	101	o		
7	16	10000	e	4	10000
8	1	1	o		

Formally, this simplification of Collatz conjecture can be define as

$$I_{n+1} = \begin{cases} \frac{I_n}{2^{p_n}} & for,\ I_n \ \ even, \\ \\ 3 \ \cdot I_n + 1 & for,\ I_n \ \ odd, \end{cases} \tag{3.1}$$

where $2^{p_n} = lsb(I_n)$ is the least significant nonzero bit of I_n.

Symbol I_n is kept as a representation of elements in the series, even if some elements are omitted in comparison to the original Collatz conjecture proposition (1.1).

Since now each even integer is producing odd integer and each odd integer is producing even integer, we can consolidate both operations into one. This time, we process only odd positive integers, so we substitute I_n with O_n using definition (2.14). We define this simplification of the Collatz conjecture as

$$O_{n+1} = \frac{3 \ \cdot O_n + 1}{2^{p_n}}, \tag{3.2}$$

where $2^{p_n} = lsb(3 \cdot O_n + 1)$ is the least significant nonzero bit of $(3 \cdot O_n + 1)$.

Notice that $(3 \cdot O_n + 1)$ is always even, so $2^{p_n} \geq 2$ for every n. This simplification of Collatz conjecture results in iterations of odd integers only. To start from an even integer, we simply reduce it to an odd integer, by dividing it by 2 as many times as needed to achieve an odd result.

TABLE 3. Collatz iterations simplified to odd integers only.

n	O_n	$(O_n)_b$	e/o	$3O_n + 1$	$(3O_n + 1)_b$	p_n	$(2^{p_n})_b$
0	11	1011	o	34	100010	1	10
1	17	10001	o	52	110100	2	100
2	13	1101	o	40	101000	3	1000
3	5	101	o	16	10000	4	10000
4	1	1	o				

There is one more simplification we can do.

The process introduced below differs from the original Collatz proposition, however, it produces the same results. To distinguish it from the above explanations, symbol X_n is used as an element of the iterations.

Starting from any positive integer X_0, we do not need to constantly divide by 2^{p_n}. To keep this process aligned with the orginal Collatz conjecture, instead of always adding 1, we have to add the least significant nonzero bit of X_n. By this, we allow X_n to increase, ultimately reaching, instead of 1, integer in the form of 2^p, where p is a positive integer.

Table 4. Improved Collatz conjecture - iterations without divisions.

n	X_n	$(X_n)_b$	$3X_n$	$(3X_n)_b$	p_n	$(2^{p_n})_b$	$3X_n + 2^{p_n}$	$(3X_n + 2^{p_n})_b$	O_n
0	11	1011	33	100001	0	1	34	**10001**0	**17**
1	34	100010	102	1100110	1	10	104	**1101**000	**13**
2	104	1101000	312	100111000	3	1000	320	**101**000000	**5**
3	320	101000000	960	1111000000	6	1000000	1024	**1**0000000000	**1**
4	1024	10000000000							

Notice that corresponding odd integers are still present in such iterations in column O_n in Table 4. They are also visible in column $(3X_n + 2^{p_n})_b$ in bold, but for each iteration they are multiplied by constantly increasing powers of 2.

Formal definition of this improved Collatz conjecture is presented below.

Definition 3.2. For any positive integer X_0 if iterating

$$X_{n+1} = 3X_n + lsb(X_n), \tag{3.3}$$

where $lsb(X_n)$ is the least significant nonzero bit of X_n, ultimately we get $X_n = 2^p$, where p is positive integer.

This way we have two equivalent methods of iterating the Collatz conjecture. The first one, proposed in (3.2), is a simplified version of (1.1) that only skips all even numbers and, as original, finally reaches 1. The second one, without any divisions by 2, proposed in (3.3), ultimately reaches 2^p, where p is a positive integer. In this case, the result in binary representation is just 1 followed by the sequence of zeros. **Each of these two methods have exactly the same number of steps as they are strictly connected**.

Example 3.3. In Table 5, we see a comparison of iterations through both methods side by side; without divisions (3.3) as X_n and with divisions (3.2) as O_n, starting from 11.

Table 5. Equivalence of Collatz iterations without divisions X_n and with divisions O_n starting from 11.

n	X_n	$(X_n)_b$	O_n	$(O_n)_b$
0	11	**1011**	11	1011
1	34	**10001**0	17	10001
2	104	**1101**000	13	1101
3	320	**101**000000	5	101
4	1024	**1**0000000000	1	1

Example 3.4. In Table 6, we see a comparison of iterations through both methods side by side; without divisions (3.3) as X_n and with divisions (3.2) as O_n, starting from 57.

TABLE 6. Equivalence of Collatz iterations without divisions X_n and with divisions O_n starting from 57.

n	X_n	$(X_n)_b$	O_n	$(O_n)_b$
0	57	**111001**	57	111001
1	172	**101011**00	43	101011
2	520	**1000001**000	65	1000001
3	1568	**110001**00000	49	110001
4	4736	**100101**0000000	37	100101
5	14336	**111**00000000000	7	111
6	45056	**1011**000000000000	11	1011
7	139264	**10001**000000000000	17	10001
8	425984	**1101**000000000000000	13	1101
9	1310720	**101**000000000000000000	5	101
10	4194304	**1**0000000000000000000000	1	1

Example 3.5. Relations between X_n, O_n, $lsb(X_n)$ and $msb(X_n)$ are shown in the example below:

$$\underbrace{\overbrace{\overbrace{\uparrow\mathbf{10010}\mathbf{1}}^{O_n}\underbrace{0000000}_{lsb(X_n)}}^{X_n}}_{msb(X_n)}$$

$\boldsymbol{X_n}$ is the entire integer, all bits in binary notation,
$\boldsymbol{O_n}$ is the odd base of X_n, which are only bits between first and last nonzero bits,
$\boldsymbol{lsb(X_n)}$ is the least significant nonzero bit of X_n in the form of 2^p,
$\boldsymbol{msb(X_n)}$ is the most significant bit of X_n in the form of 2^q,
where p, q are positive integers.

4. Elaboration on Improved Collatz conjecture

Considering iterations of X_n through the improved Collatz conjecture proposed in (3.3) a very interesting feature can be seen. The least significant nonzero bit $lsb(X_n)$ is almost always just a small fraction of X_n. Therefore, the most significant bit $msb(X_n)$ tends to grow with coefficient on average close to 3 with each iteration. Using the improved Collatz conjecture

$$X_{n+1} = 3X_n + lsb(X_n),$$

we usually get

$$\frac{lsb(X_n)}{X_n} \approx 0, \tag{4.1}$$

therefore, we can say that on average

$$msb(X_{n+1}) \approx 3 \cdot msb(X_n). \tag{4.2}$$

Small deviations from this rule can be observed, when interactions with other bits of lower significance occur (especially when O_n is small), which can temporarily make this coefficient slightly higher.

