

# Reduced Collatz Dynamics is Periodical and the Period Equals 2 to the Power of the Count of $x/2$

Wei Ren<sup>a,b,c</sup>

<sup>a</sup>*School of Computer Science*

*China University of Geosciences, Wuhan, P.R. China*

<sup>b</sup>*Key Laboratory of Network Assessment Technology, CAS*

*(Institute of Information Engineering,*

*Chinese Academy of Sciences, Beijing, P.R. China 100093)*

<sup>c</sup>*Guizhou Provincial Key Laboratory of Public Big Data*

*Guizhou University, Guizhou, P.R. China*

---

## Abstract

We propose Reduced Collatz Conjecture that is equivalent to Collatz Conjecture but is easier to explore, because reduced dynamics is more primitive than original dynamics and presents better structures (e.g., period and ratio). Reduced dynamics (that are occurred computation sequence from a starting integer to the first integer less than the starting integer) is the component of original dynamics (from a starting integer to 1). Reduced dynamics of  $x$  is represented by a sequence of computation that is either  $(3*x+1)/2$  or  $x/2$ , because  $3*x+1$  is always even and followed by  $x/2$ . We prove that reduced dynamics is periodical and its period equals 2 to the power of the count of  $x/2$ . More specifically, if there exists reduced dynamics of  $x$ , then there exists reduced dynamics of  $x+P$ , where  $P$  equals  $2^L$  and  $L$  is the total count of  $x/2$  computations in reduced dynamics of  $x$  (equivalently,  $L$  is the length of the sequence).

*Keywords:* Collatz Conjecture,  $3x+1$  Problem, Period, Reduced Dynamics  
*2000 MSC:* 11Y55, 11B85, 11A07

---

The Collatz conjecture can be stated simply as follows: Take any positive integer number  $x$ . If  $x$  is even, divide it by 2 to get  $x/2$ . If  $x$  is odd, multiply it by 3 and add 1 to get  $3 * x + 1$ . Repeat the process again and again. The

---

*Email address:* [weirencs@cug.edu.cn](mailto:weirencs@cug.edu.cn) (Wei Ren)

*Preprint submitted to ...*

*January 17, 2020*

Collatz conjecture is that no matter what the number (i.e.,  $x$ ) is taken, the process will always eventually reach 1.

The current known integers that have been verified are about 60 bits by T.O. Silva using normal personal computers [1, 2]. They verified all integers that are less than 60 bits.

Wei Ren et al. [3] verified  $2^{100000} - 1$  can return to 1 after 481603 times of  $3 * x + 1$  computation, and 863323 times of  $x/2$  computation, which is the largest integer being verified in the world. Wei Ren [4] also pointed out a new approach for the possible proof of Collatz conjecture. Wei Ren [5] proposed to use a tree-based graph to reveal two key inner properties in reduced Collatz dynamics: one is ratio of the count of  $x/2$  over the count of  $3 * x + 1$  (for any reduced Collatz dynamics, the count of  $x/2$  over the count of  $3 * x + 1$  is larger than  $\ln 3 / \ln 2$ ), and the other is partition (all positive integers are partitioned regularly corresponding to ongoing dynamics). Wei Ren et al. [6] also proposed an automata method for fast computing Collatz dynamics. All source code and output data by computer programs in those related papers can be accessed in public repository [8].

## 1. Preliminaries

### Notation 1.1.

- (1)  $\mathbb{N}^*$ : positive integers;
- (2)  $\mathbb{N} = \mathbb{N}^* \cup \{0\}$ ;
- (3)  $[1]_2 = \{x | x \equiv 1 \pmod{2}, x \in \mathbb{N}^*\}$ ;  $[0]_2 = \{x | x \equiv 0 \pmod{2}, x \in \mathbb{N}^*\}$ .
- (4)  $[i]_m = \{x | x \equiv i \pmod{m}, x \in \mathbb{N}^*, m \geq 2, m \in \mathbb{N}^*, 0 \leq i \leq m - 1, i \in \mathbb{N}\}$ .

**Proposition 1.2.**  $x/2$  always follows after  $3 * x + 1$ .

**Proof** When  $x \in [1]_2$ , then next computation is  $3 * x + 1$ . Obviously,  $3 * x + 1 \in [0]_2$ , thus the next computation must be  $x/2$  consequently.  $\square$

We thus can represent required computation as  $(3 * x + 1)/2$  and  $x/2$ , which are denoted by  $I(x)$  and  $O(x)$ , respectively.

**Notation 1.3.**  $I(x) = (3 * x + 1)/2$ ,  $O(x) = x/2$ .

Note that,  $I(x)$  and  $O(x)$  can be simply denoted as  $I(\cdot)$  and  $O(\cdot)$ , or  $I$  and  $O$ , respectively. Obviously,  $\forall x \in \mathbb{N}^*$ ,  $I(x) = (3 * x + 1)/2 > x$ ,  $O(x) = x/2 < x$ . That is the reason of notation -  $I$  represents “Increase” and  $O$  represents “dOwn”.

**Definition 1.4.** *Collatz transformation, denoted as  $f(\cdot)$ , where  $f(\cdot) = I(\cdot) = (3 * x + 1)/2$  if  $x \in [1]_2$ , and  $f(\cdot) = O(\cdot) = x/2$  if  $x \in [0]_2$ .*

**Remark 1.5.**

(1) We assume  $f^0(x) = x$ .

(2) Obviously,  $f_1 || f_2 || \dots || f_n(x) = f_n(f_{n-1}(\dots f_2(f_1(x))))$ , where  $f_i(\cdot) \in \{I(\cdot), O(\cdot)\}$ ,  $i = 1, 2, \dots, n$ , and “||” is the concatenation of Collatz transformations. For simplicity, we just denote  $f_i(\cdot)$  as  $f \in \{I, O\}$ .

(3)  $f^n(x) = \underbrace{f \dots f}_n(x)$ ,  $f^n(x) = f(f^{n-1}(x))$ ,  $n \in \mathbb{N}^*$ . Note that, whether  $f$  is  $I$  or  $O$  in  $f(f^{n-1}(x))$ , is determined by  $f^{n-1}(x) \in [1]_2$  or  $[0]_2$ .

**Definition 1.6.** *Collatz Conjecture.  $\forall x \in \mathbb{N}^*$ ,  $\exists L \in \mathbb{N}^*$ , such that  $f^L(x) = 1$  where  $f \in \{I, O\}$ .*

Obviously, Collatz conjecture is held when  $x = 1$ . In the following, we mainly concern  $x \geq 2$ ,  $x \in \mathbb{N}^*$ .

**Definition 1.7.** *Reduced Collatz Conjecture.  $\forall x \in \mathbb{N}^*$ ,  $x \geq 2$ ,  $\exists L \in \mathbb{N}^*$ , such that  $f^L(x) < x$  and  $f^i(x) \not< x$ ,  $i = 0, 1, \dots, L - 1$ ,  $f \in \{I, O\}$ .*

Obviously,  $L$  must be the minimal positive integer such that  $f^L(x) < x$ .

**Theorem 1.8.** *Collatz Conjecture is equivalent to Reduced Collatz Conjecture.*

**Proof**  $\forall x, L \in \mathbb{N}^*$ ,  $x \geq 2$ , it is obvious that  $f^L(x) \in \mathbb{N}^*$ , i.e.,  $f^L(x) \geq 1$ .

