Research Article

The $3x + 1$ Problem as a String Rewriting System

Joseph Sinyor

Quatronet Corporation, 50 Almond Ave, Thornhill, Ontario, Canada L3T 1L2

Correspondence should be addressed to Joseph Sinyor, jsinyor@quatronet.com

The $3x + 1$ problem can be viewed, starting with the binary form for any $n \in \mathbf{N}$, as a string of "runs" of 1s and 0s, using methodology introduced by Błażewicz and Pettorossi in 1983. A simple system of two unary operators rewrites the length of each run, so that each new string represents the next odd integer on the $3x + 1$ path. This approach enables the conjecture to be recast as two assertions. (I) Every odd $n \in \mathbf{N}$ lies on a distinct $3x + 1$ trajectory between two Mersenne numbers $(2^k - 1)$ or their equivalents, in the sense that every integer of the form $(4m + 1)$ with m being odd is equivalent to m because both yield the same successor. (II) If $T^r(2^k - 1) \rightarrow (2^l - 1)$ for any $r, k, l > 0$, $l < k$; that is, the $3x + 1$ function expressed as a map of k's is monotonically decreasing, thereby ensuring that the function terminates for every integer.

1. Introduction

The $3x + 1$ problem is also known under various other names including Collatz's problem, Ulam's problem, and the Syracuse problem. It considers the following transformation of integers:

$$f(x) = \begin{cases} (3x + 1) & \text{if } x \text{ is odd,} \\ \dfrac{x}{2} & \text{if } x \text{ is even.} \end{cases} \tag{1.1}$$

The conjecture states that, starting with any integer n (which for our purposes we will take as positive, although there is a negative version), $f(x)$ will eventually reach 1 under iteration of the function. Thereafter it will enter the short cycle 4, 2, 1. The conjecture specifies that there are no other cycles and also that the function is not divergent. Another commonly used notation is $T^k(n)$ for the value after k iterations.

The conjecture has been verified up to 26×2^{55} as of April 2010 in independent computations by Oliveira e Silva [1] and Roosendaal [2].

In this paper we will use another version of the problem

$$f(x_i) = \frac{3x_{i-1} + 1}{2^{m_{i-1}}}, \quad x_0 = n, \tag{1.2}$$

such that $2^{m_{i-1}} \mid 3x_{i-1} + 1$ but $2^{m_{i-1}+1}$ does not divide it.

This version iterates directly from one odd integer to another, and the conjecture states that $x_i = 1$ for some finite value of i. However, this form is less often used by researchers because of the difficulty of determining *a priori* the values of m_{i-1}.

For a comprehensive review of the research to date, see Lagarias' annotated bibliography and generalizations of the problem [3–5] and Chamberland's 2003 update [6]. Wirsching's lecture notes [7] consider the dynamical system generated by the function. Shallit [8] treats the function from the viewpoint of finite automata.

Our approach builds on Błażewicz and Pettorossi's 1983 paper [9] in which they introduce the notion of viewing the binary representation as an alternating series of "runs" of 1s and 0s. These can be rewritten using two unary operators and include appending a carry 1 as necessary to the output string to achieve the iterations correctly according to the $3x + 1$ algorithm.

Clearly the binary notation can be expressed as lengths of runs. For example, we express $n = 27$ as $\mathbf{1}^2\mathbf{0}^1\mathbf{1}^2$. We point out that this particular starting point is especially interesting because—as noted by several writers—it generates one of two of the longest trajectories for values of $n < 10^3$. As Table 1 shows, 41 steps are required to reach 1 in terms of odd integers. The other long trajectory in that range is for $n = 703$.

We will dispense with explicitly showing 1s and 0s unless otherwise indicated because, as we will show, the same rewriting rules apply to both. Hence $n = 27$ becomes {2 1 2}, and all positive odd integers are represented in this way, the only constraint being that the string is of odd length. Even values of n appear as trailing 0s and are simply deleted in the rewriting process.

2. The String Rewriting System (SRS) for the $3x + 1$ Problem

We will now list the String Rewriting System (SRS) process specific to the $3x + 1$ problem and then prove that it replicates the algorithm for odd integers. Since the rewriting will involve deletions as well as additions, for convenience, we will invert the normal sequence, so that deletions occur on the left (least significant digits) and additions of carry 1s at right (most significant digits).

