A Space-Limited Algorithm for Computing Collatz Sequences

John F. Kolen
New Mexico Institute of Mininig and Technolgy
Socorro, NM

June 15, 2025

Abstract

This paper describes a space-bounded algorithm for computing Collatz sequences. It relies on performing calculation using a double-base numbers system [?], specifically $2,3$-integers. This representational scheme simplifies the Collatz operations on x of $3x+1$ and $x/2$ to left-shift plus 1 and down-shift. These operations do not destroy terms and will consume unbounded space if left unchecked. It is possible to create boundaries and fold terms back onto a constained playing field. For any positive integer, x, it is possible to perform Collatz iterations within an area $log_2 x + 2$ by $log_2 x$ bits using at most an additional $2log_2 n$ bits of additional storage to hold the escaping bits before processing. Therefore, any Collatz sequence starting from a positive integer will eventually cycle.

1 Introduction

The Collatz conjecture is known by many names: the Ulam problem, the Syracuse problem, and the $3n+1$ conjecture. It is an unsolved problem in mathematics that has been around for over eighty years regarding sequences of number that follow a simple set of rules.

$$C(x) = \begin{cases} 3x+1 & \text{if } x \text{ is odd} \\ x/2 & \text{otherwise} \end{cases} \tag{1}$$

In this paper, a similar version will be used that recognizes that every $3x+1$ operation is always followed by a division since the value is even ($3(2x+1)+1 = 6x+4$).

$$C(x) = \begin{cases} (3x+1)/2 & \text{if } x \text{ is odd} \\ x/2 & \text{otherwise} \end{cases} \tag{2}$$

It has been observed that for positive integer, iterating either of of the functions eventually produces cyclic behavior ($4 \to 2 \to 1 \to 4$ and $2 \to 1 \to 2$, respectively.

This paper describes a space-bounded algorithm for computing the Collatz sequences. By utilizing a double-base number system to carry out calculations, it is possible to construct boundaries that limit the number of bits used to represent sequence elements. The constraints are determined *a priori* from the length of the binary representation of the starting number. The technique involves returning any escaping bits back into the fixed region.

Number Representation A double-base number system using combations of powers of two relatively prime numbers. Here is formulation of 2,3-integers

$$\sum_{i=0}^{n} \sum_{j=0}^{n} x_{ij} 2^i 3^j \tag{3}$$

where x_{ij} are either zero or one. This number system is not unique, as it is possible to represent integers in multiple forms (e.g. 4 is also $1 + 3$).

TODO Diagram showing number representation

Under this system, multiplication and division by either base is a shift. Thus, $3x$ is a right-shift ($x \rightarrow$) and $x/2$ is a down-shift ($x \downarrow$). In the binary system, division is still a shift, but multiplication by three requires a shift and add. Now, the Collatz transformation becomes

$$C(x) = \begin{cases} (x \rightarrow +1) \downarrow & \text{if } x \text{ is odd} \\ x \downarrow & \text{otherwise} \end{cases} \tag{4}$$

Determining parity of 23-intgers is a bit more involved. The only terms that impact parity are those of the form $x_{0j} 3^j$. It is necessary to count these terms modulo two in order to compute parity.

It should be clear that under this representation, the Collatz function never destroys a bit once it has been added to the value. By extending the notation to cover fractional values (where $i < 0$), bit patterns propagate unchanged down and to the right across the representational plane. In fact, the final 2-1 oscillation of the system is one of value and not representation.

TODO Diagram showing 1-2 oscilation

2 Boundaries