On the other hand, the least significant bit $lsb(X_n)$, being a part of X_n, is each time multiplied by 3 and additionally increased by adding $lsb(X_n)$. Therefore, the least significant bit of X_n tends to grow with coefficient on average close to 4 with each iteration.

When iterating

$$X_{n+1} = 3X_n + lsb(X_n) \tag{4.3}$$

on average, we have

$$\begin{aligned} lsb(X_{n+1}) &\approx 3 \cdot lsb(X_n) + lsb(X_n) \\ &\approx 4 \cdot lsb(X_n). \end{aligned} \tag{4.4}$$

A deviation from this rule can occur through interactions with other bits of X_n. The coefficient can be temporary much higher than 4, when a sequence of bits in the form of "...10101010101" appears at the end of O_n which is a part of X_n (see Figure 2 for X_0). In this case, we can observe a rapid shortening of X_n. This coefficient can also be temporarily smaller, when a sequence of consecutive 1's appears at the end of O_n. In this case, this coefficient is temporarily equal 2, until number of 1's is reduced one by one in the following iterations.

Even if both described dependencies can be temporarily disturbed, eventually in a large number of iterations they become very evident. As a result of their interactions, the distance between the most significant bit $msb(X_n)$ and the least significant nonzero bit $lsb(X_n)$ gets shortened.

Notice that a difference in lengths between $msb(X_n)$ and $lsb(X_n)$ represents the length of O_n in bits.

We see

$$msb(X_n) - lsb(X_n) \to 0, \text{ as } n \to \infty, \tag{4.5}$$

and

$$msb(X_n) \;/\; lsb(X_n) \to 1, \text{ as } n \to \infty. \tag{4.6}$$

Example 4.1. Comparison of growth trends between the most significant bit $msb(X_n)$ and the least significant nonzero bit $lsb(X_n)$.

Figure 1. Comparison of growth trends between $msb(X_n)$ and $lsb(X_n)$ starting from 57.

n	X_n	$(X_n)_b$	O_n	$(O_n)_b$
0	57	**111001**	57	111001
1	172	**101011**00	43	101011
2	520	**1000001**000	65	1000001
3	1568	**110001**00000	49	110001
4	4736	**100101**0000000	37	100101
5	14336	**111**00000000000	7	111
6	45056	**1011**000000000000	11	1011
7	139264	**10001**0000000000000	17	10001
8	425984	**1101**000000000000000	13	1101
9	1310720	**101**000000000000000000	5	101
10	4194304	**1**0000000000000000000000	1	1

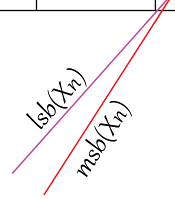
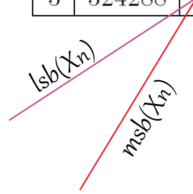


Figure 2. Comparison of growth trends between $msb(X_n)$ and $lsb(X_n)$. Special case when X_n contains a sequence of bits "...1010101".

n	X_n	$(X_n)_b$	O_n	$(O_n)_b$
0	1877	**11101010101**	1877	11101010101
1	5632	**1011**000000000	11	1011
2	17408	**10001**0000000000	17	10001
3	53248	**1101**000000000000	13	1101
4	163840	**101**000000000000000	5	101
5	524288	**1**0000000000000000000	1	1



When initial integer X_0 is very big, on average

$$\frac{msb(X_{n+1})}{msb(X_n)} = 3 \tag{4.7}$$

and on average

$$\frac{lsb(X_{n+1})}{lsb(X_n)} = 4, \tag{4.8}$$

we can propose a formula to estimate the number of iterations required to reach $O_n = 1$, which means $X_n = 2^p$, where p is a positive integer.
When using binary numbers, we know that each position represents a power of 2. Multiplication by 3 extends the length of a number by

$$\log_2(3) = 1.584963. \tag{4.9}$$

By continuous multiplication of a binary number by 3, its length increases on average by 1.584963 bits(positions) per operation.

We check how fast the least significant bit $lsb(X_n)$ increases its length, we have

$$\log_2(4) = 2. \tag{4.10}$$

We see that by continuous multiplication of the least significant nonzero bit by 4, its length increases on average by 2 bits(positions) per operation. We calculate how fast $lsb(X_n)$ approaches $msb(X_n)$.

We have

$$2 - 1.584963 = 0.415037, \tag{4.11}$$

thus $lsb(X_n)$ is on average 0.415037 bits(positions) closer to $msb(X_n)$ per iteration. Note that a number of needed iterations can be bigger, when at the end of X_0 we have a sequence of consecutive 1's "...1111111", or it can be dramatically smaller, when at the end we have a sequence of alternating 0 and 1 "...01010101".

Example 4.2. Starting from X_0, which is 20000 bits long, we can predict how many times we have to iterate, through the improved version of the Collatz conjecture (3.3), until we finally reach $O_n = 1$ (which means $X_n = 2^p$, where p is a positive integer). To approximate a number of iterations, we have to divide the length of X_0 in bits by 0.415037, in this case

$$\frac{20000}{0.415037} \approx \mathbf{48188}. \tag{4.12}$$

Exact number of required operations depends on detailed structure of bits in a particular initial integer. However, for big initial integers that do not end with consecutive 1's or alternating sequences of 0 and 1, exact number of iterations should be very close to an estimated one. In practice, starting from X_0, which was created as randomly generated 20000 bits, the exact number of operations needed to reach 1 was **48043**, which is only around 0.3% different from the estimated one.

On Figure 3, we see how length of O_n, in number of bits, decreases when iterating initial integer X_0 consisting of 20000 random bits.

Length of O_n, is the difference in bits between the length of $msb(X_n)$ and the length of $lsb(X_n)$ and decreases with almost perfect accuracy (see Figure 3). However, when we look closer at first 1000 iterations on Figure 4, we see local fluctuations. It is even more visible on Figure 5, where only first 100 iterations are presented.

Above elaboration, together with analysis of ending sequences of 1's, "...111111" described in Section 7 of this work, can be enough to proof the Collatz conjecture, however it is not used for this purpose in this work. It is only presented for better understanding how integers are processed iterating through the Collatz formula and what we can observe when analyzing their binary representations.

