Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 7, pp. 2621–2640
DOI : 10.47974/JDMSC-2232

Collatz conjecture–driven cryptographic framework

Rajitha Ranasinghe *
Vikum Bandara †
Department of Mathematics
University of Peradeniya
Peradeniya 20400
Sri Lanka

Abstract

The Collatz conjecture, introduced by Lothar Collatz in 1937, a longstanding unsolved problem, centers on a simple yet a puzzling question: *Does repeatedly applying a specific rule to any positive integer always lead to the number* 1 ? This paper explores the Collatz conjecture from two directions. First, we propose a novel approach to understanding the conjecture's behavior, which involves constructing a framework where all Collatz sequences converge upwards (meaning their terms get larger) and define a function, g, for odd numbers. This function, however, generates sequences that diverge downwards (terms get smaller) within the structure. Notably, for each term in the divergent sequence produced by g, there exists a corresponding convergent sequence generated by the Collatz function itself. The paper then demonstrates that g diverges for any positive odd integer. These findings provide strong evidence in favor of the Collatz conjecture holding true for all positive integers. The second part of the paper delves into cryptography proposing a novel cryptosystem based on the chaotic nature of the Collatz sequence. In this system, a secret key pair is shared between the sender and the receiver. The message is encrypted using a large, odd number (part of the secret key) and a specific number of iterations of the Collatz sequence applied to that number. The chaotic property of the sequence ensures that even slight changes in the starting number significantly alter the resulting sequence, making the encryption process secure. This cryptosystem boasts advantages in its simplicity – it relies solely on multiplication and addition – and its efficiency. Furthermore, its potential lies in building even more secure cryptosystems through repeated encryption or integration with other encryption methods.

Subject Classification (2010): *11B83, 11Y16, 94A60.*

Keywords: *Chaos-based cryptography, Collatz sequence, Cryptography.*

* *E-mail:* **rajithamath@sci.pdn.ac.lk (Corresponding Author)**
† *E-mail:* **vikumb@sci.pdn.ac.lk**

1. Introduction

1.1 *Mathematical background of the Collatz conjecture*

In 1937, the German mathematician Lothar Collatz formulated a conjecture asserting that, regardless of the initial positive integer, the iterative sequence defined by the rule "divide by 2 if even, multiply by 3 and add 1 if odd" will eventually reach 1 [1]. What appears to be a trivial observation endures as a profound enigma, with neither proof nor counterexample discovered to date. Paul Erdös famously remarked that mathematics may not be prepared for such problems [3], a sentiment reinforced by Jeffrey Lagarias's comprehensive survey on related results [6]. More recent results includes significant theoretical advances: Ren [8] demonstrated that validating his Reduced Collatz Conjecture (RCC) would suffice to prove the original conjecture, while Terence Tao, perhaps the most notable recent development, demonstrated that most Collatz map orbits achieve nearly bounded values [10]. Also, recently [2] presented an algebraic construction of near-bent functions, emphasizing their cryptographic applications due to properties like high nonlinearity and resilience against attacks. It explores the use of these functions in designing secure cryptographic systems. The findings contribute to the broader understanding of Boolean functions in cryptography, particularly in enhancing security measures.

1.2 *The* $3x + 1$ *Problem* [6]

Consider the set $\mathbb{Z}^+$ of positive integers and let $n \in \mathbb{Z}^+$. Collatz introduced the map

$$T(n) = \begin{cases} \frac{n}{2} & \text{if } n \text{ is even,} \\ 3n+1 & \text{if } n \text{ is odd.} \end{cases}$$

By repeatedly applying T, one obtains an orbit of n:

$$T(n), T^2(n) = T(T(n)), \cdots, T^k(n) = T(T^{k-1}(n))`, k \in \mathbb{Z}^+.$$

The Collatz conjecture asserts that the sequence obtained by iterating T always reaches 1, no matter from which positive integer n begins the sequence with. One way to think about the $3x + 1$ problem is as a directed graph with positive integer vertices and directed edges from n to T. In honor of Collatz, we call this graph the *Collatz graph of* $T(n)$. Figure 1 depicts a section of the Collatz graph of $T(n)$.

A directed graph is said to be *weakly connected*, if it is connected when viewed as an undirected graph, i.e., for any two vertices, there is a path of edges joining them, ignoring the directions on the edges. The $3x + 1$ Conjecture can be formulated in terms of the Collatz graph as follows.

1.3 *The* $3x + 1$ *Conjecture (First form)*

Consider the Collatz graph of $T(n)$ over the set of positive integers. This graph is weakly connected, and the sequence of iterates

$$(n, T(n), T^{(2)}(n), T^{(3)}(n), \cdots)$$

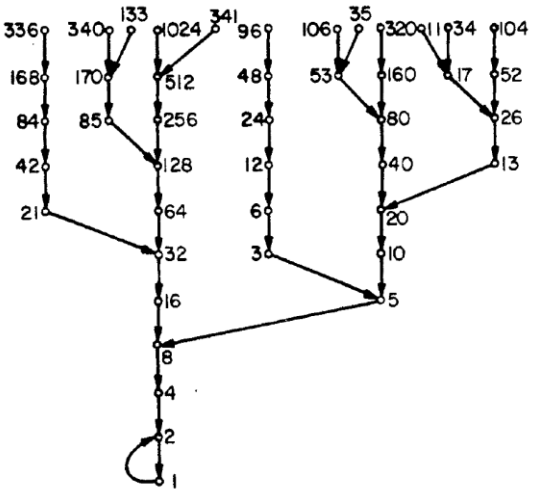

Figure 1
A portion of the Collatz graph

is called the trajectory of n. For $n > 0$, there are three possible long-term behaviors of such trajectories:

- Convergent trajectory. For some $n \in \mathbb{Z}^+$, $T^{(k)}(n) = 1$, then the trajectory of n eventually reaches 1.
- Non-trivial cyclic trajectory. The sequence $\{T^{(k)}(n)\}$ becomes periodic but does not reach 1. In this case, the trajectory falls into a repeating cycle other than the trivial loop $\{1,4,2\}$.
- Divergent trajectory. $\lim_{k\to\infty} T^{(k)}(n) = \infty$.

