一种约化前向科拉茨算法：3x+1作用下二进制字符串长度如何变化+1

Richard Kaufmanrdkaufman01 at gmail dot com

摘要

我们开发了一种算法，可用二进制表示法轻松地从约化前向科拉茨映射（锡拉库扎函数）中的一个奇数求得下一个奇数。该算法根据二进制位的模式，指出一个奇数在变为下一个奇数时会增大还是减小。我们还利用该算法，提出了一种比文献中已有方法更简单的方式，用于确定约化映射下二进制字符串长度的变化。因此，只需检查一个奇数的二进制位，就能确定后续奇数的二进制位数。我们还给出了一些简单的结果，探讨科拉茨猜想的反例可能达到的最小数值。

1. 引言

科拉茨猜想指出，对于任意正整数 n，函数

$$f(n) = \begin{cases} \dfrac{n}{2} & \text{if } n \text{ is even} \\ \dfrac{3n+1}{2} & \text{if } n \text{ is odd} \end{cases}$$

总会生成一个最终到达 1 的正整数序列。

这种函数的另一种形式称为约化科拉茨函数，它将一个奇数映射到下一个奇数，因此考拉兹序列中只包含奇数。本文提出了一种称为算法 1 的算法，可轻松地用二进制记法执行约化科拉茨函数。随后，我们利用该算法提出一种称为方法1的方法，用于确定约化科拉茨函数作用下二进制字符串长度的变化。方法1似乎比 Hew 提出的一种方法更简单，Hew 将其称为定理 1。[1]

Hew（第 481 页）指出：

> 关于比特串的长度在约化科拉茨映射作用下如何变化，文献中尚未出现完整的描述……本文提供了一种方法，只需观察即可判断比特串在约化科拉茨映射作用下长度将发生怎样的变化。

Hew（第 481 页）给出的定理 1 如下：

Theorem 1. *Let x be an odd positive integer and write the binary expansion of x as $u[01]^n$ where u is non-empty, lacks leading zeroes, and does not end with 01. Put $\hat{s} = (x+\frac{1}{3})/D(x)$ where $D(x) = 2^{\lfloor \log_2 x \rfloor + 1}$. Then $R(x)$ will be shorter than x by the following number of bits:*

	u ends with 0 (so $n \geq 1$)	*u ends with 1 (so $n \geq 0$)*
$\hat{s}<2/3$	$2n-1$ *bits*	$2n$ *bits*
$\hat{s} = 2/3$	–	$2n$ *bits*
$\hat{s} > 2/3$	$2n-2$ *bits*	$2n-1$ *bits*

Hew 的说法似乎是正确的：他的论文是首篇探讨约化科拉茨函数二进制长度变化的论文。不过，本文提出的方法可能更容易实现和检验。在撰写本文的过程中，我们还发现了另一个结果，涉及科拉茨猜想的反例可能达到的最小数值；该结果也将在下一节中介绍。

2. 主要结果

我们提出一种简单的算法，称为算法 1，作为计算约化正向科拉茨函数的替代方法。约化正向科拉茨函数将考拉兹序列中的一个奇数 d 映射到下一个奇数 n。该算法对一个数的二进制表示进行操作，且可以通过检查验证。

算法 1。

取奇数 d 的二进制表示，在开头的“1”前面加上两个“0”，然后从右向左遍历这个二进制字符串。首先写入的数字始终为 0。设置标志 = 1。如果标志 = 0 ，则

- 如果最后两位是“00”，则写入 0；否则，如果最后两位是“01”，则写入 1；否则，如果最后两位是“10”，则写入 1；否则，如果最后两位是“11”，则写入 0，并 设置标志 = *1* 结束 if。否则，如果标志 = 1 ，则

- 如果最后两位是“00”，则写入 1 并设置标志 = *0* 否则，如果最后两位是“01”，则写入 0；如果最后两位是“10”，则写入 0；如果最后两位是“11”，则写入 1。结束判断。

