Universal Pattern Generation by Cellular Automata

Jarkko Kari
Department of Mathematics and Statistics
FI-20014 University of Turku, Finland
Email: jkari@utu.fi

Abstract—Cellular automata are mathematical models for massively parallel processing of information by a large number of identical, locally interconnected tiny processors on a regular grid. Extremely simple processors, or cells, are known to be able to generate together complex patterns. In this talk we consider the problem of designing a cellular automaton that can generate all patterns of states from a finite initial seed. We describe a one-dimensional solution that is based on multiplying numbers by a suitable constant. The automaton to multiply by constant \(3/2\) is shown to be related to some difficult open questions in number theory. We discuss these connections and pose several questions concerning pattern generation in cellular automata.

I. GENERATING ALL PATTERNS

Cellular Automata (CA) are theoretical models for massively parallel information processing where a large number of tiny processors (the cells) are organized into a regular grid. The cells co-operate synchronously using local communications. More precisely, each cell stores a single symbol (its state) that is updated according to a local update rule that specifies the new state depending on the local pattern of states around the cell. The grid is uniform: all processors use the same update rule.

Despite the apparent simplicity of the system, the global behavior can remarkably complicated. Famous examples include the Game-of-Life cellular automaton by John Conway and Rule 110, a one-dimensional cellular automaton with binary alphabet and radius-one local update rule. Both of these are known to support universal computation [1], [2].

Cellular automata were introduced by John von Neumann to demonstrate a universal constructor [3]. Following a suggestion by Stanislaw Ulam, he envisioned a cellular automaton rule on the two-dimensional grid that can assemble arbitrary patterns, including the possibility of self-replication. In this presentation we consider a related question by Stanislaw Ulam about generating all patterns from one finite seed [4, page 30]. The problem is to find a local update rule and an initial configuration with all but finitely many cells in quiescent background states, such that during the evolution all finite patterns over the state set will show up. We describe a solution with six states that works on the one-dimensional line of cells [5]. The automaton \(F_{3/2}\) performs multiplication by constant \(3\) of numbers written in base \(6\). The following illustration shows the time evolution from a single digit \(1\). Consecutive patterns are drawn successively one after the other with time increasing downwards. It is easy to show that all finite sequences of digits get eventually produced at the left border of the generated patterns.

II. NUMBER THEORETIC CONSIDERATIONS

Now suppose we want to generate all sequences of digits at all positions of the space, not only along the left expanding border. A possible candidate solution is the automaton \(F_{3/2}\) that multiplies by constant \(3/2\) numbers written in base \(6\). Its single step consists of executing \(F_{3/2}\) twice, followed by a right shift. Whether it is able to generate all patterns everywhere depends on some difficult open problems in number theory. Namely, for such universal pattern generation to happen one needs to find an integer \(m\) such that the fractional parts of \(\xi(3/2)^i\) are dense in the interval \([0, 1]\) for all \(\xi = m/6^k\).

There is another closely related open problem in number theory. Mahler’s \(Z\)-numbers are defined as positive real numbers \(\xi\) with the property that, for any positive integer \(i\), the fractional part of \(\xi(3/2)^i\) is always less than one half [6]. It is not known whether any \(Z\)-numbers exist. Note that when \(\xi(3/2)^i\) is written in base \(6\), the first digit after the radix point is required to be 0, 1 or 2, with the single exception that the fractional part \(.2555\ldots\) is not allowed. Mahler’s question can hence be rephrased as a problem concerning the possible temporal state sequences of a single cell under \(F_{3/2}\). We can, for example, use the cellular automaton presentation to easily reprove the result published in [6] that there is at most one \(Z\)-number between any two consecutive integers.

Research supported by the Academy of Finland Grant 131558.

978-1-4799-0838-7/13/$31.00 ©2013 IEEE 302
A third related question is the Collatz-problem. By adding a new state to represent the floating radix point, we can modify the cellular automaton $F_{x,3}$ to simulate the function

$$n \mapsto \begin{cases} 
\frac{n}{2}, & n \text{ even}, \\
3n + 1, & n \text{ odd}, 
\end{cases}$$

in base 6 [7]. It is not known whether every orbit eventually reaches $n = 1$.

III. CONCLUSIONS

Several problems about universal pattern generation by cellular automata remain open:

- Does there exist a universal pattern generator in two- and higher dimensional grids?
- Does there exist a solution with fewer than 6 states? In particular, is there a universal pattern generator with just two states?
- Does there exist a strong universal pattern generator that would generate all patterns at all positions of the grid? Does the cellular automaton $F_{x,3/2}$ that multiplies by $3/2$ have this property?

REFERENCES