The $3x + 1$ Conjecture asserts that all trajectories of n are convergent. It is certainly true that $T^{(k)}(n) = 1$ cannot occur without some $T^{(k)}(n) \leq n$ occurring. The least positive integer k for which $T^{(k)}(n) \leq n$, is called the *stopping time* $\sigma(n)$ of n, and set $\sigma(n) = \infty$ if no k occurs with $T^{(k)}(n) \leq n$. Also, the least positive k for which $T^{(k)}(n) = 1$ is called the *total stopping time* $\sigma_\infty(n)$ of n, and set $\sigma_\infty(n) = \infty$ if no such k occurs.

Stopping time $\boldsymbol{\sigma(n)}$ =: Number of steps on the Collatz sequence of n until $T^{(k)}(n) \leq n$.

Total stopping time $\boldsymbol{\sigma_\infty(n)}$ =: Number of steps on the Collatz sequence of n until $T^{(k)}(n) = 1$.

1.3.1 Conjecture on the Divergent trajectories

The function $T: \mathbb{Z} \to \mathbb{Z}$ has no divergent trajectories, i.e., there exists no integer n_0 for which,

$$\lim_{k\to\infty} |T^{(k)}(n_0)| = \infty.$$

2. Materials and Methods

2.1 *The Collatz function and the conjecture*

The Collatz function is defined for any positive integer n as

$$T(n) = \begin{cases} \frac{n}{2} & \text{If } n \text{ is even,} \\ 3n+1 & \text{If } n \text{ is odd.} \end{cases}$$

By repeatedly applying this function to an initial positive integer n, we generate a sequence in which each new term is determined by the previous one. The Collatz conjecture asserts that, irrespective of the starting value, the sequence will eventually reach the integer 1. Once the value 1 is attained, the sequence enters a repeating cycle of numbers:

$$\{\cdots, 1,4,2,1,4,2,1, \cdots\}.$$

According to the conjecture, this is the only eventual cycle possible for sequences arising from any positive integer input.

The function can also be reformulated for some odd number k and positive integer m,

$$T(n) = \begin{cases} k \cdot 2^{m-1} & \text{If } n = k \cdot 2^m, \\ 3k+1 & \text{If } n = k. \end{cases}$$

First, we can observe that any number is connected to some other number. Hence, we can construct a structure. We approach the Collatz conjecture in a constructive way such that we get a connected and divergent number sequence starting with the number 1.

2.2 *Connected Sub-Sequences*

We construct a sequence starting with an odd number k and all the other terms are even such that the $(n+1)^{th}$ term of the sequence $\{k \cdot 2^n\}_{n=0}^{\infty}$ is $k \cdot 2^n$.

Since the Collatz function connects every odd number k' with a unique mapping; $3k'+1$ to another even number $k \cdot 2^n$, we define $\{k' \cdot 2^n\}$ as a sub-sequence of a main-sequence $\{k \cdot 2^n\}$. Then,

$$k \cdot 2^n = 1 + 3k'$$

$$k \cdot 2^n \equiv 1 (\text{mod } 3)$$

Thus, for every $k' (\neq 1)$ number in a sub-sequence, there exists an odd number k in a main-sequence such that $k \neq k'$. Here, $k = k'$ if and only if $k' = 1$. Then we check the converse, which odd number k in a main sequence has sub-sequences starting with odd numbers k'. For each $n \in \mathbb{Z}^+$, we consider three cases:

- **Case 1:** If $k \equiv 0(\mathrm{mod}3)$

  $0 \cdot 2^n \not\equiv 1(\mathrm{mod}3)$. Since no k value satisfies the congruent, there are no sub-sequences.
- **Case 2:** If $k \equiv 1(\mathrm{mod}3)$

  $1 \cdot 2^n \equiv 1(\mathrm{mod}3)$, Since the congruence is satisfied when n is an even number. There are sub-sequences when the number in the main-sequence is $k \cdot 2^{2n} \in \{k \cdot 2^n\}$.
- **Case 3:** If $k \equiv 2(\mathrm{mod}3)$

  $2 \cdot 2^n \equiv 1(\mathrm{mod}3)$, Since the congruent is satisfied when n is odd. There are sub-sequences when the number in the main-sequence is $k \cdot 2^{2n-1} \in \{k \cdot 2^n\}$.

Consequently, k' odd elements in the sub-sequence connect to $k \cdot 2^n$ elements in the main sequence. We call this the Collatz structure.

The structure described above is illustrated in Figure 2, which visualizes the connected sub-sequences and their relation to the main sequence within the Collatz structure.

Collatz-conjecture-visualized.png Visual diagram of connected sub-sequences(The Collatz structure)

We can define a function f from n^{th} level to $(n-1)^{th}$ level, mapping consecutive odd numbers k and k' in some generated Collatz sequence as,

$$f(k') = k.$$

As an example, $f(3) = 5$. We call this the odd Collatz function.

