The Collatz Singularity as a Global Binary Structural Attractor

Ammar HAMDOUS

September 3, 2026

Abstract

In our fifth work [3] entitled *A Structural Approach to the Collatz Conjecture via the Binary Singularity*, we introduced the notion of *binary singularity of Collatz sequence* as a structural invariant and proposed a structural reformulation of the Collatz conjecture: every positive odd integer either is, or eventually reaches, a singularity configuration under a finite number of iterations. We further hypothesized that this property could be characterized through an appropriate metric defined on binary structures.

In the present work, we develop this idea by introducing a **global binary structural attractor** governing the evolution of odd Syracuse iterates. This attractor is based on the **canonical binary decomposition** of the binary structure of odd integers.

We prove that every odd Syracuse iterate that is not a singularity admits a unique decomposition into a finite sequence of nested concatenation of suffixes of generators of the suffix of singularity followed by a single suffix of singularity, $\mathcal{O}_n = G_{(p,n)}\|G_{(p-1,n)}\|\cdots\|G_{(1,n)}\|S_n$, and define a structural complexity function $P(\mathcal{O}_n)$ equal to the number of suffix generators contained in this decomposition.

The evolution of P is entirely determined by the complementarity of the overflow generated by the most significant suffix generator $G_{(p,n)}$, leading to a complete structural classification of increasing, stationary, and decreasing transitions. This decomposition provides a symbolic description of the odd Syracuse dynamics and establishes the structural foundation for an eventual Lyapunov function whose minimal state is precisely the binary singularity $P = 0$.

Rather than studying the arithmetic growth of Collatz trajectories, the proposed approach describes the dynamics as the progressive elim-

ination of nested binary suffixes of generators toward a unique global structural attractor.

1 Introduction

We proposed in our previous work [3] a structural reinterpretation of the Collatz dynamics based on the notion of *binary singularity*. Within that framework, convergence to 1 or the trivial cycle is no longer regarded as the fundamental phenomena but rather as a trivial consequence of reaching a particular binary configuration, called a singularity.

The previous paper [3] established the structural role of singularity and argued that it constitutes the only mechanism allowing the Collatz trajectory to escape the alternating growth–decay behavior generated by the map. It concluded with the hypothesis that such a property should be characterized by an appropriate metric defined on binary structures whose minimal states are precisely this singularity.

The present work develops this hypothesis, instead of analyzing odd Syracuse iterates through their arithmetic values, we investigate their binary structure organization. This analysis reveals that every odd iterate that is not a singularity possesses a canonical structure composed of concatenated binary blocks that progressively generate suffixes of singularity.

Consequently, the Collatz dynamics admits a symbolic interpretation in which arithmetic complexity is replaced by structural complexity. The binary singularity becomes the unique minimal state of the decomposition, while the function P naturally emerges as a candidate for an eventual Lyapunov function associated with the global binary structural attractor.

From this perspective, the Collatz conjecture is transformed into a structural problem: proving that every canonical binary decomposition is eventually absorbed by this unique attractor through the successive elimination of its suffixes of the generators.

The objective of this paper is therefore not only to extend the structural theory introduced in our previous work but also to establish a unified binary framework in which singularity, canonical decomposition, overflow complementarity, and structural complexity appear as different manifestations of the same underlying dynamical mechanism. This global viewpoint provides a new perspective on the odd Syracuse map and suggests that its long-term behavior is governed by an intrinsic binary attractor rather than by its arithmetic growth.

2 Notion of a Suffix of a Binary Structure

Let $A = (a_k a_{k-1} \cdots a_{j+1} a_j a_{j-1} \cdots a_2 a_1)_2$ be a finite or infinite binary structure, where

$$k \in \{1, 2, \ldots, \infty\}.$$

The finite binary structure $(b_j b_{j-1} \cdots b_2 b_1)_2$ is called a suffix of the binary structure A if and only if $j \leq k$, and

$$\forall i \in \{1, 2, \ldots, j\}, \qquad b_i = a_i,$$

with either $b_{j+1} \neq a_{j+1}$ or $j = k$.

3 The Odd Syracuse Sequence

Within the framework of the binary-structure approach to the Collatz conjecture, and since the Collatz singularity is an odd term of the form

$$\sum_{i=0}^{n} 2^{2i},$$

It is possible to reduce the Collatz sequence to the odd Syracuse sequence.

This sequence maps odd natural numbers to odd natural numbers, and all of its terms remain odd. This is achieved by dividing $3n + 1$ by

$$2^{\nu_2(3n+1)}.$$

Let $2\mathbb{N} + 1 = \{n \in \mathbb{N} : n \equiv 1 \pmod 2\}$ denote the set of odd natural numbers.

The odd Syracuse sequence is formally defined by $\mathcal{O} : 2\mathbb{N} + 1 \to 2\mathbb{N} + 1$ and $\mathcal{O}_{n+1} = \frac{3\mathcal{O}_n+1}{2^{\nu_2(3\mathcal{O}_n+1)}}$.

The theorem 5.11.4 in our first work [1] becomes as follows:

3.1 Theorem

The Collatz conjecture holds if and only if every term of the odd Syracuse sequence is either a singularity or eventually reaches a singularity under odd Syracuse iterations.

4 Structural Binary Reduction of the Odd Syracuse Iteration

In the odd Syracuse sequence, the division by $2^{\nu_2(3\mathcal{O}_n+1)}$ directly yields the next odd term, since this division removes all trailing zero bits from the binary representation of $3\mathcal{O}_n + 1$.

Let $(a_k a_{k-1} \cdots a_2 a_1)_2 = \mathcal{O}_n$ be an odd term of the odd Syracuse sequence, and let

$$(b_{k+2} b_{k+1} b_k \cdots b_2 b_1)_2$$

be the binary representation of $3\mathcal{O}_n + 1$.

Let $r_1, r_2, \ldots, r_k, r_{k+1}$ be the carries obtained when adding the bits of $\mathcal{O}_n$, the bits of $2\mathcal{O}_n$, and the extra $+1$ in

$$3\mathcal{O}_n + 1 = 1 + \mathcal{O}_n + 2\mathcal{O}_n.$$

		r_{k+1}	r_k	r_{k-1}	$\ldots$	r_j	r_{j-1}	$\ldots$	r_2	r_1
$+$			a_k	a_{k-1}	$\ldots$	a_j	a_{j-1}	$\ldots$	a_2	a_1
$+$		a_k	a_{k-1}	$\ldots$	a_j	a_{j-1}	$\ldots$	a_2	a_1	0
$=$	b_{k+2}	b_{k+1}	b_k	b_{k-1}	$\ldots$	b_j	$b_{j-1} \ldots$		b_2	b_1

For every $i \in \{2, 3, \ldots, k\}$, the carry r_{i+1} is generated by $r_i + a_i + a_{i-1}$.

The carry r_1 is a pseudo-carry representing the extra $+1$ in

$$3\mathcal{O}_n + 1 = 1 + \mathcal{O}_n + 2\mathcal{O}_n.$$

We have b_1 obtained from $r_1 + a_1 + 0$, and b_2 obtained from $r_2 + a_2 + a_1$.

More generally, b_j is obtained from $r_j + a_j + a_{j-1}$.

Hence the bits $b_1, b_2, \ldots, b_j$ are generated sequentially using $a_1, a_2, \ldots, a_j$, ,the shifted bits $0, a_1, a_2, \ldots, a_{j-1}$, and the carry $r_1, r_2, \ldots, r_j$.

Consequently, the bits $b_1, b_2, \ldots, b_j$ depend exclusively on the first j least significant bits of $\mathcal{O}_n$.

Therefore, all bits equal to zero in the least significant bits of $3\mathcal{O}_n + 1$ are obtained by involving exactly the first $\nu_2(3\mathcal{O}_n + 1) = j$ least significant bits of $\mathcal{O}_n$ in the operation $3\mathcal{O}_n + 1$.

Furthermore, from Theorem 7.3.5 of our fourth work [2], we conclude that the bits of $\mathcal{O}_n$ generating the successive least significant zero bits of the result $3\mathcal{O}_n+1$ are precisely the alternating least significant bits of $\mathcal{O}_n$, whose number is $\nu_2(3\mathcal{O}_n + 1)$.

According to Theorem 7.3.5 in our fourth work [2] and the definition of a suffix of a binary structure in section 2, the binary structure $(a_j, a_{j-1}, \ldots, a_2, a_1)_2$ with $j = \nu_2(3\mathcal{O}_n + 1)$, Is a suffix of the binary structure of the singularity.

Since a singularity has an alternating binary structure over $\{0, 1\}$ and begins with a bit equal to 1, and since every singularity is odd and ends with a bit equal to 1, the binary structure

$$(a_j, a_{j-1}, \ldots, a_2, a_1)_2 \text{ with } j = \nu_2(3\mathcal{O}_n + 1)$$

is called a *a suffix of the Collatz singularity.*

Hence the bits that generate the binary structure of $\mathcal{O}_{n+1}$ are

$$(a_k, a_{k-1}, \ldots, a_{j+2}, a_{j+1})_2 = m,$$

with one iteration $f(m) = \mathcal{O}_{n+1}$.

Thus we can perform a binary structural reduction of the odd Syracuse iteration as follows:

$$\mathcal{O}_n = m \| S,$$

where $S = (a_j, a_{j-1}, \ldots, a_2, a_1)_2$ is the suffix of the singularity and

$$m = (a_k, a_{k-1}, \ldots, a_{j+2}, a_{j+1})_2.$$

Therefore $\mathcal{O}_n = m\, 2^j + S$.

We have $a_j \in \{0, 1\}$ And according to the definition of a suffix of a binary structure in section 2,
we have $(a_j = 0 \Rightarrow a_{j+1} = 0)$ and $(a_j = 1 \Rightarrow a_{j+1} = 1)$.

For every term $\mathcal{O}_n$ of the odd Syracuse sequence, $3\mathcal{O}_n + 1$ is even, hence

$$\nu_2(3\mathcal{O}_n + 1) \geq 1.$$

Since $\nu_2(3\mathcal{O}_n + 1)$ equals the number of bits of the suffix of singularity of the binary structure of $\mathcal{O}_n$ then we conclude that every term of the odd Syracuse sequence possesses a suffix of singularity.

Since our objective is to prove the Collatz conjecture using Theorem 3.1, the decomposition $\mathcal{O}_n = m \| S$ is used only when $\mathcal{O}_n$ is not itself a singularity.

5 Lemma

Let the odd Syracuse term $\mathcal{O}_n = m \| S$

And let $S = (a_j, a_{j-1}, \ldots, a_2, a_1)_2$ be a suffix of singularity.

If $a_j = 0$, then $3S + 1 = 2^j$.

If $a_j = 1$, then $3S + 1 = 2^{j+1}$.

Proof

Since S is a suffix of singularity,

$$a_1 = 1, \qquad a_j \in \{0, 1\},$$

and for every

$$i \in \{1, \ldots, j-1\},$$
$$a_i = 0 \Rightarrow a_{i+1} = 1,$$
$$a_i = 1 \Rightarrow a_{i+1} = 0.$$

We compute $3S = S + 2S$.

Thus

$$3S = (a_j a_{j-1} \cdots a_2 a_1)_2 + (a_j a_{j-1} \cdots a_2 a_1 0)_2.$$

Hence

$$3S = a_j 2^j + \sum_{i=2}^{j} (a_i + a_{i-1}) 2^{i-1} + 1.$$

But since

$$a_i + a_{i-1} = 1 \qquad \text{for } (2 \leq i \leq j),$$

we obtain

$$3S = a_j 2^j + \sum_{i=2}^{j} 2^{i-1} + 1.$$

Therefore

$$3S = a_j 2^j + \sum_{i=0}^{j-1} 2^i.$$

Using $\sum_{i=0}^{j-1} 2^i = 2^j - 1$, It follows that $3S = (2^j - 1) + a_j 2^j$.

If $a_j = 0$, then $3S = 2^j - 1$, hence $3S + 1 = 2^j$.

If $a_j = 1$, then $3S = (2^j - 1) + 2^j$, and therefore $3S + 1 = 2^{j+1}$.

Case 1: $a_j = 0$

So $a_{j-1} = 1$.

Therefore, the binary structure

$$(a_{j-1}, a_{j-2}, \ldots, a_2, a_1)_2$$

has the form of a Collatz singularity. However, it is not itself a singularity since it is not a complete term of the Collatz or odd Syracuse sequence.

From the lemma 5,

$$3S + 1 = 2^j.$$

Hence

$$3\mathcal{O}_n + 1 = 3(2^j m + S) + 1 = 2^j(3m) + (3S + 1) = 2^j(3m) + 2^j = 2^j(3m + 1).$$

Therefore when the suffix of the singularity ends with a bit equal to 0, witch means that the binary structure m starts with a bit equal to 0, we have:

$$\mathcal{O}_{n+1} = f(m) = 3m + 1,$$

since

$$\frac{3\mathcal{O}_n + 1}{2^{\nu_2(3\mathcal{O}_n+1)}} = \frac{2^j(3m + 1)}{2^j}.$$

Case 2: $a_j = 1$

Then the binary structure S has the form of a Collatz singularity. By the lemma 5 we have:

$$3S + 1 = 2^{j+1}.$$

Thus

$$3\mathcal{O}_n + 1 = (2^j 3m) + (3S + 1) = 2^j 3m + 2^{j+1} = 2^j(3m + 2).$$

Hence

$$\mathcal{O}_{n+1} = f(m) = 3m + 2,$$

since

$$\frac{3\mathcal{O}_n + 1}{2^{\nu_2(3\mathcal{O}_n+1)}} = \frac{2^j(3m + 2)}{2^j}.$$

Therefore, $\mathcal{O}_{n+1} = f(m) = 3m + 2$ whenever the suffix of singularity ends with a bit equal to 1, witch means that the binary structure m starts with a bit equal to 1, we have.

Dependence of f on m

We now formally show that the function f depends only on m.

Consider the binary operation $3\mathcal{O}_n + 1 = 1 + \mathcal{O}_n + 2\mathcal{O}_n$.

		r_{k+1}	r_k	r_{k-1}	$\dots$	r_j	r_{j-1}	$\dots$	r_2	r_1
$+$			a_k	a_{k-1}	$\dots$	a_j	a_{j-1}	$\dots$	a_2	a_1
$+$		a_k	a_{k-1}	$\dots$	a_j	a_{j-1}	$\dots$	a_2	a_1	0
$=$	b_{k+2}	b_{k+1}	b_k	b_{k-1}	$\dots$	b_j	$b_{j-1}\dots$		b_2	b_1

Let $\mathcal{O}_n = m \| S$, with $m = (a_k, a_{k-1}, \ldots, a_{j+2}, a_{j+1})_2$ and $S = (a_j, a_{j-1}, \ldots, a_2, a_1)_2$.