FIGURE 3. Decrease of O_n length for 20000 bits long initial X_0

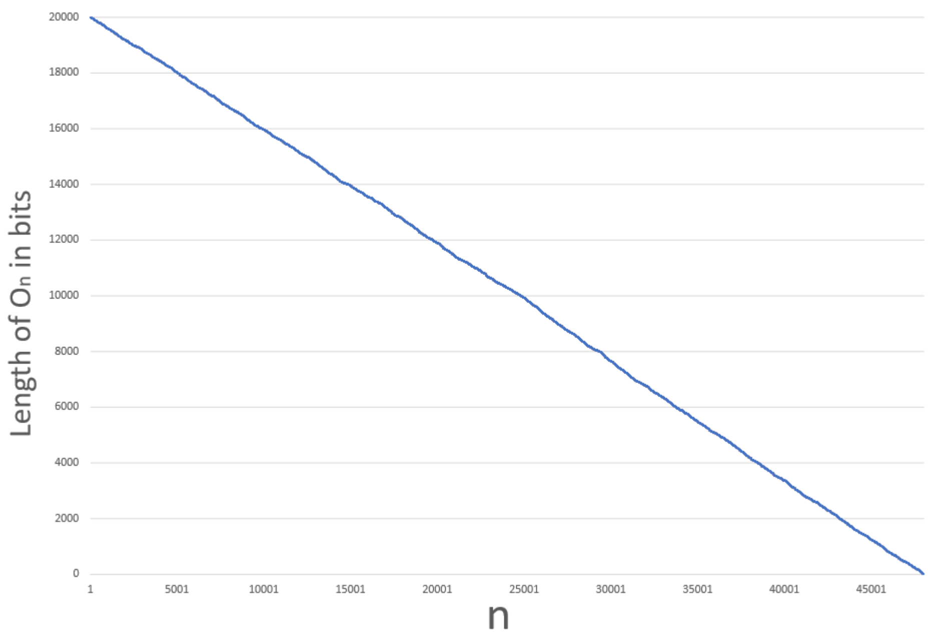


FIGURE 4. Decrease of O_n length for 20000 bits long initial X_0 *(first 1000 iterations).*

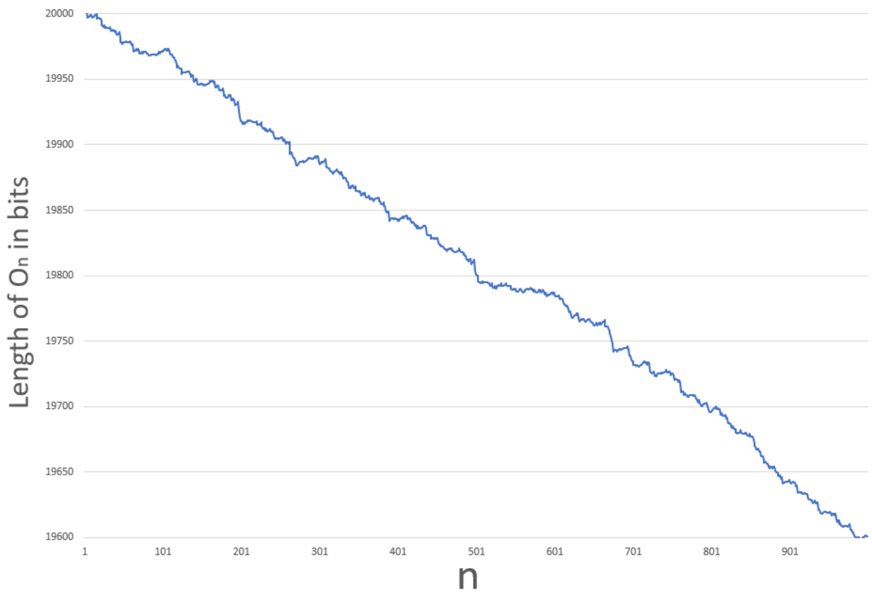
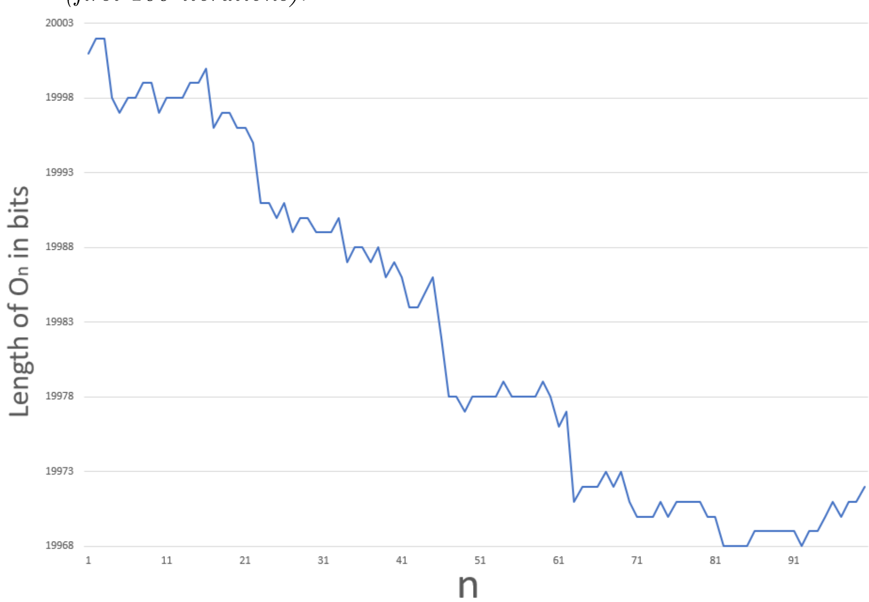


FIGURE 5. Decrease of O_n length for 20000 bits long initial X_0 *(first 100 iterations).*



5. Proof of Theorem 1.1

Proof. For any positive integer I_0, we find its base odd integer using (2.7) and it is

$$O_0 = \frac{I_0}{lsb(I_0)}. \tag{5.1}$$

Value of $lsb(I_0)$ is in the form of 2^p, where $p \geq 0$ and $p = 0$ when I_0 is odd, thus

$$O_0 = \frac{I_0}{2^p}, \tag{5.2}$$

where $p \geq 0$.

