Code 1599 Cellular Automaton New Patterns

Minjie Lei
University of Michigan

Code 1599 is a cellular automaton rule that produces complex patterns that resemble the phenomenon of free will under simple initial conditions. However, by expanding the range of initial conditions considered to include complex backgrounds, we find that code 1599 can produce patterns that are entirely different from its ordinary structure, but similar or even structurally identical to some elementary cellular automata rules.

1. Code 1599 Introduction

Code 1599 was first introduced by Stephen Wolfram in 1983 [1]. It is a three-color totalistic cellular automaton rule, which means each cell has only three possible colors and depends only on the total sum of the values of its neighboring cells. Figure 1 shows the rule plot for code 1599.

Figure 1. Rule plot for code 1599. The plot shows that each color is given a number, and the color number of the next cell is determined by the sum of the color numbers of the previous cell and its neighboring cells.

As simple as the rule seems, the patterns produced by code 1599 with simple initial conditions can be very interesting and distinct from other cellular automata rules. The typical code 1599 behavior as discussed in Wolfram's paper and subsequent discussions about the rule is shown in Figure 2.

We can see that the rule evolves into a complex tree-like structure. Yet what makes code 1599 even more fascinating is that for many initial conditions if not all, after very large steps, the complex behavior turns into simple repetitive patterns as shown in Figure 3.

Complex Systems, **25** © 2016 Complex Systems Publications, Inc.
https://doi.org/10.25088/ComplexSystems.25.4.285

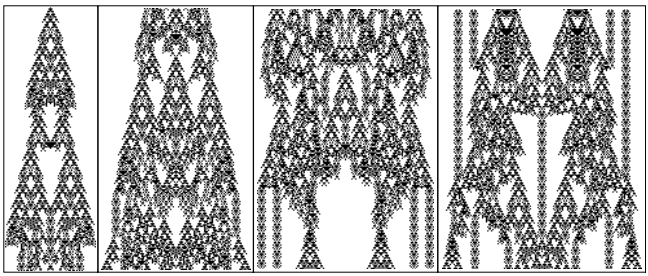

Figure 2. Code 1599, initial condition: a single gray cell in a white background.

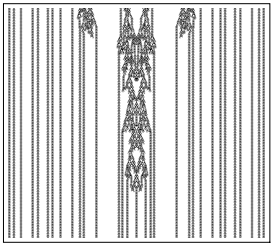

Figure 3. Code 1599, initial condition: a single gray cell in a white background, evolved from step 7800 to 8400.

Whether or not this unusual behavior applies to all initial conditions is still an open problem, which is included in "A New Kind of Science Open Projects and Problems" [2]. The problem was studied during the 2005 Wolfram Summer School from an artist's perspective. Three million specified initial conditions of code 1599 were searched; only 11% were dead within 300 steps, while others exhibited complex evolving behaviors. Of the remaining interesting initial conditions, two equivalent initial conditions reached a definite pattern after 8282 steps of evolution. Because of its complex and unpredictable long-term behavior, Wolfram used code 1599 in his book *A New Kind of Science* [3] as an example to illustrate his idea of computational irreducibility.

In this paper, we will not attempt to solve the long-term behavior of code 1599. Instead, we will discuss some new behaviors of code 1599 we found that are different from the typical code 1599 behavior discussed in Wolfram's original paper and other existing literature. By expanding our search of initial conditions to include initial conditions of complex backgrounds, we were able to generate behaviors that resemble elementary cellular automaton rules, instead of the typical code 1599 behavior. This behavior had not been found before in other cellular automata rules. For the remainder of the paper, we will

https://doi.org/10.25088/ComplexSystems.25.4.285

first introduce the various new patterns of code 1599 we found. Then we will make some qualitative observations about the initial conditions that produce the new patterns, as well as comment on the behavior of code 1599. Finally, we will conclude and discuss some possible limitations and future directions for research.