(1) Suppose Collatz Conjecture is true. That is,  $\forall x \in \mathbb{N}^*$ ,  $x \geq 2$ ,  $\exists L \in \mathbb{N}^*$ ,  $f^L(x) = 1$ . Thus,  $f^L(x) < x$ . Hence, Reduced Collatz Conjecture is true.

(2) Inversely, suppose Reduced Collatz Conjecture is true. That is,  $\forall x \in \mathbb{N}^*$ ,  $x \geq 2$ ,  $\exists q_0 \in \mathbb{N}^*$ ,  $f^{q_0}(x) < x$ .

If  $f^{q_0}(x) = 1$ , then Collatz Conjecture is true.

If  $f^{q_0}(x) > 1$ , then let  $y_1 = f^{q_0}(x)$ . As Reduced Collatz Conjecture is true,  $\exists q_1 \in \mathbb{N}^*$ ,  $f^{q_1}(y_1) < y_1$ .

For better notation, let  $y_0 = x$ . Iteratively, if  $y_i = f^{q_{i-1}}(y_{i-1}) = 1, i \in \mathbb{N}^*$ , then Collatz Conjecture is true. If  $y_i = f^{q_{i-1}}(y_{i-1}) > 1$ , then  $\exists q_i \in \mathbb{N}^*$ ,  $y_{i+1} = f^{q_i}(y_i) < y_i$ .

Thus,  $y_{i+1} < y_i < \dots < y_1 < y_0 = x$ .  $y_i (i \in \mathbb{N}^*)$  is a *strictly* decreasing sequence.

Besides,  $y_{i+1} = f^{q_0+q_1+q_2+\dots+q_i}(x) \geq 1$ .

Therefore, after finite times of iterations,  $\exists n \in \mathbb{N}^*$ ,  $y_n = 1$ .

That is,  $\exists q = q_0 + q_1 + \dots + q_{n-1} = \sum_{i=0}^{n-1} q_i, q \in \mathbb{N}^*, f^q(x) = 1$ .

Thus, Collatz Conjecture is true. □

**Remark 1.9.**

(1) We call an ordered sequence  $f^q \in \{I, O\}^q$  in above proof as original dynamics (referring to  $f^q(x) = 1$ ), which consists of  $q$  occurred Collatz transformations during the computing procedure from a starting integer to 1. For example, the original dynamics of 5 is IOOO due to  $5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$ .

(2) In contrast, we call  $f^{q_0}$  in above proof as reduced dynamics (referring to  $f^{q_0}(x) < x$ ), which is represented by a sequence of occurred Collatz transformations during the computing procedure from a starting integer (i.e.,  $x$ ) to the first transformed integer that is less than the starting integer (i.e.,  $f^{q_0}(x)$ ). For example, the reduced dynamics of 5 is IO due to  $5 \rightarrow 16 \rightarrow 8 \rightarrow 4$ .

(3) Obviously, reduced dynamics is more primitive than original dynamics, because original dynamics consists of reduced dynamics. Simply speaking, reduced dynamics are building blocks of original dynamics.

Due to above theorem, we concentrate on reduced dynamics.

**Notation 1.10.**  $RD[x]$ . It denotes reduced dynamics of  $x$  that are represented by  $f \in \{I, O\}$ . Formally,  $\forall x \in \mathbb{N}^*, x \geq 2$ , if  $\exists L \in \mathbb{N}^*$  such that  $f^L(x) < x$  and  $f^i(x) \not< x, i = 0, 1, \dots, L - 1$ , where  $f \in \{I, O\}$ , then let  $s = f^L \in \{I, O\}^L$  and  $s$  is called as reduced dynamics of  $x$ , denoted as  $RD[x] = f^L = s$ .

**Remark 1.11.**

(1) Simply speaking (or recall that),  $RD[x]$  represents occurred Collatz transformations in terms of  $I$  and  $O$  during the computing process from starting integer  $x$  to the first transformed integer that is less than  $x$ .

(2) Roughly speaking,  $f^L \in \{I, O\}^L$  is an ordered sequence consisting of  $I$  and  $O$ . Besides,  $f^L = f^{L-1} \circ f$ ,  $f^L(x) = f(f^{L-1}(x))$ , and  $f^0(x) = x$ . Furthermore, this sequence implicitly matches the parity of all occurred intermediate transformed integers that are taken as input of  $f(\cdot)$ .

(3) Recall that, in  $\text{RD}[x] = f^L$ ,  $x$  is called starting integer.  $f^i(x), i = 1, 2, \dots, L$  are called transformed integers.  $f^L(x)$  is the first transformed integer that is less than the starting integer  $x$ . In other words,  $f^i(x) \not\leq x, i = 0, 1, \dots, L - 1$ , and  $f^L(x) < x$ . ( $f^0(x) = x$ .) Besides, the parity of  $f^i(x)$  determines the selection of the intermediately next  $f \in \{I, O\}$  after  $f^i$ .

(4) Obviously,  $\text{RD}[x \in [0]_2] = O$ .

(5) For example,  $\text{RD}[3] = IIOO$ ,  $\text{RD}[5] = IO$ ,  $\text{RD}[7] = IIIIOIOO$ ,  $\text{RD}[9] = IO$ ,  $\text{RD}[11] = IIOIO$ . Indeed, we design computer programs that output all  $\text{RD}[x], \forall x \in [1, 99999999]$  [8]. From the data we discover the property - period and its relation to the number of computing  $x/2$  in reduced dynamics - will be proved in the following of this paper.

(6) In fact, we proved some results on  $\text{RD}[x]$  for specific  $x$ , e.g.,  $\text{RD}[x \in [1]_4] = IO$ ,  $\text{RD}[x \in [3]_{16}] = IIOO$ ,  $\text{RD}[x \in [11]_{32}] = IIOIO$ , and so on [5].

(7)  $IIOO$  can be denoted in short as  $I^2O^2$ .  $IIIIOIOO$  can be denoted in short as  $I^3OIO^2$ . In other words, we denote  $\underbrace{I \dots I}_n$  as  $I^n$ , and we denote  $\underbrace{O \dots O}_n$  as  $O^n$  where  $n \in \mathbb{N}^*, n \geq 2$ . We also assume  $I^1 = I$ ,  $O^1 = O$ .  $I^0 = O^0 = \emptyset$  means no transformation occurs.

(8) In fact, we formally proved that the ratio exists in any reduced Collatz dynamics. That is, the count of  $x/2$  over the count of  $3 * x + 1$  is larger than  $\log_2 3$  [7]. The ratio can also be observed and verified in our proposed tree-based graph [5].

**Example 1.12.**  $\text{RD}[5] = IO$ . It implies following results:

“ $I$ ” is due to  $5 \in [1]_2$ ;

$I(5) = (3 * 5 + 1)/2 = 8 \not\leq 5$ , thus continue;

“ $O$ ” is due to  $I(5) = 8 \in [0]_2$ ;

$IO(5) = O(I(5)) = O(8) = 8/2 = 4 < 5$ , thus end.