We iterate this odd positive integer O_0 through simplified Collatz conjecture presented in equation (3.2). We have

$$\frac{3\frac{3\frac{3\frac{3\frac{3O_0+1}{2^{p_0}}+1}{2^{p_1}}+1}{2^{p_2}}+1}{\cdots}+1}{2^{p_{n-2}}}+1}{2^{p_{n-1}}} = 1, \tag{5.3}$$

and O_n is odd for every n, so $(3O_n + 1)$ is always even, therefore

$$p_0, p_1, p_2, ..., p_{n-2}, p_{n-1} \geq 1. \tag{5.4}$$

Equation (5.3) can be also presented like this

$$\left(\left(\left(\left(\left((3O_0+1)\frac{3}{2^{p_0}}+1\right)\frac{3}{2^{p_1}}+1\right)\frac{3}{2^{p_2}}+1\right)...\right)\frac{3}{2^{p_{n-2}}}+1\right)\frac{1}{2^{p_{n-1}}} = 1. \tag{5.5}$$

By performing simple algebraic transformations we get

$$\begin{aligned} 3^n O_0 \ = \ & (2^{p_{n-1}} 2^{p_{n-2}} 2^{p_{n-3}} ... \ 2^{p_1} 2^{p_0}) - (2^{p_{n-2}} 2^{p_{n-3}} ... \ 2^{p_1} 2^{p_0})\, 3^0 - \\ & - (2^{p_{n-3}} ... \ 2^{p_1} 2^{p_0})\, 3^1 - \cdots - 2^{p_1} 2^{p_o} 3^{n-3} \ - \ 2^{p_o} 3^{n-2} \ - \ 3^{n-1}. \end{aligned} \tag{5.6}$$

Now, we can substitute O_0 from (5.2)

$$\begin{aligned} 3^n \tfrac{I_0}{2^p} \ = \ & (2^{p_{n-1}} 2^{p_{n-2}} 2^{p_{n-3}} ... \ 2^{p_1} 2^{p_0}) - (2^{p_{n-2}} 2^{p_{n-3}} ... \ 2^{p_1} 2^{p_0})\, 3^0 - \ ... \\ & - (2^{p_{n-3}} ... \ 2^{p_1} 2^{p_0})\, 3^1 - \cdots - 2^{p_1} 2^{p_o} 3^{n-3} \ - \ 2^{p_o} 3^{n-2} \ - \ 3^{n-1}, \end{aligned}$$

and multiply both sides by 2^p

$$\begin{aligned} 3^n I_0 \ = \ & (2^{p_{n-1}} 2^{p_{n-2}} 2^{p_{n-3}} ... \ 2^{p_1} 2^{p_0} 2^p) - (2^{p_{n-2}} 2^{p_{n-3}} ... \ 2^{p_1} 2^{p_0} 2^p)\, 3^0 - \ ... \\ & - (2^{p_{n-3}} ... \ 2^{p_1} 2^{p_0} 2^p)\, 3^1 - \cdots - 2^{p_1} 2^{p_o} 2^p 3^{n-3} \ - \ 2^{p_o} 2^p 3^{n-2} \ - \ 2^p 3^{n-1}. \end{aligned}$$

We substitute the following:

$$
\begin{aligned}
2^{p_{n-1}}2^{p_{n-2}}2^{p_{n-3}}...\ 2^{p_1}2^{p_0}2^{p} &= 2^{m_n},\\
2^{p_{n-2}}2^{p_{n-3}}...\ 2^{p_1}2^{p_0}2^{p} &= 2^{m_{n-1}},\\
2^{p_{n-3}}...\ 2^{p_1}2^{p_0}2^{p} &= 2^{m_{n-2}},\\
&...\\
2^{p_1}2^{p_o}2^{p} &= 2^{m_2},\\
2^{p_o}2^{p} &= 2^{m_1},\\
2^{p} &= 2^{m_0},
\end{aligned}
\tag{5.7}
$$

where all $p_0, p_1, p_2, ..., p_{n-2}, p_{n-1} \geq 1$ and $p \geq 0$.

We finally have

$$3^nI_0 \ = \ 2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \ \cdots \ - \ 2^{m_1}3^{n-2} \ - \ 2^{m_0}3^{n-1}, \tag{5.8}$$

where $m_n > m_{n-1} > m_{n-2} > ... > m_1 > m_0 \geq 0$ and m_0 can eventually be 0, when I_0 is odd.

□

We prove in opposite direction.

Proof. We start from integer I_0 that fulfils equation

$$3^nI_0 \ = \ 2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \ \cdots \ - \ 2^{m_1}3^{n-2} \ - \ 2^{m_0}3^{n-1},$$

where n is a positive integer and $m_n > m_{n-1} > m_{n-2} > ... > m_1 > m_0 \geq 0$ are integers.

We divide both sides by 3^n. We have

$$I_0 \ = \ \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \ \cdots \ - \ 2^{m_1}3^{n-2} \ - \ 2^{m_0}3^{n-1}}{3^n}.$$

Now we iterate through the simplified Collatz equation in the form of (3.1), which combines multiple divisions by 2 into single division by 2^p. In each iteration we receive odd integer from even integer or even integer from odd integer.

We divide I_0 by 2^{m_0}, where $m_0 \geq 0$ to receive odd integer. If I_0 is already odd then $m_0 = 0$, so $2^0 = 1$ and division by 1 does not affect the result. If I_0 is even, $m_0 > 0$ and m_0 is the number that represents how many times I_0 has to be divided by 2 to become odd. We have

$$I_1 \ = \ \frac{I_0}{\mathbf{2^{m_0}}} = \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \ \cdots \ - \ 2^{m_1}3^{n-2}}{3^n\mathbf{2^{m_0}}} - \frac{3^{n-1}}{3^n}$$

which is an odd integer. For odd integer

$$I_1 \ = \ \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \ \cdots \ - \ 2^{m_1}3^{n-2}}{3^n2^{m_0}} - \frac{1}{3}$$

we multiply by 3

$$I_1 \cdot \mathbf{3} \ = \ \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \ \cdots \ - \ 2^{m_1}3^{n-2}}{\mathbf{3^{n-1}}2^{m_0}} - \frac{\mathbf{3}}{3}$$

and add 1

$$I_2 \ = \ I_1 \cdot 3 + \mathbf{1} = \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \ \cdots \ - \ 2^{m_1}3^{n-2}}{3^{n-1}2^{m_0}}.$$

We put the last term to separate quotient

$$I_2 = \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \dots - 2^{m_2}3^{n-3}}{3^{n-1}2^{m_0}} - \frac{2^{m_1}3^{n-2}}{2^{m_0}3^{n-1}}.$$

We know that I_2 is even (from Remark 2.4), so it can be divided by 2^{p_0}, where $p_0 > 0$, to get an odd integer. We know that $m_1 > m_0$, so to make the right side of equation odd, we need $m_1 = p_0 + m_0$, which gives $2^{p_0} = \frac{2^{m_1}}{2^{m_0}}$ and then divide I_2 by 2^{p_0}

$$I_3 = \frac{I_2}{\mathbf{2^{p_0}}} = \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \dots - 2^{m_2}3^{n-3}}{3^{n-1}\mathbf{2^{m_1}}} - \frac{1}{3}.$$