2. Code 1599 New Patterns

During the study of the code 1599 patterns of different initial conditions, we expanded our search space to initial conditions with repetitive background and found surprisingly that some of them produce patterns entirely different from normal complex code 1599 behavior, but instead resemble elementary cellular automata. Below is a summary of the new patterns we found.

2.1 Rule 4-Like Structure

This new behavior rises without a complex repetitive background, but instead has relatively complex starting cells. The notation for the initial conditions used in the figure captions is Mathematica syntax. It is composed of two parts. The first array is the starting cells, superimposed on the repetitive background, which is specified by the second array [4]. We will use the same notation for the following patterns.

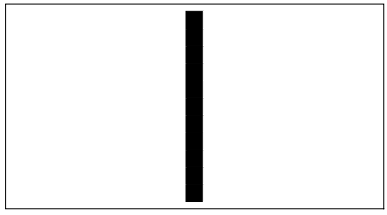

Figure 4. Rule 4, initial condition {{1}, 0}.

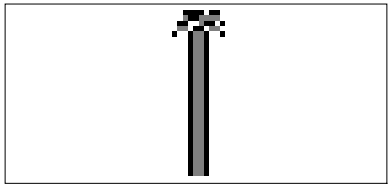

Figure 5. Code 1599, initial condition {{0, 0, 2, 2, 2, 2, 0, 2, 2}, {0}}.

https://doi.org/10.25088/ComplexSystems.25.4.285

2.2 Rule 50/250-Like Structure

With a complicated initial condition of a repetitive background, code 1599 produces a surprisingly simple pattern that resembles cellular automata rules 50 and 250. Because code 1599 is a three-color rule, the details of the patterns are different, but they are structurally similar and nothing like regular code 1599 behavior.

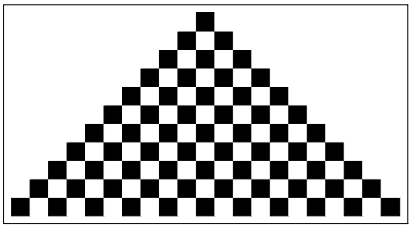

Figure 6. Rule 50, initial condition {{1}, 0}.

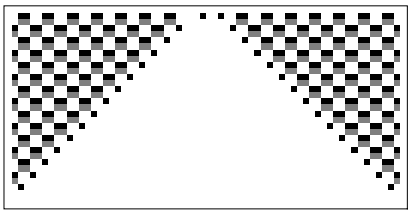

Figure 7. Code 1599, initial condition {{0, 0, 0, 0, 2, 0, 0},{0, 0, 2, 2}}.

2.3 Rule 54-Like Structure

Code 1599 with some special initial conditions can also produce a pattern that resembles a rule 54 structure. Here the color and background are different, but the main structure is similar.

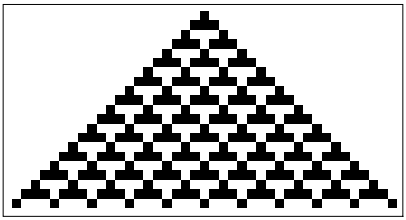

Figure 8. Rule 54, initial condition {{1}, 0}.

https://doi.org/10.25088/ComplexSystems.25.4.285

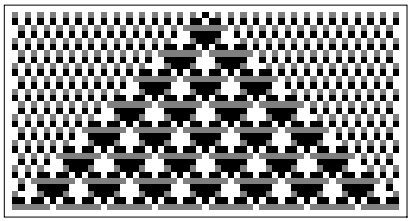

Figure 9. Code 1599, initial condition {{0, 1, 0, 2},{0, 1}}.

2.4 Rule 94-Like Structure

Rule 94-like pattern produced by code 1599 cellular automaton with complex initial conditions.

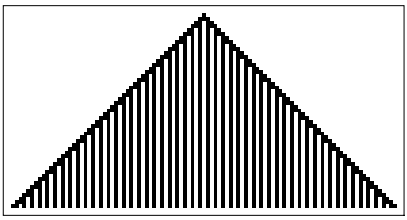

Figure 10. Rule 94, initial condition {{1}, 0}.