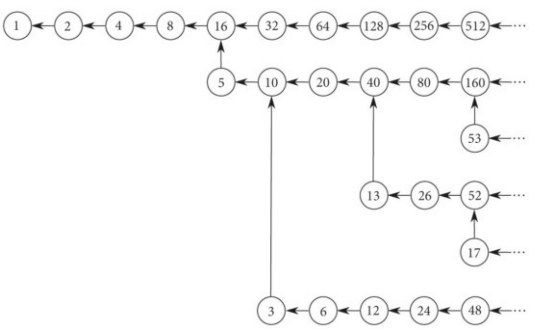

Figure 2

Visual diagram of connected sub-sequences (The Collatz structure)

We can define a function f from n^{th} level to $(n-1)^{th}$ level, mapping consecutive odd numbers k and k' in some generated Collatz sequence as,

$$f(k') = k.$$

As an example, $f(3) = 5$. We call this the odd Collatz function.

2.3 *Sequences connected to* $\{1 \cdot 2^m\}$ *sequence*

Suppose $\{1 \cdot 2^m\}$ is the 1^{st} Level sequence and a p^{th} Level sub-sequence be $\{k_p \cdot 2^m\}$.

1^{st} Level $\Rightarrow \{1,2,4,8,16,\cdots\}$

2^{nd} Level $\Rightarrow \{5 \cdot 2^m\}$, $\{21 \cdot 2^m\}$,$\{85 \cdot 2^m\},\cdots$

.

.

.

$(p-1)^{th}$ Level $\Rightarrow \{k_{p-1} \cdot 2^m\}$

p^{th} Level $\Rightarrow \{k_p \cdot 2^m\}$

We use the principle of strong induction at the n^{th} level to prove that any number starting from any level will reach 1 eventually. Since all are even numbers (except 1) in the sequence of level 1 and by dividing an element $1 \cdot 2^m$(for some positive integer m), m times by 2, we get 1. Every element in 1^{st} level will eventually reach to 1. Thus, it is true for $n = 1^{st}$ level. As the induction hypothesis, we assume that it is true for $n^{th}(< p^{th})$ level and let $n = p^{th}$ level contains sub-sequences of a main sequence at the $(p-1)^{th}$ Level. Thus, there exists an odd number k_{p-1} such that $k_p \cdot 2^n = 1 + 3k_{p-1}$. This implies that it is true for the $n = p^{th}$ level. The assertion still holds even if there are no sub-sequences at $n = p^{th}$ level, by the induction hypothesis. We can conclude that by the principle of strong induction, the assertion is true for all levels.

2.4 *Odd number sequences*

We already know that there exists a sub-sequence for any odd number k in the main sequence:

- If $k \equiv 1(\mathrm{mod}3)$, then the sub-sequences are generated when $k \cdot 2^{2n} \in \{k \cdot 2^n\}$ for $n = 1,2,3,\cdots$.
- If $k \equiv 2(\mathrm{mod}3)$, then the sub-sequences are generated when $k \cdot 2^{2n-1} \in \{k \cdot 2^n\}$ for $n = 1,2,3,\cdots$.

We can define a function g as follows;

$$k'_1 = g(k) = \begin{cases} \frac{k \cdot 2^2 - 1}{3} & \text{If } k \equiv 1 \pmod 3, \\ \frac{k \cdot 2 - 1}{3} & \text{If } k \equiv 2 \pmod 3. \end{cases}$$

So, we can define an odd number sequence generated from each odd number k, which is the sequence starting with k'_1,

$$\{k'_1 \cdot 4^{n-1} + \textstyle\sum_{i=0}^{n-2} 4^i\}_{n=2}^{\infty}$$

and it satisfies $4k'_n + 1 = k'_{n+1}$ for $n = 1,2,3,\cdots$.

Since it is connected to k by the Collatz function, we can denote it as,

$$[k] = k'_1 \mathrm{for} n = 1 \mathrm{and} \{k'_1 \cdot 4^{n-1} + \textstyle\sum_{i=0}^{n-2} 4^i\}_{n=2}^{\infty}$$

We know that every even number of the form $k \cdot 2^m \in \{k \cdot 2^m\}$ for $m = 1,2,3,\cdots$ will converge to the starting odd number k. To prove the conjecture holds for all positive integers, we only need to show that, all Collatz sequences generated for all odd positive integers converge to 1, through the Collatz structure. We can modify collatz function using these odd number sequences for any odd number k as follows;

$$T(k) = \begin{cases} k'_1 & \text{If } k = k'_1 \cdot 4^{m-1} + \sum_{i=0}^{m-2} 4^i \ (m \geq 2), \\ f(k'_1) & \text{Else } k = k'_1. \end{cases}$$

We generate a new sequence starting with $g(k)$ for each k elements in the previous level odd sequence.

We start from $[k = 1]$ and since $k \equiv 1(\mathrm{mod}3)$ we get $1 = g(1)$. Here $k = g(k)$ only for $k = 1$.

1^{st} Generation $\Longrightarrow [1] = \{1,5,21,85,341,\cdots\}$
2^{nd} Generation $\Longrightarrow [1], [5] = \{3,13,\cdots\}, [85] = \{113,453,\cdots\}, [341],\cdots$
3^{rd} Generation $\Longrightarrow [1], [5], [85], [341],\cdots [13] = \{17,69,\cdots\},\cdots [113],\cdots$
.
.
.

We can identify that the 1^{st} generation of odd numbers contains all the odd numbers of first two levels of collatz structure and the n^{th} generation contains all odd numbers of first $n+1$ levels of the Collatz structure. Suppose, we have two similar odd k numbers in some generation, but $k = g(k)$ only for $k = 1$ and the function g is one-to-one, then there should be two similar odd k numbers in previous generation also, which is a contradiction. We always get numbers different to previous all levels in the new level in the Collatz structure, except for the number 1. Most importantly, we would not have two similar odd numbers on two levels of the Collatz structure. Thus, the n^{th} generation contains all odd numbers of first $n+1$ levels of the Collatz structure and each generation contains union of disjoint odd sequences.