We have: $m \equiv 0 \pmod 2$, implies that $a_{j+1} = 0$.

From the definition of a suffix of a binary structure, and the definition of a Collatz singularity,

$$a_{j+1} = 0 \Longrightarrow a_j = 0.$$

By the Lemma 5,

$$a_j = 0 \Longrightarrow 3S + 1 = 2^j.$$

So all the information needed to deduce that $3S + 1 = 2^j$ is contained in the binary structure of m

Similarly,

$$m \equiv 1 \pmod 2 \Longrightarrow a_{j+1} = 1 \Longrightarrow a_j = 1,$$

and therefore $3S + 1 = 2^{j+1}$.

For $3S + 1 = 2^j$, this is equivalent to setting $r_{j+1} = 1$.

For $3S + 1 = 2^{j+1}$, this is equivalent to setting $r_{j+2} = 1$.

Since

$$b_1 = b_2 = \cdots = b_j = 0,$$

we obtain the binary representation corresponding to $f(m)$.

Since the maximal gain in binary length of $3\mathcal{O}_n + 1$ is $+2$ bits section 2.3 in our fifth work [3],then the maximal gain in binary length of $f(m) = \mathcal{O}_{n+1}$ is also $+2$ bits.

Moreover,

$$a_{j+1} = 0 \Longrightarrow r_{j+1} = 1,$$

and

$$a_{j+1} = 1 \Longrightarrow (r_{j+1} = 0) \wedge (r_{j+2} = 1).$$

Therefore, the function f depends only on m.

5.1 Lemma

Let the odd Syracuse term $\mathcal{O}_n = m \| S$ then the next term is given by:

$$\mathcal{O}_{n+1} = f(m) = \begin{cases} 3m + 1, \text{ if } m \equiv 0 \pmod 2, \\ 3m + 2, \text{ if } m \equiv 1 \pmod 2. \end{cases}$$

6 Lemma

Let $A = (a_k a_{k-1} \dots a_2 a_1)_2, \qquad k \geq 2$, be a binary structure.

If A undergoes the binary operation

$$3A + d, \qquad d \in \{0, 1, 2\}.$$

Then for every $i \in \{2, 3, \dots, k\}$, the binary structure

$$B = (a_k a_{k-1} \dots a_{i+1} a_i)_2$$

undergoes an operation of the form $3B + d', \qquad d' \in \{0, 1, 2\}$, and produces the binary result $(b_{k+2} b_{k+1} b_k \dots b_{i+1} b_i)_2$.

Proof

		r_{k+1}	r_k	r_{k-1}	$\dots$	r_i	r_{i-1}	$\dots$	r_2	r_1
$+$			a_k	a_{k-1}	$\dots$	a_i	a_{i-1}	$\dots$	a_2	a_1
$+$		a_k	a_{k-1}	$\dots$	a_i	a_{i-1}	$\dots$	a_2	a_1	0
$=$	b_{k+2}	b_{k+1}	b_k	b_{k-1}	$\dots$	b_i	$b_{i-1} \dots$		b_2	b_1

If $d = 0$, then $r_1 = 0$, is the only pseudo-carry.

If $d = 1$, then $r_1 = 1$, and r_1 is the only pseudo-carry.

If $d = 2$, then $r_1 = 0$ and $r_2 = 1$, and r_2 are the two pseudo-carries.

According to the binary addition diagram of the operation $3A + d$, the binary structure

$$(b_{k+2} b_{k+1} \dots b_{i+1} b_i)_2$$

is produced by

$$3B + r_i + a_{i-1}.$$

To make the operation depend only on B, we replace a_{i-1} by 0 and transfer its value to the pseudo-carry r_i.

If $a_{i-1} = 0$, then r_i keeps its value and is renamed a pseudo-carry.

If $a_{i-1} = 1$, there are two cases:

1. If $r_i = 0$, define the pseudo-carry $r_i = 1$.

2. If $r_i = 1$, define the pseudo-carries r_i, r_{i+1}

$$r_i = 0, \qquad r_{i+1} = 1.$$

Hence the binary structure B undergoes the operation $3B+r_i$ or $3B+r_{i+1}$, which is equivalent to $3B + d', \qquad d' \in \{0, 1, 2\}$.

Furthermore, in the operation $3A + d, \qquad d \in \{0, 1, 2\}$, for every $i \in \{1, 2, \ldots, k\}$, the bits $a_1, a_2, \ldots, a_i$ are generated by $r_i + a_i + a_{i-1}$, with $a_{i-1} = 0$ when $i = 1$.

Therefore, for ever $i \in \{1, 2, \ldots, k\}$, the binary structure $(a_i a_{i-1} \ldots a_2 a_1)_2$ is obtained using exactly the first i least significant bits of the complete binary structure A.

7 Lemma

Let the binary structure $A = (a_k a_{k-1} \ldots a_2 a_1)_2$.

For any $i \in \{1, 2, \ldots, k\}$

the first i least significant bits of the binary structure obtained from

$$3A + d \text{ with } d \in \{0, 1, 2\}$$

are generated using exactly the first **i** least significant bits of A.

8 Generator of the suffix of singularity

Let $\mathcal{O}_0$ be a term of the Odd Syracuse sequence that is not a singularity, according to the binary structural reduction established in Section 4, we have: $\mathcal{O}_0 = m_0 \| S_0$, where S_0 is a singularity suffix and m_0 contains at least one bit.

Then according to Lemma 5.1:

$$\mathcal{O}_1 = \begin{cases} 3m_0 + 1, & \text{if } m_0 \equiv 0 \pmod 2, \\ 3m_0 + 2, & \text{if } m_0 \equiv 1 \pmod 2. \end{cases}$$

According to section 4 if $\mathcal{O}_1$ is not a singularity, then it possesses a suffix of singularity S_1, and thus

$$\mathcal{O}_1 = m_1 \| S_1,$$

where m_1 consists of a very large number of bits.

And using Lemma (7), the suffix S_1 is generated by an operation

$$3m_0 + d, \qquad d \in \{0, 1, 2\},$$

involving the first least significant bits of m_0, and the number of these bits is equal to the number of bits in the binary structure S_1.

Moreover, by Lemma 5.1,

$$d \in \{1, 2\}.$$

Since S_1 is a binary structure with a fixed length, the least significant bits of m_0 that generate S_1 are called the suffix of the generator of the suffix of singularity S_1.

The binary pattern that generates a suffix of singularity of infinite length is called a generator of the suffix of singularity.

8.1 The Binary Structure of a Generator of the Suffix of Singularity

Let S_1 be a suffix of singularity containing a very large number of bits and generated from the least significant bits of m_0 through the operation $3m_0 + 1$ when m_0 is even, and through $3m_0 + 2$ when m_0 is odd.

Case 1: $m_0 \equiv 0 \pmod 2$

Let

$$m_0 = (a_k a_{k-1} \cdots a_2 a_1)_2.$$

We have the diagram of the operation $3m_0 + 1 = 1 + m_0 + 2m_0$

$$\begin{array}{cccccccc} & & r_{k+1} & r_k & r_{k-1} & \dots & r_2 & r_1 \\ + & & & a_k & a_{k-1} & \dots & a_2 & a_1 \\ + & & a_k & a_{k-1} & \dots & a_2 & a_1 & 0 \\ \hline = & b_{k+2} & b_{k+1} & b_k & b_{k-1} & \dots & b_2 & b_1 \end{array}$$

Since $m_0 \equiv 0 \pmod 2$, we have $a_1 = 0$.

There is a unique pseudo-carry $r_1 = 1$, representing the added $+1$ in the operation $3m_0 + 1$.

Assume $k \gg 1$.

Step 1

$$b_1 = 1$$

since it is the least significant bit of the suffix of singularity S_1.

$$b_1 = 1 \Rightarrow \begin{cases} r_1 + a_1 + 0 = (1)_2 \\ r_1 + a_1 + 0 = (11)_2 \end{cases} \Rightarrow \begin{cases} 1 + a_1 + 0 = (1)_2 \\ 1 + a_1 + 0 = (11)_2 \end{cases}$$

since $1 + a_1 + 0 = (11)_2$ is impossible, the only possibility to examine is $1 + a_1 + 0 = (1)_2$.

$$1 + a_1 + 0 = (1)_2 \Rightarrow a_1 = 0 \quad \text{and} \quad r_2 = 0.$$

Step 2

$b_2 = 0$ since $b_1 = 1$.

$$b_2 = 0 \Rightarrow \begin{cases} r_2 + a_2 + a_1 = (0)_2 \\ r_2 + a_2 + a_1 = (10)_2 \end{cases} \Rightarrow \begin{cases} 0 + a_2 + 0 = (0)_2 \\ 0 + a_2 + 0 = (10)_2 \end{cases}$$

since $0 + a_2 + 0 = (10)_2$ is impossible, the only possibility to examine is $0 + a_2 + 0 = (0)_2$.

$$0 + a_2 + 0 = (0)_2 \Rightarrow a_2 = 0 \quad \text{and} \quad r_3 = 0.$$

Step 3

$b_3 = 1$ since $b_2 = 0$.

$$b_3 = 1 \Rightarrow \begin{cases} r_3 + a_3 + a_2 = (1)_2 \\ r_3 + a_3 + a_2 = (11)_2 \end{cases} \Rightarrow \begin{cases} 0 + a_3 + 0 = (1)_2 \\ 0 + a_3 + 0 = (11)_2 \end{cases}$$

since $0 + a_3 + 0 = (11)_2$ is not possible, the only possibility to examine is $0 + a_3 + 0 = (1)_2$.

$$0 + a_3 + 0 = (1)_2 \Rightarrow a_3 = 1 \quad \text{and} \quad r_4 = 0.$$

Step 4

$b_4 = 0$ since $b_3 = 1$.

$$b_4 = 0 \Rightarrow \begin{cases} r_4 + a_4 + a_3 = (0)_2 \\ r_4 + a_4 + a_3 = (10)_2 \end{cases} \Rightarrow \begin{cases} 0 + a_4 + 1 = (0)_2 \\ 0 + a_4 + 1 = (10)_2 \end{cases}$$

since $0 + a_4 + 1 = (0)_2$ is not possible, the only possibility to examine is $0 + a_4 + 1 = (10)_2$.

$$0 + a_4 + 1 = (10)_2 \Rightarrow a_4 = 1 \quad \text{and} \quad r_5 = 1.$$

Step 5

$b_5 = 1$ since $b_4 = 0$.

$$b_5 = 1 \Rightarrow \begin{cases} r_5 + a_5 + a_4 = (1)_2 \\ r_5 + a_5 + a_4 = (11)_2 \end{cases} \Rightarrow \begin{cases} 1 + a_5 + 1 = (1)_2 \\ 1 + a_5 + 1 = (11)_2 \end{cases}$$

since $1 + a_5 + 1 = (1)_2$ is impossible, the only possibility to examine is $1 + a_5 + 1 = (11)_2$.

$$1 + a_5 + 1 = (11)_2 \Rightarrow a_5 = 1 \quad \text{and} \quad r_6 = 1.$$

Step 6

$b_6 = 0$ since $b_5 = 1$.

$$b_6 = 0 \Rightarrow \begin{cases} r_6 + a_6 + a_5 = (0)_2 \\ r_6 + a_6 + a_5 = (10)_2 \end{cases} \Rightarrow \begin{cases} 1 + a_6 + 1 = (0)_2 \\ 1 + a_6 + 1 = (10)_2 \end{cases}$$

since $1 + a_6 + 1 = (0)_2$ is impossible, the only possibility to examine is $1 + a_6 + 1 = (10)_2$.

$$1 + a_6 + 1 = (10)_2 \Rightarrow a_6 = 0 \quad \text{and} \quad r_7 = 1.$$

Step 7

$b_7 = 1$ since $b_6 = 0$.

$$b_7 = 1 \Rightarrow \begin{cases} r_7 + a_7 + a_6 = (1)_2 \\ r_7 + a_7 + a_6 = (11)_2 \end{cases} \Rightarrow \begin{cases} 1 + a_7 + 0 = (1)_2 \\ 1 + a_7 + 0 = (11)_2 \end{cases}$$

since $1 + a_7 + 0 = (11)_2$ is impossible, the only possibility to examine is $1 + a_7 + 0 = (1)_2$.

$$1 + a_7 + 0 = (1)_2 \Rightarrow a_7 = 0 \quad \text{and} \quad r_8 = 0.$$

Step 8

$b_8 = 0$ since $b_7 = 1$.

$$b_8 = 0 \Rightarrow \begin{cases} r_8 + a_8 + a_7 = (0)_2 \\ r_8 + a_8 + a_7 = (10)_2 \end{cases} \Rightarrow \begin{cases} 0 + a_8 + 0 = (0)_2 \\ 0 + a_8 + 0 = (10)_2 \end{cases}$$

since $0 + a_8 + 0 = (10)_2$ is impossible, the only possibility to examine is $0 + a_8 + 0 = (0)_2$.

$$0 + a_8 + 0 = (0)_2 \Rightarrow a_8 = 0 \quad \text{and} \quad r_9 = 0.$$

Step 9

$b_9 = 1$ since $b_8 = 0$.

$$b_9 = 1 \Rightarrow \begin{cases} r_9 + a_9 + a_8 = (1)_2 \\ r_9 + a_9 + a_8 = (11)_2 \end{cases} \Rightarrow \begin{cases} 0 + a_9 + 0 = (1)_2 \\ 0 + a_9 + 0 = (11)_2 \end{cases}$$

since $0 + a_9 + 0 = (11)_2$ is impossible, the only possibility to examine is $0 + a_9 + 0 = (1)_2$.

$$0 + a_9 + 0 = (1)_2 \Rightarrow a_9 = 1 \quad \text{and} \quad r_{10} = 0.$$

For the first time, we obtain a step that is exactly identical to a previous step (Step 3).

Step 9:

$$r_9 = 0, \quad a_9 = 1, \quad r_{10} = 0, \quad b_9 = 1.$$

Step 3:

$$r_3 = 0, \quad a_3 = 1, \quad r_4 = 0, \quad b_3 = 1.$$

and r_{10}, r_4 are both real carries

Therefore, $(b_8\, b_7\, b_6\, b_5\, b_4\, b_3)_2$ is a binary pattern that will repeat with period $9 - 3 = 6$.