从右向左处理完所有二进制位后停止。

消除下一个奇数 n 的前导和末尾 0。

以下示例展示了如何将算法 1 应用于数字 467。示例 A：

467_{10} =	1 1 1 0 1 0 0 1 1	Write number in binary
Number row	0 0 1 1 1 0 1 0 0 1 1	Put 00 in front of binary. This is the "number row"
Steps:	**0**	First number to write is **0**. Set Flag = 1.
	1 0	Flag = 1. Last two digits in number row were 11. So write **1**.
	0 1 0	Flag = 1. Last two digits in number row were 01. So write **0**.
	1 0 1 0	Flag = 1. Last two digits in number row were 00. So write **1**. Set Flag = 0.
	1 1 0 1 0	Flag = 0. Last two digits in number row were 10. So write **1**.
	1 1 1 0 1 0	Flag = 0. Last two digits in number row were 01. So write **1**.
	1 1 1 1 0 1 0	Flag = 0. Last two digits in number row were 10. So write **1**.
	0 1 1 1 1 0 1 0	Flag = 0. Last two digits in number row were 11. So write **0**. Set Flag = 1.
	1 0 1 1 1 1 0 1 0	Flag = 1. Last two digits in number row were 11. So write **1**.
	0 1 0 1 1 1 1 0 1 0	Flag = 1. Last two digits in number row were 01. So write **0**.
	1 0 1 0 1 1 1 1 0 1 0	Flag = 1. Last two digits in number row were 00. So write **1**. Set Flag = 0. End
Next Number	1 0 1 0 1 1 1 1 0 1	Eliminate leading and trailing 0. This is 701 in base 10.

以下示例展示了约化科拉茨函数从 31 开始得到的后续每个奇数。示例 B：

Sequence of odd Collatz numbers	Binary	Number of binary digits	Difference in number of binary digits from previous
31	11111	5	
47	101111	6	1
71	1000111	7	1
107	1101011	7	0
161	10100001	8	1
121	1111001	7	-1
91	1011011	7	0
137	10001001	8	1
103	1100111	7	-1
155	10011011	8	1
233	11101001	8	0
175	10101111	8	0
263	100000111	9	1
395	110001011	9	0
593	1001010001	10	1
445	110111101	9	-1
167	10100111	8	-1
251	11111011	8	0
377	101111001	9	1
283	100011011	9	0
425	110101001	9	0
319	100111111	9	0
479	111011111	9	0
719	1011001111	10	1
1079	10000110111	11	1
1619	11001010011	11	0
2429	100101111101	12	1
911	1110001111	10	-2
1367	10101010111	11	1
2051	100000000011	12	1
3077	110000000101	12	0
577	1001000001	10	-2
433	110110001	9	-1
325	101000101	9	0
61	111101	6	-3
23	10111	5	-1
35	100011	6	1
53	110101	6	0
5	101	3	-3
1	1	1	-2

以下示例展示了从 63 开始的每个后续奇数。示例 C:

Base 10	Binary (Start 63)	12	11	10	9	8	7	6	5	4	3	2	1
63	111111							1	1	1	1	1	1
95	1011111						1	0	1	1	1	1	1
143	10001111					1	0	0	0	1	1	1	1
215	11010111					1	1	0	1	0	1	1	1
323	101000011				1	0	1	0	0	0	0	1	1
485	111100101				1	1	1	1	0	0	1	0	1
91	1011011						1	0	1	1	0	1	1
137	10001001					1	0	0	0	1	0	0	1
103	1100111						1	1	0	0	1	1	1
155	10011011					1	0	0	1	1	0	1	1
233	11101001					1	1	1	0	1	0	0	1
175	10101111					1	0	1	0	1	1	1	1
263	100000111				1	0	0	0	0	0	1	1	1
395	110001011				1	1	0	0	0	1	0	1	1
593	1001010001			1	0	0	1	0	1	0	0	0	1
445	110111101				1	1	0	1	1	1	1	0	1
167	10100111					1	0	1	0	0	1	1	1
251	11111011					1	1	1	1	1	0	1	1
377	101111001				1	0	1	1	1	1	0	0	1
283	100011011				1	0	0	0	1	1	0	1	1
425	110101001				1	1	0	1	0	1	0	0	1
319	100111111				1	0	0	1	1	1	1	1	1
479	111011111				1	1	1	0	1	1	1	1	1
719	1011001111			1	0	1	1	0	0	1	1	1	1
1079	10000110111		1	0	0	0	0	1	1	0	1	1	1
1619	11001010011		1	1	0	0	1	0	1	0	0	1	1
2429	100101111101	1	0	0	1	0	1	1	1	1	1	0	1
911	1110001111			1	1	1	0	0	0	1	1	1	1
1367	10101010111		1	0	1	0	1	0	1	0	1	1	1
2051	100000000011	1	0	0	0	0	0	0	0	0	0	1	1
3077	110000000101	1	1	0	0	0	0	0	0	0	1	0	1
577	1001000001			1	0	0	1	0	0	0	0	0	1
433	110110001				1	1	0	1	1	0	0	0	1
325	101000101				1	0	1	0	0	0	1	0	1
61	111101							1	1	1	1	0	1
23	10111								1	0	1	1	1
35	100011							1	0	0	0	1	1
53	110101							1	1	0	1	0	1
5	101										1	0	1
1	1												1