Now that I_3 is odd, we multiply it by 3 again

$$I_3 \cdot \mathbf{3} = \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \dots - 2^{m_2}3^{n-3}}{\mathbf{3^{n-2}}2^{m_1}} - \frac{\mathbf{3}}{3}$$

and add 1

$$I_4 = I_3 \cdot 3 + \mathbf{1} = \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \dots - 2^{m_3}3^{n-4} - 2^{m_2}3^{n-3}}{3^{n-2}2^{m_1}}.$$

We put the last term to separate quotient

$$I_4 = \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \dots - 2^{m_3}3^{n-4}}{3^{n-2}2^{m_1}} - \frac{2^{m_2}3^{n-3}}{2^{m_1}3^{n-2}}.$$

We know that I_4 is even, so it can be divided by 2^{p_1}, where $p_1 > 0$, to become an odd integer. We know that $m_2 > m_1$, so to make the right side of equation odd, we need $m_2 = p_1 + m_1$, which gives $2^{p_1} = \frac{2^{m_2}}{2^{m_1}}$ and then divide I_4 by 2^{p_1}

$$I_5 = \frac{I_4}{2^{p_1}} = \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \dots - 2^{m_3}3^{n-4}}{3^{n-2}2^{m_2}} - \frac{3^{n-3}}{3^{n-2}}.$$

We can continue this process till

$$I_{k-4} = \frac{2^{m_n} - 2^{m_{n-1}}3^0}{3^2 2^{m_{n-2}}} - \frac{3^1}{3^2}.$$

We know I_{k-4} is odd, we multiply it by 3

$$I_{k-4} \cdot 3 = \frac{2^{m_n} - 2^{m_{n-1}}3^0}{\mathbf{3^1}2^{m_{n-2}}} - \frac{\mathbf{3^2}}{3^2}$$

and add 1

$$I_{k-3} = I_{k-4} \cdot 3 + 1 = \frac{2^{m_n} - 2^{m_{n-1}}3^0}{3^1 2^{m_{n-2}}}.$$

We put the last term to separate quotient

$$I_{k-3} = \frac{2^{m_n}}{3^1 2^{m_{n-2}}} - \frac{2^{m_{n-1}}3^0}{2^{m_{n-2}}3^1}.$$

We know that I_{k-3} is even, so it can be divided by $2^{p_{n-2}}$, where $p_{n-2} > 0$, to become an odd integer. We know that $m_{n-1} > m_{n-2}$, so to make the right side of equation odd, we need $m_{n-1} = p_{n-2} + m_{n-2}$, which gives $2^{p_{n-2}} = \frac{2^{m_{n-1}}}{2^{m_{n-2}}}$ and then divide I_{k-3} by $2^{p_{n-2}}$

$$I_{k-2} = \frac{I_{k-3}}{2^{p_{n-2}}} = \frac{2^{m_n}}{3^1 \mathbf{2^{m_{n-1}}}} - \frac{3^0}{3^1}.$$

We know I_{k-2} is odd, we multiply it by 3

$$I_{k-2} \cdot 3 = \frac{2^{m_n}}{\mathbf{3^0} 2^{m_{n-1}}} - \frac{\mathbf{3^1}}{3^1}$$

and add 1

$$I_{k-1} = I_{k-2} \cdot 3 + 1 = \frac{2^{m_n}}{3^0 2^{m_{n-1}}}.$$

We know that I_{k-1} is even, so it can be divided by $2^{p_{n-1}}$, where $p_{n-1} > 0$, to become an odd integer. We know that $m_n > m_{n-1}$, so to make the right side of equation odd, we need $m_n = p_{n-1} + m_{n-1}$, which gives $2^{p_{n-1}} = \frac{2^{m_n}}{2^{m_{n-1}}}$ and then divide I_{k-1} by $2^{p_{n-1}}$

$$I_k = \frac{I_{k-1}}{2^{p_{n-1}}} = \frac{2^{m_n}}{\mathbf{2^{m_n}}} = 1.$$

□

Notice that for any initial positive integer I_0 that fulfils equation

$$3^n I_0 = 2^{m_n} - 2^{m_{n-1}} 3^0 - 2^{m_{n-2}} 3^1 - \cdots - 2^{m_1} 3^{n-2} - 2^{m_0} 3^{n-1},$$

where n is a positive integer and $m_n > m_{n-1} > m_{n-2} > ... > m_1 > m_0 \geq 0$ are integers, we have a sequence of integers

$$I_0, I_1, I_2, I_3, ...I_{k-3}, I_{k-2}, I_{k-1}, I_k$$

and $I_k = 1$.

6. Procedure

We consider the following procedure.

Procedure 1.

Step 1. Take any positive integer I_0 and define

(6.1) $$A_0 = 2^p, \text{where } p \in \mathbb{Z}^+ \textit{ and } A_0 > I_0,$$

(6.2) $$B_0 = A_0 - I_0,$$

(6.3) $$C_0 = 0.$$

We have

(6.4) $$\boldsymbol{3^0 I_0 = A_0 - B_0 - C_0}.$$

Step 2. Multiply both sides of the equation by 3 using the following transformations

$$3^n I_0 = 3 \cdot 3^{n-1} I_0, \tag{6.5}$$

$$A_n = 4 \cdot A_{n-1}, \tag{6.6}$$

$$B_n = 3 \cdot B_{n-1} + A_{n-1} - lsb(B_{n-1}), \tag{6.7}$$

$$C_n = 3 \cdot C_{n-1} + lsb(B_{n-1}). \tag{6.8}$$

We have general formula for n^{th} iteration

$$3^n I_0 = A_n - B_n - C_n. \tag{6.9}$$

Step 3. Iterate Step 2 forever.

Remark 6.1. Notice that in Procedure 1 we have

$$A_n > B_n, \tag{6.10}$$

$$B_n > 0, \; for \; all \; n \; and \tag{6.11}$$

$$C_n > 0, \; for \; n > 0. \tag{6.12}$$

Lemma 6.2. *When iterating Procedure 1, for any initial positive integer I_0 such iteration number k exists that starting from this iteration and for all the following iterations, when $n \geq k$,*

$$A_n = B_n + lsb(B_n). \tag{6.13}$$

Proof. We iterate Procedure 1 for any initial positive integer I_0. From (6.9) we have

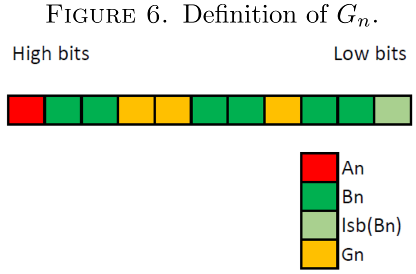
$$A_n = 3^n I_0 + B_n + C_n \tag{6.14}$$

therefore we see that

$$A_n > B_n, \; for \; all \; n. \tag{6.15}$$

We define G_n as a sum of all bits between $lsb(B_n)$ and A_n that are not part of B_n.