Since, the Collatz sequence is conjectured for any odd number, we can conjecture that, if we let the n^{th} generation to the infinity, then we get all the odd positive integers as a disjoint union of sequences residing in this structure.

2.5 *Infinite loops of sequences and the inverse Collatz function*

2.5.1 Infinite loops of sequences

We define the process between two consecutive odd numbers in the Collatz sequence, as an iteration. The Collatz structure we created is the opposite direction to Collatz sequence and for one iteration in the Collatz sequence, for some odd number in n^{th} level reaches the $(n-1)^{st}$ level. Suppose there is a an odd number k_0 such that after repeating the process, it never comes to the 1^{st} level containing the sequence, $\{1 \cdot 2^n\}$. Then there exist a parallel system which

is not connected the Collatz structure, we constructed. Also in that system, there exists an infinite loop of sequences such that there are two similar odd $\boldsymbol{k_0}$ values after q iterations ($k_q = k_0$),

$k_0 \cdot 2^{n_0} = 1 + 3k_{q-1} = T(k_{q-1})\ q^{th}$ iteration
. . .
. . .
. . .
$k_2 \cdot 2^{n_2} = 1 + 3k_1 = T(k_1)\ 2^{nd}$ iteration
$k_1 \cdot 2^{n_1} = 1 + 3k_0 = T(k_0)\ 1^{st}$ iteration

We know that this loop of numbers contained in the system generated by odd number k_0 in the sequence $\{k_0 \cdot 2^n\}$. Without loss of generality, let $k_0 = max\{k_0, k_1, \ldots, k_{q-1}\}$.

2.5.2 The inverse odd Collatz function

We know that there are infinitely many ways that odd numbers converge to 1, through the Collatz structure and there is a leftmost path for some particular odd number k. Since, when $k \equiv 0(\mathrm{mod}3)$ the Collatz structure doesn't have sub-sequences, we map those k's to $k \cdot 2^2 + 1$.

We can modify function g for all odd numbers,

$$g(k) = \begin{cases} \frac{k \cdot 2^2 - 1}{3}, & \text{if } k \equiv 1 \pmod 3, \\ \frac{k \cdot 2 - 1}{3}, & \text{if } k \equiv 2 \pmod 3, \\ k \cdot 2^2 + 1, & \text{if } k \equiv 0 \pmod 3. \end{cases}$$

Since $k = g(k)$ only for $k = 1$, we have a convergent sequence $\{1,1,1,1,\cdots\}$ for $k = 1$. Thus, for any odd number k not equal to 1, we can conjecture that the sequence generated from this function is divergent. We can observe that,

- for odd numbers of the form $k \equiv 1(\mathrm{mod}\ 3)$ and $k \equiv 0(\mathrm{mod}\ 3)$, we get $k < g(k)$.
- for odd numbers of the form $k \equiv 2(\mathrm{mod}\ 3)$, we get $k > g(k)$. Here $k = 3m + 2$ for some positive odd integer m. Since k is an odd number, m should be an odd number. Let $m = 2r - 1$ for some positive integer $r = 1,2,3,\cdots$. Then, $k = 6r - 1 \equiv 2(\mathrm{mod}\ 3)$ for some positive integer $r = 1,2,3,\cdots$.

2.6 *Odd numbers of the form* $6n - 1$ *and* $6n + 1$

For some positive integer n, we have

- $6n - 1 = 2^{\alpha_1}3^{\alpha_2}5^{\alpha_3}P_4^{\alpha_4} \cdots P_j^{\alpha_j} - 1$ for $\alpha_j, j \in \mathbb{Z}^+$ and $P_j \in$ Primes
  $6n - 1 = 2^l 3^m q - 1$; for some positive integers l, m, and an odd number q.
- $6n + 1 = 2^l 3^m 5^{\alpha_3} P_4^{\alpha_4} \cdots P_j^{\alpha_j} + 1$ for $\alpha_j, j \in \mathbb{Z}^+$ and $P_j \in$ Primes

$6n + 1 = 2^l 3^m q + 1$; for some positive integers l, m, and an odd number q.

We have already defined the function g(inverse odd Collatz function).

We can apply g only if $l, m \neq 0$ and we know that, $k = 6n - 1 \equiv 2(\text{mod}3)$ and $k = 6n + 1 \equiv 1(\text{mod}3)$.

$$g(k) = \begin{cases} \frac{k \cdot 2^2 - 1}{3}, & \text{if } k \equiv 1 \ (\text{mod } 3), \\ \frac{k \cdot 2 - 1}{3}, & \text{if } k \equiv 2 \ (\text{mod } 3), \\ k \cdot 2^2 + 1, & \text{if } k \equiv 0 \ (\text{mod } 3). \end{cases}$$

We have defined the function f in section 2.2, for two consecutive odd numbers (k' and k) in the Collatz sequence $f(k') = k$.

2.6.1 Odd numbers of the form $6\text{n} + 1 = 2^{\text{l}} 3^{\text{m}} \text{q} + 1$ for some odd number q

First we apply g once,

$$g(2^l 3^m q + 1) = \frac{(2^l 3^m q + 1) \cdot 2^2 - 1}{3} = 2^{l+2} 3^{m-1} q + 1 \equiv 1(\text{mod } 3)$$

Here we lose one factor of 3 and gain 2 factors of 2 from $2^l 3^m q$, but still $2^{l+2} 3^{m-1} q + 1 \equiv 1 \ (\text{mod } 3)$

We can apply g, m number of times

$$g^m(2^l 3^m q + 1) = 2^{l+2m} q + 1 \not\equiv 1(\text{mod } 3)$$

and $f^m(2^{l+2m} q + 1) = 2^l 3^m q + 1$. As long as $m \neq 0$, for some finite number m, the sequence generated by g increases m number of times. Also, we can go back using the function f.