To better present above the implicit in reduced dynamics, we introduce two functions as follows:

**Definition 1.13.**  $IsMatched : x \times c \rightarrow bool$ . It takes as input  $x \in \mathbb{N}^*$  and  $c \in \{I, O\}$ , and outputs  $bool \in \{True, False\}$ . If  $x \in [1]_2$  and  $c = I$ , or if  $x \in [0]_2$  and  $c = O$ , then output  $bool = True$ ; Otherwise, output  $bool = False$ .

**Remark 1.14.** Simply speaking, this function checks whether the forthcoming Collatz transformation (i.e.,  $c \in \{I, O\}$ ) matches with the current transformed integer  $x$ .

**Definition 1.15.**  $GetS : s \times i \times j \rightarrow s'$ . It takes as input  $s, i, j$ , where  $s \in \{I, O\}^{|s|}$ ,  $1 \leq i \leq |s|$ ,  $1 \leq j \leq |s| - (i - 1)$ , and outputs  $s'$  where  $s = s_a || s' || s_b$ ,  $|s_a| = i - 1$ ,  $|s'| = j$ ,  $|s_b| = |s| - |s_a| - |s'|$  and “ $|\cdot|$ ” returns length.

**Remark 1.16.**

(1) For example,  $GetS(IIOO, 1, 4) = IIOO$ ,  $GetS(IIOO, 1, 3) = IIO$ .

(2) Especially,  $GetS(s, 1, |s|) = s$ .  $GetS(s, |s|, 1)$  returns the last transformation in  $s$ .  $GetS(s, 1, 1)$  returns the first transformation in  $s$ .  $GetS(s, j, 1)$  returns the  $j$ -th transformation in  $s$ .

(3) In other words,  $s'$  is a selected segment in  $s$  that starts from the location  $i$  and has the length of  $j$ . Indeed, that is the reason we call this function as “Get Substring”.

(4) Simply speaking, this function can obtain the Collatz transforms from  $i$  to  $i + j - 1$  from a given inputting transform sequence (e.g., reduced dynamics) in terms of  $s \in \{I, O\}^{|s|}$ .

(5) Note that,  $GetS(\cdot)$  itself is a function. In other words, it can be looked as  $GetS(\cdot)(\cdot)$ . E.g.,  $GetS(IIOO, 1, 1)(3) = I(3) = (3 * 3 + 1)/2 = 5$ ,  
 $GetS(IIOO, 1, 2)(3) = II(3) = I(I(3)) = I(5) = (3 * 5 + 1)/2 = 8$ ,  
 $GetS(IIOO, 1, 3)(3) = IIO(3) = O(II(3)) = O(8) = 8/2 = 4$ ,  
 $GetS(IIOO, 1, 4)(3) = IIOO(3) = O(IIO(3)) = O(4) = 4/2 = 2 < 3$ .

(6) It is worth to stress that, although in above definition  $j \geq 1$ , it can be extended to  $j \geq 0$  by assuming  $GetS(\cdot, \cdot, 0)(x) = x$ .

**Example 1.17.** If  $RD[x]$  ( $x \in \mathbb{N}^*$ ,  $x \geq 2$ ) exists, then

(1)  $s(x) < x$ , where  $s = RD[x]$ ;

(2)  $GetS(s, 1, i)(x) \not\leq x$ , where  $i = 1, 2, \dots, |s| - 1$ ;

(3)  $IsMatched(GetS(s, 1, j-1)(x), GetS(s, j, 1)) = True$ , where  $j = 1, 2, \dots, |s|$ .

**Remark 1.18.**

(1)  $s(x)$  is the last transformed integer, or the first transformed integer that is less than the starting integer.

(2)  $GetS(s, 1, i)(x)$  ( $i = 1, 2, \dots, |s| - 1$ ) are all intermediate transformed integers.

(3) When  $j = 1$ ,  $GetS(s, 1, j - 1)(x) = GetS(s, 1, 0)(x) = x$ .  $GetS(s, j, 1)$  is the first transformation.

(4) If  $GetS(s, 1, j - 1)(x)$  ( $j = 2, \dots, |s|$ ) is current transformed integer, then  $GetS(s, j, 1)$  is the next intermediate Collatz transformation.

**Proposition 1.19.**  $\forall x \in \mathbb{N}^*$ ,  $x \geq 2$ , if  $RD[x]$  exists, then  $RD[x]$  is unique.

**Proof** Straightforward. Given  $x$ , either  $I(x)$  or  $O(x)$  is deterministic and unique. Similarly, given  $x$ ,  $s'(x)$  is deterministic and unique, where  $s' = GetS(s, 1, i)$ ,  $s = RD[x]$ ,  $i = 1, 2, \dots, |s|$ . Thus,  $s$  is unique for any given  $x$ .  $\square$

**Remark 1.20.**

We assume  $RD[x = 1] = IO$ , although  $IO(1) = O((3 * 1 + 1)/2) = O(2) = 2/2 = 1 \not\prec x$ . In other words, we assume the reduced dynamics of  $x = 1$  is  $IO$ . In the following, we always concern  $x \geq 2, x \in \mathbb{N}^*$ .

**Proposition 1.21.** Given  $x \in \mathbb{N}^*$ , if  $RD[x]$  exists, then  $RD[x]$  ends by  $O$ .

**Proof** Straightforward due to  $I(x) = (3 * x + 1)/2 > x$ . Suppose  $\exists x \in \mathbb{N}^*$ ,  $x \geq 2$ ,  $s(x) \not\prec x$ ,  $RD[x] = s||I$ . Then,  $\{s||I\}(x) = I(s(x)) > s(x)$ , thus  $RD[x] = \{s||I\}(x) \not\prec x$ . Contradiction occurs.  $\square$

**Proposition 1.22.**  $RD[x \in [0]_2] = O$ ,  $RD[x \in [1]_4] = IO$ .

**Proof** (1)  $x \in [0]_2$ , thus  $O$  occurs.  $x/2 < x$ , thus  $RD[x] = O$ .

(2) If  $x = 1$ ,  $RD[1] = IO$  (by assumption).

If  $x \geq 2$ ,  $x = 4t + 1 \in [1]_2$ , where  $t \in \mathbb{N}^*$ . Thus,  $I$  occurs.  $I(x) = (3 * x + 1)/2 = (3 * (4t + 1) + 1)/2 = (12t + 4)/2 = 2 * (3t + 1) \in [0]_2$ .  $2 * (3t + 1) > x = 4t + 1$ , thus further transformation occurs.  $O(I(x)) = 2 * (3t + 1)/2 = 3t + 1 < 4t + 1 = x$  ( $\because t \in \mathbb{N}^*$ ), thus  $RD[x] = IO$ .  $\square$

## 2. Period Theorem

In this section, we will formally prove  $\text{RD}[x + 2^L] = \text{RD}[x]$ ,  $L = |\text{RD}[x]|$ ,  $\text{RD}[x] \in \{I, O\}^{\geq 1}$  in this section. Note that, interestingly,  $L$  is indeed the count of  $x/2$  computations in reduced dynamics.

### 2.1. Notations and Observations

#### Notation 2.1.

(1)  $\min(S = \{\dots\})$  : The minimal element in a set  $S$ .

(2)  $\max(S = \{\dots\})$  : The maximal element in a set  $S$ .

