Drawdown Automata, Part 1: Basic Concepts

Cellular Automata

A cellular automaton is an array of identical, interacting cells. There are many possible geometries for cellular automata; the most commonly used are shown in Figure 1.

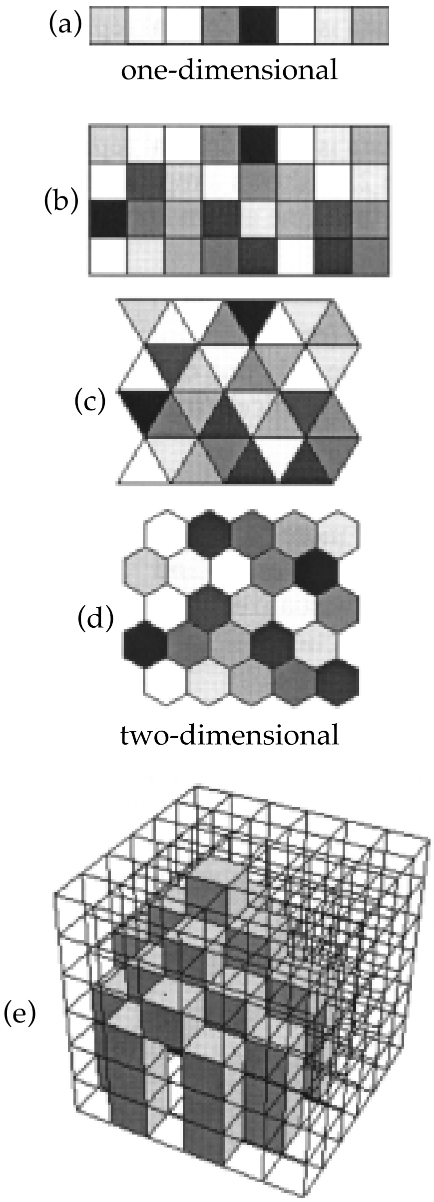


Figure 1. Cellular Automata Geometries

We'll confine our attention, at least initially, to square cells in two dimensions, as shown in Figure 1b.

The cells in cellular automata have states, indicated in Figure 1 by different colors. We'll confine our attention to cellular automata in which the cells have only two states, 1 and 0, indicated by black and white respectively. Figure 2 shows an example.

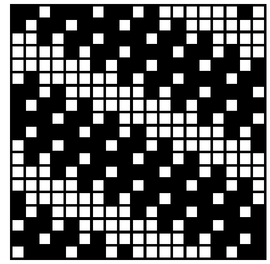


Figure 2. Drawdown Automata

The choice of the term drawdown automata is deliberate; that is exactly the purpose for which we'll use them.

A cellular automaton, as a whole, passes through a succession of configurations corresponding to the states of its cells. The automaton goes from one configuration to another at discrete intervals of time, the states of all its cells changing in parallel. The change of state of a cell is determined by a transition rule that depends on the neighbors of the cell and is the same for all cells in the automaton.

The neighborhood of a cell can be defined in different ways. Figure 3 shows one of the most frequently used neighborhoods, which is named after John von Neumann, who used it in his studies of self-reproducing machines. See the side bar on the next page.

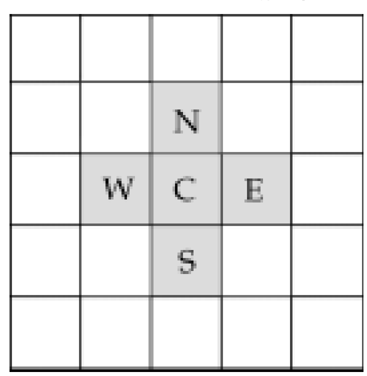


Figure 3. Von Neumann 5-Neighborhood

The cell itself is labeled C. Its four neigh-

Cellular Automata Applications

John von Neumann, who played a major role in the design of modern computers, was among the first to use cellular automata as models for abstract machines.