2.6.2 Odd numbers of the form $6\text{n} - 1 = 2^{\text{l}} 3^{\text{m}} \text{q} - 1$ for some odd number q

First we apply g once,

$$g(2^l 3^m q - 1) = \frac{(2^l 3^m q - 1) \cdot 2 - 1}{3} = 2^{l+1} 3^{m-1} q - 1 \equiv 2(\text{mod } 3).$$

Here we lose one factor of 3 and gain one factor of 2 from $2^l 3^m q$, but still $2^{l+2} 3^{m-1} q - 1 \equiv 2(\text{mod}3)$.

We can apply g, m times to obtain

$$g^m(2^l 3^m q - 1) = 2^{l+m} q - 1 \not\equiv 2(\text{mod } 3)$$

and

$$f^m(2^{l+m} q - 1) = 2^l 3^m q - 1.$$

As long as $m \neq 0$, for some finite number m, the sequence generated by g decreases m times.

2.7 *The sequences generated by the function g are divergent for all positive odd integers*

We use the principle of strong induction on r to prove that the sequences generated for all odd numbers $k \leq 6r-1$, we have an odd number $g^t(k) > 6r-1$ for some integer t.

First, we take $r = 1$. Then we get $6r-1 = 5$ and the sequence generated for $k = 3$ is $\{13,17,11,7,9,37,\cdots\}$ and the sequence generated for $k = 5$ is $\{3,13,17,11,7,9,37,\cdots\}$. Since we have $g^2(k) > 5$ for all $k \leq 5$, the assertion is true for $r = 1$.

Then, we take $r = 2,3,4$, so we get $6r-1 = 11,17,23$. Since we have $g^t(k) > 23$ for some t for the sequences generated for all $k \leq 23$, the assertion is true for $r = 1,2,3,4$.

As the induction hypothesis, assume the assertion is true for $4 < r < n$, then for all $k \leq 6n-7$, we have odd numbers $g^t(k) > 6n-7$ for some integer t. After that, we consider $r = n$. We want to show that for all $k \leq 6n-1$, there exist an odd number $g^t(k) > 6n-1$ for some integer t.

- If $k = 6n-5 \equiv 1 \pmod 3$, then we get $g(6n-5) = \frac{(6n-5)\cdot 2^2-1}{3} = \frac{24n-21}{3} = 8n-7 > 6n-1$ for $n > 4$.
- If $k = 6n-3 \equiv 0 \pmod 3$, then we get $g(6n-3) = (6n-3)\cdot 2^2 + 1 = 24n-11 > 6n-1$ for $n > 4$.
- If $k = 6n-1 \equiv 2 \pmod 3$, then we get $g(6n-1) = \frac{(6n-1)\cdot 2-1}{3} = \frac{12n-3}{3} = 4n-1 < 6n-1$. We want to show that $g^t(k) > 6n-1$ for some t, but this case is not true if and only if $g^t(6n-1) = 6n-1$ for some t and there is a loop in the sequence generated for $6n-1$ such that $k_0 = 6n-1 = max\{k_0, k_1, \cdots, k_{q-1}\}$. Also after q (finite) number of iterations, we get $g^q(6n-1) = 6n-1$. In the section 2.4, we found that, there are no similar odd numbers in two levels in the Collatz structure. Hence, $g^t(6n-1) \neq 6n-1$ for any t. The assertion is true for $r = n$. By the principle of strong induction the assertion is true for all n.

Since we arbitrarily chose $k = 6r-1$ for any odd number $k \equiv 2(\mathrm{mod}3)$, the sequence generated by the function g is divergent. We can conclude that the sequences generated from the function are divergent for all odd numbers such that the Collatz structure contains all the positive odd integers. In addition, there cannot be a loop such that for the odd number $k_0 = 6n-1 = max\{k_0, k_1, \cdots,, k_{q-1}\}$ which we are applying the induction hypothesis.

We can conclude that there cannot exist a loop of odd numbers in a separate structure, which is not connected to the Collatz structure. Thus, the Collatz conjecture is true for any positive integer.

3. Results

3.1 *Similar Sequences to the Collatz sequence*

By *The Fundamental Theorem of Arithmetic*, for $n \in \mathbb{Z}^+$ can be written as

$$n = 2^{\alpha_1}3^{\alpha_2}5^{\alpha_3}P_4^{\alpha_4}\cdots P_j^{\alpha_j} \text{ for } \alpha_j, j \in \mathbb{Z}^+ \text{ and } P_j \in \text{Primes}$$

which can again be written as

$$n = 2^{\alpha_1}3^{\alpha_2}\cdots P^{\alpha_n}k;$$

for some odd number k. Now the Collatz function yields

$$f(n) = \begin{cases} \frac{n}{2}, & \text{if } n = 2^{\alpha_1}3^{\alpha_2}5^{\alpha_3}k, \\ \frac{n}{3}, & \text{if } n = 3^{\alpha_2}5^{\alpha_3}k;\ \alpha_1 = 0, \\ Pn+1, & \text{if } n = k;\ \alpha_1, \alpha_2, \cdots, \alpha_n = 0. \end{cases}$$

Similarly we can define functions, $\boldsymbol{Pn+1}$, for any prime P.