**Definition 2.2.** *Period.*  $\min(\{P | \text{RD}[x + P] = \text{RD}[x], x, P \in \mathbb{N}^*\})$  is called the period of  $x$ .

#### Remark 2.3.

(1)  $\forall x \in [0]_2$ ,  $\text{RD}[x + 2] = \text{RD}[x] = O$ , period  $P = 2^{|\text{RD}[x]|} = 2^{|O|} = 2^1 = 2$ ;

(2)  $\forall x \in [1]_4$ ,  $\text{RD}[x + 4] = \text{RD}[x] = IO$ , period  $P = 2^{|\text{RD}[x]|} = 2^{|IO|} = 2^2 = 4$ .

(3) We thus concentrate on  $x \in [3]_4$  in the following.

For easily understanding, we point out two concerns in the forthcoming proof.

1. Obviously,  $\text{RD}[3 + 16] = IIOO = \text{RD}[3]$ . We *observed* that during the computing of  $IIOO(3 + 16)$ , intermediate transformed integers (i.e.,  $I(3 + 16)$ ,  $II(3 + 16)$ ,  $IIO(3 + 16)$ ) are odd or even, if and only if  $I(3)$ ,  $II(3)$ ,  $IIO(3)$  are odd or even. Besides, 16 is the minimal integer to satisfy above requirements.
2. Formally speaking, the parity of  $s(x + P)$  and  $s(x)$  are always identical, where  $s = \text{GetS}(\text{RD}[x], 1, i)$ , and  $i = 1, 2, \dots, |\text{RD}[x]| - 1$ . Also, the parity of  $x + P$  and  $x$  are identical. That is, the parity sequence during computing for the reduced dynamics of starting integer  $x + P$  is identical with that of starting integer  $x$ , which results in the occurred Collatz transformations of both are exactly identical. Besides,  $P$  is the minimal integer to satisfy above requirements. This is one concern.
3. The other concern is to prove  $s(x + P) < x + P$  and  $s(x) < x$  where  $s = \text{RD}[x]$ ; Also,  $s(x + P) \not\prec x + P$  and  $s(x) \not\prec x$  where  $s = \text{GetS}(\text{RD}[x], 1, i)$ ,  $i = 1, 2, \dots, |\text{RD}[x]| - 1$ .

A new notation  $I'(\cdot)$  is introduced hereby to reveal the relations among  $I(x + P)$ ,  $I(x)$  and  $I'(P)$ .

**Notation 2.4.**  $I'(x) = (3 * x)/2$ .

**Example 2.5.**

(1)  $I(3 + 16) = (3(3 + 16) + 1)/2 = (3 * 3 + 1)/2 + 3 * 16/2 = I(3) + I'(16)$ ,  
 $I(3) = (3 * 3 + 1)/2 = 5$ ,  $I'(16) = 3 * 16/2 = 24 \in [0]_2$ ,  $5 > 3$ ,  $5 + 24 > 3 + 16$ .  
*Thus, either next transformation for  $3+16$  and  $3$  is  $I$ .*

(2)  $II(3 + 16) = I(I(3) + I'(16)) = (3 * (I(3) + I'(16)) + 1)/2 = (3I(3) + 1)/2 + 3I'(16)/2 = II(3) + I'I'(16)$ ,  
 $II(3) = I(5) = (3 * 5 + 1)/2 = 8$ ,  $I'I'(16) = I'(24) = 3 * 24/2 = 36 \in [0]_2$ ,  
 $8 > 3$ ,  $8 + 36 > (3 + 16)$ . *Thus, either next transformation is  $O$ .*

(3)  $IIO(3 + 16) = O(II(3) + I'I'(16)) = IIO(3) + I'I'O(16)$ ,  
 $IIO(3) = 8/2 = 4$ ,  $I'I'O(16) = 36/2 = 18 \in [0]_2$ ,  $4 > 3$ ,  $4 + 18 = 22 > (3 + 16)$ . *Thus, either next transformation is  $O$ .*

(4)  $IIOO(3 + 16) = O(IIO(3) + I'I'O(16)) = IIOO(3) + I'I'OO(16)$   
 $IIOO(3) = 4/2 = 2$ ,  $I'I'OO(16) = 18/2 = 9$ ,  $2 < 3$ ,  $2 + 9 = 11 < (3 + 16)$ .  
*Thus, either reduced dynamics ends.*

**Remark 2.6.** *In above example we can observe that  $I'(16)$ ,  $I'I'(16)$ ,  $I'I'O(16)$  are always remained even. Thus, they do not influence the resulting next Collatz transformation (i.e., “ $I$ ” or “ $O$ ”) during computing for the reduced dynamics of starting integer 3. Therefore, the parity of  $s(x + 16)$  and  $s(x)$  always maintain to be identical, where  $x = 3$ ,  $s = \text{GetS}(\text{RD}[x], 1, i)$ , and  $i = 1, 2, \dots, |\text{RD}[x]| - 1$ .*

For better presentation, we thus introduce two functions as follows:

**Definition 2.7.**  $IsEven : x \rightarrow \text{bool}$ . *It takes as input  $x \in \mathbb{N}^*$ , and outputs  $\text{bool} \in \{\text{True}, \text{False}\}$ , where  $\text{bool} = \text{True}$  if  $x \in [0]_2$  and  $\text{bool} = \text{False}$  if  $x \in [1]_2$ .*

**Definition 2.8.**  $Replace : s \rightarrow s'$ . *It takes as input  $s \in \{I, O\}^{\geq 1}$ , and outputs  $s' \in \{I', O\}^{\geq 1}$ , where  $\text{GetS}(s', i, 1) = I'$  if  $\text{GetS}(s, i, 1) = I$ , and  $\text{GetS}(s', i, 1) = O$  if  $\text{GetS}(s, i, 1) = O$ , for  $i = 1, 2, \dots, |s|$ .*

**Remark 2.9.**

(1) Simply speaking, replacing all “I” in “s” respectively by “I’” will result in “s’”.

(2) Obviously,  $\forall s \in \{I, O\}^{\geq 1}$ ,  $|s'| = |s|$  where  $s' = \text{Replace}(s)$ .

(3) By using above introduced functions, we can restate the reason in Remark 2.6 as follows:  $\text{GetS}(s', 1, i) \in [0]_2$ , where  $s' = \text{Replace}(s)$ ,  $i = 1, 2, \dots, |s'| - 1$ , thus the parity of  $s(x + 16)$  and  $s(x)$  are always identical, where  $x = 3$ ,  $s = \text{GetS}(\text{RD}[x], 1, i)$ , and  $i = 1, 2, \dots, |\text{RD}[x]| - 1$ .