Since the period in the binary structure of the singularity is 2 (a divisor of 6), then the generator of the suffix of singularity is $((000111)^\omega 00)_2$, where $(00)_2$ are the least significant bits, and $(000111)_2$ is the binary pattern that repeats infinitely.

Case 2

$$m_0 \equiv 1 \pmod 2.$$

We consider the binary diagram of the operation $3m_0 + 2$, which is exactly the same diagram used in Case 1, except that in Case 2 there are two pseudo-carries $r_1 = 0$ and $r_2 = 1$.

Since $m_0 \equiv 1 \pmod 2$, we have $a_1 = 1$,

and since S_1 consists of a very large number of bits, $k \gg 1$.

Step 1

We have $b_1 = 1$ since it is the least significant bit of the suffix of singularity s_1.

$$b_1 = 1 \Rightarrow \begin{cases} r_1 + a_1 + 0 = (1)_2 \\ r_1 + a_1 + 0 = (11)_2 \end{cases} \Rightarrow \begin{cases} 0 + a_1 + 0 = (1)_2 \\ 0 + a_1 + 0 = (11)_2 \end{cases}$$

Since $0 + a_1 + 0 = (11)_2$ is impossible, the only possibility is

$$0 + a_1 + 0 = (1)_2.$$

$$0 + a_1 + 0 = (1)_2 \Rightarrow a_1 = 1.$$

And we have $r_2 = 1$ since r_2 is a pseudo-carry representing the $(+2)$ in the operation $3m_0 + 2$.

Step 2

We have $b_2 = 0$ since $b_1 = 1$.

$$b_2 = 0 \Rightarrow \begin{cases} r_2 + a_2 + a_1 = (0)_2 \\ r_2 + a_2 + a_1 = (10)_2 \end{cases} \Rightarrow \begin{cases} 1 + a_2 + 1 = (0)_2 \\ 1 + a_2 + 1 = (10)_2 \end{cases}$$

Since $1 + a_2 + 1 = (0)_2$ is impossible, the only possibility to examine is $1 + a_2 + 1 = (10)_2$.

$$1 + a_2 + 1 = (10)_2 \Rightarrow a_2 = 0 \text{ and } r_3 = 1.$$

Step 3

$b_3 = 1$ since $b_2 = 0$.

$$b_3 = 1 \Rightarrow \begin{cases} r_3 + a_3 + a_2 = (1)_2 \\ r_3 + a_3 + a_2 = (11)_2 \end{cases} \Rightarrow \begin{cases} 1 + a_3 + 0 = (1)_2 \\ 1 + a_3 + 0 = (11)_2 \end{cases}$$

Since $1 + a_3 + 0 = (11)_2$ is impossible, the only possibility to examine is $1 + a_3 + 0 = (1)_2$.

$$1 + a_3 + 0 = (1)_2 \Rightarrow a_3 = 0 \text{ and } r_4 = 0.$$

Step 4

$b_4 = 0$ since $b_3 = 1$.

$$b_4 = 0 \Rightarrow \begin{cases} r_4 + a_4 + a_3 = (0)_2 \\ r_4 + a_4 + a_3 = (10)_2 \end{cases} \Rightarrow \begin{cases} 0 + a_4 + 0 = (0)_2 \\ 0 + a_4 + 0 = (10)_2 \end{cases}$$

Since $0 + a_4 + 0 = (10)_2$ is impossible, the only possibility to examine is $0 + a_4 + 0 = (0)_2$.

$$0 + a_4 + 0 = (0)_2 \Rightarrow a_4 = 0 \text{ and } r_5 = 0.$$

Step 5

$b_5 = 1$ since $b_4 = 0$.

$$b_5 = 1 \Rightarrow \begin{cases} r_5 + a_5 + a_4 = (1)_2 \\ r_5 + a_5 + a_4 = (11)_2 \end{cases} \Rightarrow \begin{cases} 0 + a_5 + 0 = (1)_2 \\ 0 + a_5 + 0 = (11)_2 \end{cases}$$

Since $0 + a_5 + 0 = (11)_2$ is impossible, the only possibility to examine is $0 + a_5 + 0 = (1)_2$.

$$0 + a_5 + 0 = (1)_2 \Rightarrow a_5 = 1 \text{ and } r_6 = 0.$$

Step 6

$b_6 = 0$ since $b_5 = 1$.

$$b_6 = 0 \Rightarrow \begin{cases} r_6 + a_6 + a_5 = (0)_2 \\ r_6 + a_6 + a_5 = (10)_2 \end{cases} \Rightarrow \begin{cases} 0 + a_6 + 1 = (0)_2 \\ 0 + a_6 + 1 = (10)_2 \end{cases}$$

Since $0 + a_6 + 1 = (0)_2$ is impossible, the only possibility to examine is $0 + a_6 + 1 = (10)_2$.

$$0 + a_6 + 1 = (10)_2 \Rightarrow a_6 = 1 \text{ and } r_7 = 1.$$

Step 7

$b_7 = 1$ since $b_6 = 0$.

$$b_7 = 1 \Rightarrow \begin{cases} r_7 + a_7 + a_6 = (1)_2 \\ r_7 + a_7 + a_6 = (11)_2 \end{cases} \Rightarrow \begin{cases} 1 + a_7 + 1 = (1)_2 \\ 1 + a_7 + 1 = (11)_2 \end{cases}$$

Since $1 + a_7 + 1 = (1)_2$ is impossible, the only possibility to examine is $1 + a_7 + 1 = (11)_2$.

$$1 + a_7 + 1 = (11)_2 \Rightarrow a_7 = 1 \text{ and } r_8 = 1.$$

Step 8

$b_8 = 0$ since $b_7 = 1$.

$$b_8 = 0 \Rightarrow \begin{cases} r_8 + a_8 + a_7 = (0)_2 \\ r_8 + a_8 + a_7 = (10)_2 \end{cases} \Rightarrow \begin{cases} 1 + a_8 + 1 = (0)_2 \\ 1 + a_8 + 1 = (10)_2 \end{cases}$$

Since $1 + a_8 + 1 = (0)_2$ is impossible, the only possibility to examine is $1 + a_8 + 1 = (10)_2$.

$$1 + a_8 + 1 = (10)_2 \Rightarrow a_8 = 0 \text{ and } r_9 = 1.$$

Step 9

$b_9 = 1$ since $b_8 = 0$.

$$b_9 = 1 \Rightarrow \begin{cases} r_9 + a_9 + a_8 = (1)_2 \\ r_9 + a_9 + a_8 = (11)_2 \end{cases} \Rightarrow \begin{cases} 1 + a_9 + 0 = (1)_2 \\ 1 + a_9 + 0 = (11)_2 \end{cases}$$

Since $1 + a_9 + 0 = (11)_2$ is impossible, the only possibility to examine is $1 + a_9 + 0 = (1)_2$.

$$1 + a_9 + 0 = (1)_2 \Rightarrow a_9 = 0 \text{ and } r_{10} = 0.$$

For the first time, we have a step exactly identical to a previous step (Step 3).

Step 9: $r_9 = 1, \quad a_9 = 0, \quad r_{10} = 0, \quad b_9 = 1$

Step 3: $r_3 = 1, \quad a_3 = 0, \quad r_4 = 0, \quad b_3 = 1$ and r_{10}, r_4 are both real carries

Therefore, there will be a repetition of the binary pattern $(a_8, a_7, a_6, a_5, a_4, a_3)_2$ with a periodicity equal to $9 - 3 = 6$,

and since the periodicity in the binary structure of the singularity is 2 (a divisor of the period 6), therefore the generator of the suffix of singularity is

$$((011100)^\omega 01)_2$$

where $(01)_2$ are the least significant bits, and $(011100)_2$ is the binary pattern that repeats infinitely.

8.1.1 Remark

We have the generator of the suffix of singularity S_1 $((000111)^\omega 00)_2$ if $m_0 \equiv 0 \pmod 2$

and $((011100)^\omega 01)_2$ if $m_0 \equiv 1 \pmod 2$.

But we can express $((000111)^\omega 00)_2$ as: $((011100)^\omega)_2$.

And we can express the generator $((011100)^\omega 01)_2$ as $((110001)^\omega)_2$.

We call these two formulas the **variants** of the generator of the suffix of singularity.

The fact that one variant begins with a bit equal to 0 and the other variant begins with a bit equal to 1 is crucial, because it allows the first bit immediately following (on the left) the suffix of singularity to be integrated as the first bit of the generator suffix of singularity, for every term of the odd Syracuse sequence that is not itself a singularity.

9 Canonical binary decomposition of odd Syracuse terms

9.1 Overflow of the reduced odd Syracuse operation

Let $\mathcal{O}_n$ be a term of the odd Syracuse sequence that is not a singularity. Then we have $\mathcal{O}_n = m_n \parallel S_n$,

where S_n is a suffix of singularity and m_n is composed of at least two bits.

According to Lemma 5.1, we have

$$\mathcal{O}_{n+1} = \begin{cases} 3m_n + 1 & \text{if } m_n \equiv 0 \pmod 2, \\ 3m_n + 2 & \text{if } m_n \equiv 1 \pmod 2. \end{cases}$$

Let $m_n = (a_k a_{k-1} \cdots a_2 a_1)_2, \qquad k \geq 2,$

and let $\mathcal{O}_{n+1} = (b_{k+2} b_{k+1} \cdots b_2 b_1)_2$.

We call *overflow of the odd Syracuse operation* the appearance of the bits b_{k+2} and b_{k+1}, with $b_{k+2} = 0 \Rightarrow b_{k+1} = 1$.

Possible values for $(b_{k+1} b_{k+2})_2$

We have the following binary operation $3m_n + d, \qquad d \in \{1, 2\}$,

with $r_1 = 1$ the pseudo-carry if $d = 1$, and $r_1 = 0$, $r_2 = 1$ the pseudo-carries if $d = 2$.

$$\begin{array}{cccccccc}
 & & r_{k+1} & r_k & r_{k-1} & \dots & r_2 & r_1 \\
+ & & & a_k & a_{k-1} & \dots & a_2 & a_1 \\
+ & & a_k & a_{k-1} & \dots & a_2 & a_1 & 0 \\
\hline
= & b_{k+2} & b_{k+1} & b_k & b_{k-1} & \dots & b_2 & b_1
\end{array}$$

r_k	a_{k-1}	r_{k+1}	b_{k+1}	b_{k+2}
0	0	0	1	0
1	0	1	0	1
0	1	1	0	1
1	1	1	0	1

Therefore, there is either an overflow of one bit, $(1)_2$, or an overflow of two bits, $(10)_2$.

9.2 Binary structural conditions on $\mathcal{O}_n$ so that $\mathcal{O}_{n+1}$ will be a singularity

Let $\mathcal{O}_n$ be a term of the odd Syracuse sequence that is not a singularity. Thus

$$\mathcal{O}_n = m_n \parallel S_n,$$

where S_n is a suffix of singularity and m_n is composed of at least one bit.

According to Remark 8.1.1, m_n has a suffix generator of a suffix of singularity G_n. Therefore

$$m_n = R_n \parallel G_n,$$

where R_n is the remainder of the binary structure of $\mathcal{O}_n$.

We know that

$$\mathcal{O}_{n+1} = f(m_n) = \begin{cases} 3m_n + 1 & \text{if } m_n \equiv 0 \pmod 2, \\ 3m_n + 2 & \text{if } m_n \equiv 1 \pmod 2. \end{cases}$$

We know that G_n, under the operation of f, will produce the suffix of singularity of $\mathcal{O}_{n+1}$, and R_n will produce the suffix of the generator of the suffix of singularity G_{n+1}.

Since $G_{n+1} \parallel S_{n+1}$ is not a suffix of singularity, then $\mathcal{O}_{n+1}$ is not a singularity.

Therefore, for $\mathcal{O}_{n+1}$ to be a singularity, it is necessary that m_n be composed only of G_n, that is,

$$\mathcal{O}_n = m_n \parallel S_n = G_n \parallel S_n.$$

If it is the case, then since the function f produces an overflow during the iteration $\mathcal{O}_{n+1} = f(G_n)$.

Let $G_n = (a_k a_{k-1} \cdots a_2 a_1)_2$, then
$\mathcal{O}_{n+1} = (b_{k+2} b_{k+1} \cdots b_2 b_1)_2$.
The overflow of f is b_{k+1}, b_{k+2}, with $b_{k+2} = 0 \Rightarrow b_{k+1} = 1$.
According to Section 9.1, the possible values of $(b_{k+2} b_{k+1})_2$ are $(1)_2$ and $(10)_2$.
Therefore, for $\mathcal{O}_{n+1}$ to be a singularity, the overflow of f must be complementary to the suffix of singularity of $\mathcal{O}_{n+1}$, whose binary structure is

$$(b_k b_{k-1} \cdots b_2 b_1)_2.$$

In other words:

- if the suffix of singularity ends with the bit $b_k = 0$, then the overflow must be $(b_{k+1})_2 = (1)_2$;
- if $b_k = 1$, then the overflow must be $(b_{k+2} b_{k+1})_2 = (10)_2$.

9.3 Lemma

If the odd syracuse term $\mathcal{O}_n$ is not a singularity, then the term $\mathcal{O}_{n+1}$ is a singularity if and only if

- $\mathcal{O}_n = G_n \parallel S_n$, where G_n is the suffix of the generator of the suffix of the singularity;
- the overflow of $f(G_n) = \mathcal{O}_{n+1}$ is complementary to the suffix of singularity of $\mathcal{O}_{n+1}$ produced by G_n, that is:
  - either the suffix of singularity of $\mathcal{O}_{n+1}$ ends with 0 and the overflow is $(1)_2$;
  - or the suffix of singularity of $\mathcal{O}_{n+1}$ ends with 1 and the overflow is $(10)_2$.

9.4 Decomposition of $\mathcal{O}_n$ into suffix of generator of suffix of singularity and one suffix of singularity

Here we will prove that, for any term $\mathcal{O}_n$ of the odd Syracuse sequence, if $\mathcal{O}_n$ is not a singularity, then $\mathcal{O}_n$ can be decomposed into suffix of the generator of the suffix of singularity and one suffix of singularity, namely

$$\mathcal{O}_n = G_p \parallel G_{p-1} \parallel \cdots \parallel G_2 \parallel G_1 \parallel S, \qquad p \geq 1.$$

A Delphi program *Decomposition* that shows the decomposition of the binary structure of odd integers is provided with this preprint as supplementary material

Proof

Let $\mathcal{O}_1$ be a term of the odd Syracuse sequence that is not a singularity.