这些彩色单元格展示了一些在所有考拉兹序列中也能观察到的对角线模式:

- 黄色表示第一个 0（从右向左读取）
- 蓝色表示对角线上相邻的交替比特
- 红色表示相邻的比特均为 0 或均为 1，直到每条对角线终止。

该算法根据二进制位的模式，指出一个奇数何时会增长或缩小为下一个奇数。以下是基于算法 1 的一些简单观察:

1. 从左向右读取二进制表示时：a. 从序列中的一个奇数 d 到下一个奇数 n，最高位二进制数字 1 只能向左移动一位。当 d 中从最高位 1 开始出现“11”且前面没有“00”时，就可能发生这种情况。

b. 从一个数 d 到下一个奇数 n，最高位二进制数字 1 可以向右移动多位（稍后将在方法1中说明）。c. 当一个奇数以“01”结尾时，下一个奇数的位数可能相同或更少。

2. 从右向左读取二进制表示时：a. 对于每个后续奇数，第一个二进制位 0 都向右移动一位，直到出现“01”。

与仅使用二进制加法相比，这种算法能提供更多见解[2]。

我们还可以跳过考拉兹序列中的一些数，直接从一个奇数跳到紧接着下一个偶数之后的奇数。为此，我们将使用定理 A，接下来会对此进行讨论。

定理 A：每个正奇数都可以表示为 $n = g2^k - 1$，其中 g 是某个正奇数，且 k是某个正整数。对于正向考拉兹序列（而非仅包含奇数的简化正向考拉兹序列）中的奇数，在持续递增的正向考拉兹序列中，其后续最高数（该数为偶数）将是 $g3^k - 1$。

在证明这一点之前，我们先给出一些例子：

例1：对于奇数 $n = 79 = 5(2)^4 - 1$，在持续递增的正向考拉兹序列中，后续最高数为 $5(3)^4 - 1 = 404$。

示例 2：对于奇数 $n = 127 = (2)^7 - 1$，在持续递增的正向考拉兹序列中，后续数值的最大值为 $(3)^7 - 1 = 2186$。

下面的例子展示了这种方法如何减少从 63 开始到达 1 所需的计算次数。例 D：

Base 10	Binary (Start 63)	12	11	10	9	8	7	6	5	4	3	2	1
63	111111							1	1	1	1	1	1
91	1011011						1	0	1	1	0	1	1
103	1100111						1	1	0	0	1	1	1
175	10101111					1	0	1	0	1	1	1	1
445	110111101				1	1	0	1	1	1	1	0	1
167	10100111					1	0	1	0	0	1	1	1
283	100011011				1	0	0	0	1	1	0	1	1
319	100111111				1	0	0	1	1	1	1	1	1
911	1110001111			1	1	1	0	0	0	1	1	1	1
577	1001000001			1	0	0	1	0	0	0	0	0	1
433	110110001				1	1	0	1	1	0	0	0	1
325	101000101				1	0	1	0	0	0	1	0	1
61	111101							1	1	1	1	0	1
23	10111								1	0	1	1	1
5	101										1	0	1
1	1												1

这只需要 16 行，而不是之前使用的 40 行。此方法基于定理 A，并进一步揭示了奇偶序列中使用的数字。[3]

定理 A 的归纳证明:

