Modeling Braids, Cables, and Weaves with Stranded Cellular Automata

Joshua Holden*
Department of Mathematics, Rose-Hulman Institute of Technology
5500 Wabash Ave., Terre Haute, IN 47803, USA
holden@rose-hulman.edu

Lana Holden
Skew Loose, LLC, Terre Haute, IN 47803, USA
http://www.ravelry.com/designers/lana-holden
lana@knittinglaboratory.com

Abstract

The mathematical study of braids involves with mathematical representations of one-dimensional strands in three-dimensional space. These strands are also sometimes viewed as representing the movement through a time dimension of points in two-dimensional space. On the other hand, the study of cellular automata usually involves a one- or two-dimensional grid of cells which evolve through a time dimension according to specified rules. This time dimension is often represented as an extra spacial dimension. The ideas of representing both strands in space and cellular automata have also been explored in many artistic media, including drawing, sculpture, knitting, crochet, and weaving. This paper presents a system of Stranded Cellular Automata which realistically captures the behavior of strands in certain media, such as knitting and crochet.

Introduction

Knots and other forms of interlacing strands have been used for decorative and artistic purposes for thousands of years. These can be actual knots, such as in Chinese knotwork or macramé, or woven textiles, such as loom weaving, basket weaving, or plaiting. They can also be two-dimensional depictions of interlaced strands, which can be found in Roman mosaics, in the "Celtic Knot" decorations of Christian manuscripts from the British Isle, and in Islamic art, among many other places. And three-dimensional materials which are not actually interlaced can be used to give an illusion of interlacing, as in bas-relief, in knitted or crocheted "cables", or in the "traveling eyelets" used in knitted lace.

The complex but structured nature of many of these patterns suggests the possible use of cellular automata to model them. A cellular automaton is a mathematical construct which models a system evolving in time. It is characterized by a discrete set of cells, finite or infinite, in a regular grid, with a finite number of states that a cell can be in. Each cell has a well-defined finite neighborhood which determines how the cell will evolve through the different states. Time moves in discrete steps, and the state of each cell at time t is determined by the states of its neighbors at time $t-1$. Finally, each cell uses the same rule to determine its state. Examples include the "Game of Life", invented by John Conway [1] and the "Elementary Cellular Automata" popularized by Stephen Wolfram [9].

Given a grid, a rule, and a starting state, the evolution of a cellular automaton is often depicted graphically. In the case of the Game of Life, the grid is two-dimensional and the evolution is generally portrayed using animation. In the case of Elementary Cellular Automata, the grid is one-dimensional and the evolution is often shown using a second dimension to represent time.

Cellular Automata for Braids, Cables, and Weaves

In the examples noted above, each cell can only be in one of two states. Since we want to represent "stranded" designs, we will need more states. Each cell can hold no strands, only a strand starting on the left, only a strand starting on the right, or strands starting on both sides. The strands can be upright or slant from one side of the cell to the other. If there are two slanting strands they will cross, and we need to specify which strand is on "top". (For now, we are not considering the possibility where one strand is slanted and the other is upright.) All of this amounts to four pieces of information but only eight distinct states, as shown in Figure 1.

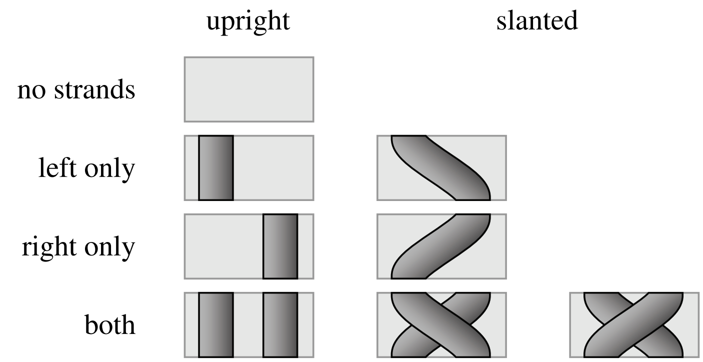

Figure 1 : *Each cell will store four pieces of information in eight states.*

The neighborhood we will use will be a one-dimensional version of the Margolus neighborhood studied in [7, Chapter 12], as shown in Figure 2. Unlike the standard for Elementary Cellular Automata, we will represent time as moving from the bottom of our pictures to the top, in order to make it resemble a knitting or crochet pattern. Therefore the state of a given cell at time t depends on the state of two neighbor cells at time $t-1$, where those cells are represented as the two cells which touch it from below in the picture.