Figure 6. Definition of G_n.



We know that

$$G_n > lsb(B_n), \tag{6.16}$$

when there are some gaps between bits in B_n, or

$$G_n = 0,\ G_n < lsb(B_n) \tag{6.17}$$

if there are no gaps between bits in B_n and all bits are next to each other.

We have

$$A_n = B_n + G_n + lsb(B_n) \tag{6.18}$$

(*see* Figure 6).

From (6.7) we have

$$B_{n+1} = 3B_n + A_n - lsb(B_n), \tag{6.19}$$

we substitute A_n from (6.18)

$$B_{n+1} = 3B_n + B_n + G_n + lsb(B_n) - lsb(B_n) \tag{6.20}$$

therefore

$$B_{n+1} = 4B_n + G_n. \tag{6.21}$$

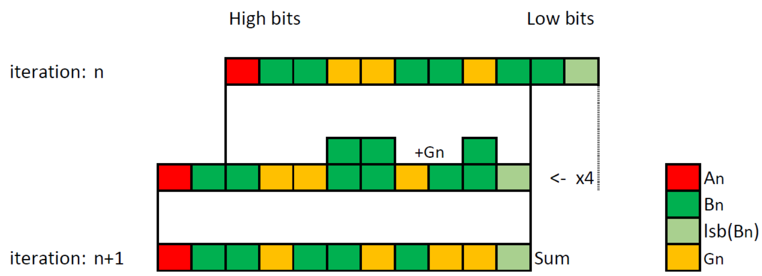
Example 6.3. When we multiply number by 4, we shift the binary representation of such number by two positions towards higher bits.

FIGURE 7. Multiplication by 4 in binary notation - all bits shifted by 2 positions.

We multiply (shift) all bits in B_n as well as all bits in G_n.

Example 6.4. In (6.21) when we change from iteration n to $n+1$, we shift all of the bits in B_n, but we also add G_n to have B_{n+1}.

Figure 8. Binary operations for $B_{n+1} = 4B_n + G_n$.



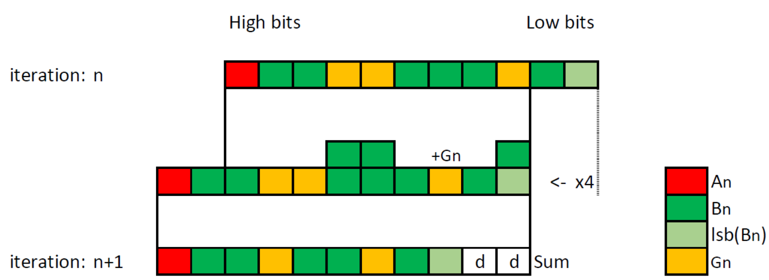
Notice how one G_n was added to $4B_n$ therefore when iterating from n to $n+1$ in this example we have

$$\begin{aligned} G_{n+1} &= 4G_n - G_n \\ &= 3G_n, \end{aligned} \tag{6.22}$$

(*compare to Figure 7*).

Example 6.5. In general case, after the operation $B_{n+1} = 4B_n + G_n$, $lsb(B_n)$ can be shifted more then two bits.

Figure 9. Binary operations for $B_{n+1} = 4B_n + G_n$, bigger shift of $lsb(B_n)$.



By comparing with Example 6.4, notice a bigger shift of $lsb(B_n)$ (by 4 positions). Some bits transfered from G_n were left behind $lsb(B_{n+1})$ and G_{n+1} is lower than $3G_n$

$$\begin{aligned} G_{n+1} &= 4G_n - G_n - d \\ &= 3G_n - d, \end{aligned} \tag{6.23}$$

where d represents all bits that are left below $lsb(B_{n+1})$.

We continue with the proof.

We divide both sides of (6.21) by B_n

$$\frac{B_{n+1}}{B_n} = \frac{4B_n}{B_n} + \frac{G_n}{B_n} \tag{6.24}$$

thus

$$\frac{B_{n+1}}{B_n} = 4 + \frac{G_n}{B_n}. \tag{6.25}$$

We evaluate $\frac{G_n}{B_n}$ for $n+1$. From (6.21) we know the change of B_n and from (6.23) we know the change of G_n therefore

$$\frac{G_{n+1}}{B_{n+1}} = \frac{3G_n - d}{4B_n + G_n}. \tag{6.26}$$

We see that

$$\lim_{n \to \infty} \frac{G_n}{B_n} = 0. \tag{6.27}$$

Remark 6.6. Notice that from (6.25) we have

$$\lim_{n \to \infty} \frac{B_{n+1}}{B_n} = 4. \tag{6.28}$$

From (6.16) we know that when there are gaps between bits in B_n, we have

$$G_n > lsb(B_n), \tag{6.29}$$

we divide both sides by B_n

$$\frac{G_n}{B_n} > \frac{lsb(B_n)}{B_n}. \tag{6.30}$$

We know from (6.27) that

$$\lim_{n \to \infty} \frac{G_n}{B_n} = 0. \tag{6.31}$$

This means that such iteration k exists that starting from this iteration and for all the following iterations n, where $n \geq k$

$$\frac{G_n}{B_n} < \frac{lsb(B_n)}{B_n} \tag{6.32}$$

this means

$$G_n < lsb(B_n) \tag{6.33}$$

therefore from (6.17)

$$G_n = 0. \tag{6.34}$$

Substituting G_n to (6.18) we have

$$A_n = B_n + lsb(B_n). \tag{6.35}$$

□

7. Proof of Theorem 1.2

Proof. We start Procedure 1 for any positive integer I_0. From Lemma 6.2 we know that such iteration number k exists that for all next iterations when $n \geq k$