John von Neumann
1903-1957

He proved that it is possible, in principle, to design machines that not only are capable of reproduction but also of evolving into more complicated machines.

Cellular automata are widely used as discrete models of physical systems and have been used to simulate a wide range of natural processes such as turbulent fluid flow, gas diffusion, forest fires, and avalanches. Cellular automata can even be used to generate pseudo-random numbers.

Considered abstractly, cellular automata exhibit a wide variety of behaviors: self organization, chaos, pattern formation, and fractals.

John Conway's Game of Life is the best known abstract application of cellular automata. In it, a wide variety of patterns with life-like properties are born, interact, and die in fascinating and complex ways. Vast amounts of human and computer time have been expended exploring this strange world.

bors are labeled according to their relative positions according to the points of the compass.

Figure 4 shows another commonly used neighborhood, named after Edward F. Moore, an early pioneer in studies of cellular automata.

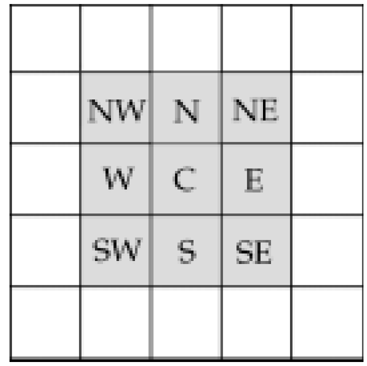


Figure 4. Moore 9-Neighborhood

Subscripts are used to denote times, which proceed 1, 2, 3, … . For example C_{10} is the state of C at time 10.

A typical transition rule is the "parity rule" for the 5-neighborhood:

$$C_{i+1} = (C_i + N_i + E_i + S_i + W_i) \bmod 2$$

That is, $C_{i+1} = 1$ if the sum of the neighborhood states (including C itself) is odd and 0 otherwise.

Another interesting rule is the "voter rule" for the 5-neighborhood:

$$C_{i+1} = 1 \text{ if } (N_i + E_i + S_i + W_i) > 2$$
$$C_{i+1} = 0 \text{ if } (N_i + E_i + S_i + W_i) < 2$$
$$C_{i+1} = \sim C_i \text{ otherwise}$$

where ~C is the complement of C: 1 if C = 0, 0 if C = 1.

Note that in the voter rule, the result may depend on the value of C, while in the parity rule, it does not: In the parity rule, C is treated no differently than its neighbors.

Cellular Automata Topology

Before we can go further, we need to deal with a sticky issue: What happens to the cells at the edge of an automaton? What are their neighbors?

This problem can be dealt with in several ways. The way chosen depends on the context in which the cellular automaton is considered.

One way is to consider the cellular automaton to be infinite without edges, with cells extending off indefinitely in all four directions. Another way is to treat the cells at the edges as unchanging, serving as a kind of static border.

A less obvious but natural and useful way in the context of drawdowns is to consider the cellular automaton to wrap around from edge to edge. See Figure 5.

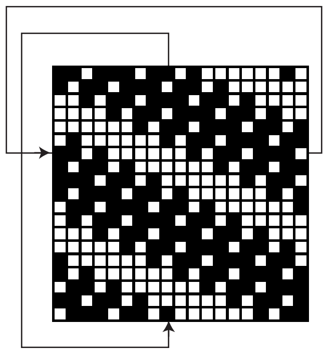


Figure 5. Neighborhood Wraparound

Thus, the N neighbor of a cell on the top edge is the cell in the corresponding row on the bottom edge, and so on.

From a topological point of view, this constitutes wraparound of the horizontal and vertical edges and also of the top and bottom edges. The result is a three-dimensional surface known as a torus, as suggested by Figure 6.