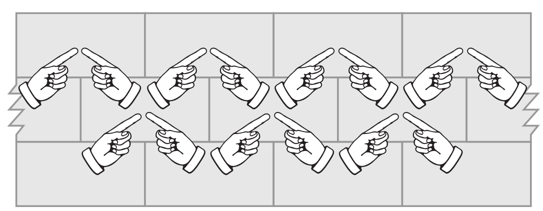

Figure 2 : *The neighbors of each cell are the two cells which touch it from below.*

In theory, our cellular automaton could use any of the $8^{8\cdot 8} \approx 6\cdot 10^{57}$ rules which map the states of the two neighbor cells to the state of the new cell. However, the Stranded Cellular Automata system we use restricts this rule set for aesthetic and practical reasons. First, we specify that if the left neighbor has a strand ending on the right, the new state will have a strand starting on the left, and similarly for the right neighbor. Figure 3 shows examples of this. In order to preserve the continuity of the strands, we will not consider changing these conditions here. This still leaves $2^{3\cdot 5}2^{5\cdot 3}3^{5\cdot 5} \approx 9\cdot 10^{20}$ possible rules, which is a very large rule set to explore. In order to provide a practical starting point for exploration, we have chosen to break our system into two simpler cellular automata controlling different aspects of the state.

The first cellular automata controls whether strands are upright or slanted, based on whether the strands in the neighbor cells are upright, slanted, or absent. This can be thought of as a function which takes nine

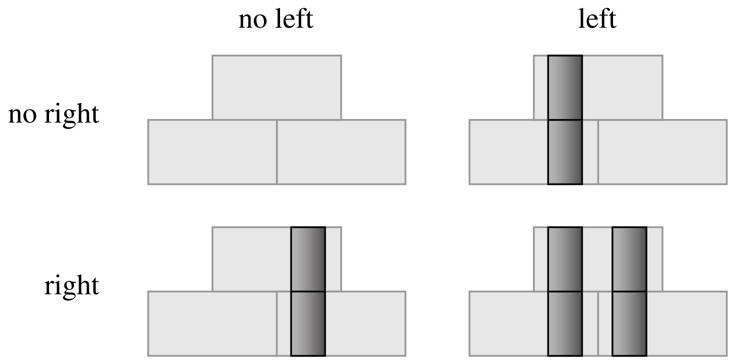

Figure 3 : *The conditions controlling whether strands are present or not.*

possibilities as input (three for each neighbor) and for each possible input chooses one of two possibilities as output (upright or slanted). This gives $2^9 = 512$ possible functions. We code these functions using binary numbers as a "Turning Rule", similarly to the coding for Elementary Cellular Automata in [9]. Figure 4 shows Turning Rule 39, which in binary is 000100111. (Note that in fact bit 4 always controls an empty cell and thus its value does not affect the final output.)