基本情形。对于 $k = 1$，则 $n = g2^1 - 1$ 是奇数。正向Collatz函数为 $f(n)$ =%&'(!)$*+$ ~~$(= g3^1 =$~~ 1。这个数是偶数，因为 g、 $g3^k$ 和 1 都是奇数。因此，正向考拉兹序列中的下一个数会更小。所以，基本情形成立。归纳步骤。假设对于某个正整数 k 和奇数 g，$n = g2^k - 1$，使得在不断递增的正向考拉兹序列中，后续出现的最大数为$g3^k - 1$。证明对于 $k + 1$，这也成立。根据归纳假设，$f(g2^k - 1) = g3^k - 1$。因此 $f(g2^{k+1} - 1) = f((2g)2^k - 1) = (2g)3^k - 1$，这是一个等于 $(3^k g)2^1 - 1$ 的奇数。同样，根据归纳假设，f?$(3^k g)2^1 - 1$@ $= (3^k g)3^1 - 1 = g3^{k+1} - 1$。由于 g 是奇数，因此得到所需结果 $g3^{k+1} - 1$，它是偶数，所以下一个数会更小。证毕。

通过观察（也就是说，参见示例 C 和 D），将奇数 n 写成二进制形式时，显而易见，k 是从右向左连续的 1 的个数（也就是说，在简化正向考拉兹序列中，需要经过 k 步，才能让从右向左数遇到的第一个 0 位一路移到最右边）。因此 $k > 0$。

从奇数 n_i 到序列中紧随下一个偶数之后的奇数 n_{i+1}，然后到 $n_{i+1} = g3^k - 1$，其中所有 2 的幂都已被约去。
由于 $n_i + 1 = g2^k$，因此：$n_{i+1} = g3^k - 1 = (g2k)(g3k)\,(g2k) - 1 = (n_i + 1)\,?^{3^k}\,{}_{2k}@ - 1 =$

$$(n_i + 1)\left(\frac{3}{2}\right)^k - 1.$$

因此，以下方程 1 将一个奇数 n_i 变为序列中下一个偶数之后的奇数 n_{i+1}，然后：

$$n_{i+1} = (n_i + 1)\left(\frac{3}{2}\right)^k - 1, \text{ where all powers of 2 have been removed.} \tag{1}$$

由于 n_{i+1} 是一个已去除所有 2 的幂的奇数，因此，将 $(n_i + 1)$ 写成二进制时，从右向左数，w就是连续的 1 的个数? $^3{}_2@^k - 1$，或者，

$$n_{i+1} = \frac{(n_i + 1)\left(\frac{3}{2}\right)^k - 1}{2^w} \tag{2}$$

这可以改写为：

$$\frac{2^w n_{i+1} + 1}{n_i + 1} = \left(\frac{3}{2}\right)^k \tag{3}$$

其中 $k > 0$ 且 $w > 0$。

Hew 提出了前文引用的定理 1，用于确定约化科拉茨映射中二进制字符串长度的变化。这里我们使用算法 1 来推导一种更简单的方法，称为方法1，以区别于 Hew 的定理 1。

方法1。
确定考拉兹序列中从一个奇数 d 到下一个奇数n之间的二进制位数：

- 将奇数 d 转换为二进制表示，在开头的“1”前面加上一个“0”，然后从右向左遍历这个二进制字符串。
- 设置标志 = 1
- 如果标志 = 0 且遇到“11”，则设置标志 = 1；否则，
如果标志 = 1 且遇到“00”，则设置标志 = 0。结束判断

- 从右向左遍历完所有二进制位后停止。
- Num = 从d右端开始的连续交替数字的数量（注意：第一个数字 1 始终计为 1）。

n 的位数将比d多（1+ 标志 – Num）位。

方法1易于实现。它表明，通过检查一个奇数的各位数字，可以确定简化的考拉兹序列中下一个奇数的位数变化。

3. 其他结果

在撰写本文的过程中，我们还观察到科拉茨猜想的一些其他结果，现于本节中展示。

定理 *S*：如果科拉茨猜想不成立，那么该序列中存在一个最小的正整数n。那么n 不可能属于以下任何一种形式：

- $2w$
- $1 + 4w$
- $2 + 3w$
- $3 + 16w$

对于任意整数w。对于这样的n，第一条要点表明n 必定是奇数。

让我们用反证法来证明科拉茨猜想。不失一般性，我们假设所有小于整数n 的正整数都能到达 1。但正整数n 无法到达 1。[换句话说，n 是第一个无法到达 1 的正整数。] 我们将试图证明这会导致矛盾，从而证明该猜想。