As an example for $P = 5$, we can define the function as,

$$f(n) = \begin{cases} \frac{n}{2}, & \text{if } n = 2^{\alpha_1}3^{\alpha_2}k \\ \frac{n}{3}, & \text{if } n = 3^{\alpha_2}\ \alpha_1 = 0 \\ \frac{n}{3}, & \text{if } n = k;\ \alpha_1, \alpha_2, = 0 \end{cases}$$

3.2 *Encryption and Decryption algorithm using the Collatz sequence*

We present a symmetric cryptographic scheme based on the iterative application of the Collatz sequence for plaintext encryption in which the sender and receiver establish a shared secret key pair (k, l), where k represents a large odd integer and l denotes the number of iterations, both parameters being generated through a cryptographically secure random process. This approach represents a departure from conventional symmetric encryption methods by leveraging the mathematical properties of the Collatz conjecture to establish a deterministic yet computationally complex transformation of the input plaintext.

Encryption Algorithm

Suppose we want to encrypt some random plaintext M, with r number of characters and the secret key (k, l) is chosen randomly.

- **Step 1:** Convert each character of M into 3-digit ASCII representation, such that we get $3r$ digit length code.
- **Step 2:** Make l number of partitions from the $3r$-digit ASCII code. If $3r$ is not divisible by l, we pad t number of zeros in front of the code, such that $3r + t$ is divisible by l, for some positive integer t. Thus, $3r + t$ digits can be partitioned into l partitions with $(3r + t)/l$-digit lengths.

  - We denote the partitioned ASCII code as $[m_1, m_2, \cdots m_l]$.
- **Step 3:** Calculate the Collatz sequence of k.

- **Step 4:** From the sequence generated in Step 3, We obtain the first l consecutive odd numbers, which are denoted as $[k_0, k_1, \cdots, k_l]$ and the even number count between those numbers, which are denoted as $[n_1, n_2, \cdots, n_l]$.
- **Step 5:** Each n_i is padded in front of each m_i for $i = 1,2, \cdots, l$.
  - Partitioned and padded ASCII code is denoted as $[n_1m_1, n_2m_2, \cdots, n_lm_l]$.
- **Step 6:** Finally, the partitions we created in the last step are taken together as one number, which is denoted as M' and we multiply that number with the l^{th} consecutive odd number, which is k_l.
  - The encrypted message is $M' \cdot k_l = [(n_1m_1n_2m_2 \cdots n_lm_l) \cdot k_l]$

Illustration using Python

Suppose we take the plaintext as `mathematics', which is an 11-letter word, and the secret key $(1111,5)$ is chosen randomly.

- **Step 1:** Convert each letter into a 3-digit ASCII representation such that we get $33(= 11 \cdot 3)$ digit code.
  - ASCII representations of each letter in `mathematics' are $[109,097,116,104,101,109,097,116,105,099,115]$ such that $[10909711610410110909711610509911 5]$ is the 33 digit code.
- **Step 2:** Make $l\ (= 5)$ number of partitions from the 33-digit ASCII code. Since 33 is not divisible by 5, we pad two zeros in front of the code. Thus, $35\ (33 + 2)$ digits can partition into 5 partitions with 7-digit lengths.
  - Partitioned ASCII code is $[m_1, m_2, m_3, m_4, m_5] =$ $[0010909,7116104,1011090,9711610,5099115]$
- **Step 3:** Then, we calculate the Collatz sequence of 1111, which is [1111, 3334, 1667, 5002, 2501, 7504, 3752, 1876, 938, 469, 1408, 704, 352, 176, 88, 44, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1].
- **Step 4:** After that, from the sequence we generated, We obtain first 5 consecutive odd numbers, which are $[1667,2501,469,11,17]$ and the even number count between those 5 consecutive odd numbers, which are [1,1,4,7,1].
- **Step 5:** Each n_i is padded in front of each m_i for $i = 1,2,3,4,5$.
  - Partitioned and padded ASCII code is $[10010909,17116104,41011090,79711610,15099115]$.
- **Step 6:** Finally, the partitions we created in the last step are taken together as a one number and we multiply that number with $l^{th}(= 5^{th})$ consecutive odd number which is 17.
  - The encrypted message is $M' \cdot k_5 = [1001090917116104410110907971161015099115 \cdot 17]$ and denoted by $C = [3170185455909737749718854355 0973725668 4955]$.

Decryption Algorithm Using the Collatz sequence

Suppose we want to decrypt the encrypted message, C using the shared Secret key, (k, l) for some odd number k, and l number of iterations in the Collatz sequence, starting with k.

- **Step 1:** First we generate the Collatz sequence starting with k. So, we can obtain first n consecutive odd numbers, which are $[k_1, k_2, \cdots, k_l]$ and the even number count between those l consecutive odd numbers which are $[n_1, n_2, \cdots, n_l]$.
- **Step 2:** Next, we divide C by l^{th} consecutive odd number, which is k_l. Then we get $M' = C/k_l$.
- **Step 3:** After that, we need to remove the padded n_i's. Since we know the number of digits in M' and l of them are n_is, we can get l partitions without them. Thus, we get the partitioned ASCII code as $[n_1 m_1, n_2 m_2, \cdots, n_l m_l]$.
- **Step 4:** In this step, we take all $l)$ partitions together as a one number and remove all the zeros in front of it. Thus, we can get 3-digit ASCII representations of the encrypted plaintext(M).
- **Step 5:** Finally, we can get the plaintext(M) using the ASCII table.
  - Decrypted text is denoted as $D(C) = M$

Illustration using Python

Suppose we receive the encrypted message as,

$$C = [3170185455909737749718854355097372566849 55]$$

and the shared Secret key is $(1111,5)$.

