The Collatz Conjecture - a proof

Richard L. Hudson 4-24-2021

Abstract

Originated by Lothar Collatz in 1937 [1], the conjecture states: given the recursive function, y=3x+1 if x is odd, or y=x/2 if x is even, for any positive integer x, y will equal 1 after a finite number of steps. This analysis examines number form and uses a tree type graph to prove the process.

1. examples

An example for a random selection of 7, using the original method:

S=(7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1)

An example for a random selection of 12, using the original method:

S= (12, 6, 3, 10, 5, 16, 8, 4, 2, 1)

2. functions

The recursive function is replaced with function d for odd values (2n-1), with

$$d(x) = 3x+1 = u = 2^k y \qquad (2.0)$$

and function e for even values, which removes all factors of 2,

$$e(u) = y \qquad (2.1)$$

The function e can be defined as a short program with a loop that repeatedly divides u by 2 until the output is an odd integer. This eliminates the redundancy and clutter of repeated division by 2.
After k divisions by 2, u = y, an odd integer. The value of y becomes the input x, and the cycle is repeated until y=1. The application of e(d(7)) results in S=(7 11 17 13 5 1), the revised format used in this analysis, with the understanding of a 2^k factor between each pair of odd integers. Notation is upper case for sets, lower case for elements of a set.

3. reverse sequences

If all sequences converge to the value 1, then it should be possible to form all reverse sequences, diverging from 1. For this purpose the odd integers are classified into 3 subsets, 0 mod 3, 1 mod 3, and 2 mod 3, labeled as Y_0, Y_1, and Y_2.

Y_0 = {3 9 15 21 27 ...} or y = 6n-3
Y_1 = {1 7 13 19 25 ...} or y = 6n-5
Y_2 = {5 11 17 23 29 ...} or y = 6n-1

Rearranging (2.0), we can find x, given y, while requiring y to be a (1 mod 3) value.
If y ≡ 1 mod 3, then k is even and if y ≡ 2 mod 3, then k is odd.

$$x = (2^k y-1)/3 = (u-1)/3. \qquad (2.2)$$

Varying k in (2.2) with y = 1, is shown in fig.1.

k	2	4	6	8	...
u	4	16	64	256	...
x	1	5	21	85	...

fig.1

Varying k in (2.2) with y = 5, is shown in fig.2.

k	1	3	5	7	...
u	10	40	160	640	...
x	3	13	53	213	...

fig.2

There are multiple combinations of x and k, that produce a given y. The x terms for each y, form an unlimited set and transform to y via the function e(d(x)). They are labeled as branching or b-terms, and indexed as generation 1, 2, 3, ... etc. in order of increasing values. The B notation for y=1 is B_1 = {1 5 **21** 85 341 ...}, meaning, y=1 for any x value in the set B_1. The B notation for y=5 is B_5 = {**3** 13 53 **213** ...}, meaning, y=5 for any x value in the set B_5. As shown in fig.(1 & 2), the u terms are related by a factor of 4, since that maintains the (1 mod 3) state of u. Bold fonts are (0 mod 3) terms.

3.1 defining a branch

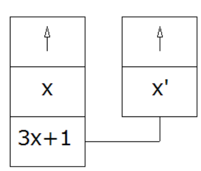

fig.3

If 3x'+1 = 4(3x+1), then x' = 4x+1.

Then the relation of successive b-terms is

$$x_{r+1} = 4x_r+1 \qquad (2.3)$$

There is a corresponding set B_y for each y, except elements of Y_0.
Since the function d cannot produce (0 mod 3) output, an element from Y_0 can only begin a descending sequence S of odd integers, which implies, a reverse sequence R will terminate. The one exception being (1 1), a simple loop.
A complete branch is one that begins with a (0 mod 3) term and ends with a b-term.
In ascending mode, using B_1 = {1 5 **21** 85 341 ...}, the next term is 1. R = (1 1) and terminates.

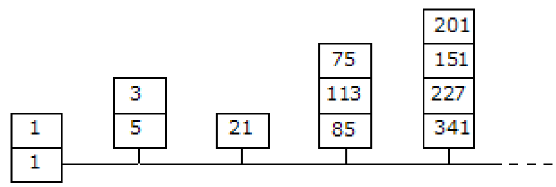

fig. 4

Fig.4 shows available options via B_1. The b-terms allow bypassing the next cell by forming a new branch. Using the next available term from B_1, R = (1 5).
From B_5 = {**3** 13 53 **213** ...}, R = (1 5 **3**), the sequence R terminates with a (0 mod 3) term.