(i) At the start of each iteration, we delete any leading 1s *in pairs*. This action corresponds to the removal of $(4k + 1)$ factors whenever k is odd, which has no impact on the calculation.

(ii) Reset the output string to a null string.

Table 1: $3x + 1$ path (odd integers) for $n = 27$ expressed in integer and run notations.

No.	Integer	Run notation	No.	integer	Run notation
0	27	2 1 2	21	283	2 1 2 3 1
1	41	1 2 1 1 1	22	425	1 2 1 1 1 1 2
2	31	5	23	319	6 2 1
3	47	4 1 1	24	479	5 1 3
4	71	3 3 1	25	719	4 2 2 1 1
5	107	2 1 1 1 2	26	1079	3 1 2 4 1
6	161	1 4 1 1 1	27	1619	2 2 1 1 1 2 2
7	121	1 2 4	28	2429	1 1 5 1 1 2 1
8	91	2 1 2 1 1	29	911	4 3 3
9	137	1 2 1 3 1	30	1367	3 1 1 1 1 1 1 1 1
10	103	3 2 2	31	2051	2 9 1
11	155	2 1 2 2 1	32	3077	1 1 1 7 2
12	233	1 2 1 1 3	33	577	1 5 1 2 1
13	175	4 1 1 1 1	34	433	1 3 2 1 2
14	263	3 5 1	35	325	1 1 1 3 1 1 1
15	395	2 1 1 3 2	36	61	1 1 4
16	593	1 3 1 1 1 2 1	37	23	3 1 1
17	445	1 1 4 1 2	38	19	2 3 1
18	169	3 2 1 1 1	39	53	1 1 1 1 2
19	251	2 1 5	40	5	1 1 1
20	377	1 2 4 1 1	41	1	1

(iii) Apply the following rewriting rules, one integer at a time, and concatenate to the output string, using the unary operators f and g. Every iteration starts with f:

$$\text{(i) for } k = 1 \quad f1 \rightarrow +1g \quad \text{or} \quad g1 \rightarrow +1f,$$

$$\text{(ii) for } k = 2 \quad g2 \rightarrow 2g \quad \text{or} \quad f2 \rightarrow 1\ 1g,$$

$$\text{(iii) for } k \geq 3 \quad gk \rightarrow 1\ (k-2)\ 1g \quad \text{or} \quad fk \rightarrow (k-1)\ 1g, \tag{2.1}$$

$$\text{(iv) for } k = \phi \ (\text{end of input string}), \quad g\phi \longrightarrow 1f.$$

Note that, while in general we concatenate to the output string, the + sign (first rule) requires that we add 1 to the last integer on the output string. If there is no previous integer (output string is empty), then the 1 is simply discarded.

Proof. For this proof only, we will revert to the usual binary representation (least significant digits at right) and perform the $3n + 1$ calculation by adding a 1 to the right of n and add n. We summarize some findings of Błażewicz and Pettorossi as follows.

(i) A run ρ of 1s is translated by the function into $0\rho0$, and a run of 0s changes to $1\rho1$, so long as the length of ρ satisfies $|\rho| \geq 2$. For example, runs of $|\rho| = 5$ are shown pictorially as follows; the operation $f\,5$ is on the left and $g\,5$ is on the right:

$$
\begin{array}{ll}
\cdots\cdots\cdots 000111111 & \cdots\cdots 110000011 \\
\cdots\cdots\cdots 00011111 & \cdots\cdots\cdots 11000001 \\
\cdots\cdots\cdots 01\underline{011110} & \cdots\cdots 100\underline{1000100}
\end{array}
\tag{2.2}
$$

(ii) An alternating series of 1 and 0 results in a string of 0s

$$
\begin{array}{l}
\cdots\cdots\cdots\cdots 010101011 \\
\cdots\cdots\cdots 101010101 \\
\cdots\cdots\ 1000000000
\end{array}
\tag{2.3}
$$

this corresponds to a succession of $(4k + 1)$ factors which can be removed if they occur at right.