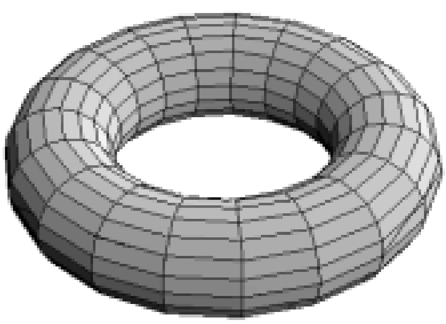


Figure 6. Torus

The cells on this torus are distorted because the "horizontal" circumference is larger than the "vertical" circumference so that the general shape to be seen more easily. Perspective causes the shapes of the cells to be skewed.

It is not necessary to actually make a toroidal cellular automaton. It is only necessary, when applying rules, to determine the neighbors according to the wraparound topology.

It is worth noting that edge wraparound is equivalent to an infinite plane of repeats.

Pattern Sequences

When a cellular automaton is started in a specific configuration and a rule is applied repeatedly, a pattern sequence results.

Figure 7 shows the beginning of the pattern sequence that results from applying the 5-neighborhood parity rule to the pattern shown in Figure 2. The complete sequence has 511 distinct patterns; at the next iteration, the original pattern reappears; after this, there are no new patterns.

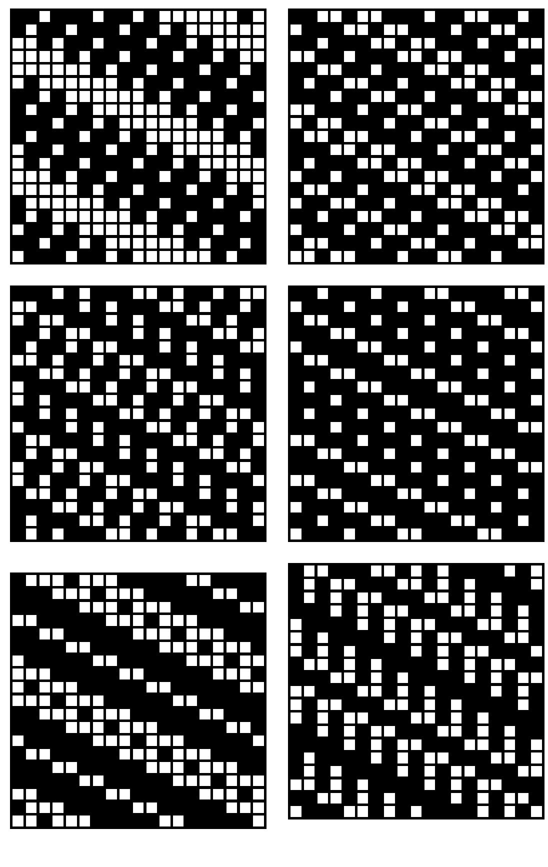


Figure 7. Parity Rule Sequence

Figure 8 shows the pattern sequence that results from applying the voter rule to the pattern shown in Figure 2. In this case, there are only three distinct patterns; the fourth is the same as the second.

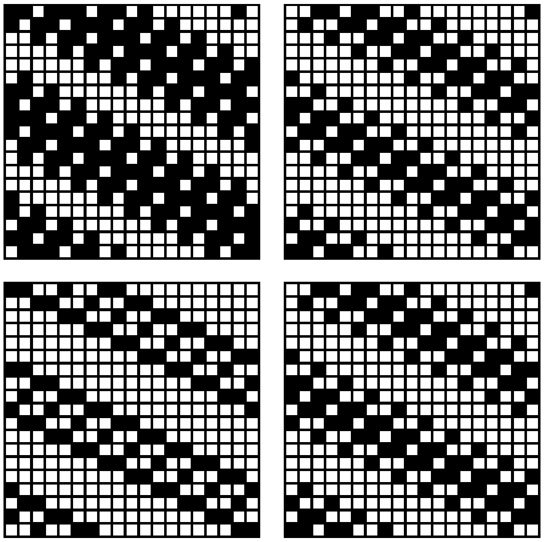


Figure 8. Voter Rule Example

Figure 9 shows the beginning of the pattern sequence for the 5-neighborhood parity rule starting with a symmetric pattern. There are 511 distinct patterns in all, the 512th being the same as the first.