If $\mathcal{O}_2$ is a singularity, then according to Lemma 9.3, $\mathcal{O}_1$ is composed of a suffix of singularity and a single suffix of a generator of suffix of singularity. Formally, $\mathcal{O}_1 = G_1 \parallel S_1$, with $\mathcal{O}_2 = f_1(G_1)$

generating an overflow complementary to the suffix of the singularity S_2 of $\mathcal{O}_2$.

In this case we have

$$\mathcal{O}_1 = G_p \parallel G_{p-1} \parallel \cdots \parallel G_2 \parallel G_1 \parallel S_1, \qquad p \geq 1,$$

therefore $\mathcal{O}_1$ satisfies the decomposition rule.

If, on the contrary, $\mathcal{O}_2$ is not a singularity, two cases arise.

Case 1 $f_1(G_1) = \mathcal{O}_2$

The operation does not produce an overflow complementary to the singularity suffix of $\mathcal{O}_2$.

Case 2 $\mathcal{O}_1$ is not composed only of a suffix of singularity and a single suffix of the generator of the suffix of singularity.

Instead, $\mathcal{O}_1$ contains R_1, which is the remainder of the binary structure of $\mathcal{O}_1$ apart from G_1 and S_1. Thus we write $\mathcal{O}_1 = R_1 \parallel G_1 \parallel S_1$,

where R_1 contains at least one bit.

To avoid confusion, we index the suffix of the generator of the suffix of singularity appearing in the binary structure of $\mathcal{O}_i$ as follows:

$$G_{(\text{position},i)},$$

where position denotes the position of G among all the suffixes of the generators contained in $\mathcal{O}_i$, counting from right to left, and i is the index of $\mathcal{O}_i$.

Using this notation, we obtain $\mathcal{O}_2 = f_1(R_1 \parallel G_{(1,1)}) = R_2 \parallel G_{(1,2)} \parallel S_2$.

According to Lemma 7, $G_{(1,1)}$ generates S_2, and R_1 generates $R_2 \parallel G_{(1,2)}$.

To avoid confusion, we also introduce a new notation for the remainder part of a term $\mathcal{O}_i$:

$$R_{(\text{position},i)},$$

where position is the position of the remainder R among all the parts of the binary structure of $\mathcal{O}_i$, counting from right to left beginning with the first suffix of the generator of the suffix of singularity $G_{(1,i)}$.

According to Lemma 7, the least significant bits of R_1 contain the bits that generated $G_{(1,2)}$. We denote these bits by $G_{(2,1)}$.

Hence $\mathcal{O}_1 = R_1 \parallel G_{(1,1)} \parallel S_1 = R_{(3,1)} \parallel G_{(2,1)} \parallel G_{(1,1)} \parallel S_1$.

Furthermore, $\mathcal{O}_2 = f_1\big(R_{(3,1)} \parallel R_{(2,1)} \parallel G_{(1,1)}\big) = R_{(2,2)} \parallel G_{(1,2)} \parallel S_2$.

It should also be noted that the most significant bits of $R_{(2,2)}$ contain the overflow bits.

9.4.1 Definition of the Function L

We define the function L that associates with every binary structure A the number of bits that constitute the structure A:

$$A \longmapsto L(A).$$

Unlike the function that associates with a binary structure, the minimum number of bits required to represent a binary number having the value of A, the function L does not take the value of A into consideration.

Example. If $A = (0101)_2$, then $L(A) = 4$.

We have if $L(R_{(3,1)}) \geq 1$, then $R_{(2,2)} \geq 1$, with

$$L(R_{(3,1)}) = L(R_{(2,1)}) - L(G_{(1,2)}),$$

since $G_{(1,2)}$ originates from the least significant bits of $R_{(2,1)} = R_1$.

Therefore,

$$L(R_{(3,1)}) \geq 1 \Rightarrow \mathcal{O}_3 = f_2\big(R_{(2,2)} \parallel G_{(1,2)}\big) = R_{(2,3)} \parallel G_{(1,3)} \parallel S_3$$

Is not a singularity.

S_3 originates from $G_{(1,2)}$, and $R_{(2,3)} \parallel G_{(1,3)} = f(R_{(2,2)})$.

Since $R_{(2,2)}$ originates from $R_{(3,1)}$, we may decompose $R_{(3,1)} = R_{(4,1)} \parallel G_{(3,1)}$ such that

$$L(G_{(3,1)}) = L(G_{(1,2)}) = L(G_{(1,3)}).$$

Hence

$\mathcal{O}_1 = R_{(4,1)} \parallel G_{(3,1)} \parallel G_{(2,1)} \parallel G_{(1,1)} \parallel S_1$. therefore, in general, if $L(R_{(i,1)}) \geq 1$, then $\mathcal{O}_{i+1}$ is not a singularity and $R_{(i,1)}$ can be decomposed as

$$R_{(i+1,1)} \parallel G_{(i,1)}.$$

And we have $\mathcal{O}_1 = R_{(i+1,1)} \parallel G_{(i,1)} \parallel \cdots \parallel G_{(2,1)} \parallel G_{(1,1)} \parallel S_1$.

However, since our objective is to prove that the binary structure of $\mathcal{O}_1$ can be decomposed without involving the overflow bits that are produced, we must add the following constraint:

$$\sum_{j=2}^{i} L(G_{(j,1)}) + L(R_{(i+1,1)}) \leq L(R_1).$$

By continuing the decomposition process, we necessarily reach

$$\mathcal{O}_1 = R_{(i+1,1)} \parallel G_{(i,1)} \parallel \cdots \parallel G_{(2,1)} \parallel G_{(1,1)} \parallel S_1,$$

with

$$\sum_{j=2}^{i} L(G_{(j,1)}) + L(R_{(i+1,1)}) \leq L(R_1),$$

and $\mathcal{O}_1 = R_{(i+2,1)} \parallel G_{(i+1,1)} \parallel \cdots \parallel G_{(2,1)} \parallel G_{(1,1)} \parallel S_1$,
with $\sum_{j=2}^{i+1} L(G_{(j,1)}) + L(R_{(i+2,1)}) > L(R_1)$.

Case 1

$$\sum_{j=2}^{i} L\big(G_{(j,1)}\big) + L\big(R_{(i+1,1)}\big) = L(R_1)$$

and

$$\sum_{j=2}^{i+1} L\big(G_{(j,1)}\big) + L\big(R_{(i+2,1)}\big) > L(R_1).$$

This occurs when the operation $f_1\big(R_{(2,1)} \parallel G_{(1,1)}\big) = \mathcal{O}_2$ produces an overflow that is not complementary to $G_{(1,2)}$.

We then obtain the decomposition of $\mathcal{O}_1$ as follows:

$$\mathcal{O}_1 = G_{(i+1,1)} \parallel G_{(i,1)} \parallel \cdots \parallel G_{(2,1)} \parallel G_{(1,1)} \parallel S_1$$

where $G_{(i+1,1)}$ is derived from $R_{(i+1,1)}$.

Case 2

$$\sum_{j=2}^{i} L\big(G_{(j,1)}\big) + L\big(R_{(i+1,1)}\big) < L(R_1)$$

and

$$\sum_{j=2}^{i+1} L\big(G_{(j,1)}\big) + L\big(R_{(i+2,1)}\big) > L(R_1).$$

This occurs when the operation $f_1\big(R_{(2,1)} \parallel G_{(1,1)}\big) = \mathcal{O}_2$ produces an overflow whose one bit or both bits are complementary to $G_{(1,2)}$.

We have that $R_{(i+2,1)}$ contains the most significant bits of $\mathcal{O}_1$ within its least significant bits, and the overflow bits of $\mathcal{O}_2 = f\big(R_{(2,1)} \parallel G_{(1,1)}\big)$ which complete $R'_{(i+2,1)}$ to form $R_{(i+2,1)}$.

In this case we have

$$\mathcal{O}_1 = G'_{(i+2,1)} \parallel G_{(i+1,1)} \parallel \cdots \parallel G_{(2,1)} \parallel G_{(1,1)} \parallel S_1$$

where $G'_{(i+2,1)}$ is derived from $R'_{(i+2,1)}$.

Therefore, in both cases we obtain a complete decomposition of $\mathcal{O}_1$ in the form

$$\mathcal{O}_1 = G_{(p,1)} \parallel G_{(p-1,1)} \parallel \cdots \parallel G_{(2,1)} \parallel G_{(1,1)} \parallel S_1, \qquad p \geq 2.$$

However, we have already proven that this decomposition also works for $p = 1$ (in case where $\mathcal{O}_2$ is a singularity term).

Therefore, for any term $\mathcal{O}_n$ of the odd Syracuse sequence, if $\mathcal{O}_n$ is not a singularity, then $\mathcal{O}_n$ admits a decomposition into a single suffix of singularity and suffix of the generators of the suffix of singularity.

9.4.2 Uniqueness of the Decomposition

The decomposition of odd Syracuse terms $\mathcal{O}_n \in 2\mathbb{N}+1$ that are not singularities is based on the suffix of singularity and the suffixes of generators of the suffix of singularity. These two mathematical concepts are suffixes of binary structures.

According to the definition of a suffix of a binary structure given in Section 1, for every term $\mathcal{O}_n \in 2\mathbb{N}+1$ that is not a singularity, the singularity suffix and the suffixes of generators of the suffix of singularity forming the decomposition of $\mathcal{O}_n$ are unique.

By induction, at each step the extracted suffix is unique according to the definition of a suffix of a binary structure in section 2; therefore the remaining part is unique. Hence, the complete decomposition is unique.

9.5 Lemma

Let $\mathcal{O}_n$ be any term of the odd Syracuse sequence, $\mathcal{O}_n \in 2\mathbb{N}+1$.

If $\mathcal{O}_n$ is not a singularity, then the binary structure of $\mathcal{O}_n$ can be uniquely decomposed as $\mathcal{O}_n = G_{(p,n)} \parallel G_{(p-1,n)} \parallel \cdots \parallel G_{(2,n)} \parallel G_{(1,n)} \parallel S_n, \qquad p \geq 1.$

where

- S_n is the suffix of singularity;
- for every $i \in \{1, 2, \ldots, p\}$, $G_{(i,n)}$ is a suffix of the generator of the suffix of singularity.

such that

For $p > 1$ we have: For $i = 1$, $G_{(i,n)}$ generates S_{n+1}
with $f_i(G_{(i,n)}) = 3G_{(i,n)} + d_i, \qquad d_i \in \{1, 2\}$.
For $i > 1$, $G_{(i,n)}$ generates $G_{(i-1,n+1)}$
with

$$f_i(G_{(i,n)}) = 3G_{(i,n)} + d_i, \qquad d_i \in \{0, 1, 2\}.$$

If $i = p$ and there is an overflow complementary to $G_{(i-1,n+1)}$, then

$$G_{(i,n)} \text{ generates } G'_{(i-1,n+1)}$$

with $f_i(G_{(i,n)}) = 3G_{(i,n)} + d_i, \qquad d_i \in \{0, 1, 2\}$.

For $p = 1$ we have: $G_{(i,n)}$ generates S_{n+1} with

$$f_i(G_{(i,n)}) = 3G_{(i,n)} + d_i, \qquad d_i \in \{1, 2\}.$$

If thre is an overflow complementary to S_{n+1}, then $G_{(i,n)}$ generates S'_{n+1} with $f_i(G_{(i,n)}) = 3G_{(i,n)} + d_i, \qquad d_i \in \{1, 2\}$.

9.6 Growth and Decrease of the function P

According to Lemma 9.5, for any term $\mathcal{O}_n$ of the odd Syracuse sequence, if $\mathcal{O}_n$ is not a singularity, then $\mathcal{O}_n$ can be represented by the unique decomposition

$$\mathcal{O}_n = G_{(p,n)}\|G_{(p-1,n)}\| \cdots \|G_{(2,n)}\|G_{(1,n)}\|S_n, \qquad p \geq 1.$$

We define the function assigning to every odd Syracuse term $\mathcal{O}_n$ of the sequence starting from $\mathcal{O}_0$

the number of Suffixes of the Generators of the Suffix of Singularity contained in the decomposition of $\mathcal{O}_n$:

$P(\mathcal{O}_n) =$ the number of blocks $G_{(i,n)}$ in the decomposition of $\mathcal{O}_n$ with $P(\mathcal{O}_n) \in \mathbb{N}$.

We have $\mathcal{O}_{n+1} = f_n\Big(G_{(p,n)}\|G_{(p-1,n)}\| \cdots \|G_{(2,n)}\|G_{(1,n)}\Big)$ and six cases arise.

Case 1 The operation f_n produces a two-bit overflow, both bits being complementary to $G_{(p-1,n+1)}$ when $p > 1$, and complementary to S_{n+1} when $p = 1$.

This yields

$$\mathcal{O}_{n+1} = G'_{(p-1,n+1)}\|G_{(p-2,n+1)}\| \cdots \|G_{(2,n+1)}\|G_{(1,n+1)}\|S_{n+1}$$

for $p > 1$,

where $G'_{(p-1,n+1)}$ is the result of the complementarity operation between the overflow bits and $G_{(p-1,n+1)}$.

For $p = 1$, $\mathcal{O}_{n+1} = S'_{n+1}$,

where S'_{n+1} is the result of the complementarity operation between the overflow bits and S_{n+1}.

Therefore, $P(\mathcal{O}_{n+1}) = P(\mathcal{O}_n) - 1$.

Case 2 The operation f_n produces a two-bit overflow, but only the bit immediately to the left of $G_{(p-1,n+1)}$ is complementary, or the bit immediately to the left of S_{n+1} when $p = 1$.

Then

$$\mathcal{O}_{n+1} = G_{(p,n+1)} \| G'_{(p-1,n+1)} \| \cdots \| G_{(2,n+1)} \| G_{(1,n+1)} \| S_{n+1}$$

for $p > 1$,

where $G'_{(p-1,n+1)}$ is the result of the complementarity operation between the overflow bit and $G_{(p-1,n+1)}$.

For $p = 1$, $\mathcal{O}_{n+1} = G_{(1,n+1)} \| S'_{n+1}$,

where S'_{n+1} is the result of the complementarity operation between the overflow bit and S_{n+1}.