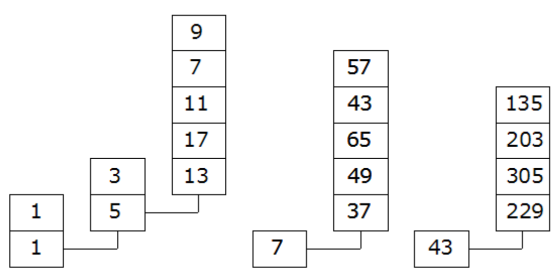

fig. 5

Remaining with the current R and B_5, the next (gen-2) term 13, allows a new branch and extension of R, as shown in fig.5. A new branch can be formed from any term in the current branch except (0 mod 3). In the example, the next to last term is arbitrarily selected. Using the b-terms for each successive x, extends the branch vertically to the next termination value 9. This process is repeated with 7, 43, 203, etc., and can be extended without limit.

B_1 ={1 5 **21** 85 341 **1365** ...}, (1 5)
B_5 ={**3** 13 53 **213** ...}, (1 5 13)
B_{13} = {17 **69** 277 **1109**...}, (1 5 13 17)
B_{17} = {11 **45** 181 **725**...}, (1 5 13 17 11)
B_{11} = {7 29 **117** 469...}, (1 5 13 17 11 7)
B_7 = {**9** 37 149 **597**...}, (1 5 13 17 11 7 37)
... (1 5 13 17 11 7 37...)

A reverse sequence R of any length can be formed using the b-terms which allow tree expansion.

4. the range of 2^k

	u=6n-2										
div	4	10	16	22	28	34	40	46	52	58	64
2		5		11		17		23		29	
4	1				7				13		
8							5				
16			1								
↓											

fig.6

Fig.6 shows a uniform distribution of divisors relative to the u-terms. The portion of u terms divisible by 2^k is $1/2^k$ with y an element of Y_1 or Y_2.

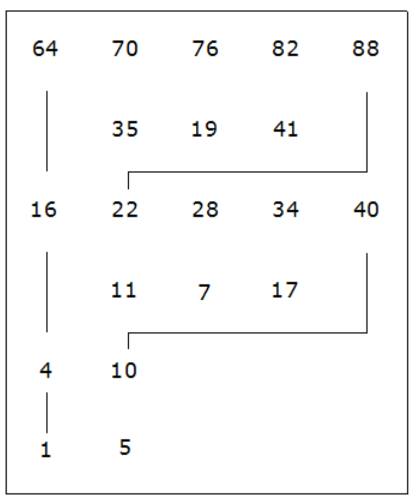

fig.7

As the value of u moves into larger ranges of 2^k, each pair of adjacent terms is expanded by a factor of 4 with 3 additional terms between them. This allows longer sequences in a branch, and larger divisors, as shown in fig.7.

4.1 odd integers in descending mode

Using the definition of a complete branch section (3.1), the descending sequence S_9 is
(9 7 11 17 13
3 5
1),
where 13 and 5 are b-terms.
S_9 is actually one branch joined to S_3 joined to S_1, and 2 branches distant from the trunk.

The x terms are classified into 3 sets.
X_1 = {3 7 11 15 ... } or x=4n-1, all x divisible by 2, with output of Y_2.
X_2 = {1 9 17 25 ... } or x=8n-7, all x divisible by 4, with output of Y_1.
B_x = {5 13 21 29 ... } or x=8n-3.
8n-3 can be rearranged as 4(2n-1)+1, the same form as eq(2.3).
Thus B_3 is a set of b-terms, one for every odd integer.
Using eq.(2.0), if x=2n-1 then u=6n-2.
The function d(8n-3)=24n-8, which =4(6n-2).
The set B_x as input is thus redundant and is used specifically in the process of branch formation.

5. even integer selection

All reverse sequences for even integer selection, can be formed by appending a 2^k progression times an odd integer, presented here as a list, using 1, 3, 5, 7, 9, ...

(2 4 8 16 32 ...)
(6 12 24 48 ...)
(10 20 40 80 ...)
(14 28 56 122 ...)
(18 36 72 144 ...)
...

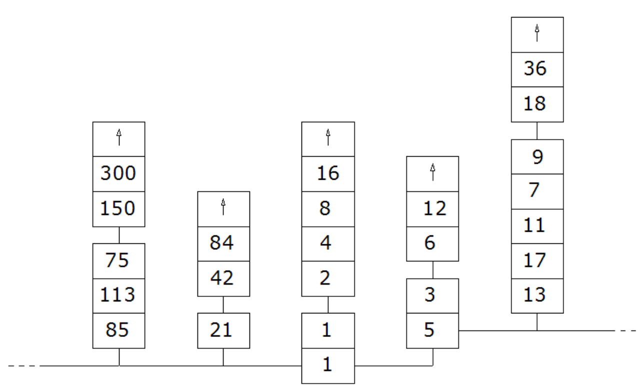

fig.8

This provides a means of extending the Y_0 termination values to sequences of unlimited length as shown in fig.8.

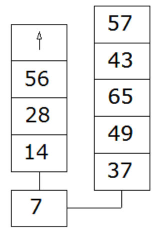

fig.9

Each term from Y_1 and Y_2 now have an extended sequence of even integers as in fig.9.