(iii) Single 0s or 1s disappear into the structures of their neighbors

$$
\begin{array}{ll}
\cdots\cdots\cdots 1110111\cdots & \cdots\cdots 0001000\cdots \\
\cdots\cdots\cdots 1110111\cdots & \cdots\cdots 0001000\cdots \\
\cdots\cdots\cdots 11\underline{00}111\cdots & \cdots\cdots 000\underline{11}000\cdots
\end{array}
\tag{2.4}
$$

(iv) It is easily seen that the rewriting rules (ii) and (iii) maintain the parity of the string for operator g and change it to even for f. Rule 1 has no effect on the parity. Hence, if the output string (which always starts with f) ends with g, we must add a 1 (the carry bit) to it to preserve the requirement that its length be odd.

Rule (ii) can similarly be proved. Using the above, one can confirm that the rewriting rules (2.1) satisfy all the requirements of the $3x + 1$ algorithm.   □

The above SRS has some useful properties.

A Mersenne number $(2^k - 1)$ which in binary is a row of k 1s is represented in our notation simply as $\{k\}$.

It is easy to see that every integer of the form $(4m + 1)$ with m being odd is equivalent to m because both yield the same successor. In binary, the string for an odd integer m is followed by a 0 and a 1 to produce $(4m + 1)$. In our notation this translates even more simply as a pair of leading 1s. Given any odd integer one can write or immediately recognize an infinite number of equivalents simply by adding (or removing) pairs of 1s. Only adding 1s in pairs will maintain the correct parity of the string.

3. The $3x + 1$ Problem and Mersenne Numbers

It is rather surprising that the Mersenne numbers should play a helpful role in the $3x + 1$ problem because—as several writers have noted, for example, Kuttler [10]—it is easy to show that $T^k(2^k - 1) = (3^k - 1)$; that is, the values appear to grow quickly (we caution here that Kuttler uses a variant of our definition (1.1) which results in a different set of iterations).
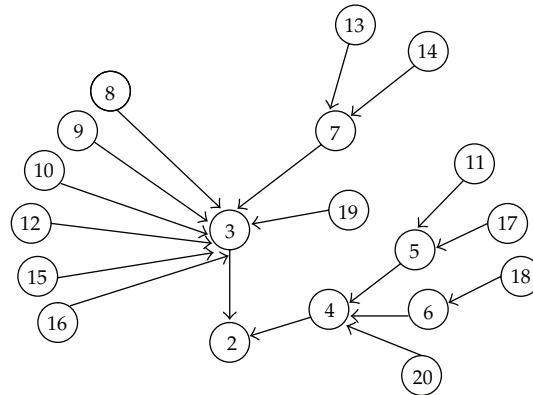
Figure 1: The $3x + 1$ map expressed as values of $k \leq 20$ for endpoints $(2^k - 1)$.

However, if we complete the numerical evidence for small values of k (≤ 20), an interesting structure emerges. Figure 1 shows each vertex containing the kth Mersenne number and each edge representing the $3x + 1$ trajectory from that kth Mersenne number to the next. For example, the trajectory for $n = 27$ is largely represented by the single edge between $k = 5$ and $k = 4$, that is, $n = 31$ and $n = 61$, where the latter is the equivalent of $n = 15$ which cannot have a predecessor since it is 0 (mod 3). Note that all the values of k in Figure 1 are monotonically decreasing in the direction of the arrows.

Define as usual the stopping time of n, denoted by $\sigma(n)$, as being the least k such that $T^k(n) < n$. It is easy to verify, as in Garner [11], the following stopping times; the $\sigma(n)$ have been adjusted to count odd integers only:

$$\sigma(n) = 1 \quad \text{if } n \equiv 1 \ (\text{mod } 4),$$

$$\sigma(n) = 2 \quad \text{if } n \equiv 3 \ (\text{mod } 16),$$

$$\sigma(n) = 3 \quad \text{if } n \equiv 11 \quad \text{or} \quad 23 \ (\text{mod } 32), \tag{3.1}$$

$$\sigma(n) = 4 \quad \text{if } n \equiv 7,\ 15 \quad \text{or} \quad 59 \ (\text{mod } 128),$$

$$\sigma(n) = 5 \quad \text{if } n \equiv 39, 79, 95, 123, 175, 199, \quad \text{or} \quad 219 \ (\text{mod } 256).$$