Therefore, $P(\mathcal{O}_{n+1}) = P(\mathcal{O}_n)$.

Case 3

The operation f_n produces a two-bit overflow that is not complementary to $G_{(P-1,n+1)}$ for $p > 1$, and to S_{n+1} in the case $p = 1$. In addition, the two overflow bits each form a new suffix of a generator of the suffix of singularity. Thus, we have

$$\mathcal{O}_{n+1} = G_{(P+1,n+1)} \parallel G_{(P,n+1)} \parallel G_{(P-1,n+1)} \parallel \cdots \parallel G_{(2,n+1)} \parallel G_{(1,n+1)} \parallel S_{n+1},\ p > 1,$$

and $\mathcal{O}_{n+1} = G_{(2,n+1)} \parallel G_{(1,n+1)} \parallel S_{n+1}, \qquad p = 1.$

Therefore, $P(\mathcal{O}_{n+1}) = P(\mathcal{O}_n) + 1$.

Case 4

The operation f_n produces a two-bit overflow that is not complementary to $G_{(P-1,n+1)}$ in the case $p > 1$, or to S_{n+1} in the case $p = 1$, but the two overflow bits form only one new suffix of a generator of the suffix of singularity.

Thus,

$$\mathcal{O}_{n+1} = G_{(P,n+1)} \parallel G_{(P-1,n+1)} \parallel \cdots \parallel G_{(2,n+1)} \parallel G_{(1,n+1)} \parallel S_{n+1}, \qquad p > 1,$$

or $\mathcal{O}_{n+1} = G_{(1,n+1)} \parallel S_{n+1}, \qquad p = 1.$, Therefore, $P(\mathcal{O}_{n+1}) = P(\mathcal{O}_n)$.

Case 5

The operation f_n produces a one-bit overflow that is not complementary to $G_{(P-1,n+1)}$ for $p > 1$, or to S_{n+1} for $p = 1$.

Thus,

$$\mathcal{O}_{n+1} = G_{(P,n+1)} \parallel G_{(P-1,n+1)} \parallel \cdots \parallel G_{(2,n+1)} \parallel G_{(1,n+1)} \parallel S_{n+1}, \qquad p > 1,$$

and $\mathcal{O}_{n+1} = G_{(1,n+1)} \parallel S_{n+1}, \qquad p = 1.$, Therefore, $P(\mathcal{O}_{n+1}) = P(\mathcal{O}_n)$.

Case 6

The operation f_n produces a one-bit overflow that is complementary to $G_{(P-1,n+1)}$ for $p > 1$, or to S_{n+1} for $p = 1$.

Thus,

$$\mathcal{O}_{n+1} = G'_{(P-1,n+1)} \parallel G_{(P-2,n+1)} \parallel \cdots \parallel G_{(2,n+1)} \parallel G_{(1,n+1)} \parallel S_{n+1}, \qquad p > 1,$$

where $G'_{(P-1,n+1)}$ is the result of the complementarity between the overflow bit and $G_{(P-1,n+1)}$.

Or, in the case $p = 1$, $\mathcal{O}_{n+1} = S'_{n+1}$,

where S'_{n+1} is the result of the complementarity between the overflow bit and S_{n+1}.

Therefore, $P(\mathcal{O}_{n+1}) = P(\mathcal{O}_n) - 1$.

Hence, from the study of these six possible cases, we deduce that after each operation of the Odd Syracuse sequence, the number of generators of the suffix of singularity may decrease, increase, or remain unchanged. Whenever a decrease occurs, it is always equal to -1; whenever an increase occurs, it is always equal to $+1$.

Lemma 9.7

Let $\mathcal{O}_0$ be a term of the odd Syracuse sequence that is not a singularity that contains $P(\mathcal{O}_0) \geq 1$ suffixes of generators of the suffix of singularity.

If the number of suffixes of generators of the suffix of singularity reaches for the first time a value $P(\mathcal{O}_s)$ at the term $\mathcal{O}_s$, such that

$$\forall i \in \{0, 1, \ldots, s-1\}, \qquad P(\mathcal{O}_i) > P(\mathcal{O}_s),$$

then $P(\mathcal{O}_{s-1}) = P(\mathcal{O}_s) + 1$.

And if the number of suffixes of generators of the suffix of singularity reaches for the first time a value $P(\mathcal{O}_g)$ at the term $\mathcal{O}_g$, such that

$$\forall i \in \{0, 1, \ldots, g-1\}, \qquad P(\mathcal{O}_i) < P(\mathcal{O}_g),$$

then $P(\mathcal{O}_{g-1}) = P(\mathcal{O}_g) - 1$.

Lemma 9.8

The Collatz conjecture holds if and only if every Odd Syracuse term $\mathcal{O}_n \in 2\mathbb{N}+1$ is either a singularity, that is, the number of suffixes of the generators of suffixe of the singularity is: $P(\mathcal{O}_n) = 0$. Or, $\mathcal{O}_n$ is eventually transformed, after a finite number of iterations, into a singularity. Formally, this can be written as $P(\mathcal{O}_n) > 0 \quad \Longrightarrow \quad \exists\, i > 0 \ : \ P(\mathcal{O}_{n+i}) = 0$.

9.8.1 Remark

Lemma 9.8 is directly derived from Theorem 5.11.4 in our work [1]

10 The Function P as an Eventual Lyapunov Function

The function P is an eventual Lyapunov function if and only if $\forall O_n \in 2\mathbb{N}+1, \quad$ then $P(\mathcal{O}_n) > 0 \quad \Longrightarrow \quad \exists\, k \geq 1$ such that $P(\mathcal{O}_{n+k}) < P(\mathcal{O}_n)$.

Let $\mathcal{O}_0$ be a term of the odd Syracuse sequence that is not a singularity such that $P(\mathcal{O}_0) = 1.$, we will prove that

$P(\mathcal{O}_0) = 1 \Rightarrow \exists i > 0 \ ; \ P(\mathcal{O}_i) = 0$, in other words, $\mathcal{O}_i$ is a singularity.

Proof

According to Lemma 9.5, we have

$\mathcal{O}_0 = G_{(p,0)} \| G_{(p-1,0)} \| \cdots \| G_{(2,0)} \| G_{(1,0)} \| S_0$ with $p = 1$.

Hence, $\mathcal{O}_0 = G_{(1,0)} \| S_0$.

According to Lemma 7, $G_{(1,0)}$ generates S_1 if there is no overflow complementary to S_1. If the overflow is complementary to S_1, then $G_{(1,0)}$ generates S_1'.

However, according to Section 9.1, the overflow of the operation

$f_0(G_{(1,0)}) = \mathcal{O}_1$ with $f_0(G_{(1,0)}) = 3G_{(1,0)} + d_0, \qquad d_0 \in \{1, 2\}$,

produces the two possible overflows $(1)_2$ or $(10)_2$.

We observe that for the overflow $(10)_2$,if the bit $(0)_2$ is complementary to S_1, then since S_1 is a suffix of singularity consisting of alternating bits $\{0,1\}$ all $(10)_2$ binary pattern is complimentary, in other words:

$$f_0(G_{(1,0)}) = S_1' \Rightarrow P(O_1) = 0.$$

Therefore $\mathcal{O}_1$ is a singularity.

Thus we shall study the cases where $f_0(G_{(1,0)})$ produces a non-complementary overflow.

10.1 Property

$G_{(1,0)}$ is a suffix of the generator of the suffix of singularity, and the generator of the suffix of singularity has two variants: $((011100)^{\omega}01)_2$ if the most significant bit of S_0 is equal to 1, $((000111)^{\omega}00)_2$, if the least significant bit of S_0 is equal to 0.

10.2 Property

The most significant bit **MSB** of $G_{(1,0)}$ is always equal to 1.

10.3 Property

$L(G_{(1,0)}) \equiv 1 \pmod 2 \Rightarrow$ the most significant bit of S_1 is equal to 1

$L(G_{(1,0)}) \equiv 0 \pmod 2 \Rightarrow$ the most significant bit of S_1 is equal to 0.

10.4 Property

The two variants of the generator of the suffix of singularity have a periodicity of 6 bits.

10.5 Possible values of $L(G_{(1,0)})$

a- If the most significant bit of S_0 is equal to 1, then $G_{(1,0)}$ is a suffix of the generator $((011100)^{\omega}01)_2$,, and since $G_{(1,0)}$ must end with a bit equal to 1 (Property 10.2), we have the following possible cases:

$$L(G_{(1,0)}) = 1$$

$$L(G_{(1,0)}) = 5 + 6i, \qquad i \in \mathbb{N}$$

$L(G_{(1,0)}) = 6 + 6i, \qquad i \in \mathbb{N}$ hence $L(G_{(1,0)}) = 6i, \qquad i \in \mathbb{N}^*$
$L(G_{(1,0)}) = 7 + 6i, \qquad i \in \mathbb{N}$ hence $L(G_{(1,0)}) = 1 + 6i, \qquad i \in \mathbb{N}^*$

10.6 Parity of the most significant bit of S_1 generated by $G_{(1,0)}$

We know that $L(G_{(1,0)}) = L(S_1)$ According to Lemma 7.

Note carefully that $S_1 \neq f_0(G_{(1,0)})$ in the case where there is a complementary overflow.

Since S_1 is composed of alternating bits with period 2, we have:

- If $L(G_{(1,0)}) = 1$, then S_1 ends with a 1.
- If $L(G_{(1,0)}) = 5 + 6i, \qquad i \in \mathbb{N}$,

  and since the period of the generator suffix of singularity is a multiple of the period of S_1, then for $L(G_{(1,0)}) = 5$, we have $S_1 = (10101)_2$,

  therefore S_1 ends with a 1.

  Hence $L(G_{(1,0)}) = 5 + 6i, \qquad i \in \mathbb{N}$, implies that S_1 ends with a 1.
- If

  $$L(G_{(1,0)}) = 6i, \qquad i \in \mathbb{N}^*,$$

  First, if $L(G_{(1,0)}) = 6$, then $S_1 = (010101)_2$.

  Therefore S_1 ends with a 0.

  Hence $L(G_{(1,0)}) = 6i, \qquad i \in \mathbb{N}^*$ implies that S_1 ends with a 0.
- If

  $$L(G_{(1,0)}) = 1 + 6i, \qquad i \in \mathbb{N}^*,$$

  First, if $L(G_{(1,0)}) = 1$, then $S_1 = (1)_2$,

  thus S_1 ends with a 1.

  Hence $L(G_{(1,0)}) = 1 + 6i, \qquad i \in \mathbb{N}^*$ implies that S_1 ends with a 1.

b- If the most significant bit of S_0 is equal to 0, then $G_{(1,0)}$ is a suffix of the generator $((000111)^\omega 00)_2$, and since $G_{(1,0)}$ must end with a 1 (Property 10.2), we have the following possible cases:

$$L(G_{(1,0)}) = 3 + 6i, \qquad i \in \mathbb{N}$$

$$L(G_{(1,0)}) = 4 + 6i, \qquad i \in \mathbb{N}$$

$$L(G_{(1,0)}) = 5 + 6i, \qquad i \in \mathbb{N}.$$

10.7 Parity of the most significant bit of S_1 generated by $G_{(1,0)}$

We know that

$L(G_{(1,0)}) = L(S_1)$ according to Lemma 7.

Note carefully that $S_1 \neq f(G_{(1,0)})$

In the case where there is a complementary overflow, and since S_1 is composed of alternating bits with a period of 2 bits, then:

- If $L(G_{(1,0)}) = 3 + 6i, \qquad i \in \mathbb{N}$ then S_1 ends with a 1.
- If $L(G_{(1,0)}) = 4 + 6i, \qquad i \in \mathbb{N}$ then S_1 ends with a 0.
- If $L(G_{(1,0)}) = 5 + 6i, \qquad i \in \mathbb{N}$ then S_1 ends with a 1.

10.8 Proof that the odd Syracuse sequence reaches a singularity when $P(\mathcal{O}_0) = 1$

10.8.1 $L(G_{(1,0)}) = 1$

If $L(G_{(1,0)}) = 1$ then we have $G_{(1,0)} = (1)_2$.

Since $L(G_{(1,0)}) = 1$, then $G_{(1,0)} = (1)_2$, therefore $G_{(1,0)}$ can only be the suffix of the generator $((011100)^\omega 01)_2$.

We have $\mathcal{O}_1 = f_1(G_{(1,0)}) = 3G_{(1,10)} + 2$ according to Section 5.1.

Therefore

$\mathcal{O}_1 = 2 \cdot (1)_2 + (1)_2 + 2 = (10)_2 + (11)_2 = (101)_2$, which is a singularity.

We have

$$S_1 = (1)_2,$$

The overflow is $(10)_2$, which is complementary to S_1, and forms $S_1' = (101)_2$, which is a singularity.

10.8.2 $L(G_{(1,0)}) > 1$

Let $A = (a_k\, a_{k-1}\, \ldots\, a_2\, a_1)_2$ be a binary structure undergoing the operation $3A + d, \qquad d \in \{0, 1, 2\}$. we will determine the iteration undergone by the binary structure $B = (a_k\, a_{k-1}\, \ldots\, a_{j+1}\, a_j)_2, \qquad j > 1$, because in Section 5.1 we proved a special case where $\mathcal{O}_n = m \parallel S_n$ and $\mathcal{O}_{n+1} = f_n(m)$
with $f_n(m) = 3m + d, \qquad d \in \{1, 2\}$.

Case 1

A undergoes the operation $3A + d, \qquad d = 0$.

		r_{k+1}	r_k	r_{k-1}	$\ldots$	r_j	r_{j-1}	$\ldots$	r_2	r_1
$+$			a_k	a_{k-1}	$\ldots$	a_j	a_{j-1}	$\ldots$	a_2	a_1
$+$		a_k	a_{k-1}	$\ldots$	a_j	a_{j-1}	$\ldots$	a_2	a_1	0
$=$	b_{k+2}	b_{k+1}	b_k	b_{k-1}	$\ldots$	b_j	$b_{j-1} \ldots$		b_2	b_1

Therefore there is only one pseudo-carry: $r_1 = 0$.
Let r_i be the first carry obtained during the operation $3A + 0$, with $i \leq j$.
We know that r_i is obtained during the addition $r_{i-1} + a_{i-1} + a_{i-2}$,
and since r_i is the first carry obtained, then $r_{i-1} = 0$.
Therefore, the only way to obtain $r_i = 1$ is that $a_{i-1} = a_{i-2} = 1$.