6. x-y correspondence

		↑	↑		↑	↑	↑		↑	↑	↑		↑	↑	...	gen
		85	213		469	597	725		981	1109	1237		1493	1621	...	4
	X3	21	53		117	149	181		245	277	309		373	405	...	3
A		5	13		29	37	45		61	69	77		93	101	...	2
	X2		3		7		11		15		19		23		...	1
	X1	1				9				17				25	...	1
	u	4	10		22	28	34		46	52	58		70	76	...	1
B	4u	16	40		88	112	136		184	208	232		280	304	...	2
	16u	64	160		352	448	544		746	832	928		1120	1216	...	3
	↓	↓	↓		↓	↓	↓		↓	↓	↓		↓	↓	...	
C	Y2		5		11		17		23		29		35		...	
	Y1	1				7				13				19	...	

fig.10

Fig.10 is a summary of x to y correspondence. The B_x extend vertically in the X3 section. Remaining in the same column, an odd integer x is selected from section A. An application of d(x) yields u in section B, with a matching generation index. An application e(u) yields y in section C.

7. the tree

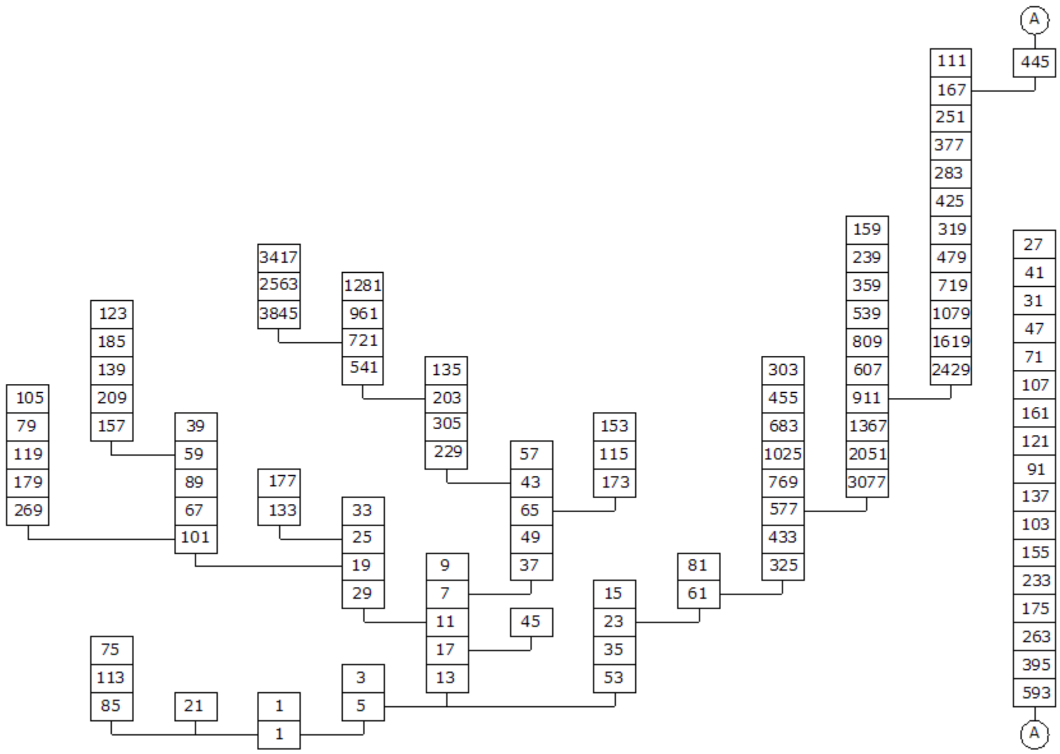

fig.11

Fig.11 shows the initial growth of the tree for odd integers only, from a 'trunk' of 1, vertically with each branch terminating in a (0 mod 3) value, and horizontally via the B_x as demonstrated in section 3. The sequence for x=27 is revealed as a composite of 7 branches to the right, S_{27} S_{111} S_{159} S_{303} S_{81} S_{15} S_5. Note the 3rd and 4th are partial branches.
To visualize a partial tree with all branches would require 3 dimensions.

conclusion

The reverse sequences are intended to answer the question,
Is a network possible that produces the specified results, using the specified rules?
Section 3 shows it is possible.
Descending in any branch, the values reflect x movement within the 2^k ranges, whereas the horizontal movement using b-terms, moves a sequence of x closer to x=1, with decreasing values. Therefore there is no (simple) distance function for any sequence of values relative to the trunk. The distance is determined by number of branches.

In ascending mode, choices were made in forming the 'one to many' network of paths diverging from the trunk, based on the Collatz rules. If a path can be formed from x=1 to any integer using a reverse engineering method, then a randomly selected x must return

to x=1 via a 'one to one' predetermined path.
An analogy would be a multi-story building with stairways between all floors. A person placed on any floor, has a path to the ground floor, by design.
Therefore all sequences merge at x=1 in descending mode. The Collatz conjecture applies only to finite length sequences, in the descending mode

reference

1. Wikipedia.org/Collatz Conjecture, Mar 2018