- **Step 1:** Generate the Collatz sequence of 1111. So, we can obtain first 5 consecutive odd numbers, which are $[1667,2501,469,11,17]$ and the even number count between those 5 consecutive odd numbers which are $[1,1,4,7,1]$. Thus, $n_1 = 1, n_2 = 1, n_3 = 4, n_4 = 7, n_5 = 1$ and $k_5 = 17$.
- **Step 2:** Divide C by $l^{th}(= 5^{th})$ consecutive odd number, which is 17. Then we get
  $$M' = C/k_5 = [1001090917116104410110907971161015099115]$$
- **Step 3:** After that, we need to remove the padded n_i's. Since there are 40 digits in M' and 5 of them are n_is, we can get 5 partitions without them. Thus, we get the partitioned ASCII code as
  $[0010909,7116104,1011090,9711610,5099115]$.
- **Step 4:** Take all $l(= 5)$ partitions together as one number and remove all the zeros in front of it. Thus, we can get 3-digit ASCII representations of the encrypted plaintext(M), which are
  $[109,097,116,104,101,109,097,116,105,099,115]$.
- **Step 5:** Finally, we can get the plaintext(M) using the ASCII table.
  - Decrypted text is denoted as $D(C) =$ 'mathematics'.

4. Discussion

Security[4], [9]

For any secret key $= (\mathbf{k}, \mathbf{n})$, we obtain a unique set of n_i values with unique k_n. So we get a unique encryption. Padding concept is similar to Secure Hashing Algorithm (SHA 256), the hash function and mining algorithm of the Bitcoin protocol. For more security, We can use more than one security key. Also we can modify this system such that $\mathbf{E}(\mathbf{M}) = C = M'^{k_l}$ in the last step of Encryption Algorithm. This can be implemented with other known cryptography systems.

Compared with traditional methods (such as AES and DES), chaos-based encryption[7] schemes have shown superior performance[11].

5. Conclusions

5.1 *The approach on the Collatz conjecture*

We approached the Collatz conjecture in a constructive perspective. We know that the Collatz sequence is conjectured for any positive integer n and all the even numbers of the form $n = k \cdot 2^m$ (for some odd number k and positive integer m) converge to the odd number k, through a sequence generated by the Collatz function. Thus, we only need to prove the Collatz conjecture is true for all odd positive integers.

Also, we have showed that for any level in the Collatz structure, odd numbers reside in a sequence of the form

$$[k] = k'_1 \text{ for } n = 1 \text{ and } \{k'_1 \cdot 4^{n-1} + \textstyle\sum_{i=0}^{n-2} 4^i\}_{n=2}^{\infty}$$

and if k'_1 is convergent to 1, then all the terms on the sequence of the form $k'_1 \cdot 4^{n-1} + \sum_{i=0}^{n-2} 4^i$ for all n are convergent to 1, through a sequence generated from the odd Collatz function $f(f(k') = k)$. Thus, we only need to prove that the Collatz conjecture is true for all odd k'_1 numbers.

In this constructed Collatz structure, the Collatz sequences converge upward and we have defined a function $g(g(k) = k'_1$ and $g(k) = 4k + 1$ for $k \equiv 0(\mathrm{mod}3))$, which diverges downward. Although the function g is not the exact inverse of the odd Collatz function $f(f(k') = k)$, if $g^t(k) = k_t$ for some positive integer t and odd numbers k and k_t, we have $f^{t'}(k_t) = k$ for some $t'(\leq t)$ such that we can go from k_t to k using the odd Collatz function. Hence, for every divergent sequence generated from the function g, we have a convergent Collatz sequence generated from the function f, starting with a term of the sequence generated by g and true for any term of the sequence generated by g. Thus, our present study provides a good justification for why the Collatz conjecture holds for all positive integers.

5.2 *The cryptographic scheme based on the Collatz sequence*

In the proposed cryptographic scheme, we employ the Collatz sequence to symmetrically encrypt plaintext. The system is based on a secret key pair (k, l), where k is a large, randomly chosen odd integer and l denotes the number of

iterations in the Collatz sequence, beginning with k. A plaintext message is encrypted by iteratively applying the Collatz function according to these parameters.

The Collatz sequence is inherently chaotic [5]: for every positive integer there exists a unique trajectory, and even a slight perturbation in the initial value results in a significant change in the generated sequence. This sensitivity to initial conditions enhances the security of the proposed cryptosystem. Moreover, the transformations involved are simple, consisting only of multiplication and addition, which makes the system both computationally efficient and easy to implement.

Abbreviations

The following abbreviations are used in this manuscript:

AES Advanced Encryption Standard

DES Data Encryption Standard

5. Appendices

Appendix A. Python Code for Encryption Algorithm

```
collseed = 1111
lenn = 5
message_to_enc='mathematics'

def collatz(number,n):
    count = 0
    evencount = 0
    lst=[]
    oddll = []
    lst.append(number)
    while(number!=1):
        if(number %2==0):
            number=number//2
            lst.append(number)
            # if count < n:
        else:
            oddll.append(number)
            number=number*3+1
            lst.append(number)
        count = count+1

        if(len(oddll)==n+1):
            return [lst,oddll]
    print(lst)
    return [lst,oddll]

cooll = collatz(collseed,lenn)
```

```
ncolatz = cooll[0]
oddcolatz = cooll[1]
nthoddcolatz = cooll[1][lenn]

evencount = []
evencountpass = 0
for ev in ncolatz[1:]:
    if ev%2 == 0:
        evencountpass = evencountpass+1
    else:
        evencount.append(evencountpass)
        evencountpass = 0

    print(cooll)
    print(ncolatz)
    print(oddcolatz)
    print(nthoddcolatz)
    print(evencount)
    def strtoascii(srting):
        strlist = [ord(c) for c in srting]
        intll = []
        for i in strlist:
            n = str(i)
            filll = n.zfill(3)
            intll.append(filll)

        longstr = "".join(intll)
        print(longstr)
        print(intll)

        return longstr
    longstr = strtoascii(message_to_enc)
    lenmod = len(longstr)%lenn

    print(lenmod)

    if(lenmod!=0):
        paddedlstr = longstr.zfill(len(longstr)+(lenn-lenmod))
    else:
        paddedlstr = longstr

    print(paddedlstr)
    print(len(paddedlstr))
    partition_size = len(paddedlstr)//lenn
    print(partition_size)
    out = [(paddedlstr[i:i+partition_size]) for i in range(0, len(paddedlstr),
        partition_size)]
```

```
print(out)
encylst = []
for idx , nthsss in enumerate(out):
    encccc = str(evencount[idx]%10)+nthss
    # encccc = int(nthsss)+oddcolatz[idx]
    encylst.append(encccc)

print(encylst)
longencystr = "".join(encylst)
encmulssint = int(longencystr)*nthoddcolatz

print(encmulssint)
```