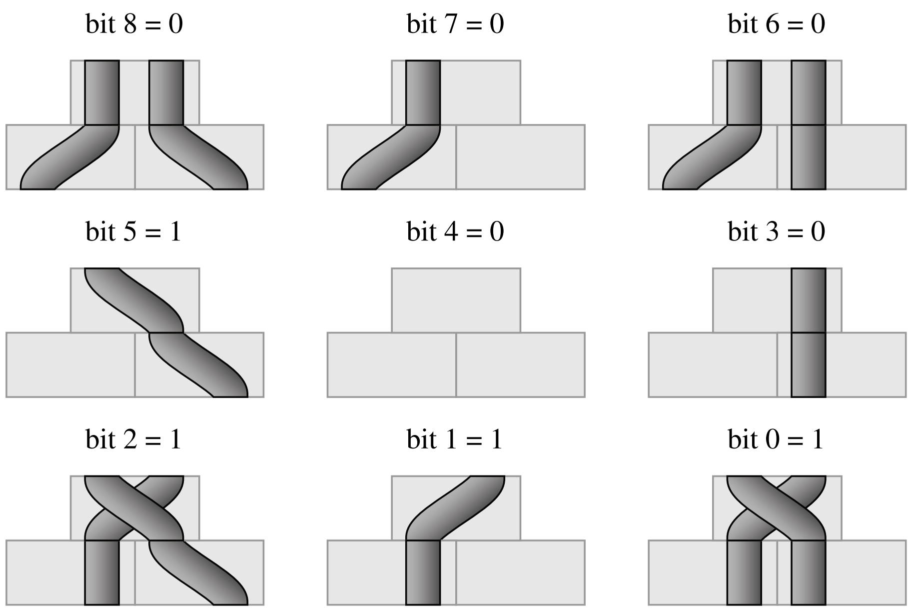

Figure 4 : *Turning Rule 39.*

The second cellular automata controls which strand is on top if the strands cross. Each neighbor again is considered to have three possibilities: the strand going towards the new cell is on top, the strand going towards the new cell is on the bottom, or strands do not cross. (In the last case either there is only one strand or the two strands are upright.) Again, there are two possible outputs and $2^9 = 512$ rules, coded in binary as "Crossing Rules". Figure 5 shows Crossing Rule 39.

The remaining thing to consider is what happens at the edge of the grid. We could make the grid infinite, as in [9]. This seems somewhat unsatisfying for artistic purposes, especially fiber arts. We could have a special kind of state for edge cells that does not allow strands to propagate through the edge. (This

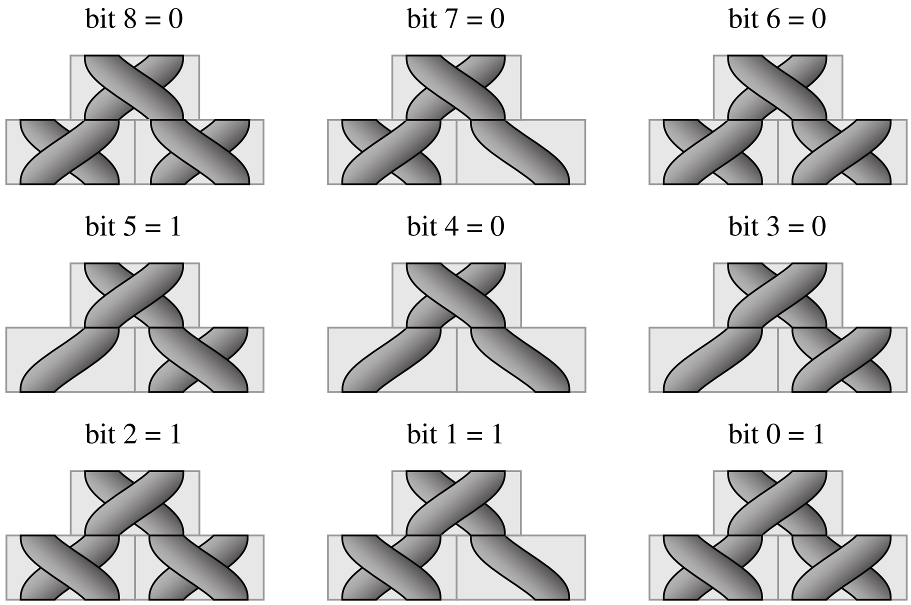

Figure 5 : *Crossing Rule 39.*

could be thought of like a selvage on a piece of fabric.) Or we could achieve a similar effect by modifying the topology of the grid so that cells are reflected upon themselves at the edges. In that case, we would need to choose where to put the axis of reflection. Finally, we could make the grid cylindrical so that the left and right edges are identified. (In the cellular automata literature, this is known as "periodic boundary conditions". See, for example, [4].) So far we have only explored the cylindrical case. For physical representations, one could "cut" the cylinder in a convenient place and unroll it, or use it in a tubular context such as a sock or the sleeve of a garment.