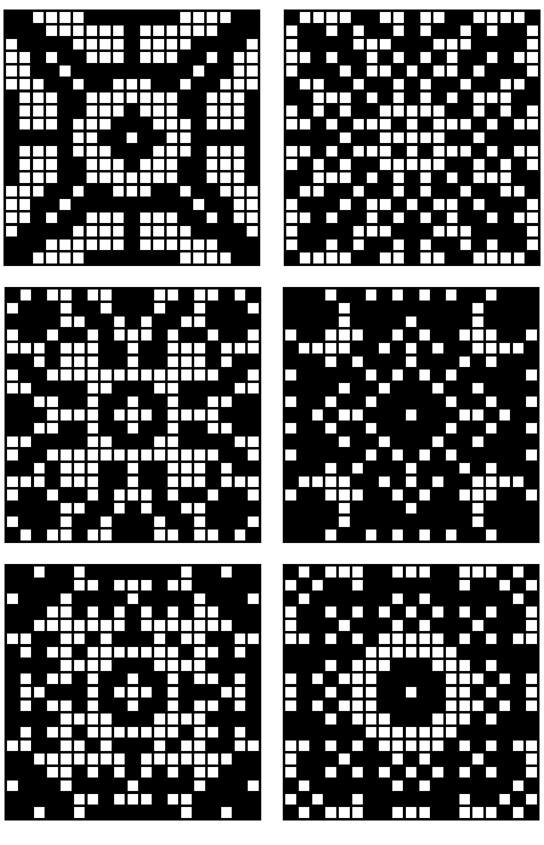


Figure 9. Parity Rule with Symmetric Pattern

The voter rule, as in the previous example, yields fewer distinct patterns starting with this initial pattern, the seventh being the same as the first. See Figure 10.

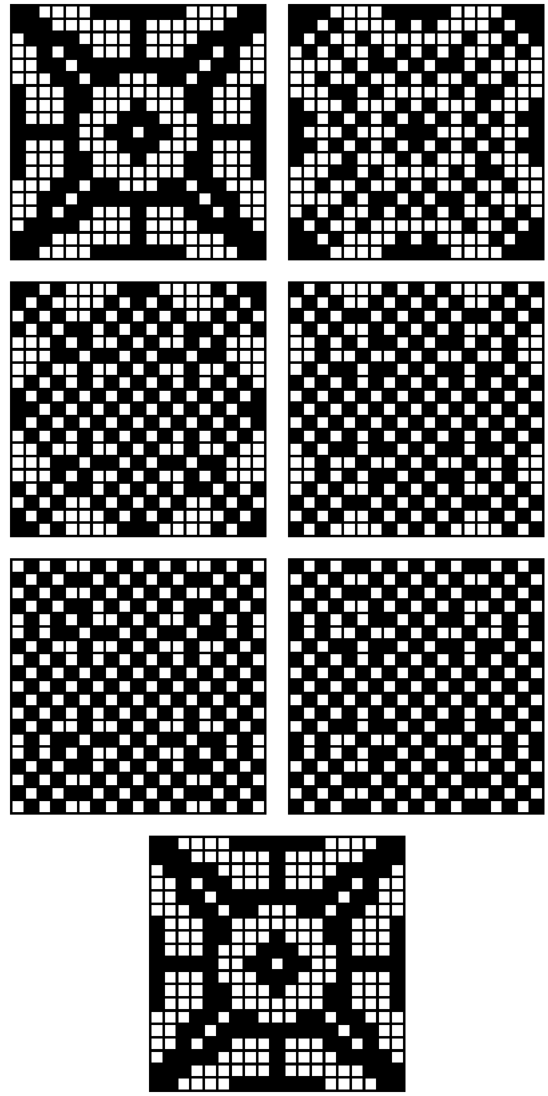


Figure 10. Voter Rule with Symmetric Pattern

An interesting way to explore the effects of a rule is to start with a “seed”, a single black cell in a field of white ones.