Appendix B. Python Code for the Decryption Algorithm

```
decstr = '17018545590973774971881154355011973725668495512321 1'
collseed = 1111
lenn = 5

def collatz(number,n):
    count = 0
    lst=[]
    oddll = []
    lst.append(number)
    while(number!=1):
        if(number%2==0):
            number=number//2
            lst.append(number)
            # if count < n:
        else:
            oddll.append(number)
            number=number*3+1
            lst.append(number)
            count = count+1

            if(len(oddll)==n+1):
                return [lst,oddll]
        print(lst)
        return [lst,oddll]

cooll = collatz(collseed,lenn)
ncolatz = cooll[0]
oddcolatz = cooll[1]
nthoddcolatz = cooll[1][lenn]
```

```
evencount = []

evencountpass = 0
for ev in ncolatz[1:]:
    if ev%2==0:
        evencountpass = evencountpass+1
    else:
        evencount.append(evencountpass)
        evencountpass = 0

print(cooll)
print(ncolatz)
print(oddcolatz)
print(nthoddcolatz)
print(evencount)

intmsg = int(decstr)//nthoddcolatz
print(intmsg)
strmsg = str(intmsg)
partition_size = len(strmsg)//lenn
split_list=list(map(''.join, zip(*[iter(strmsg)]*partition_size)))

print(split_list)

encoded_list = []
for item in split_list:
    encoded_list.append(item[1:])

print(encoded_list)
combine_str = ''.join(encoded_list).lstrip('0')
# combine_str =
print(combine_str)

decoded_ascii_list = []
if len(combine_str)%3 ==0:
    decoded_ascii_list = list(map(''.join, zip(*[iter(combine_str)]*3)))
else:
    decoded_ascii_list = list(map(''.join, zip(*[iter('0'+combine_str)]*3)))

print(decoded_ascii_list)

decoded_char_list = []
for item in decoded_ascii_list:
    decoded_char_list.append(chr(int(item)))

decoded_msg = ''.join(decoded_char_list)
```

```
print(decoded_msg)
```

Appendix C. Python codes for the number of steps in the $\boldsymbol{Pn+1}$ *Sequence*

```
def getprimes(n):
    primelist = []
for num in range(2,n):
    prime = True
    for i in range(2,num):
        if (num%i==0):
            prime = False
    if prime:
        # print (num)
        primelist.append(num)
return primelist
```

C.2. Python code for the repetitions occurred in the $\boldsymbol{Pn+1}$ *sequence*

```
def collatzwprime(x,primes): initial = x a = [x] repeats = []
while x != 1:
    nner = False
    # print(x)
    for item in primes:
        if x % item == 0:
            x = x/item
            nner = True
            break
        if nner is not True:
            if x in repeats:
                print(initial,x)
                break
            repeats.append(x)
            x = P*x + 1
        a.append(x)
    return(a)
```

References

[1] Ş. Andrei and C. Masalagiu, "About the collatz conjecture," *Acta Informatica*, vol. 35, no. 2, pp. 167–179 (1998).

[2] R. Guy, Unsolved problems in number theory. Springer Science & Business Media, vol. 1 (2024).

[3] J. C. Lagarias, The ultimate challenge: The 3x+ 1 problem. American Mathematical Soc. (2010).

[4] W. Ren, "Reduced collatz dynamics data reveals properties for the future proof of collatz conjecture," *Data*, vol. 4, no. 2, p. 89 (2019).

[5] T. Tao, "Almost all orbits of the collatz map attain almost bounded values," in Forum of Mathematics, Pi, vol. 10. Cambridge University Press (2022).

[6] V. G. Bhatta, H. Panackal, P. Poojary, and S. K. Pandey, "Algebraic construction of near-bent function with application to cryptography," *Journal of Discrete Mathematical Sciences and Cryptography,* vol. 24, no. 2, pp. 527–540 (2021).

[7] J. H. S. Jeffrey Hoffstein, Jill Pipher, An Introduction to Mathematical Cryptography. Berlin: Springer (2008).

[8] G. J. Simmons and M. J. Norris, "Preliminary comments on the mit public-key cryptosystem," *Cryptologia*, vol. 1, no. 4, pp. 406–414 (1977).

[9] B. Furht and D. Kirovski, Multimedia Encryption and Authentication Techniques and Applications. Boca Raton, FL, USA: CRC Press (2006).

[10] C. Zhu, "A novel image encryption scheme based on improved hyperchaotic sequences," *Optics Communications*, vol. 285, no. 1, pp. 29–37 (2012).

[11] L. Kocarev, "Chaos-based cryptography: A brief overview," Circuits and Systems Magazine, *IEEE*, vol. 1, pp. 6 – 21 (2002).

Received May, 2024