Examples from Fiber Arts

Cellular automata have been used in fiber arts in the past. Many artists have represented Elementary Cellular Automata using knitting and crochet, including Debbie New [6, Section 7], Jake Wildstrom [8], and Nora Gaughan [2, Chapter 5]. In fact New has incorporated cables into cellular automata designs, but in her case the cellular automata only control the existence of crossings between parallel strands, not the directions of crossings or the turning of the cables.

There are at least two ways to simulate Elementary Cellular Automata using Stranded Cellular Automata. One is to let all cells have two strands and all of the crossings be the same. The turning rule then takes one of two possible states for each neighbor and outputs one of two possible new states, just like an Elementary Cellular Automata. Using Turning Rule 68 and Crossing Rule 0 produces the same Sierpinski triangle as Elementary Cellular Automaton Rule 90, as shown in Figure 6.

Another way is to let all of the cells be filled with crossings. Then the crossing rule takes one of two possible types of crossings for each neighbor and outputs one of two possible new crossings. Using Turning Rule 0 and various crossing rules produces the weaving patterns shown in Figure 7.

Using both turning and crossing rules lets us reproduce the traditional braid patterns shown in Figure 8. The first two pictures show traditional three- and four-strand flat braids and the third shows a four-strand round braid.

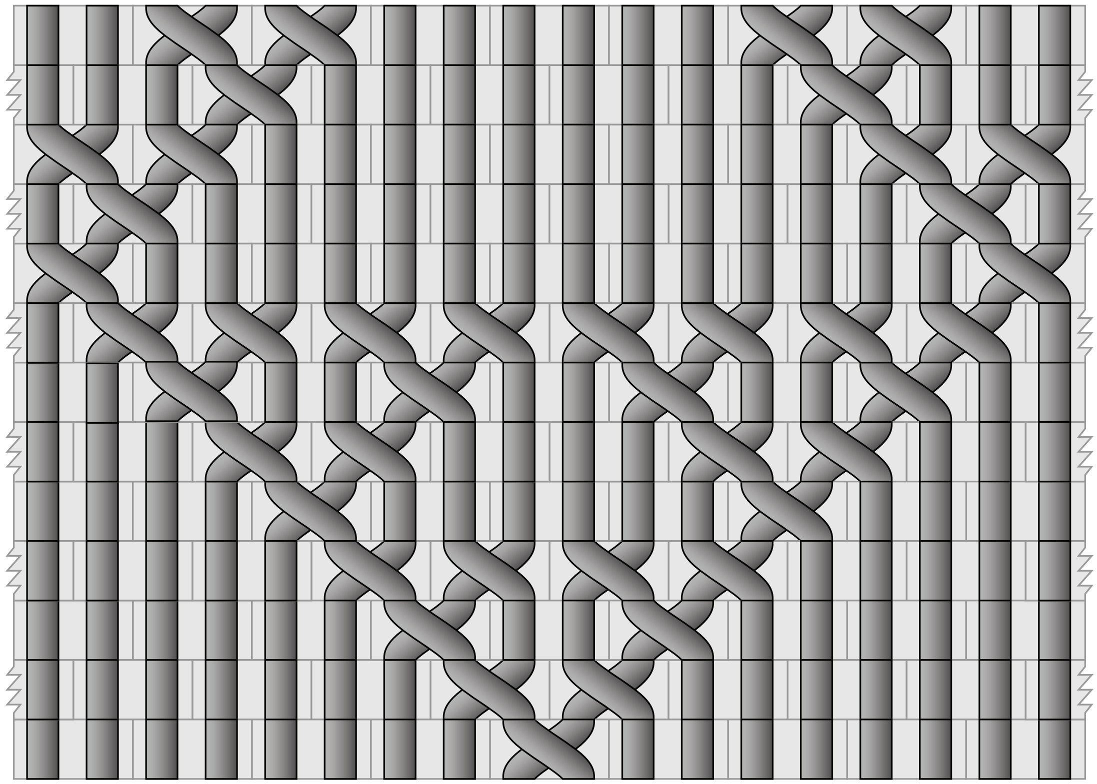

Figure 6 : *Turning Rule 68 and Crossing Rule 0.*

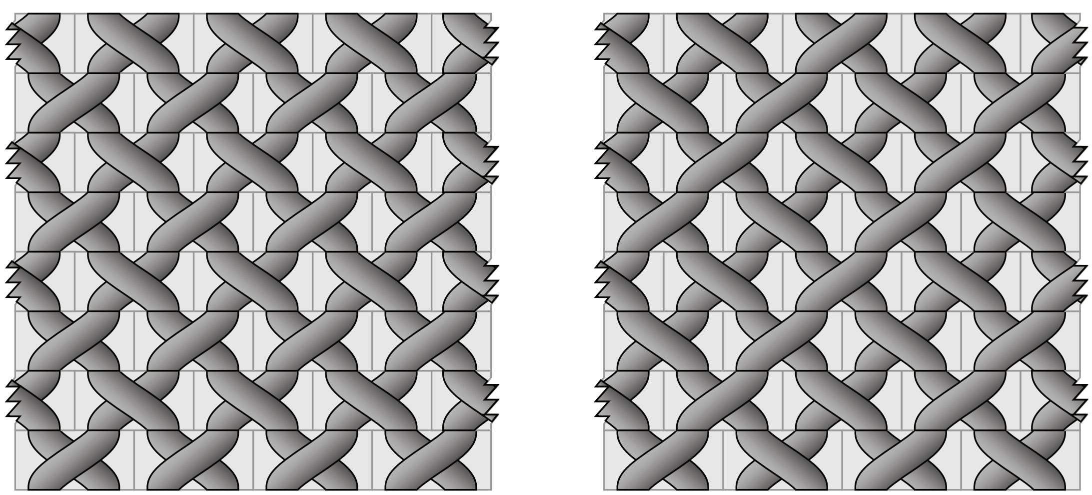

Figure 7 : *Left: Rules 0 and 47. Right: Rules 0 and 448.*

Using both turning and crossing rules also lets us generate patterns reminiscent of traditional cable designs from knitting and crochet. An exploration of the space of rules and starting states reveals many many new patterns, some of which are quite aesthetically pleasing. Two of these are shown in Figures 9 and 10. (The first of these also appeared as a knitting pattern in [3].)