## 2.2. The Proof of Period Theorem

**Lemma 2.10.** *If  $P \in [0]_2, x \in \mathbb{N}^*$ , then  $\text{IsEven}(x + P) = \text{IsEven}(x)$ .*

**Proof** Straightforward. Due to  $P \in [0]_2$ , if  $x \in [1]_2$ , then  $x + P \in [1]_2$ ; If  $x \in [0]_2$ , then  $x + P \in [0]_2$ . Thus,  $\text{IsEven}(x + P) = \text{IsEven}(x)$ .  $\square$

**Remark 2.11.** *Above lemma states that if  $P \in [0]_2$ , the first Collatz transformation of  $x + P$  is identical with that of  $x$ .*

**Lemma 2.12.**  *$s(x + P) = s(x) + s'(P)$ , where  $s \in \{I, O\}$ ,  $s' = \text{Replace}(s)$ ,  $x \in \mathbb{N}^*, P \in [0]_2$ .*

**Proof**  $\text{IsEven}(x + P) = \text{IsEven}(x)$  because  $P \in [0]_2$ , due to Lemma 2.10. Thus, the first Collatz transformation of  $x + P$  and the first Collatz transformation of  $x$  are identical.

(1) Suppose  $x \in [1]_2$ , so  $s = I$ . Thus,  $s' = \text{Replace}(s) = I'$ .

$s(x + P) = I(x + P) = 3((x + P) + 1)/2 = (3x + 1)/2 + 3 * P/2 = I(x) + I'(P) = s(x) + s'(P)$ .

(2) Suppose  $x \in [0]_2$ , so  $s = O$ . Thus,  $s' = \text{Replace}(s) = O$ .

$s(x + P) = O(x + P) = (x + P)/2 = x/2 + P/2 = O(x) + O(P) = s(x) + s'(P)$ .

Summarizing (1) and (2),  $s(x + P) = s(x) + s'(P)$ .  $\square$

**Lemma 2.13. (Separation Lemma.)** *Suppose  $x \in \mathbb{N}^*$ ,  $s \in \{I, O\}^{\geq 2}$ ,  $s' = \text{Replace}(s)$ . If  $\text{GetS}(s', 1, j)(P) \in [0]_2$ ,  $j = 0, 1, 2, \dots, |s| - 1$ , then*

(1)  $\text{IsEven}(\text{GetS}(s, 1, j)(x + P)) = \text{IsEven}(\text{GetS}(s, 1, j)(x))$ ;

(2)  $\text{GetS}(s, 1, j + 1)(x + P) = \text{GetS}(s, 1, j + 1)(x) + \text{GetS}(s', 1, j + 1)(P)$ .

**Proof** (1)  $j = 0$ .

(1.1)  $GetS(s', 1, j)(P) \in [0]_2$ . Besides,  $GetS(s', 1, j)(P) = GetS(s', 1, 0)(P) = P$ . Thus,  $P \in [0]_2$ . Thus,  $IsEven(x + P) = IsEven(x)$  due to Lemma 2.10. Thus, the intermediate next Collatz transformation of  $x + P$  and  $x$  are identical.

$$\begin{aligned}
& (1.2) \quad GetS(s, 1, j + 1)(x + P) \\
& = GetS(s, 1, 1)(x + P) \quad \because j = 0 \\
& = GetS(s, 1, 1)(x) + GetS(s', 1, 1)(P) \quad \because \text{Lemma 2.12} \\
& = GetS(s, 1, j + 1)(x) + GetS(s', 1, j + 1)(P).
\end{aligned}$$

(2)  $j = 1$ .

(2.1) Due to (1),  $GetS(s, 1, 1)(x + P) = GetS(s, 1, 1)(x) + GetS(s', 1, 1)(P)$ .

Besides,  $GetS(s', 1, 1)(P) \in [0]_2$ . Thus,

$IsEven(GetS(s, 1, 1)(x + P)) = IsEven(GetS(s, 1, 1)(x))$ . Thus, the intermediate next Collatz transformation of  $x + P$  and  $x$  are identical.

(2.2) There exists two cases as follows:

(2.2.1) If  $GetS(s, 1, j + 1) = GetS(s, 1, j) \parallel I$ , then

$$\begin{aligned}
& GetS(s, 1, j + 1)(x + P) \\
& = (GetS(s, 1, j) \parallel I)(x + P) \\
& = I(GetS(s, 1, j)(x + P)) \\
& = I(GetS(s, 1, j)(x) + GetS(s', 1, j)(P)) \quad \because (1.2) \\
& = (3(GetS(s, 1, j)(x) + GetS(s', 1, j)(P)) + 1)/2 \\
& = (3 * GetS(s, 1, j)(x) + 1)/2 + 3 * GetS(s', 1, j)(P)/2 \\
& = I(GetS(s, 1, j)(x)) + I'(GetS(s', 1, j)(P)) \\
& = (GetS(s, 1, j) \parallel I)(x) + (GetS(s', 1, j) \parallel I')(P) \\
& = GetS(s, 1, j + 1)(x) + GetS(s', 1, j + 1)(P).
\end{aligned}$$

(2.2.2) If  $GetS(s, 1, j + 1) = GetS(s, 1, j) \parallel O$ , then

$$\begin{aligned}
& GetS(s, 1, j + 1)(x + P) \\
& = GetS(s, 1, j) \parallel O(x + P) \\
& = O(GetS(s, 1, j)(x + P)) \\
& = O(GetS(s, 1, j)(x) + GetS(s', 1, j)(P)), \quad \because (1.2) \\
& = (GetS(s, 1, j)(x) + GetS(s', 1, j)(P))/2 \\
& = O(GetS(s, 1, j)(x)) + O(GetS(s', 1, j)(P)) \\
& = O(GetS(s, 1, j)(x)) + O(GetS(s', 1, j)(P)) \\
& = (GetS(s, 1, j) \parallel O)(x) + (GetS(s', 1, j) \parallel O)(P) \\
& = GetS(s, 1, j + 1)(x) + GetS(s', 1, j + 1)(P).
\end{aligned}$$

(Note that, here  $j + 1 = 2$ . Recall that “ $\parallel$ ” is concatenation.)

(3) Similarly,  $j = 2$ .

Due to (2),  $GetS(s, 1, 2)(x + P) = GetS(s, 1, 2)(x) + GetS(s', 1, 2)(P)$ .

Besides,  $GetS(s', 1, 2)(P) \in [0]_2$ . Thus,  
 $IsEven(GetS(s, 1, j)(x + P)) = IsEven(GetS(s, 1, j)(x))$ . Thus, the intermediate next Collatz transformation of  $x + P$  and  $x$  are identical.

(Note that, here  $j = 2$ ).

Again, we can prove the following similar to (2.2).

$GetS(s, 1, j + 1)(x + P) = GetS(s, 1, j + 1)(x) + GetS(s', 1, j + 1)(P)$ .

(Note that, here  $j + 1 = 3$ .)

(4) Similarly, we can prove  $j = 3, \dots, |s'| - 1$ , respectively and especially in an order.  $\square$

**Remark 2.14.**

(1) Obviously, above conclusion can be extended to include  $|s| = 1$  by Lemma 2.10 and Lemma 2.12.

(2) Separation Lemma states the sufficient condition (i.e.,  $GetS(s', 1, j)(P) \in [0]_2$ ,  $j = 0, 1, 2, \dots, |s| - 1$ ) for guaranteeing that all intermediate parities of transformed integers  $x + P$  are exactly identical with those of  $x$ .

(3) Separation Lemma is general, as  $s$  could be either original dynamics or reduced dynamics of certain  $x \in \mathbb{N}^*$ . That is the reason we give a special name to this lemma for emphasizing its importance.