In such pattern sequences, it usually takes some time for the seed to spread results to a sufficient extent that useful patterns result. Figure 11 shows the pattern sequence for a single seed and the 5-neighborhood parity rule. There are 511 different patterns in all. The first eight are shown in this Figure. Figure 12 shows four of the more interesting patterns from the first 64.

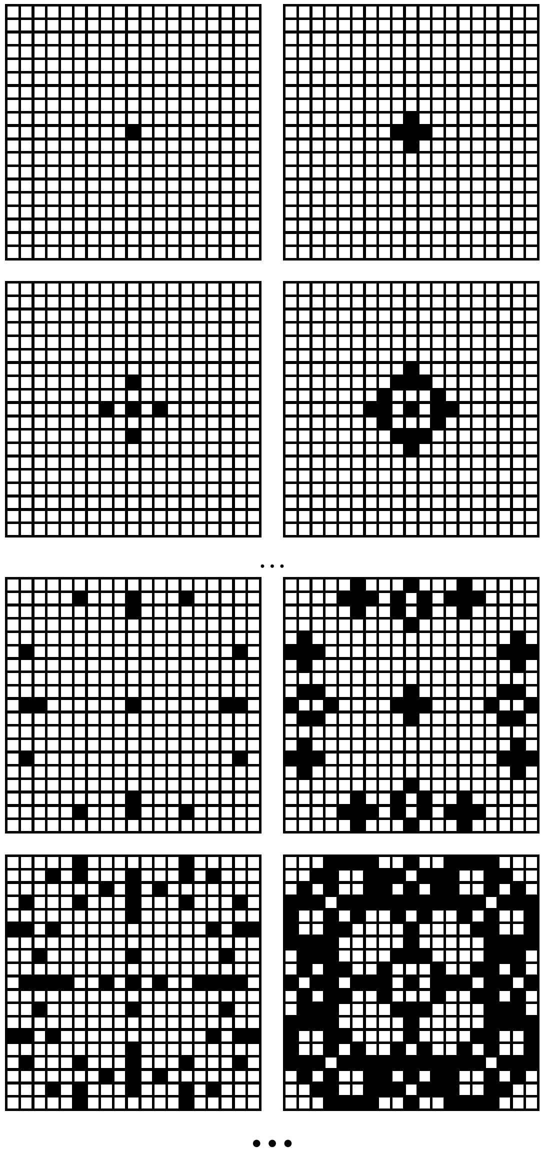


Figure 11. Parity Pattern Sequence Start-up

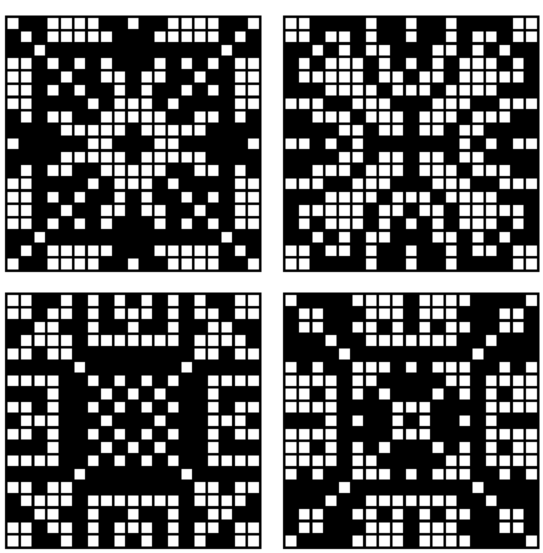


Figure 12. Selections from First 64

An apparently uninteresting 9-neighborhood rule, called "1-of-8", is

$$C_{i+1} = 1 \text{ if } (NW_i + N_i + NE_i + E_i + SE_i + S_i + SW_i + S_i) = 1$$

$$C_{i+1} = C_i \text{ otherwise}$$

This rule, starting with a single seed, produces a fascination fractal pattern. See Figure 13.