根据假设，从n 出发的考拉兹整数序列（正向或反向）中不可能出现小于n 的整数，否则它就会到达 1（这与初始假设矛盾）。因此，n 是其考拉兹序列中的最小整数。

现在，如果n 是偶数，那么$n/2$ 就会是序列中的下一个数，而且更小，这与初始假设矛盾。因此，n 必定是奇数。

限制 1。n 是奇数。因此，n 不是形如 $2w$的数，其中 w为整数。

由于 n 是其正向或反向序列中的最小数，因此序列中位于 n 之前的数必定更大。如果前一个数 m 经 Collatz函数计算后是奇数（也就是说，m 满足 $f(m) = (3m + 1)/2 = n$），那么 $m = (2n - 1)/3$ 小于 n，这是不可能的。因此，m 必定是偶数。由于 m 不可能是奇数，根据 Collatz函数，$m = 2n - 1$ $3 \neq 1 + 2w$ $n \neq 3 + 6w$ +( )/，其中 1 2，$n \neq 2 + 3w$ n 2, 5, 8, 11, 14, [w 为整数。这意味着 (( ))/ 或 。因此，它不可能是诸如 … 这样的数，即形如 $2 + 3w$ 的数，其中 w] 为整数：

限制 2。n 不是形如 $2 + 3w$ 的数，其中 w 为整数。

将奇数 n 代入 Collatz函数，则 $o = 3n+1\ 2$ 是序列中的下一个正整数。现在，如果 o 是偶数，那么后续的数将是 (($3n + 1$ 2 $2 = 3n +$)/)/ ( )/，这会小于 ，而这与最初的假设矛盾。 1 4, n

因此，o 必须是奇数。由于o 不能是偶数，因此$o = (3n + 1)/2 \neq 2w$，其中w 是一个正整数。这意味着 $n \neq 4w - 1$ 3 n 1, 5, 9, 13, ( )/ 。因此，它不能是……这类数：对 17, [$1 + 4w$ w] 于整数 :，形式为

限制3：n 的形式不是$1+4w$，其中w为整数。

将奇数o 代入Collatz函数，然后$p = 3\cdot o+1\ 2 = \left(3\left(3n+1\ 2\right)+1\right)\ 2 = (9n+5)/4$就是序列中的下一个正整数。现在，$p$ 不能被2^2 整除，否则序列将得到一个小于n的数。因此，正整数$p = (9n+5)/4 \neq 4w$，其中w 是一个整数。这意味着 $n \neq 16w-5$ 9 n 3, 19, 35, ( )/ 51, 67, [$3+16w$ w]
。因此，不能是诸如 ..... 的数；整数的此类形式。

限制4：n 不是形如$3+16w$的数，其中w为整数。

结合前面的限制，则n 不是任何整数w对应的以下形式：
$2w$、 $1+4w$、 $2+3w$、 $3+16w$。

在这些限制下，n 就不可能是 3、5、9、11、13、17、19、21、23、25、29、33、35、37、41.....这样的奇数。

或者，采用这种方法，n 就可能是 7、15、27、31、39、43、55、63、75、79、87、91、95、103、107、111、117、119.....这样的奇数。当然，这意味着，从这些数中的任何一个开始的正向考拉兹序列都不可能包含小于初始数 n 的奇数，因为这最终会到达 1（根据最初的假设，n 是违反科拉茨猜想的最小数）。

4. 结论

我们提出了一种执行正向Collatz函数的简单算法，并借此开发出一种比 Hew 提出的方法更简单的方法，用于确定二进制字符串长度的变化。Hew（第 486 页）指出：“这种重新表述带来了有用而优美的见解”，并且还“推测，从这种角度看待 Collatz函数会产生更多结果”。我们对此表示赞同。

致谢

作者感谢 Pablo Castañeda 对本文提出修改意见。他已同意合作撰写一篇运用这些研究成果的新论文。

资助详情： 本文未获得任何资助。**利益披露声明：**作者声明不存在需要披露的竞争性利益。

[1] Patrick Chisan Hew（2021）[0中的二进有理数上的 Collatz：在 .5, 1) 上的分形——比特串长度如何在 3x + 1下发生变化，《实验数学》，30:4，481-488，DOI：10.1080/10586458.2019.1577765

[2] ”科拉茨猜想。” 作为一种以二进制计算的抽象机。维基百科，2020年9月9日，en.wikipedia.org/wiki/Collatz_conjecture。（访问于2022年6月19日）。[3] ”科拉茨猜想。” 作为奇偶