Conclusion

For there to be a carry r_i, with $i \leq j$, during the operation $3A + 0$, there must exist two or more consecutive bits equal to 1, namely

$$a_{i-1} = a_{i-2} = 1, \qquad i \leq j.$$

If $i = j$

Then we have the first carry $r_i = r_j = 1$ and $a_{j-1} = a_{j-2} = 1$. Since $a_{j-1} = 1$ and $r_j = 1$ imply that $r_{j+1} = 1$, then by transferring the two values $r_j = 1$ and $a_{j-1} = 1$, we form the two pseudo-carries $r_j = 0$ and $r_{j+1} = 1$.

Therefore B undergoes the operation $3B + 2$, where the term $(+2)$ represents the pseudo-carry $r_{j+1} = 1$.

Now we will determine the propagation conditions of the last carry resulting from the last two consecutive bits equal to 1, and forming either r_j or r_{j+1} during the operation $3A + 0$.

Let the last carry produced be r_i, by the addition $r_{i-1} + a_{i-1} + a_{i-2}$, with $r_{i-1} = 0$ and $a_{i-1} = a_{i-2} = 1$, with $i < j$.

According to Section 2.3.1 in our fifth work [3], the binary structure $(a_{j-1}\, a_{j-2}\, \ldots\, a_{i+1}\, a_i)_2$ must not contain two or more consecutive bits equal to 0 to alow the propagation of r_i.

But since a_{i-1}, a_{i-2} are the last consecutive bits equal to 1, then the binary structure $(a_{j-1}\, a_{j-2}\, \ldots\, a_{i+1}\, a_i)_2$ must not contain two or more consecutive bits equal to 1.

Therefore this structure must be exactly an alternation of bits from the set $\{0, 1\}$, with $a_i = 0$.

Hence, if $(a_{j-1}\, a_{j-2}\, \ldots\, a_{i+1}\, a_i)_2$ is an alternation of bits, then $r_j = 1$, and we have two cases.

a) : $a_{j-1} = 0$ Then we have $r_j = 1$, and it is the only pseudo-carry in the operation undergone by B, and it represents the term $(+1)$ in the operation

$$3B + 1.$$

b) : $a_{j-1} = 1$ We transfer the values $a_{j-1} = 1$ and $r_j = 1$ to form $r_j = 0$ and $r_{j+1} = 1$ which will be the two pseudo-carries of the operation undergone by B, namely $3B + 2$, and the term $(+2)$ represents the pseudo-carry $r_{j+1} = 1$.

Case 2

A undergoes the operation $3A + d$, with $d = 1$, thus, there is a single pseudo-carry, namely $r_1 = 1$. We have two cases:

- Either this is r_1 that propagates and induces r_j.
- Or r_1 is stopped because $a_1 = 0$, or because there exist two or more consecutive bits equal to 0: a_x, a_{x+1} with $j - 1 > x > 1$.

If this is r_1 that propagates and induces r_j, then $a_1 = 1$, and furthermore the binary structure $(a_{j-1}\ a_{j-2}\ \cdots\ a_2\ a_1)_2$ must not contain two or more consecutive bits equal to 0. In this case, there are two possibilities:

- If $a_{j-1} = 0$, then $r_j = 1$, is the only pseudo-carry, and B undergoes the operation $3B + 1$.

- If $a_{j-1} = 1$, then the two values $a_{j-1} = 1$ and $r_j = 1$ are transferred to form the two pseudo-carries $r_j = 0$ and $r_{j+1} = 1$, and B undergoes the operation $3B + 2$.

If r_1 is stopped, then we have:

- If r_1 is stopped because $a_1 = 0$, then we define the structure

$$A' = (a_k\, a_{k-1}\, \cdots\, a_3\, a_2)_2$$

which undergoes the operation $3A' + 0$, and we apply the rule already defined in **Case 1** to deduce the operation undergone by

$$B = (a_k\, a_{k-1}\, \cdots\, a_{j+1}\, a_j)_2.$$

- If r_1 is stopped by the existence of $a_x = a_{x+1} = 0, \qquad j - 1 > x > 1,$

Then we define the binary structure $A' = (a_k\, a_{k-1}\, \cdots\, a_{x+3}\, a_{x+2})_2$

which undergoes the operation $3A' + 0$, and we apply the rule already defined in **Case 1** to deduce the operation undergone by

$$B = (a_k\, a_{k-1}\, \cdots\, a_{j+1}\, a_j)_2.$$

Case 3: A undergoes the operation $3A + d$, with $d = 2$

Therefore, there are two pseudo-carries given by $r_2 = 1$ and $r_1 = 0$.

There are two cases:

- either this is r_2 that propagates and induces r_j,
- or r_2 is stopped by the fact that $a_1 = a_2 = 0$, or by the existence of two or more consecutive bits equal to 0, a_x, a_{x+1}, with $j - 1 > x > 2$.

If this is r_2 that propagates and induces r_j, then there is at least one of the bits a_1, a_2 equal to 1, and the binary structure $(a_{j-1}a_{j-2}\ldots a_2a_1)_2$ also contains no sequence of two or more consecutive bits equal to 0. In this case there are two possibilities:

- if $a_{j-1} = 0$, Then the only pseudo-carry is $r_j = 1$, and therefore B undergoes the operation $3B + 1$.

- if $a_{j-1} = 1$, Then the two values $r_j = 1$ and $a_{j-1} = 1$ are transferred to form the two pseudo-carries $r_{j+1} = 1$ and $r_j = 0$, and B undergoes the operation

$$3B + 2.$$

If r_2 is stopped, then:

- if $a_1 = a_2 = 0$, We define the structure $A' = (a_k\, a_{k-1} \dots a_4 a_3)_2$, which undergoes the operation $3A' + 0$, and we apply the rule already defined in Case 1 to determine the operation undergone by

$$B = (a_k\, a_{k-1} \dots a_{j+1} a_j)_2.$$

- if r_2 is stopped by the existence of $a_x = a_{x+1} = 0$, with $j - 1 > x > 2$, Then we define the binary structure $A' = (a_k\, a_{k-1} \dots a_{x-2} a_{x-3})_2$, and we apply the rule already defined in Case 1 to A', which undergoes the operation $3A' + 0$, in order to determine the operation undergone by

$$B = (a_k\, a_{k-1} \dots a_{j+1} a_j)_2.$$

10.8.3

Here we will use Properties 10.1, 10.2, 10.3, 10.4, 10.5, 10.6, 10.7 and the rules established in Section 10.8.2 to continue the proof that

$$P(\mathcal{O}_0) = 1 \Rightarrow \exists i > 0 : P(\mathcal{O}_i) = 0$$

for the case where $L(G_{(1,0)}) > 1$,

because in Section 10.8.1 we proved that if $\mathcal{O}_0 = G_{(1,0)} \| S_0$ and $L(G_{(1,0)}) = 1$, then $\exists i > 0 : P(\mathcal{O}_i) = 0$.

Case 1 $G_{(1,0)}$ is a suffix of the generator $((011100)^\omega 01)_2$.

If $L(G_{(1,0)}) = 5 + 6i, \quad i \in \mathbb{N}$, then, according to Section 10.6, S_1 ends with an MSB equal to 1.

Furthermore, we may write $G_{(1,0)}$ in the form $((100011)^\omega 10001)_2$.

According to Section 5.1, $G_{(1,0)}$ undergoes the operation $3G_{(1,0)} + 2$,

But in order to determine the overflow of the iteration

$$\mathcal{O}_1 = f_0(G_{(1,0)}) = 3G_{(1,0)} + 2,$$

It is sufficient to determine it from the operation undergone by the MSB of $G_{(1,0)}$ to determine this operation undergone by the MSB of $G_{(1,0)}$, we apply the

rule established in Section 10.8.2 to the binary structure $G_{(1,0)}$ which undergoes the operation $3G_{(1,0)} + 2$.

We have the MSB of $G_{(1,0)}$ is always $(1)_2$, and it is always preceded by two consecutive bits equal to 0. Therefore, the MSB of $G_{(1,0)}$ undergoes the operation $3(1)_2 + 0 = (11)_2$.

Hence, the overflow $(1)_2$ of $f_0(G_{(1,0)}) = \mathcal{O}_1$ is not complementary to S_1, whose MSB is equal to 1.

Therefore, $\mathcal{O}_1 = G_{(1,1)} \| S_1$, which gives $(P(\mathcal{O}_1) = 1) \wedge (L(G_{(1,1)}) = 1)$.

Thus $\mathcal{O}_1$ falls under the case studied in Section 10.8.1, and therefore

$$(P(\mathcal{O}_1) = 1) \wedge (L(G_{(1,1)}) = 1) \Rightarrow P(\mathcal{O}_2) = 0,$$

Hence $\mathcal{O}_2$ is a singularity.

Case $L(G_{(1,0)}) = 6i,\ i \in \mathbb{N}^*$

If $L(G_{(1,0)}) = 6i, \qquad i \in \mathbb{N}^*$, then according to Section 10.6 the MSB of S_1 is 0. Furthermore, we have $G_{(1,0)} = \big((110001)^\omega 110001\big)_2$.

According to Section 5.1, $G_{(1,0)}$ undergoes the operation $3G_{(1,0)} + 2$.

To determine the overflow produced by the operation $f_0(G_{(1,0)}) = \mathcal{O}_1 = 3G_{(1,0)} + 2$, it is sufficient to deduce it from the operation undergone by the MSB of $G_{(1,0)}$. To determine this operation, we apply the rule established in Section 10.8.2 to the binary structure $G_{(1,0)}$, which undergoes the operation $3G_{(1,0)} + 2$.

The MSB of $G_{(1,0)}$ is always equal to $(1)_2$, and it is always preceded by $(100)_2$. Therefore, there is no carry that can propagate to reach the MSB of $G_{(1,0)}$. However, since the bit immediately preceding this MSB is equal to 1, the MSB undergoes the operation $3 \cdot (1)_2 + 1$.

Since $3 \cdot (1)_2 + 1 = (100)_2$, the overflow of the iteration $f_0(G_{(1,0)}) = \mathcal{O}_1$ is $(10)_2$, which is not complementary to S_1 because the MSB of S_1 is 0. Therefore,

$$(10)_2 = G_{(2,1)} \| G_{(1,1)},$$

and hence $P(\mathcal{O}_1) = 2$.

Moreover, $L(G_{(1,1)}) = 1$, which implies that the MSB of S_2 is 1.

Note that $\mathcal{O}_1 = G_{(2,1)} \| G_{(1,1)} \| S_1$ is not a case conforming to Case 1, where $G_{(1,0)}$ is a suffix of the generator $((011100)^\omega 01)_2$,

because $G_{(1,1)} = (0)_2$.

Therefore, it is a suffix of the generator $((000111)^\omega 00)_2$,

and furthermore, $P(\mathcal{O}_1) \neq 1$.

According to Section 5.1, $G_{(2,1)}\|G_{(1,1)}$ undergoes the operation $3\big(G_{(2,1)}\|G_{(1,1)}\big)+1$.

To determine the overflow of the operation $f_1\big(G_{(2,1)}\|G_{(1,1)}\big) = \mathcal{O}_2$,

We deduce it from the operation undergone by the MSB of $G_{(2,1)}$. According to the rule established in Section 10.8.2, this operation is $3(1)_2 + 0$, since the MSB of $G_{(2,1)}$ is preceded by two consecutive bits equal to 0, We have:

$$3 \cdot (1)_2 = (11)_2.$$

Therefore, the overflow of $f_1(G_{(2,1)}\|G_{(1,1)})$ is $(1)_2$, which is not complementary to $G_{(1,2)} = (1)_2$.

Hence, $\mathcal{O}_2 = G_{(2,2)}\|G_{(1,2)}\|S_2$,

with $G_{(2,2)} = (1)_2, \qquad G_{(1,2)} = (1)_2, \qquad S_2 = (1)_2$.

Thus, $\mathcal{O}_2 = (111)_2$, which gives

$$\mathcal{O}_3 = \frac{3\mathcal{O}_2 + 1}{2^{\nu_2(3\mathcal{O}_2+1)}},$$

without using the reduced odd Syracuse operation by binary structure, we obtain

$\mathcal{O}_3 = (1011)_2$. herefore,

$\mathcal{O}_4 = \frac{3\mathcal{O}_3+1}{2^{\nu_2(3\mathcal{O}_3+1)}} = (10001)_2$. hence,

$\mathcal{O}_5 = \frac{3\mathcal{O}_4+1}{2^{\nu_2(3\mathcal{O}_4+1)}} = (1101)_2$, and we observe that

$P(\mathcal{O}_5) = 1$ and $L(G_{(1,5)}) = 1$.

Therefore, $\mathcal{O}_5$ conforms to the case of Section 10.8.1, hence,

$$(P(\mathcal{O}_5) = 1) \wedge (L(G_{(1,5)}) = 1) \Rightarrow P(\mathcal{O}_6) = 0,$$

which means that $\mathcal{O}_6$ is a singularity, we therefore conclude that

$$L(G_{(1,0)}) = 6i, \qquad i \in \mathbb{N}^*,$$

with $G_{(1,0)}$ being a suffix of the generator $((011100)^\omega 01)_2$,

implies that $\mathcal{O}_6$ is a singularity.

- If

$$L(G_{(1,0)}) = 1 + 6i, \quad i \in \mathbb{N}^*,$$

then according to Section 10.6, the MSB of S_1 is equal to 1. Furthermore, we have $G_{(1,0)} = ((111000)^\omega 1110001)_2$.

We know from Section 5.1 that $G_{(1,0)}$ undergoes the operation $3G_{(1,0)}+2$,

But in order to determine the overflow of the operation $f_0(G_{(1,0)}) = \mathcal{O}_1$,

It is sufficient to deduce it from the operation undergone by the MSB of $G_{(1,0)}$. To determine this operation, we apply the rule established in Section 10.8.2 to the binary structure $G_{(1,0)}$, which undergoes the operation $3G_{(1,0)} + 2$

The MSB of $G_{(1,0)}$ is always equal to $(1)_2$,, and it is always preceded by two consecutive bits equal to 1. Therefore, there is a carry that reaches the MSB of $G_{(1,0)}$ according to the rule established in Section 10.8.2. Since the bit preceding the MSB is $(1)_2$, we conclude that the MSB undergoes the operation $3(1)_2 + 2 = (101)_2$.