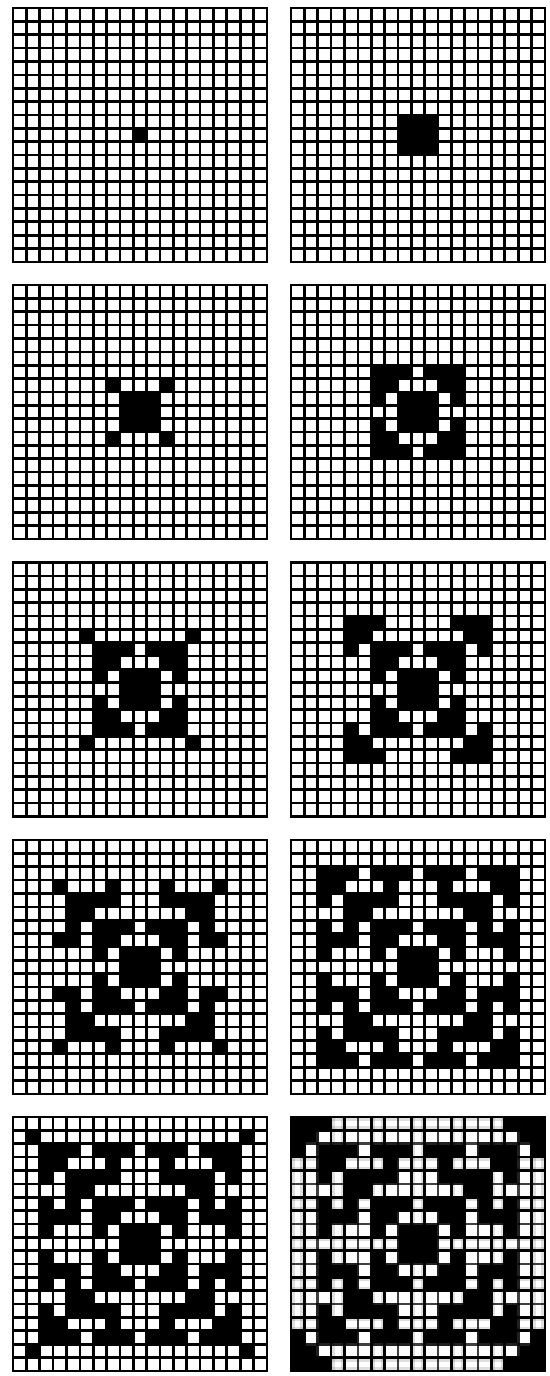


Figure 13. 1-of-8 Rule Fractal Pattern

All patterns after the 10th are the same as the 10th.

Putting the seed off center illustrates the effect of wraparound topology. See Figure 14.