$$lsb(B_n) = A_n - B_n. \tag{7.1}$$

From general formula on n^{th} iteration (6.9) we have

$$3^n I_0 \ = \ A_n \ - \ B_n - \ C_n. \tag{7.2}$$

For iterations where $n \geq k$, we substitute for $A_n - B_n$. We have

$$3^n I_0 \ = \ lsb(B_n) - \ C_n. \tag{7.3}$$

Notice that $lsb(B_n)$ is a single bit in the form of

$$lsb(B_n) = 2^{m_n}, \ where \ m_n \in \mathbb{Z}^+ \tag{7.4}$$

and C_n is created by iterating Procedure 1, based on the formula

$$C_n = 3C_{n-1} + lsb(B_{n-1}). \tag{7.5}$$

With each iteration, C_n is multiplied by 3 (all bits are multiplied by 3) and new higher bit is added, so it can be presented as

$$C_n = 3^{n-1} \cdot 2^{m_0} + 3^{n-2} \cdot 2^{m_1} + ... + 3^1 \cdot 2^{m_{n-2}} + 3^0 \cdot 2^{m_{n-1}}, \tag{7.6}$$

where

$$m_{n-1} > m_{n-2} > ... > m_1 > m_0 \geq 0.$$

We substitute in (7.3)

$$\begin{aligned} 3^n I_0 \ &= \ 2^{m_n} \ - \ (3^{n-1} \cdot 2^{m_0} + 3^{n-2} \cdot 2^{m_1} + ... + 3^1 \cdot 2^{m_{n-2}} + 3^0 \cdot 2^{m_{n-1}}) \\ &= \ 2^{m_n} \ - \ 3^{n-1} \cdot 2^{m_0} - 3^{n-2} \cdot 2^{m_1} - ... - 3^1 \cdot 2^{m_{n-2}} - 3^0 \cdot 2^{m_{n-1}} \end{aligned} \tag{7.7}$$

and we sort terms to get

$$3^n I_0 \ = \ 2^{m_n} - 2^{m_{n-1}} 3^0 - 2^{m_{n-2}} 3^1 - \ \cdots \ - \ 2^{m_1} 3^{n-2} \ - \ 2^{m_0} 3^{n-1},$$

where all m's form a sequence of integers that

$$m_n > m_{n-1} > m_{n-2} > ... > m_1 > m_0 \geq 0.$$

We conclude that for every initial positive integer I_0, when iterating Procedure 1, such positive integer k exists that for every positive integer $n \geq k$ a sequence of integers

$$m_n > m_{n-1} > m_{n-2} > ... > m_1 > m_0 \geq 0$$

exists, for which

$$3^n I_0 \ = \ 2^{m_n} - 2^{m_{n-1}} 3^0 - 2^{m_{n-2}} 3^1 - \ \cdots \ - \ 2^{m_1} 3^{n-2} \ - \ 2^{m_0} 3^{n-1}.$$

□

8. Extension of Theorem 1.2

Theorem 8.1. *For every initial positive integer I_0, an infinite number of equations exists that satisfies Theorem 1.2, therefore, it can be extended in an infinite number of ways to form the following expression*

$$\boldsymbol{I_0 = \frac{2^{m_n} - 2^{m_{n-1}}3^0 - 2^{m_{n-2}}3^1 - \cdots - 2^{m_1}3^{n-2} - 2^{m_0}3^{n-1}}{3^n}}, \tag{8.1}$$

where n is a positive integer and all m's form a sequence of integers that

$$m_n > m_{n-1} > m_{n-2} > ... > m_1 > m_0 \geq 0.$$

Proof. The proof of Theorem 1.2 confirms that. □

9. Examples

Presented below are various examples of positive integers, confirming the Theorems proven above.

$$3^6 \cdot \mathbf{9} = 2^{13} - 2^9 3^0 - 2^6 3^1 - 2^4 3^2 - 2^3 3^3 - 2^2 3^4 - 2^0 3^5 \tag{9.1}$$

$$3^7 \cdot \mathbf{9} = 2^{15} - 2^{13} 3^0 - 2^9 3^1 - 2^6 3^2 - 2^4 3^3 - 2^3 3^4 - 2^2 3^5 - 2^0 3^6 \tag{9.2}$$

$$3^8 \cdot \mathbf{9} = 2^{17} - 2^{15} 3^0 - 2^{13} 3^1 - 2^9 3^2 - 2^6 3^3 - 2^4 3^4 - 2^3 3^5 - 2^2 3^6 - 2^0 3^7 \tag{9.3}$$

$$\begin{aligned} 3^{12} \cdot \mathbf{6541} = 2^{32} - 2^{28} 3^0 - 2^{25} 3^1 - 2^{23} 3^2 - 2^{22} 3^3 - 2^{21} 3^4 - 2^{17} 3^5 \\ -2^{15} 3^6 - 2^{13} 3^7 - 2^{10} 3^8 - 2^9 3^9 - 2^3 3^{10} - 2^0 3^{11} \end{aligned} \tag{9.4}$$

$$3^7 \cdot \mathbf{435} = 2^{20} - 2^{16} 3^0 - 2^{11} 3^1 - 2^{10} 3^2 - 2^9 3^3 - 2^4 3^4 - 2^1 3^5 - 2^0 3^6 \tag{9.5}$$

$$\begin{aligned} 3^{41} \mathbf{27} = 2^{70} - 2^{66} 3^0 - 2^{61} 3^1 - 2^{60} 3^2 - 2^{59} 3^3 - 2^{56} 3^4 - 2^{52} 3^5 \\ -2^{50} 3^6 - 2^{48} 3^7 - 2^{44} 3^8 - 2^{43} 3^9 - 2^{42} 3^{10} - 2^{41} 3^{11} - 2^{38} 3^{12} \\ -2^{37} 3^{13} - 2^{36} 3^{14} - 2^{35} 3^{15} - 2^{34} 3^{16} - 2^{33} 3^{17} - 2^{31} 3^{18} - 2^{30} 3^{19} \\ -2^{28} 3^{20} - 2^{27} 3^{21} - 2^{26} 3^{22} - 2^{23} 3^{23} - 2^{21} 3^{24} - 2^{20} 3^{25} - 2^{19} 3^{26} \\ -2^{18} 3^{27} - 2^{16} 3^{28} - 2^{15} 3^{29} - 2^{14} 3^{30} - 2^{12} 3^{31} - 2^{11} 3^{32} - 2^9 3^{33} \\ -2^7 3^{34} - 2^6 3^{35} - 2^5 3^{36} - 2^4 3^{37} - 2^3 3^{38} - 2^1 3^{39} - 2^0 3^{40} \end{aligned} \tag{9.6}$$

$$\begin{aligned} 3^{34} \cdot \mathbf{121} = 2^{61} - 2^{57} 3^0 - 2^{52} 3^1 - 2^{51} 3^2 - 2^{50} 3^3 - 2^{47} 3^4 - 2^{43} 3^5 \\ -2^{41} 3^6 - 2^{39} 3^7 - 2^{35} 3^8 - 2^{34} 3^9 - 2^{33} 3^{10} - 2^{32} 3^{11} - 2^{29} 3^{12} \\ -2^{28} 3^{13} - 2^{27} 3^{14} - 2^{26} 3^{15} - 2^{25} 3^{16} - 2^{24} 3^{17} - 2^{22} 3^{18} - 2^{21} 3^{19} \\ -2^{19} 3^{20} - 2^{18} 3^{21} - 2^{17} 3^{22} - 2^{14} 3^{23} - 2^{12} 3^{24} - 2^{11} 3^{25} - 2^{10} 3^{26} \\ -2^9 3^{27} - 2^7 3^{28} - 2^6 3^{29} - 2^5 3^{30} - 2^3 3^{31} - 2^2 3^{32} - 2^0 3^{33} \end{aligned} \tag{9.7}$$