Therefore, the overflow of the operation $f_0(G_{(1,0)}) = \mathcal{O}_1$ is $(10)_2$,, and it is complementary (both bits) to S_1, since S_1 has an MSB equal to $(1)_2$.

Therefore $\mathcal{O}_1$ is a singularity. Hence, $L(G_{(1,0)}) = 1 + 6i, \quad i \in \mathbb{N}^*$,

with $G_{(1,0)}$ being a suffix of a singularity-suffix generator $((011100)^\omega 01)_2$,

implies that $\mathcal{O}_1$ is a singularity.

Conclusion for Case 1 If $P(\mathcal{O}_0) = 1$, with $G_{(1,0)}$ being a suffix of the singularity-suffix generator $((011100)^\omega 01)_2$,

Then the odd Syracuse sequence reaches a term $\mathcal{O}_i, \quad i > 0$, such that $\mathcal{O}_i$ is a singularity.

Case 2

$G_{(1,0)}$ is a suffix of the generator $((000111)^\omega 00)_2$.

If $L(G_{(1,0)}) = 3 + 6i, \quad i \in \mathbb{N}$, then according to Section 10.7, the MSB of S_1 is equal to 1, furthermore, $G_{(1,0)} = ((100011)^\omega 100)_2$

According to Section 5.1, $G_{(1,0)}$ undergoes the operation $3G_{(1,0)} + 1$,

But in order to determine the overflow of the operation $f_0(G_{(1,0)}) = \mathcal{O}_1$, it is sufficient to determine it from the operation undergone by the MSB of $G_{(1,0)}$. To determine this operation, we apply the rule established in Section 10.8.2 to the binary structure $G_{(1,0)}$, which undergoes the operation $3G_{(1,0)} + 1$.

The MSB of $G_{(1,0)}$ is always equal to $(1)_2$, and it is always preceded by two consecutive bits equal to 0. Therefore, there is no carry that can propagate and reach the MSB of $G_{(1,0)}$. Since the bit preceding the MSB is 0, the MSB undergoes the operation $3(1)_2 + 0 = (11)_2$.

Hence, the overflow is $(1)_2$, and it is not complementary to S_1, since the MSB of S_1 is $(1)_2$.

Therefore, $\mathcal{O}_1 = G_{(1,1)} \parallel S_1$,

with $G_{(1,1)} = (1)_2$.
Thus $G_{(1,1)}$ is a suffix of the generator $((011100)^\omega 01)_2$,
and since $L(G_{(1,1)}) = 1$,, $\mathcal{O}_1$ falls under Case 1 of Section 10.8.1.
Therefore, we conclude that
$(P(\mathcal{O}_1) = 1)$ and $(L(G_{(1,1)}) = 1)$ implies $P(\mathcal{O}_2) = 0$,
that is, $\mathcal{O}_2$ is a singularity.
If

$$L(G_{(1,0)}) = 4 + 6i, \quad i \in \mathbb{N},$$

then according to Section 10.7, the MSB of S_1 is equal to 0. Furthermore,

$$G_{(1,0)} = ((110001)^\omega 1100)_2.$$

According to Section 5.1 $G_{(1,0)}$ undergoes the operation,

$$f_0(G_{(1,0)}) = \mathcal{O}_1 = 3G_{(1,0)} + 1.$$

However, in order to determine the overflow of the operation $f_0(G_{(1,0)}) = \mathcal{O}_1$, it is sufficient to determine it from the operation undergone by the MSB of $G_{(1,0)}$, and to determine this operation we apply the rule established in Section 10.8.2 to the binary structure $G_{(1,0)}$, which undergoes the operation

$$3G_{(1,0)} + 1.$$

The MSB of $G_{(1,0)}$ is always equal to $(1)_2$, and it is always preceded by $(100)_2$. Therefore, there is no carry that can propagate to reach the MSB of $G_{(1,0)}$. However, since the bit immediately preceding this MSB is equal to $(1)_2$, the MSB of $G_{(1,0)}$ undergoes the operation $3 \cdot (1)_2 + 1 = (100)_2$.

Therefore, the overflow of the operation $f_0(G_{(1,0)}) = \mathcal{O}_1$

is $(10)_2$, and it is not complementary to S_1 since the MSB of S_1 is $(0)_2$. Hence,

$(10)_2 = G_{(2,1)} \| G_{(1,1)}$, and therefore

$$P(\mathcal{O}_1) = 2.$$

Moreover, $L(G_{(1,1)}) = 1$, which implies that the MSB of S_2 is $(1)_2$.

Note that $\mathcal{O}_1 = G_{(2,1)} \| G_{(1,1)} \| S_1$ is not a case conforming to **Case 1**, where $G_{(1,0)}$ is a suffix of the generator $((011100)^\omega 01)_2$,

because for $\mathcal{O}_1$, $G_{(1,1)}$ is a suffix of the generator $((000111)^\omega 00)_2$.

Furthermore, $P(\mathcal{O}_1) \neq 1$.

According to Section 5.1, $G_{(2,1)} \| G_{(1,1)}$ undergoes the operation $f_1(G_{(2,1)} \| G_{(1,1)}) = 3(G_{(2,1)} \| G_{(1,1)}) + 1 = \mathcal{O}_2$.

To determine the overflow produced during this operation, we deduce it from the operation undergone by the MSB of $G_{(2,1)}$. According to the rule established in Section 10.8.2, this operation is $3 \cdot (1)_2 + 0$,

since the MSB of $G_{(2,1)}$ is always preceded by two consecutive bits equal to 0., we have $3 \cdot (1)_2 = (11)_2$,

therefore the overflow of $f_1(G_{(2,1)} \| G_{(1,1)})$ is $(1)_2$, which is not complementary to $G_{(1,2)} = (1)_2$.

Hence, $\mathcal{O}_2 = G_{(2,2)} \| G_{(1,2)} \| S_2$,

with $G_{(2,2)} = (1)_2, \qquad G_{(1,2)} = (1)_2, \qquad S_2 = (1)_2$.

Therefore, $\mathcal{O}_2 = (111)_2$, which gives

$\mathcal{O}_3 = \frac{3\mathcal{O}_2+1}{2^{\nu_2(3\mathcal{O}_2+1)}} = (1011)_2 = G_{(2,3)} \| G_{(1,3)} \| S_3$.

Then we obtain

$$\mathcal{O}_4 = \frac{3\mathcal{O}_3 + 1}{2^{V_2(3\mathcal{O}_3+1)}} = (10001)_2 = G_{(2,4)} \parallel G_{(1,4)} \parallel S_4$$

Then we obtain

$$\mathcal{O}_5 = \frac{3\mathcal{O}_4 + 1}{2^{V_2(3\mathcal{O}_4+1)}} = (1101)_2 = G_{(1,5)} \parallel S_5$$

We observe that $P(\mathcal{O}_5) = 1$ and $L(G_{(1,5)}) = 1$, therefore $\mathcal{O}_5$ satisfies the case of Section 10.8.1, hence $(P(\mathcal{O}_5) = 1)$ and $(L(G_{(1,5)}) = 1) \Rightarrow P(\mathcal{O}_6) = 0$ that is, $\mathcal{O}_6$ is a singularity. Therefore, we conclude that $L(G_{(1,0)}) = 4{+}6i, \quad i \in \mathbb{N}$ with $G_{(1,0)}$ being a suffix of the generator $((000111)^\omega 00)_2$

implies that $\mathcal{O}_6$ is a singularity.

• If

$$L(G_{(1,0)}) = 5 + 6i, \qquad i \in \mathbb{N}$$

then, according to Section 10.7, the MSB of S_1 is $(1)_2$. Furthermore, we have $G_{(1,0)} = ((111000)^\omega 11100)_2$

We know from Section 5.1 that $G_{(1,0)}$ undergoes the operation $3G_{(1,0)} + 1$ but to determine the overflow of the operation $f_0(G_{(1,0)}) = \mathcal{O}_1$

It is sufficient to deduce it from the operation undergone by the MSB of $G_{(1,0)}$, and to determine this operation, we will apply the rule established in Section 10.8.2 to the binary structure $G_{(1,0)}$ undergoing the iteration $3G_{(1,0)}{+}1$.

We have: the MSB of $G_{(1,0)}$ is always equal to $(1)_2$, and it is always preceded by $(11)_2$, therefore there is a carry that will propagate and reach the MSB of $G_{(1,0)}$, and furthermore the bit preceding the MSB is equal to $(1)_2$, therefore the MSB undergoes the operation $3 \cdot (1)_2 + 2$.

We have $3 \cdot (1)_2 + 2 = (101)_2$ therefore the overflow of the operation $f_0(G_{(1,0)}) = \mathcal{O}_1$ is $(10)_2$, which is complementary (both bits) to S_1 since the MSB of S_1 is $(1)_2$.

Therefore, we conclude that $\mathcal{O}_1$ is a singularity.

Thus, $L(G_{(1,0)}) = 5 + 6i, \quad i \in \mathbb{N}$ with $G_{(1,0)}$ being a suffix of the generator $((000111)^\omega 00)_2$

implies that $\mathcal{O}_1$ is a singularity.

Conclusion for Case 2

If $P(\mathcal{O}_0) = 1$ with $G_{(1,0)}$ being a suffix of the generator of the suffix of the singularity $((000111)^\omega 00)_2$

then the odd Syracuse sequence will reach a term $\mathcal{O}_i$, $i > 0$, such that $\mathcal{O}_i$ is a singularity.

Conclusion for Section 10

Let $\mathcal{O}_0$ be a term of the odd Syracuse sequence that is not a singularity such that $P(\mathcal{O}_0) = 1$. Then the sequence will reach a term $\mathcal{O}_i$ with $i > 0$ such that $\mathcal{O}_i$ is a singularity; equivalently,

$$P(\mathcal{O}_0) = 1 \Rightarrow \exists i > 0 \text{ such that } P(\mathcal{O}_i) = 0.$$

10.9 Proof that the function P is decreasing for $P(\mathcal{O}_0) > 1$

We will use an induction proof to demonstrate that

$$P(\mathcal{O}_0) > 1 \Rightarrow \exists i > 0 : P(\mathcal{O}_i) < P(\mathcal{O}_0)$$

starting from the fact that we have already proved in Section 10.8 that

$$P(\mathcal{O}_0) = 1 \Rightarrow \exists i > 0 : P(\mathcal{O}_i) < P(\mathcal{O}_0)$$

and for this we must identify the crucial properties that enabled the proof presented in Section 10.8 and attempt to prove that these properties remain valid even when $P(\mathcal{O}_0) > 1$.

10.9.1 Periodicity of the generators of the suffix of singularity

Let A and B be two infinite binary structures such that

$B = 3A + d, \qquad d \in \{0, 1, 2\}$, $A = (a_k\, a_{k-1}\, \ldots\, a_2\, a_1)_2$,

which gives $A = \sum_{i=1}^{\infty} a_i \, 2^{i-1}$. and $B = (b_k \, b_{k-1} \, \ldots \, b_2 \, b_1)_2$,
which gives

$$B = \sum_{i=1}^{\infty} b_i \, 2^{i-1}.$$

We have $d = 1 \Rightarrow r_1 = 1$, and this is the only pseudo-carry. $d = 0 \Rightarrow r_1 = 0$,

implies that there is one pseudo-carry. $d = 2$
implies that $r_1 = 0, \qquad r_2 =$,
And this is the two pseudo-carries.

$$\begin{array}{cccccccc}
 & & r_{k+1} & r_k & r_{k-1} & \ldots & r_2 & r_1 \\
+ & & & a_k & a_{k-1} & \ldots & a_2 & a_1 \\
+ & & a_k & a_{k-1} & \ldots & a_2 & a_1 & 0 \\
\hline
= & b_{k+2} & b_{k+1} & b_k & b_{k-1} & \ldots & b_2 & b_1
\end{array}$$

We have $3A + d = A + 2A + d$.
The addition is performed bit by bit and is governed by the following rule:

$$2r_{k+1} + b_k = a_k + a_{k-1} + r_k.$$

To determine A from B and the operation $B = 3A + d$, we rewrite the local equation modulo 2,

$$a_k \equiv b_k \oplus a_{k-1} \oplus r_k \pmod 2$$

and the next carry r_{k+1} is

$$r_{k+1} = \left\lfloor \frac{a_k + a_{k-1} + r_k - b_k}{2} \right\rfloor$$

The computation of the bit a_k depends only on a local state triplet

$$E_k = (a_{k-1}, r_k)$$

since $a_{k-1} \in \{0, 1\}$ and $r_k \in \{0, 1, 2\}$ (read the following remark to understand why $r_k = 2$ is possible),

therefore there exist 6 possibilities for this dynamical system at each step k.

$$E_k = \{(0,0), (0,1), (0,2), (1,0), (1,1), (1,2)\}$$

at each iteration $k \to k + 1$, the system receives an incoming information (b_k) (which is periodic with period t), and transitions to a new state

$$E_{k+1} = (a_k, r_{k+1})$$

10.9.1.1 Arguments for the $3t$-periodicity of A

- since B is periodic with period t, then the incoming information b_k repeats exactly every t steps, that is $b_{k+t} = b_k$
- if we consider the state of the system at indices that are multiples of the period, that is, the sequence of states $E_0, E_t, E_{2t}, E_{3t}, \ldots$

1. Pigeonhole Principle: the state space E is finite (6 states), if one observes the sequence of states E_{mt} for $m = 0, 1, 2, 3, \ldots$

One applies a shifted transition by a linear operator, the division by 3.

2- Modulo 3

The transition mapping on the p-adic states induces a permutation; more precisely, the action of shifting by t bits on the carry structure corresponds to a multiplication by 2^t (modulo 3).

3- Since 2^t (modulo 3) can take only two values $\{1, 2\}$, the maximal multiplicative order of this operation is 3

(related to the field extension or to the cycle of residues modulo 3).

If we follow the evolution of the state E_k,

1. At step t, the state E_t has absorbed one complete period of B, and the accumulated carry has undergone a phase shift.
2. At step $2t$, the phase shift is doubled.
3. At step $3t$, the phase shift is multiplied by

$$2^{3t} = (2^3)^t \equiv 8^t \equiv 1^t \equiv 1 \pmod 3$$

for t always even,

the carry returns exactly to its initial state.

We have therefore rigorously proved that $E_{k+3t} = E_k$.