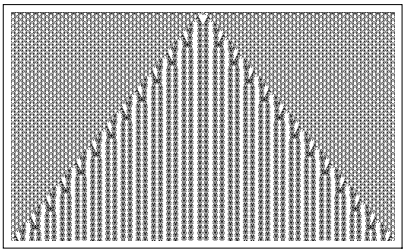

Figure 11. Code 1599, initial condition {0, 2, 0, 0},{1, 1, 1, 1, 1, 2, 2, 1, 2, 2}.

2.5 Rule 105/150-Like Structure

One of the most surprising results is that code 1599 could produce a nested pattern that is structurally identical to rules 105 and 150. Figure 12 shows what rules 105 and 150 look like; they are complement rules [5].

https://doi.org/10.25088/ComplexSystems.25.4.285

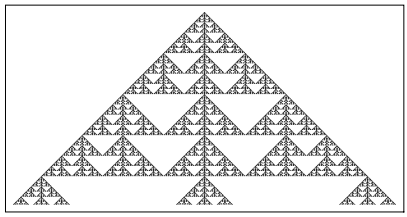

Figure 12. Rule 150, initial condition {{1}, 0}.

Figure 13 shows what is produced by code 1599, which is structurally identical to the nested elementary patterns in Figure 12. There are many more initial conditions that can produce a similar pattern for code 1599. We will discuss them in more detail in the next section.

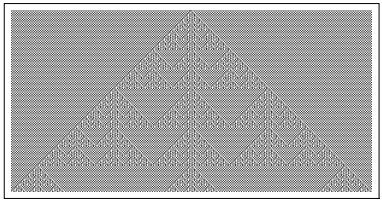

Figure 13. Code 1599, initial condition {{2, 1, 2}, {0, 1, 1, 0}}.

2.6 Rule 30/86-Like Structure

Some initial conditions of code 1599 produce a chaotic pattern alongside the usual code 1599 behavior. The patterns are similar to those produced by rules 30 and 86.

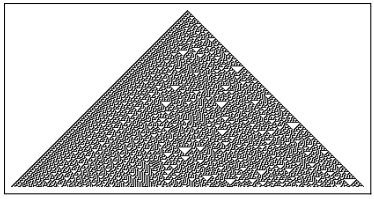

Figure 14. Rule 30, initial condition {{1}, 0}.

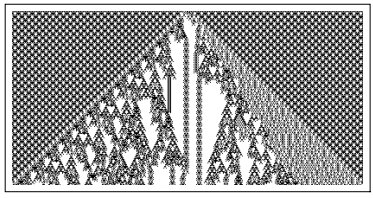

Figure 15(a). Code 1599, initial condition {{0, 2, 0, 2, 2, 0, 2},{2, 1, 1, 2, 1, 2, 1, 1}}.

https://doi.org/10.25088/ComplexSystems.25.4.285

Figure 15(b) shows a closer look at the chaotic part of the code 1599 pattern. We can clearly see the randomly distributed little triangles forming a complicated structure.

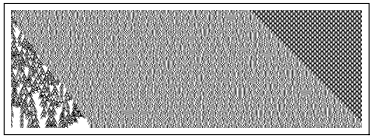

Figure 15(b). Closer look at the chaotic pattern produced by code 1599 with complex initial conditions.

Figure 16 shows a similar pattern. It has the rule 30-like pattern on the left side of the traditional code 1599 behavior.

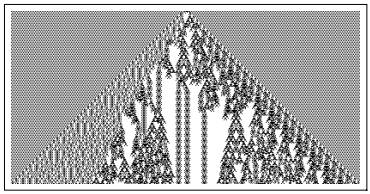

Figure 16. Code 1599, initial condition {{1, 1, 2, 1, 2, 2, 2, 1},{2, 0}}.