(9.8)

$$
\begin{aligned}
3^{174}\cdot \mathbf{8388607} = 2^{299} & - 2^{295}3^{0} - 2^{290}3^{1} - 2^{289}3^{2} - 2^{288}3^{3} - 2^{285}3^{4} \\
& - 2^{281}3^{5} - 2^{279}3^{6} - 2^{277}3^{7} - 2^{273}3^{8} - 2^{272}3^{9} - 2^{271}3^{10} - 2^{270}3^{11} - 2^{267}3^{12} \\
& - 2^{266}3^{13} - 2^{265}3^{14} - 2^{264}3^{15} - 2^{263}3^{16} - 2^{262}3^{17} - 2^{260}3^{18} - 2^{259}3^{19} - 2^{257}3^{20} \\
& - 2^{256}3^{21} - 2^{255}3^{22} - 2^{252}3^{23} - 2^{250}3^{24} - 2^{249}3^{25} - 2^{248}3^{26} - 2^{247}3^{27} - 2^{245}3^{28} \\
& - 2^{244}3^{29} - 2^{243}3^{30} - 2^{241}3^{31} - 2^{240}3^{32} - 2^{236}3^{33} - 2^{235}3^{34} - 2^{234}3^{35} - 2^{233}3^{36} \\
& - 2^{232}3^{37} - 2^{229}3^{38} - 2^{227}3^{39} - 2^{225}3^{40} - 2^{224}3^{41} - 2^{223}3^{42} - 2^{221}3^{43} - 2^{219}3^{44} \\
& - 2^{218}3^{45} - 2^{214}3^{46} - 2^{213}3^{47} - 2^{207}3^{48} - 2^{206}3^{49} - 2^{204}3^{50} - 2^{201}3^{51} - 2^{200}3^{52} \\
& - 2^{198}3^{53} - 2^{197}3^{54} - 2^{196}3^{55} - 2^{195}3^{56} - 2^{193}3^{57} - 2^{190}3^{58} - 2^{187}3^{59} - 2^{185}3^{60} \\
& - 2^{184}3^{61} - 2^{183}3^{62} - 2^{180}3^{63} - 2^{179}3^{64} - 2^{178}3^{65} - 2^{173}3^{66} - 2^{172}3^{67} - 2^{171}3^{68} \\
& - 2^{170}3^{69} - 2^{169}3^{70} - 2^{168}3^{71} - 2^{166}3^{72} - 2^{165}3^{73} - 2^{163}3^{74} - 2^{162}3^{75} - 2^{160}3^{76} \\
& - 2^{158}3^{77} - 2^{157}3^{78} - 2^{151}3^{79} - 2^{150}3^{80} - 2^{148}3^{81} - 2^{147}3^{82} - 2^{146}3^{83} - 2^{145}3^{84} \\
& - 2^{143}3^{85} - 2^{139}3^{86} - 2^{138}3^{87} - 2^{131}3^{88} - 2^{130}3^{89} - 2^{128}3^{90} - 2^{126}3^{91} - 2^{123}3^{92} \\
& - 2^{122}3^{93} - 2^{121}3^{94} - 2^{120}3^{95} - 2^{119}3^{96} - 2^{118}3^{97} - 2^{117}3^{98} - 2^{116}3^{99} - 2^{114}3^{100} \\
& - 2^{113}3^{101} - 2^{112}3^{102} - 2^{108}3^{103} - 2^{107}3^{104} - 2^{105}3^{105} - 2^{102}3^{106} - 2^{101}3^{107} \\
& - 2^{100}3^{108} - 2^{99}3^{109} - 2^{98}3^{110} - 2^{94}3^{111} - 2^{93}3^{112} - 2^{91}3^{113} - 2^{90}3^{114} - 2^{89}3^{115} \\
& - 2^{87}3^{116} - 2^{86}3^{117} - 2^{84}3^{118} - 2^{83}3^{119} - 2^{81}3^{120} - 2^{80}3^{121} - 2^{78}3^{122} - 2^{74}3^{123} \\
& - 2^{72}3^{124} - 2^{71}3^{125} - 2^{69}3^{126} - 2^{67}3^{127} - 2^{66}3^{128} - 2^{65}3^{129} - 2^{61}3^{130} - 2^{60}3^{131} \\
& - 2^{59}3^{132} - 2^{58}3^{133} - 2^{57}3^{134} - 2^{56}3^{135} - 2^{54}3^{136} - 2^{53}3^{137} - 2^{52}3^{138} - 2^{49}3^{139} \\
& - 2^{46}3^{140} - 2^{42}3^{141} - 2^{40}3^{142} - 2^{39}3^{143} - 2^{36}3^{144} - 2^{34}3^{145} - 2^{32}3^{146} - 2^{30}3^{147} \\
& - 2^{29}3^{148} - 2^{28}3^{149} - 2^{24}3^{150} - 2^{22}3^{151} - 2^{21}3^{152} - 2^{20}3^{153} - 2^{19}3^{154} - 2^{18}3^{155} \\
& - 2^{17}3^{156} - 2^{16}3^{157} - 2^{15}3^{158} - 2^{14}3^{159} - 2^{13}3^{160} - 2^{12}3^{161} - 2^{11}3^{162} - 2^{10}3^{163} \\
& - 2^{9}3^{164} - 2^{8}3^{165} - 2^{7}3^{166} - 2^{6}3^{167} - 2^{5}3^{168} - 2^{4}3^{169} - 2^{3}3^{170} - 2^{2}3^{171} - 2^{1}3^{172} \\
& - 2^{0}3^{173}
\end{aligned}
$$

Email address: mr.leszek.mazurek@gmail.com ,