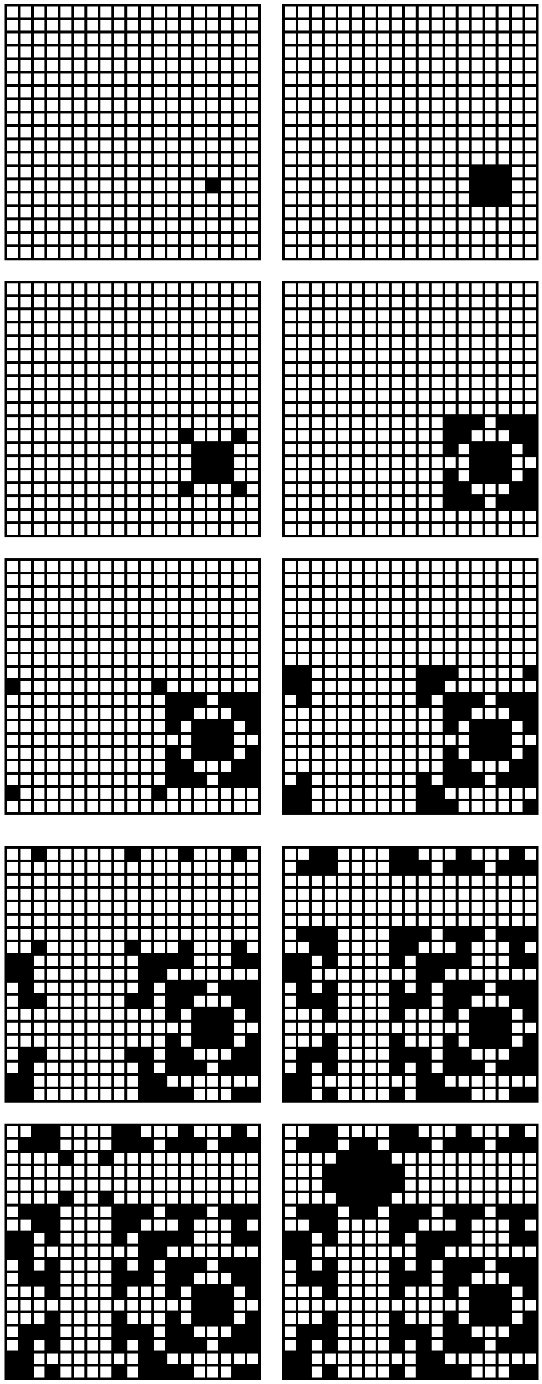


Figure 14. Offset and Wraparound

The patterns in Figure 14 are the same as those in Figure 13; they are just at different positions on the torus.

Structural and Aesthetic Concerns

Many patterns produced by drawdown automata are unsuitable for interlacement for structural reasons. Notable examples are the initial patterns in sequences starting with a single seed. Other patterns simply are unattractive.

Drawdown automata can produce thousands of patterns quickly. Even with the rejection of obviously unsuitable patterns, the problem is one of excess. How can really good patterns be found in seas of possibilities?

One approach is to start with a conventional drawdown pattern such as the one shown in Figure 2 and look for interesting examples "of type".

Another approach is to start with an attractive and structurally sound symmetric pattern and apply a symmetric rule (one, like the parity rule, in which the result does not depend on the actual positions of specific neighbors). This avoids the problem with an overwhelming cascade of chaotic patterns that may result by starting with a pattern without much structure and applying an asymmetric rule.

Size matters also. We've used 19 × 19 patterns in this article for presentation purposes. Large patterns usually lead to longer pattern sequences and allow more interesting results, as illustrated by the large 1-of-8 pattern shown at the end of this article.

Patterns need not be square, nor does symmetry need to be of the "mirror" type, as shown in the examples in this article, to produce attractive results. There is a world to explore.

Resources

You can find a large number of cellular automata programs on the Web, ranging from freeware to commercial applications. Programs for Windows, the Macintosh, and Linux are plentiful. Some Web sites have applets that allow you to experiment interactively.

Some of cellular automata programs have limited capabilities but are adequate for drawdown explorations. Others are elaborate and very capable, far exceeding the kinds of things described in this article.

The Web also contains a large amount of information on cellular automata, ranging from introductory surveys to in-depth coverage of specialized topics.

There are several books on cellular automata, and many recent books on recreational mathematics cover cellular automata to some extent.

Gaylord and Nishidate [1] deal with cellular automata in some detail but with an emphasis on simulating physical systems using the symbolic mathematics program *Mathematica*. Poundstone [2] provides extensive conceptual background, but with a focus the Game of Life. Toffoli and Margolis [3] provide extensive coverage, including some information of a technical nature. Wolfram [4] contains an extensive collection of papers by the leading current researcher in cellular automata. Although much of the material in this book is technical, there are fascinating sections for the layperson along with many drawings and pictures that are inspirational in themselves.

In the many books on recreational mathematics, Gardner [5], Dewdney [6], and Peterson [7] are accessible and stimulating.