(4) In above proof, we assume the number of transformations will be  $s$ . Note that, it does not influence the conclusion, as we can use condition “ $j = 0, 1, 2, \dots$ ” instead of “ $j = 0, 1, 2, \dots, |s| - 1$ ” to omit the assumption on the number of transformations.

We next explore how to compute  $GetS(s', 1, j)(x)$ .

**Definition 2.15.** Function  $CntI(\cdot)$ .  $CntI : s \rightarrow n$ . It takes as input  $s \in \{I, O\}^{\geq 1}$ , and outputs  $n \in \mathbb{N}$  that is the count of  $I$  in  $s$ .

**Example 2.16.**  $CntI(IIOO) = 2$ ,  $CntI(III) = 3$ . Obviously, the function name stems from “Count the number of  $I$ ”.

**Lemma 2.17.** Suppose  $s \in \{I, O\}^{\geq 1}$ ,  $s' = Replace(s)$ ,  $x \in \mathbb{N}^*$ , we have

$$GetS(s', 1, j)(x) = \frac{3^{CntI(GetS(s, 1, j))}}{2^j} * x, \quad j = 1, 2, \dots, |s|.$$

**Proof** (1)  $|s| = 1$ . Thus,  $j = 1$ .

(1.1) If  $s = I$ , then  $s' = \text{Replace}(s) = I'$ .

$$\begin{aligned} \text{GetS}(s', 1, j)(x) &= \text{GetS}(I', 1, 1)(x) = I'(x) = 3 * x/2 = 3^1 * x/2^1 \\ &= 3^{\text{CntI}(I)} * x/2^{|I|} = 3^{\text{CntI}(\text{GetS}(s, 1, j))} / 2^j * x. \end{aligned}$$

(1.2) If  $s = O$ , then  $s' = \text{Replace}(s) = O$ .

$$\begin{aligned} \text{GetS}(s', 1, j)(x) &= O(x) = x/2 = 3^0 * x/2^1 = 3^{\text{CntI}(O)} * x/2^{|O|} \\ &= 3^{\text{CntI}(\text{GetS}(s, 1, j))} / 2^j * x. \end{aligned}$$

(2)  $|s| \geq 2$ .

(2.1)  $j = 1$ .

(2.1.1) If  $\text{GetS}(s, 1, 1) = I$ , then

$$\begin{aligned} \text{GetS}(s', 1, j)(x) &= I'(x) = 3 * x/2 = 3^1 * x/2^1 = 3^{\text{CntI}(I)} * x/2^{|I|} = \\ &= 3^{\text{CntI}(\text{GetS}(s, 1, j))} / 2^j * x. \end{aligned}$$

(2.1.2) If  $\text{GetS}(s, 1, 1) = O$ , then

$$\begin{aligned} \text{GetS}(s', 1, j)(x) &= O(x) = x/2 = 3^0 * x/2^1 = 3^{\text{CntI}(O)} * x/2^{|O|} \\ &= 3^{\text{CntI}(\text{GetS}(s, 1, j))} / 2^j * x. \end{aligned}$$

(2.2) Iteratively, for  $j = 1, 2, \dots, |s'| - 1$  in an order (recall that  $|s'| = |s|$ ).

(2.2.1) If  $\text{GetS}(s, 2, 1) = I$ , then

$$\begin{aligned} &\text{GetS}(s', 1, j + 1)(x) \\ &= (\text{GetS}(s', 1, j) \| I')(x) \\ &= I'(\text{GetS}(s', 1, j)(x)) \\ &= I'(\text{GetS}(s', 1, j)(x)) \\ &= 3 * \text{GetS}(s', 1, j)(x) / 2 \\ &= 3 * \frac{3^{\text{CntI}(\text{GetS}(s, 1, j))}}{2^j} * x/2 \quad \because (2.1) \text{ for } j = 1, (2.2) \text{ for } j = 2, \dots, |s'| - 1 \\ &= \frac{3^{\text{CntI}(\text{GetS}(s, 1, j)) + 1}}{2^{j+1}} * x \\ &= \frac{3^{\text{CntI}(\text{GetS}(s, 1, j+1))}}{2^{j+1}} * x. \quad \because \text{GetS}(s, 1, j + 1) = \text{GetS}(s, 1, j) \| I \end{aligned}$$

(2.2.2) If  $\text{GetS}(s, 2, 1) = O$ , then

$$\begin{aligned} &\text{GetS}(s', 1, j + 1)(x) \\ &= (\text{GetS}(s', 1, j) \| O)(x) \\ &= O(\text{GetS}(s', 1, j)(x)) \\ &= \text{GetS}(s', 1, j)(x) / 2 \\ &= \frac{3^{\text{CntI}(\text{GetS}(s, 1, j))}}{2^j} * x/2 \\ &= \frac{3^{\text{CntI}(\text{GetS}(s, 1, j))}}{2^{j+1}} * x \\ &= \frac{3^{\text{CntI}(\text{GetS}(s, 1, j+1))}}{2^{j+1}} * x. \quad \because \text{GetS}(s, 1, j + 1) = \text{GetS}(s, 1, j) \| O \end{aligned} \quad \square$$

**Remark 2.18.**

(1) Recall that  $s' = \text{GetS}(s', 1, |s|)$  and  $s = \text{GetS}(s, 1, |s|)$ . Thus, when  $j = |s|$ , then  $s'(x) = \text{GetS}(s', 1, |s|)(x) = \frac{3^{\text{CntI}(\text{GetS}(s, 1, |s|))}}{2^{|s|}} * x = \frac{3^{\text{CntI}(s)}}{2^{|s|}} * x$ .

(2) Indeed,  $j = |\text{GetS}(s, 1, j)|$ , thus the lemma can be restated as  $\text{GetS}(s', 1, j)(x) = \frac{3^{\text{CntI}(\text{GetS}(s, 1, j))}}{2^{|\text{GetS}(s, 1, j)|}} * x$ ,  $j = 1, 2, \dots, |s|$ .

**Lemma 2.19.**

$$\min(\{P | \text{GetS}(s', 1, j)(P) \in [0]_2, j = 0, 1, 2, \dots, |s| - 1, \\ s \in \{I, O\}^{\geq 2}, s' = \text{Replace}(s), P \in \mathbb{N}^*\}) = 2^{|s|}.$$

**Proof** (1)  $j = 0$ ,  $\text{GetS}(s', 1, j)(P) \in [0]_2 \Leftrightarrow P \in [0]_2$ .

(2)  $j = 1, 2, \dots, |s| - 1$ .

By Lemma 2.17,  $\text{GetS}(s', 1, j)(x) = \frac{3^{\text{CntI}(\text{GetS}(s, 1, j))}}{2^j} * x$ . Thus,

$$\begin{aligned} \text{GetS}(s', 1, j)(P) &= \frac{3^{\text{CntI}(\text{GetS}(s, 1, j))}}{2^j} * P \in [0]_2, j = 1, 2, \dots, |s| - 1 \\ \Leftrightarrow P/2^j &\in [0]_2, j = 1, 2, \dots, |s| - 1 \\ \Leftrightarrow P/2^{j+1} &\in \mathbb{N}^*, j = 1, 2, \dots, |s| - 1 \\ \Leftrightarrow P/2^{|s|} &\in \mathbb{N}^* \\ \Leftrightarrow \min(P) &= 2^{|s|}. \end{aligned}$$

By (1) and (2),  $\min(P) = 2^{|s|}$ . □

**Notation 2.20.**  $\text{Set}_{\text{RD}} = \{s | x \in \mathbb{N}^*, \exists \text{RD}[x], s = \text{RD}[x] \in \{I, O\}^{\geq 1}\}$ .

