A Conjecture Equivalent to the Collatz Conjecture

Ashish Tiwari
Microsoft, Redmond WA 98052, USA
ashish.tiwari@microsoft.com
http://www.csl.sri.com/users/tiwari

Abstract. We present a formulation of the Collatz conjecture that is potentially more amenable to modeling and analysis by automated termination checking tools.

Keywords: Collatz Conjecture · Dynamical Systems · Rewrite Systems.

1 A Conjecture

Consider an organization where we have a linear hierarchy among the employees. Each employee can be in one of 3 different “state of mind”: sleepy, confused, or motivated. Furthermore, each employee can get a gentle-nudge or big-push (that is generated by their supervisor), and each employee can generate either a gentle-nudge or big-push for their subordinate.

Formally, an employee is a Mealy machine [3,2] with 3 states (sleepy, confused, motivated) and two input (and output) symbols (gentle-nudge, big-push). An organization is formally an ordered list of employees. Consequently, the state of an organization is a list of states of its employees; for example, a possible state of an organization with 3 employees is [motivated, confused, sleepy].

Every morning the top member of the organization gets a gentle-nudge, which is propagated down through the day to all the employees in the hierarchy based on the following rules:

1. When a sleepy gets a gentle-nudge, they continue to remain a sleepy and generate a gentle-nudge for the next person.
2. When a sleepy gets a big-push, they turn confused and generate a big-push for the next person.
3. When a confused gets a gentle-nudge, they turn sleepy and generate a big-push for the next person.
4. When a confused gets a big-push, they turn motivated and generate a gentle-nudge for the next person.
5. When a motivated gets a gentle-nudge, they turn confused and generate a gentle-nudge for the next person.
6. When a motivated gets a big-push, they continue to remain motivated and generate a big-push for the next person.
If the last person in the hierarchy generates a *big-push*, then a new junior motivated person is hired (and added to the organization).

If at the end of the day all members of an organization turn sleepy but for possibly the last member, then the organization goes bankrupt.

*Example 1.* Let us illustrate the dynamics of an organization with an example. We start an organization with 3 people, who initially are in the state motivated, confused, sleepy in that order. So, on Day 0, the state of the organization is the tuple \([\text{motivated, confused, sleepy}]\).

On Day 1, motivated gets a gentle-nudge, so, they turn confused and generate a gentle-nudge, which turns the confused into a sleepy and makes it generate a big-push for the third employee (sleepy). As a result, sleepy turns confused and hires a motivated. Thus, at the end of Day 1, the state of the organization is \([\text{confused, sleepy, confused, motivated}]\).

Continuing this way, we notice that at the end of Day 2, Day 3, Day 4, and Day 5 we get to states

- **Day2:** \([\text{sleepy, confused, motivated, confused}]\)
- **Day3:** \([\text{sleepy, sleepy, motivated, motivated}]\)
- **Day4:** \([\text{sleepy, sleepy, confused, confused}]\)
- **Day5:** \([\text{sleepy, sleepy, sleepy, motivated}]\)

The organization goes bankrupt on Day 5.

We can easily formalize the six rules enumerated above as defining the transition relation and output function of a Mealy machine. We write \(s \rightarrow t\) to denote that state \(s\) of an organization (at the start of a day) changes to state \(t\) (at the end of that day). A state \(s\) is bankrupt if only its last element is possibly not sleepy.

Conjecture 1. Every organization eventually goes bankrupt; that is, for any state \(s\) of an organization, there is a \(k \geq 0\) such that if \(s \rightarrow s_1 \rightarrow \cdots \rightarrow s_k\) is a derivation, then \(s_k\) is a bankrupt state.

**2 Conjecture**[1] is equivalent to Collatz Conjecture

We first recall the Collatz conjecture [1].

*Conjecture 2 (Collatz Conjecture).* Let \(f(n)\) be \(n/2\) if \(n\) is even and \(3n + 1\) if \(n\) is odd. Then, for every positive natural number \(n\), there exists a \(k \geq 0\) s.t. \(f^k(n) = 1\).

We now show that Conjecture[1] is equivalent to the Collatz conjecture. We use a novel representation of numbers for this purpose. Each digit in our representation is either 0, 2, or 4, and the place value is interpreted as in Base-3 representation. For example, the number 420 in our new representation denotes the decimal number \(4 \times 3^2 + 2 \times 3^1 + 0 \times 3^0\), which is 42 in base-10. We can only
represent even numbers in this new representation. In this new representation, if 0 is written as sleepy, 2 as confused, and 4 as motivated, then each number $n$ can be seen as a state of an organization. For example, 42 is represented by the organization [motivated, confused, sleepy].

We modify the iterations in the statement of Collatz conjecture to go from even number to an even number as follows: Define $g(n)$ to be $n/2$ if $n/2$ is even, and $3(n/2) + 1$ if $n/2$ is odd. Now, it is an easy exercise to see that if $m = g(n)$, and $s$ is a state representing $n$, then $s \rightarrow t$ where $t$ represents $m$. This shows the equivalence to Collatz conjecture.

Example 2. Let us map the scenario in Example 1 to numbers. At the end of Day 0, we have the number 42. At the end of the next 5 days, we get the numbers 64, 32, 16, 8, 4. Note that 4 is represented as [sleepy, sleepy, sleepy, motivated]. Note that a sleepy member at the head of a state can be dropped without affecting the numerical interpretation of that state.

There is some recent work [4] on trying to prove the Collatz conjecture using automated termination proving techniques for rewriting systems. For this purpose, one needs a rewriting system that mimics the iterations in Collatz conjecture. There is one such rewriting system in [4]. The formulation presented here can also be turned into a (different) rewrite system.

Remark 1 (Open loop interpretation.). Note that the “control input action” is fixed in our setting: the top member of the organization always gets a gentle-nudge at the start of the day. One can consider the open-loop setting where the input to the top element is a control input, and one could study the problem of synthesizing a controller. We do not do so here.

References

1. Collatz conjecture: Collatz conjecture - wikipedia (2021), [Online; accessed Aug 2021]
2. Mealy, G.H.: A method for synthesizing sequential circuits. Bell System Technical Journal 34, 1045–1079 (1955)
3. Mealy machine: Mealy machine - wikipedia (2021), [Online; accessed Aug 2021]
4. Yolcu, E., Aaronson, S., Heule, M.J.H.: An automated approach to the collatz conjecture. CoRR abs/2105.14697 (2021), [Online; accessed Aug 2021]