Equivalently (see [12]), $\sigma(n) \leq 5$ unless n is congruent mod 256 one of the following:

$$27 \quad 31 \quad 47 \quad 63 \quad 71 \quad 103 \quad 111 \quad 127 \quad 155 \quad 159 \quad 167 \quad 191 \quad 207 \quad 223 \quad 231 \quad 239 \quad 251 \quad 255. \tag{3.2}$$

Clearly we can always find an n with arbitrarily long stopping time $\sigma(n) = k$. In our run notation, this means simply choosing an integer of the form $\{k \cdots\}$. So stopping times alone will not suffice to resolve the problem. Looking at Figure 1 suggests that we consider the $3x + 1$ problem as a combination of two propositions: one for predecessors and another for successors.

(i) First we must show that every odd $n \in \mathbf{N}$ lies on the map described by Figure 1, either on the path between two Mersenne numbers ($2^k - 1$) or as the Mersenne numbers themselves. While we cannot predict where a particular n will fall on the map, we can say that this corresponds to the hypothesis that every n or an equivalent must have a Mersenne number as a predecessor. This condition is met if we can always find a valid predecessor of n smaller than n; that is, $T^{-k}(n) < n$, identical to the stopping time condition except in the reverse direction. Note that, if n is 0 (mod 3), we can convert it to an equivalent (in fact an infinite number of them) which has a valid predecessor.

(ii) The second proposition is that if $T^r(2^k - 1) \rightarrow (2^l - 1)$ for any $r, k, l > 0, l < k$; that is, the $3x + 1$ function expressed as a map of k's is monotonically decreasing, thereby ensuring that the function terminates for every integer. Note that we need only to concern ourselves here with Mersenne numbers, not all the points in between.

Hence a counterexample must satisfy $T^k(n) > n$ for all k's (positive and negative); that is, n must be a minimum with respect to its successor trajectory as well as its infinite number of predecessor paths.

It remains to be seen whether the String Rewriting System (SRS) approach can lead to further advances in the $3x + 1$ problem.

References

[1] T. 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] Eric Roosendaal, http://www.ericr.nl/wondrous/index.html.

[3] J. C. Lagarias, "The $3x + 1$ problem: an annotated bibliography (1963–1999)," http://arxiv.org/abs/math/0309224.

[4] J. C. Lagarias, "The $3x + 1$ problem: an annotated bibliography, II (2001–2009)," http://arxiv.org/abs/math/0608208.

[5] J. C. Lagarias, "The $3x + 1$ problem and its generalizations," *The American Mathematical Monthly*, vol. 92, no. 1, pp. 3–23, 1985.

[6] M. Chamberland, "An update on the $3x + 1$ problem," *Butlletí de la Societat Catalana de Matemàtiques*, vol. 18, no. 1, pp. 19–45, 2003.

[7] G. J. Wirsching, *The Dynamical System Generated by the 3n+1 Function*, vol. 1681 of *Lecture Notes in Mathematics*, Springer, Berlin, Germany, 1998.

[8] J. O. Shallit, "The $3x + 1$ problem and finite automata," *Bulletin of the AETCS*, no. 46, pp. 182–185, 1992.

[9] J. Błażewicz and A. Pettorossi, "Some properties of binary sequences useful for proving Collatz's conjecture," *Foundations of Control Engineering*, vol. 8, no. 2, pp. 53–63, 1983.

[10] J. R. Kuttler, "On the $3x + 1$ problem," *Advances in Applied Mathematics*, vol. 15, no. 2, pp. 183–185, 1994.

[11] L. E. Garner, "On the Collatz $3n + 1$ algorithm," *Proceedings of the American Mathematical Society*, vol. 82, no. 1, pp. 19–22, 1981.

[12] V. Klee and S. Wagon, *Old & New Unsolved Problems in Plane Geometry & Number Theory*, vol. 11 of *The Dolciani Mathematical Expositions*, Mathematical Association of America, Washington, DC, USA, 1991.