2.7 Rule 73-Like Structure

A rule 73-like pattern is also seen, with the wedges on two sides of the triangle being the rule 30-like chaotic pattern. The structure in the middle of the triangle is the normal code 1599 structure.

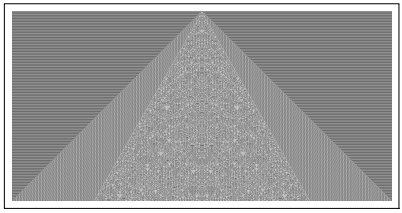

Figure 17. Rule 73, initial condition {{1}, 0}.

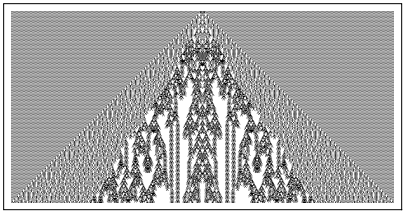

Figure 18. Code 1599, initial condition {{2,1,2,0,2,1,2},{2,0}}.

https://doi.org/10.25088/ComplexSystems.25.4.285

In summary, we have seen many examples where code 1599, when expanding from complex initial conditions, could produce patterns entirely different from its normal patterns. These patterns resemble simple elementary rules. In the next section, we will discuss some of the qualitative observations we made from the initial conditions that produce these patterns and discuss if this behavior is unique to code 1599.

3. Observations

3.1 On the Initial Conditions

Here are two qualitative observations we made about the initial conditions of code 1599 that produce the new patterns:

- Similar initial conditions produce similar new patterns.
- In some cases, a small change in initial conditions can produce dramatically different patterns.

The first observation seems intuitive, but it would allow us to find more initial conditions that generate a certain pattern and help us reach a quantitative conclusion about what kinds of initial conditions produce what types of patterns. Table 1 gives a list of initial conditions that produce the rule 105/150-like pattern.

Starting Cells	Background
{0, 0, 0, 0}	{1, 0}
{0, 0, 0, 0}	{0, 1}
{2, 0, 0, 0}	{0, 1}
{0, 2, 0, 0}	{0, 1}
{2, 1, 2, 1}	{1, 2, 2, 1}
{2, 0, 2, 0}	{0, 2, 2, 0}

Table 1. Initial conditions that produce the rule 105/150-like pattern.

From Table 1, we can see that many initial conditions that produce the same pattern differ only by a few numbers and the position of the numbers. The last two initial conditions also suggest a symmetry.

However, as the second observation suggests, it is not always the case that a small change in the initial condition would produce the same pattern. Sometimes a small change of numbers or positions can dramatically change the pattern. Figures 19 –23 show some examples.

https://doi.org/10.25088/ComplexSystems.25.4.285

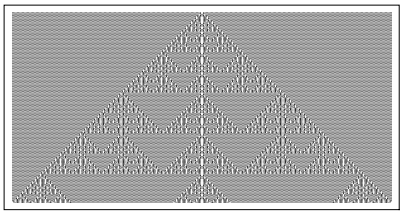

Figure 19. Initial condition {{0, 0, 0, 0}, {0, 1}}.

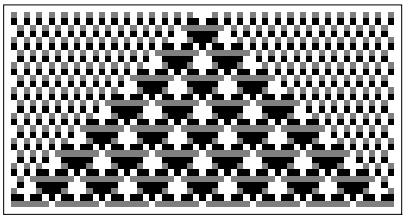

Figure 20. Initial condition {{0, 0, 0, 1}, {0, 1}}.

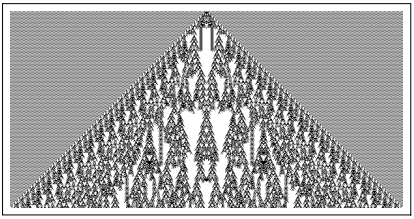

Figure 21. Initial condition {{0, 0, 1, 0}, {0, 1}}.