To explore drawdown automata, however, all you need is a computer, Web access, and an adventuresome spirit.

What's to Come?

This article only scratches the surface of the topic of drawdown automata. Subsequent articles will explore different neighborhoods and rules in more depth, look at the properties of pattern sequences, go on to different geometries, and finally (?) explore color drawdowns.

References

1. *Modeling Nature: Cellular Automata Simulations with Mathematica*, Richard J. Gaylord and Kazume Nishidate, Springer-Verlag, 1996.

2. *The Recursive Universe; Cosmic Complexity and the Limits of Scientific Knowledge*, William Poundstone, Contemporary Books, 1985.

3. *Cellular Automata Machines: A New Environment for Modeling*, Tommaso Toffoli and Norman Margolus, The MIT Press, 1991.

4. *Cellular Automata and Complexity; Collected Papers*, Stephen Wolfram, Addison-Wesley, 1994.

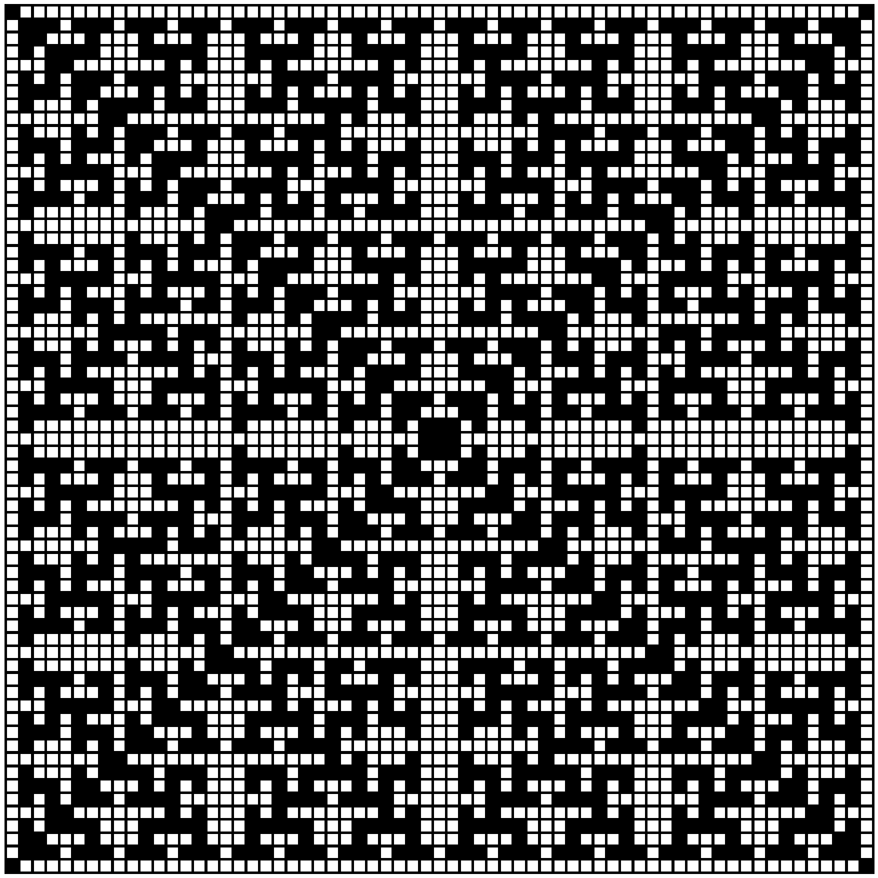
5. *Wheels, Life and Other Mathematical Amusements*, Martin Gardner, Freeman, 1983.

6. *The Armchair Universe: An Exploration of Computer Worlds*, A. K. Dewdney, Freeman, 1988.

7. *The Mathematical Tourist: Snapshots from Modern Mathematics*, Ivars Peterson, Freeman, 1988.

Ralph E. Griswold

Department of Computer Science
The University of Arizona
Tucson, Arizona



65×65 1-of-8 Pattern