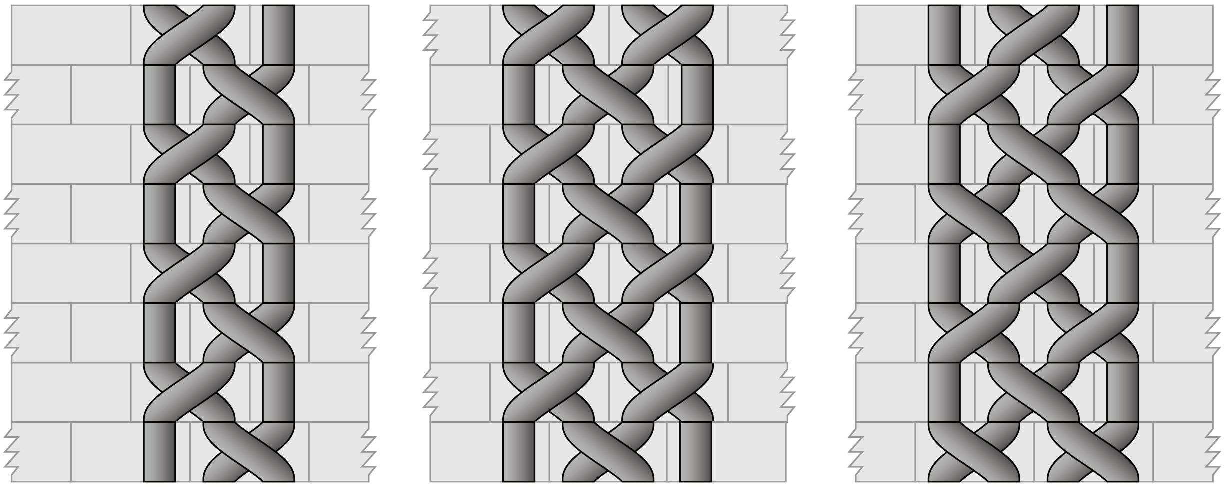

Figure 8 : *Left and Center: Rules 333 and 39 with different starting states. Right: Rules 333 and 99.*

Future Work

Much work has been done in the classification of patterns in the Game of Life and in Elementary Cellular Automata. In particular we could classify patterns which go to the same "ending state" under a given rule or category of rules. In the case of a finite width grid any pattern will eventually become periodic, so an important question is how long these periods can be. Some work on this has been done for the cellular automata described here, which will appear in a forthcoming publication. There are still many open cases. Cellular automata rules can also be classified by whether they are reversible, in the sense that each possible state of a row is generated by exactly one state of the previous row. This work has not yet been started for Stranded Cellular Automata.

Another question would be to classify which braids, in the sense of mathematical braid theory (see, e.g. [5]) can be represented by this model. Classifications based on traditional fiber arts patterns (knitting, crochet, braiding, weaving, macramé, etc.) would also be interesting.

As mentioned above, so far we have only investigated the case of cylindrical topology. It would be interesting to investigate topologies that incorporate reflection. It would also be interesting to look at grids where the topology changes with time; perhaps varying in size or in reflection vs. periodicity. Finally, one could look at a two-dimensional grid, as in the Game of Life, and use time as the third dimension instead of the second. This would result in a more sculptural structure, which could be rendered in traditional sculptural materials or in a strongly three-dimensional fiber art such as macramé.

Acknowledgments The authors would like to thank the anonymous reviewers for a number of helpful comments.

References

[1] Martin Gardner, *Mathematical Games: The Fantastic Combinations of John Conway's New Solitaire Game "Life"*, Scientific American **223** (October 1970), 120–123.

[2] Norah Gaughan, *Knitting Nature: 39 Designs Inspired by Patterns in Nature*, Stewart, Tabori and Chang, New York; London, 2012.