Simply speaking,  $\text{Set}_{\text{RD}}$  is a set of all reduced dynamics for those  $x \in \mathbb{N}^*$  if reduced dynamics of  $x$  exists. That is,  $\forall x \in \mathbb{N}^*$ , if  $\text{RD}[x]$  exists, then  $\text{RD}[x] = s$  will be included in  $\text{Set}_{\text{RD}}$ , which is a set of existing reduced dynamics.

**Lemma 2.21.**  $s'(P) < P$ , where  $s \in \text{Set}_{\text{RD}}$ ,  $s' = \text{Replace}(s)$ ,  $P \in \mathbb{N}^*$ .

**Proof** By Lemma 2.17 and Remark 2.18 (1),

$$s'(P) = 3^{\text{CntI}(s)} / 2^{|s|} * P.$$

Due to Corollary 2.31,  $\frac{3^{\text{CntI}(s)}}{2^{|s|}} < 1$ . Thus,  $3^{\text{CntI}(s)} / 2^{|s|} * P < P$ . □

**Remark 2.22.** Here we use Corollary 2.31 (Form Corollary) given in Appendix, which is formally proved by us in another paper [7].

**Lemma 2.23.**  $\text{GetS}(s', 1, j)(P) > P$ ,  $j = 1, 2, \dots, |s| - 1$ , where  $s \in \text{Set}_{\text{RD}}$ ,  $s' = \text{Replace}(s)$ ,  $P \in \mathbb{N}^*$ .

**Proof** By Lemma 2.17,

$$GetS(s', 1, j)(P) = \frac{3^{CntI(GetS(s,1,j))}}{2^j} * P.$$

Due to Corollary 2.32,  $\frac{3^{CntI(GetS(s,1,j))}}{2^j} > 1$ . Thus  $\frac{3^{CntI(GetS(s,1,j))}}{2^j} * P > P$ .  $\square$

**Theorem 2.24. (Period Theorem.)** *If  $RD[x] \in \{I, O\}^{\geq 1}$  exists, then  $RD[x + 2^L]$  exists and  $RD[x + 2^L] = RD[x]$ ,  $L = |RD[x]|$ .*

**Proof** (1) Regarding two special cases:

$$RD[x \in [0]_2] = O, RD[x + 2^L] = RD[x], L = |RD[x]| = |O| = 1.$$

$$RD[x \in [1]_4] = IO, RD[x + 2^L] = RD[x], L = |RD[x]| = |IO| = 2.$$

(2) Next, w.l.o.g., suppose  $RD[x] = s$ ,  $|s| \geq 3$ . Let  $P = 2^L = 2^{|RD[x]|} = 2^{|s|}$ ,  $s' = Replace(s)$ .

Thus,  $GetS(s', 1, j)(P) \in [0]_2, j = 0, 1, 2, \dots, |s| - 1$ , and  $P$  is the minimal integer for this requirement by Lemma 2.19.

(2.1) Regarding the ordered parity sequence of  $x + P$  and  $x$ .

$P \in [0]_2$ , thus the first transformation of  $x + P$  and  $x$  is identical by Lemma 2.10.

$GetS(s, 1, j+1)(x+P) = GetS(s, 1, j+1)(x) + GetS(s', 1, j+1)(P)$  where  $s' = Replace(s)$ , by Lemma 2.13 (i.e., *Separation Lemma*). The parity for all transformed integers for  $x + P$  and  $P$  (except for the last one) are exactly identical due to  $GetS(s', 1, j)(P) \in [0]_2, j = 0, 1, 2, \dots, |s| - 1$ .

(2.2) Regarding the comparison between transformed integers and starting integer.

(2.2.1)  $s(x) < x$  due to the definition of  $RD[x]$ .

(2.2.2)  $GetS(s, 1, j)(x) \not\prec x, j = 1, 2, \dots, |s| - 1$ , due to the definition of  $RD[x]$ .

(2.2.3)  $s'(P) < P$  by Lemma 2.21.

(2.2.4)  $GetS(s', 1, j)(P) \not\prec P, j = 1, 2, \dots, |s'| - 1$ , by Lemma 2.23.

(2.2.5)  $s(x + P) = s(x) + s'(P) < x + P$ , by Lemma 2.13, (2.2.1) and (2.2.3).

(2.2.6)  $GetS(s, 1, j)(x + P) = GetS(s, 1, j)(x) + GetS(s', 1, j)(P) \not\prec x + P, j = 1, 2, \dots, |s| - 1$ , due to (2.2.2) and (2.2.4).

Thus, only the last transformed integer  $s(x + P)$  is less than the starting integer  $x + P$ , and the other transformed integers are not less than the starting integer  $x + P$ . Thus,  $RD[x + P] = s$ .

Due to (2.1) and (2.2),  $RD[x + P] = RD[x], P = 2^L, L = |s| \geq 3$ .

Summarizing (1) and (2),  $RD[x + 2^L]$  exists and  $RD[x + 2^L] = RD[x]$  where  $L = |RD[x]| \in \mathbb{N}^*$ .  $\square$

**Remark 2.25.** *Interestingly, the period equals 2 to the power of the count of  $x/2$  in reduced dynamics, as the count of  $x/2$  equals to the length of  $\text{RD}[x]$ . The count of  $x/2$  has two folders: one equals the count of “I” due to  $(3 * x + 1)/2$ ; the other equals the count of “O” due to  $x/2$ . Obviously, the count of “I” add the count of “O” equals the length of  $\text{RD}[x]$ .*

**Corollary 2.26.** *If reduced dynamics of  $x \in \mathbb{N}^*$  exists, denoted as  $\text{RD}[x] \in \{I, O\}^L, L \in \mathbb{N}^*$ , then reduced dynamics of  $x + k * 2^L$  ( $k \in \mathbb{N}^*$ ) exists and is identical with that of  $x$ .*

**Corollary 2.27.**  $\forall s \in \text{Set}_{\text{RD}}, \|\{x | \text{RD}[x] = s\}\| = \|\mathbb{N}^*\| = \aleph_0$  where “ $\|\cdot\|$ ” returns the number of a set.

**Proof** Let  $2^{|s|} = P \in \mathbb{N}^*$ .  $\text{RD}[x + P] = s = \text{RD}[x]$ . Thus,  $\text{RD}[x + k * P] = s = \text{RD}[x], k \in \mathbb{N}^*$ . A bijective mapping from  $\{x | \text{RD}[x] = s\}$  to  $\mathbb{N}^*$  can be created as follows:  $x \leftrightarrow 1, x + k * P \leftrightarrow k + 1, k \in \mathbb{N}^*$ . Thus,  $\|\{x | \text{RD}[x] = s\}\| = \|\mathbb{N}^*\| = \aleph_0$ .  $\square$

## Acknowledgement

The research was financially supported by National Natural Science Foundation of China (No.61972366), Major Scientific and Technological Special Project of Guizhou Province (No. 20183001), the Foundation of Key Laboratory of Network Assessment Technology, Chinese Academy of Sciences (No. KFKT2019-003), and the Foundation of Guizhou Provincial Key Laboratory of Public Big Data (No. 2018BDKFJJ009, No. 2019BDKFJJ003, No. 2019BDKFJJ011).