Conclusion

Since the state $E_{k+3t} = (a_{k-1+3t}, r_{k+3t})$ is identical to the state $E_k = (a_{k-1}, r_k)$,

and since the incoming information B is identical, $b_{k+3t} = b_k$, then the automaton will produce exactly the same outputs from this point onward; consequently, $a_{k+3t} = a_k, \qquad \forall k \in \mathbb{N}^*$, that is, the bits in the binary structure A repeat every $3t$ bits, therefore (the binary period of A divides $3t$).

10.9.1.2 Remark

The value $r_k = 2$ does not appear during the forward carry propagation (the computation of B from A), but it appears when studying the inverse process (the computation of A from B), using the local equation obtained by isolating the bit a_k:

$$a_k + a_{k-1} + r_k = 2r_{k+1} + b_k$$

In this inverse automaton, the system must infer the bit a_k from the information it receives.

If the information B contains a sequence of zeros (generated by a large subtraction resulting from a particular pattern choice), the binary reconstruction algorithm is forced to assign the value 2 to the internal carry variable in order to mathematically compensate for the phase shift produced by the preceding blocks of bits.

This is why, in the theoretical phase space of the automaton (which covers all possible reconstruction trajectories), the state $r_k = 2$ exists and must be taken into account.

10.9.2 Overflows Generated According to the Operation undergone by the MSB

Let $\mathcal{O}_0$ be a term of the odd Syracuse sequence that is not a singularity. According to Lemma 9.5, the binary structure of $\mathcal{O}_0$ can be decomposed as

$\mathcal{O}_0 = G_{(p,0)} \| G_{(p-1,0)} \| \cdots \| G_{(2,0)} \| G_{(1,0)} \| S_0$, with $p \geq 1$.

We know that the overflow of the operation

$$\mathcal{O}_1 = f_0\big(G_{(p,0)} \| G_{(p-1,0)} \| \cdots \| G_{(2,0)} \| G_{(1,0)}\big)$$

can be deduced from the operation undergone by the MSB of $G_{(p,0)}$. Therefore we have:

(a) If the MSB undergoes the operation $3 \cdot (1)_2 + 0$ with MSB $= (1)_2$, then the overflow is $(1)_2$ sinc $3(1)_2 + 0 = (11)_2$.

(b) If the MSB undergoes the operation $3 \cdot (1)_2 + 1$, then the overflow is $(10)_2$ since $3(1)_2 + 1 = (100)_2$.

(c) If the MSB undergoes the operation $3 \cdot (1)_2 + 2$, then the overflow is $(10)_2$ since $3(1)_2 + 2 = (101)_2$.

10.9.2.1 Number of Successive Iterations with decreasing, constant and increasing P

It was established empirically using two Delphi programs *PDecayAnalyzer* and *PGrowthAnalyzer* that are provided with this preprint as supplementary material that over the test interval $\mathcal{O}_0 \in [1, 99999]$ with $\mathcal{O}_0$ odd,

The maximum number of successive iterations with the function P decreasing is 9.

And the maximum number of successive iterations with constant P is 14.

Now we will formally prove that the number of successive iterations of the odd Syracuse sequence with increasing P is 2 for any $\mathcal{O}_0 \in 2\mathbb{N} + 1$

where $\mathcal{O}_0$ is not a singularity.

Proof For P to increase, according to Section 9.6 there must be a two-bit overflow and the overflow must be completely non-complementary.

Furthermore, each overflow bit forms a new suffix of a generator of the suffix of singularity, and according to Section 10.9.2.1 this corresponds to cases (b) and (c).

That is, the MSB of $G_{(p,0)}$ of $\mathcal{O}_0$ witch is always equal to $(1)_2$ undergoes the operation $3 \cdot (1)_2 + 1$ or $3 \cdot (1)_2 + 2$.

a)

Assume that the MSB of $\mathcal{O}_0$ undergoes the operation $3 \cdot (1)_2 + 1$ which generates an overflow of $(10)_2$ since $3(1)_2 + 1 = (100)_2$.

Therefore, the three most significant bits of $\mathcal{O}_1$ are $(100)_2$.

According to the rule established in Section 10.8.2, the MSB of $\mathcal{O}_1$ will undergo the operation $3 \cdot (1)_2 + 0$ because it is preceded by two successive bits equal to zero, and therefore the overflow will be $(1)_2$, which means that there cannot be another iteration in which the function P is increasing.

b)

Now assume that the MSB of $\mathcal{O}_0$ undergoes the operation $3 \cdot (1)_2 + 2$ which generates an overflow of $(10)_2$ since $3(1)_2 + 2 = (101)_2$.

Therefore, the three most significant bits of $\mathcal{O}_1$ are $(101)_2$.

According to the rule established in Section 10.8.2, the MSB of $\mathcal{O}_1$ may undergo the operation $3 \cdot (1)_2 + 1$

If the other bits of $\mathcal{O}_1$ that are located to the right of the three most significant bits of $\mathcal{O}_1$, namely $(101)_2$, reach two or more consecutive bits equal to 1 before reaching two or more consecutive bits equal to 0.

This means that there is a possibility that the MSB of $\mathcal{O}_1$ undergoes the operation $3 \cdot (1)_2 + 1$ and produces the overflow $(10)_2$, which makes it possible to have another iteration of the odd Syracuse sequence with increasing P. However, if this second iteration allows P to increase for the second consecutive time, then it will be the last one because, according to part (a), the iteration that generates $\mathcal{O}_2$ will produce an overflow of $(1)_2$.

Therefore, the function P can be successively increasing for at most two iterations.

Example

$$\mathcal{O}_0 = 1\|1\|11\|1\|01\|1 \text{ with } P(\mathcal{O}_0) = 5$$

$$\mathcal{O}_1 = 1\|0\|1\|1\|11\|0\|01 \text{ with } P(\mathcal{O}_1) = 6$$

$$\mathcal{O}_2 = 1\|0\|0\|0\|1\|1\|01\|1 \text{ with } P(\mathcal{O}_2) = 7$$

$$\mathcal{O}_3 = 11\|0\|0\|0\|1\|1\|01\|1 \text{ with } P(\mathcal{O}_3) = 7$$

10.9.3 The Odd Syracuse Dynamics as a Binary Rewriting System

The structural framework developed in this work suggests that the odd Syracuse dynamics can be interpreted naturally as a *binary rewriting system*. This interpretation does not replace the arithmetic definition of the odd Syracuse map; rather, it provides an equivalent symbolic description in which the evolution of an odd integer is represented as a sequence of transformations of binary blocks.

$$\frac{3\mathcal{O}_n + 1}{2^{\nu_2(3\mathcal{O}_n+1)}}.$$

Since

$$3\mathcal{O}_n + 1 = \mathcal{O}_n + 2\mathcal{O}_n + 1,$$

The computation of $3\mathcal{O}_n + 1$ is performed by the binary addition of two shifted copies of the binary structure of $\mathcal{O}_n$, together with the additional $+1$. The resulting bits are therefore generated by a local propagation of binary carries. In particular, the bit b_i of the result is determined by the neighboring input bits and the carry entering the corresponding position Lemma 7. Thus, the arithmetic transformation $\mathcal{O}_n \mapsto 3\mathcal{O}_n + 1$ admits a local symbolic description in terms of binary rewriting rules. The binary construction established above shows that the first j least significant bits of the result depend only on the first j least significant bits of the input Lemma 7. Consequently, the transformation is locally determined at the level of binary structures. The division by

$$2^{\nu_2(3\mathcal{O}_n+1)}$$

has an equally natural interpretation in this symbolic framework. In binary representation, this division simply removes the corresponding trailing zero bits from $3\mathcal{O}_n + 1$. Therefore, one complete odd Syracuse iteration consists of two binary operations: first, a local rewriting generated by the operation $3\mathcal{O}_n + 1$, and second, the deletion of the maximal suffix of zero bits. Hence the odd Syracuse map can be represented schematically as

$$\text{binary structure of } \mathcal{O}_n \quad \longrightarrow \quad \text{binary structure of } (3\mathcal{O}_n + 1)$$

$$\longrightarrow \quad \text{binary structure of } \mathcal{O}_{n+1}.$$

$$\mathcal{O}_n = G_{(p,n)} \| G_{(p-1,n)} \| \cdots \| G_{(1,n)} \| S_n,$$

where S_n is the suffix of singularity, and each $G_{(i,n)}$ Is a suffix of a generator of the suffix of singularity. Each generator has a prescribed structural role in the subsequent evolution. In particular,

$$G_{(1,n)} \longrightarrow S_{n+1},$$

while, for $i > 1$,

$$G_{(i,n)} \longrightarrow G_{(i-1,n+1)}.$$

Therefore, the suffixes of generators behave as hierarchical binary blocks that are successively rewritten toward the singularity. This provides a symbolic rewriting interpretation of the global structural evolution.

The rewriting process is conditional rather than purely local in the sense of an isolated block, because the overflow produced by the most significant

generator may depend on the carry reaching its most significant bit. Consequently, the rewriting of a generator is influenced by its binary context. The overflow can be represented by one or two bits, and its complementarity with the generator induced at the next level determines whether the number of generators is increasing, decreasing, or preserved. This mechanism is responsible for the three possible transitions

$$P(\mathcal{O}_{n+1}) - P(\mathcal{O}_n) \in \{-1, 0, +1\}.$$

Hence, the function P measures the number of generator blocks present in the current rewriting configuration, while the overflow determines how the rewriting system modifies this configuration.

Finally, the carry propagation itself admits a finite-state symbolic representation. The local state

$$E_k = (a_{k-1}, r_k)$$

Belongs to a finite set of states, and its evolution is determined by the incoming binary information and the local rewriting rule section 10.9.1. This provides an automaton-like description of the binary transformation and further supports the interpretation of the odd Syracuse dynamics as a symbolic rewriting process.

Accordingly, the odd Syracuse dynamics may be viewed as a binary rewriting system whose configurations are finite binary structures, whose rewriting rules are generated by binary addition and carry propagation, and whose hierarchical blocks are progressively transformed toward the binary singularity. In this representation, the Collatz problem is reformulated as a termination problem for a structured binary rewriting process:

$$\begin{aligned} \text{initial binary configuration} &\Longrightarrow \text{successive binary re-writings} \\ &\Longrightarrow \text{binary singularity.} \end{aligned}$$

The singularity $P = 0$ is therefore interpreted as the terminal structural configuration of the rewriting dynamics, while the remaining problem is to establish that every Odd Syracuse term that does not exhibit the singularity configuration is eventually rewritten into this terminal state.

10.9.4 Well-Founded Lexicographic Termination and the Eventual Decrease of P

The binary structural formulation developed in this work naturally leads to a termination problem. Every odd Syracuse term that is not a singularity

admits a unique canonical decomposition $\mathcal{O}_n = G_{(p,n)} \| G_{(p-1,n)} \| \cdots \| G_{(1,n)} \| S_n$, where S_n is a suffix of singularity, and the $G_{(i,n)}$ are suffixes of generators of the suffix of the singularity. The structural objective is to show that these generators are eventually absorbed into the singularity, and that their number decreases to zero. The binary singularity corresponds to $P = 0$ and therefore plays the role of a terminal structural configuration.

However, the function P, which counts the number of suffixes of generators, cannot be assumed to decrease at every odd Syracuse iteration. The analysis of the six possible overflow-complementarity cases gives

$$P(\mathcal{O}_{n+1}) - P(\mathcal{O}_n) \in \{-1, 0, +1\}.$$

A complementary overflow may eliminate one generator and produce

$$P(\mathcal{O}_{n+1}) = P(\mathcal{O}_n) - 1,$$

whereas a non-complementary overflow may preserve the number of generators or create a new generator, giving respectively $P(\mathcal{O}_{n+1}) = P(\mathcal{O}_n)$ or $P(\mathcal{O}_{n+1}) = P(\mathcal{O}_n) + 1$. Thus, P is not a one-step monotone ranking function.

Nevertheless, the increase of P is strongly constrained. In particular, the binary analysis of the most significant generator shows that P cannot increase during more than two consecutive odd Syracuse iterations. A first increase produces a specific overflow configuration, and the structure generated by this overflow constrains the following transition; after at most a second consecutive increase, the next overflow cannot produce another increase. Hence,

$$P(\mathcal{O}_{n+2}) > P(\mathcal{O}_{n+1}) > P(\mathcal{O}_n)$$

is the longest possible strictly increasing consecutive pattern.

This property, however, does not by itself exclude arbitrarily long sequences involving alternating increases decreases and stationary transitions. Consequently, proving global attraction requires a stronger argument than one-step monotonicity. The relevant question is whether an infinite sequence of binary rewritings can avoid a strict reduction of structural complexity, and producing a drift to infinity, or a cycle. This is naturally formulated as a termination problem.

$$\big(M_1(\mathcal{O}_n), M_2(\mathcal{O}_n), \ldots, M_r(\mathcal{O}_n)\big),$$

then

$$\mathcal{M}(\mathcal{O}_{n+1}) <_{\text{lex}} \mathcal{M}(\mathcal{O}_n)$$

would provide a well-founded descent, since the lexicographic order on a finite product of well-founded orders is itself well founded.

The role of P in such a measure is fundamental but need not be sufficient by itself. The first decomposition describes the hierarchical organization of the global binary attractor, while the transition of the most significant generator is controlled by the overflow produced by its binary operation and by the generator induced at the next level. The overflow is determined by the operation undergone by the most significant bit of $G_{(p,n)}$, and its complementarity with the induced structural block determines whether P increases, decreases or stayes inchanged.

Therefore, the role of the well-founded lexicographic termination framework is to provide a mathematical mechanism capable of containing the non-monotone behavior of P. If an appropriate well-founded ranking measure can be constructed, then an infinite binary rewriting trajectory that indefinitely avoids the singularity would be impossible. Since $P = 0$ is precisely the binary singularity and the singularity is the proposed global structural attractor, termination of the binary rewriting dynamics would imply global structural attraction:

$$\text{no infinite non-singular rewriting trajectory}$$

$$\Longrightarrow \quad \text{eventual reachability of } P = 0.$$

Consequently, establishing such a well-founded ranking measure would provide the required bridge between the local overflow dynamics and the global structural attraction of the binary singularity.

References

[1] Ammar Hamdous, *Revealing a Singularity in Collatz Sequences*, Zenodo, 2026.

[2] Ammar Hamdous, *Reduction of the Collatz conjecture validation sets ROM3 and B to low-density sets*, Zenodo, 2026.

[3] Ammar Hamdous, *A Structural Approach to the Collatz Conjecture via the Binary Singularity*, Zenodo, 2026.