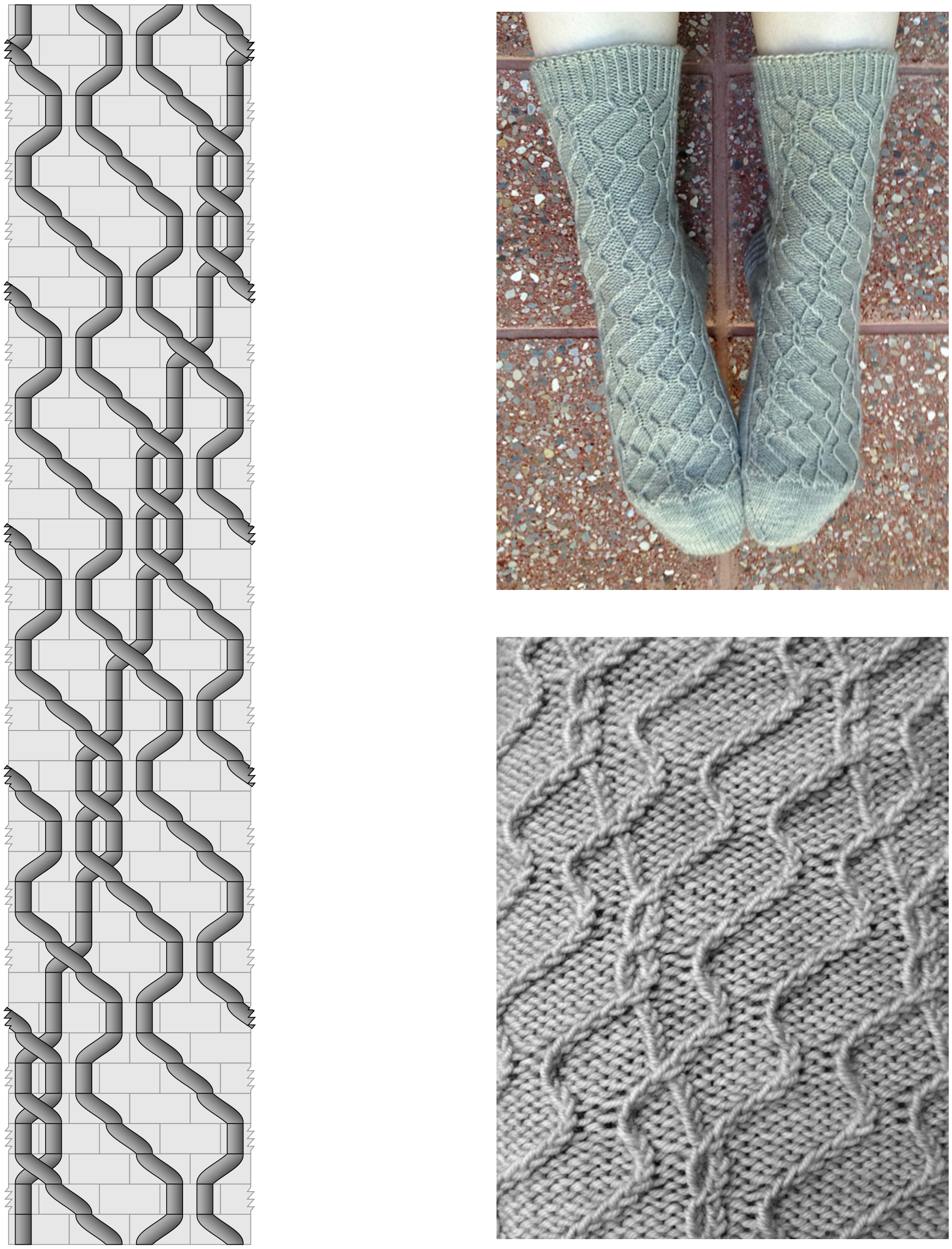

Figure 9 : *Left: Rules 47 and 0. Right: The pattern depicted as cables on a sock, overview and close-up.*

[3] Lana Holden, *Knit Stranded Cellular Automata*, Sockupied (Spring 2014), 10–11.

[4] Erica Jen, *Cylindrical Cellular Automata*, Communications in Mathematical Physics **118** (December 1988), no. 4, 569–590.

[5] Wilhelm Magnus, *Braid Groups: A Survey*, Proceedings of the Second International Conference on The Theory of Groups, Springer, Berlin; Heidelberg, 1974, pp. 463–487.

[6] Debbie New, *Unexpected Knitting*, Schoolhouse Press, Stevens Point, WI, 2003.

Figure 10 : *Left: Rules 201 and 39. Right: The pattern depicted as cables on a cowl, overview and close-up.*

[7] Tommaso Toffoli and Norman Margolus, *Cellular Automata Machines: A New Environment for Modeling*, The MIT Press, Cambridge, MA, 1987.

[8] D. Jacob Wildstrom, *The Sierpinski Variations: Self-Similar Crochet*. Chapter 3, Making Mathematics with Needlework: Ten Papers and Ten Projects, A K Peters/CRC Press, Wellesley, MA, 2007, pp. 40–52.

[9] Stephen Wolfram, *A New Kind of Science*, Wolfram Media, Champaign, IL, 2002.