## References

- [1] Tomas Oliveira e Silva, *Maximum excursion and stopping time record-holders for the  $3x+1$  problem: computational results*, *Mathematics of Computation*, vol. 68, no. 225, pp. 371-384, 1999.
- [2] Tomas Oliveira e Silva, *Empirical Verification of the  $3x+1$  and Related Conjectures*. In *The Ultimate Challenge: The  $3x+1$  Problem*, (book edited by Jeffrey C. Lagarias), pp. 189-207, American Mathematical Society, 2010.

- [3] Wei Ren, Simin Li, Ruiyang Xiao and Wei Bi, *Collatz Conjecture for  $2^{100000} - 1$  is True - Algorithms for Verifying Extremely Large Numbers*, Proc. of IEEE UIC 2018, Oct. 2018, Guangzhou, China, 411-416, 2018
- [4] Wei Ren, *A New Approach on Proving Collatz Conjecture*, Journal of Mathematics, Hindawi, April 2019, ID 6129836, <https://www.hindawi.com/journals/jmath/2019/6129836/>.
- [5] Wei Ren, *Ratio and Partition are Revealed in Proposed Graph on Reduced Collatz Dynamics*, Proc. of 2019 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCLOUD/SocialCom/SustainCom), pp. 474-483, 16-28 Dec. 2019, Ximen, China
- [6] Wei Ren, Ruiyang Xiao, *How to Fast Verify Collatz Conjecture by Automata*, Proc. of IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), pp. 2720-2729, 10-12 Aug. 2019, Zhangjiajie, China
- [7] W. Ren, *A Reduced Collatz Dynamics Maps to a Residue Class, and its Count of  $x/2$  over Count of  $3*x+1$  is larger than  $\ln 3/\ln 2$* , Journal of Number Theory, Elsevier, Submitted, 2020.
- [8] Wei Ren, *Reduced Collatz Dynamics Data Reveals Properties for the Future Proof of Collatz Conjecture*, Data, MDPI, 2019, 4, 89, doi:10.3390/data4020089, <https://www.mdpi.com/2306-5729/4/2/89/pdf>.

## Appendix

In our another paper [7], we proved the following *Form Corollary* that states the requirements on the count of “ $O$ ” and the count of “ $I$ ” in any reduced dynamics.

**Definition 2.28.**  $CntO : s \rightarrow n$ . It takes as input  $s \in \{I, O\}^{\geq 1}$ , and outputs  $n \in \mathbb{N}^*$  that is the count of “ $O$ ” in  $s$ .

E.g.,  $CntO(IIOO) = 2$ ,  $CntO(IO) = 1$ .

**Corollary 2.29. (*Form Corollary*.)**  $\forall s \in \{I, O\}^{\geq 1}$ ,  $s \in Set_{RD}$ , if and only if

(1)  $s = O$ ; Or,

(2)  $CntO(s) = \lceil \log_2 1.5 * CntI(s) \rceil$  and  $CntO(s') < \lceil \log_2 1.5 * CntI(s') \rceil$

where  $s' = GetS(s, 1, i)$ ,  $i = 1, 2, \dots, |s| - 1$ ,  $|s| \geq 2$ .

Following conclusions are all derived from above *Form Corollary*.

Following corollary states that  $Set_{RD}$  can be constructed by generating  $s \in \{I, O\}^{\geq 1}$  that satisfies aforementioned requirements instead of by conducting concrete Collatz transformations for all  $x \in \mathbb{N}^*$ .

**Corollary 2.30.**  $Set_{RD} = \{O\} \cup \{s | s \in \{I, O\}^L, L \in \mathbb{N}^*, L \geq 2,$   
 $CntO(s) = \lceil \log_2 1.5 * CntI(s) \rceil,$   
 $CntO(s') < \lceil \log_2 1.5 * CntI(s') \rceil, s' = GetS(s, 1, i), i = 1, 2, \dots, L - 1\}$ .

**Proof** It is straightforward due to Corollary 2.29. □

Following corollary states the relations between  $CntI(s)$  and  $CntO(s) + CntI(s) = |s|$ ,  $s \in Set_{RD}$ . Note that,  $CntI(s)$  is indeed equal to the count of  $3 * x + 1$  computation, and  $|s|$  is indeed equal to the total count of  $x/2$  computation in reduced dynamics.

**Corollary 2.31.**  $s \in \{I, O\}^{\geq 1}$ ,  $s \in Set_{RD}$ , we have  
 (1)  $|s| \geq \lceil \log_2 3 * CntI(s) \rceil$ ; (2)  $3^{CntI(s)} < 2^{|s|}$ .

**Proof** (1) When  $s = O$ ,  $CntI(s) = 0$ ,  $|s| = 1$ .  $1 > \lceil \log_2 3 * 0 \rceil = 0$  by Corollary 2.30.

When  $s \neq O$ ,  $s \in Set_{RD} \Rightarrow |s| = \lceil \log_2 1.5 * CntI(s) \rceil + CntI(s) = \lceil \log_2 1.5 * CntI(s) + CntI(s) \rceil = \lceil \log_2 3 * CntI(s) \rceil$  by Corollary 2.30.

In summary,  $s \in Set_{RD} \Rightarrow |s| \geq \lceil \log_2 3 * CntI(s) \rceil$ , and note that “ $>$ ” is obtained *when and only when*  $s = O$ .

(2)  $|s| \geq \lceil \log_2 3 * CntI(s) \rceil \Rightarrow |s| \geq \log_2 3^{CntI(s)} \Rightarrow 3^{CntI(s)} \leq 2^{|s|} \Rightarrow 3^{CntI(s)} < 2^{|s|}$ .  $\square$

**Corollary 2.32.**  $s \in Set_{RD}, s \in \{I, O\}^{\geq 2}$ , we have  $3^{CntI(GetS(s,1,j))} > 2^j$ ,  $j = 1, 2, \dots, |s| - 1$ .

**Proof** Let  $s' = GetS(s, 1, j), j = 1, 2, \dots, |s| - 1$ . Obviously,  $|s'| = j$ .

$s \in Set_{RD}$   
 $\Rightarrow CntO(s') < \lceil \log_2 1.5 * CntI(s') \rceil \quad \because \text{Corollary 2.29}$   
 $\Rightarrow CntO(s') < \log_2 1.5 * CntI(s') \quad \because \log_2 1.5 \notin \mathbb{Q}, CntI(s), CntO(s) \in \mathbb{N}^*$   
 $\Rightarrow CntO(s') + CntI(s') < \log_2 3 * CntI(s')$   
 $\Rightarrow |s'| < \log_2 3 * CntI(s') \Rightarrow 2^{|s'|} < 3^{CntI(s')}$   
 $\Rightarrow 3^{CntI(GetS(s,1,j))} > 2^j$ .  $\square$