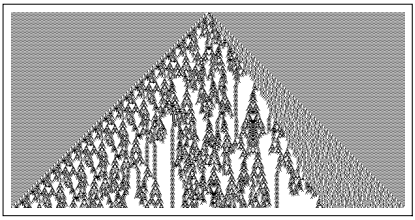

Figure 22. Initial condition {{1, 0, 1, 0},{0, 1}}.

https://doi.org/10.25088/ComplexSystems.25.4.285

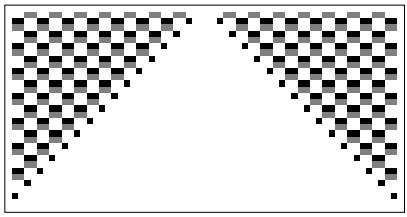

Figure 23. Initial condition {{0, 0, 0, 0},{0, 1, 1, 0}}.

By simply changing one number or the position of one number in the initial conditions, the pattern produced by code 1599 can be dramatically different in structure and complexity. This may be a unique property among cellular automata rules.

3.2 On the New Behaviors of Code 1599

The behavior that with complex initial conditions, code 1599 can produce patterns different from its usual pattern, but that instead resemble some elementary rules, had not been seen in previous studies on cellular automaton rules. However, by running the same search on other rules, we did find at least one other such case in rule 94, shown in Figures 24 and 25. This means that such behavior is not unique to code 1599, and raises the question about what properties of a rule make the behavior possible, which we wish to investigate further in future research.

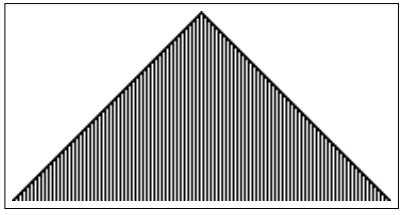

Figure 24. Rule 94 with initial condition {{1}, 0}.

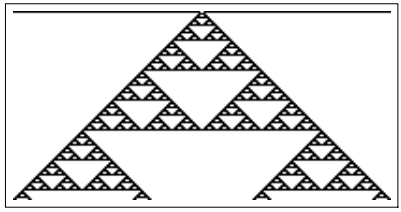

Figure 25. Rule 94 with initial condition {{1, 1, 1, 0, 0}, 1}, rule 18-like nested pattern.

https://doi.org/10.25088/ComplexSystems.25.4.285

4. Conclusion

By expanding initial conditions to include complex backgrounds, we found that code 1599 can produce patterns different from its typical structure discussed in existing research. Instead, they resemble elementary cellular automata rules. We made some further qualitative observations about the initial conditions that produce the patterns and the behavior of code 1599. There are still some open problems we will need to address in the future. First, will the elementary cellular automata-like patterns persist after large steps? Typical code 1599 patterns have a very unpredictable long-term behavior, therefore we cannot be sure some new patterns, such as the rule 30-like pattern, will persist without examining them after large steps. Also, there are elementary cellular automaton rules like rule 110 that we looked for in code 1599 but did not find. We intend to address the problems when we pursue the subject more quantitatively in the future.

References

[1] S. Wolfram, "Cellular Automata," *Los Alamos Science*, **9**, 1983 pp. 2–21. http://library.lanl.gov/cgi-bin/getfile?09-01.pdf.

[2] S. Wolfram. "A New Kind of Science Open Problems and Projects" from Wolfram Science—A Wolfram Web Resource. https://www.wolframscience.com/openproblems/NKSOpenProblems.pdf.

[3] S. Wolfram, *A New Kind of Science*, Champaign, IL: Wolfram Media, Inc., 2002.

[4] "Cellular Automaton" from Wolfram Language and System Documentation Center—A Wolfram Web Resource. http://reference.wolfram.com/language/ref/CellularAutomaton.html.

[5] E. W. Weisstein. "Rule 150" from Wolfram MathWorld—A Wolfram Web Resource. http://mathworld.wolfram.com/Rule150.html.

https://doi.org/10.25088/ComplexSystems.25.4.285