Continuous Transitions of Cellular Automata

John Pedersen

Department of Mathematics, University of South Florida,
Tampa, FL 33620, USA

Abstract. A method of continuously deforming one cellular automaton rule into another is presented. Transitions between behaviors under this model are found to share some characteristics of the dynamics of iterated self-maps.

Introduction

As usual, a one-dimensional (discrete) k-state R-neighborhood cellular automaton (CA) is a doubly infinite linear array of cells, each of which has one of k possible values (states), evolving in discrete time steps according to a (local) transition function. If the possible states are $[k] = \{0, 1, \ldots, k - 1\}$, then the transition function is any function $\tau : [k]^R \to [k]$ such that $\tau(0) = 0$ (some authors do not even make the "nothing from nothing" restriction). This transition function is applied simultaneously at all cells, thus inducing a translation invariant global transition function $g_\tau : [k]^{\mathbf{Z}} \to [k]^{\mathbf{Z}}$ (where $\mathbf{Z}$ denotes the integers).

To be precise, we must specify to which R cells surrounding a given cell the local transition function is to apply. Usually $R = 2r + 1$, and we take a symmetric neighborhood of r cells to the left and right of a given cell, although Smith [4] has shown that $R = 2$ is enough to simulate any larger R (if k is allowed to be large enough).

Even for the simple case $k = 2$, $R = 3$, interesting evolution diagrams are observed for some of the 128 possible rules, starting from an initial configuration of a single live (state 1) cell (see figure 2 and [8, App. 5]). Furthermore, the evolutionary behavior of a given initial configuration under rules that differ only slightly from each other can be vastly different. This paper was motivated by an attempt to understand better this last phenomenon.

There are some qualitative similarities in the evolutionary behavior of cellular automata and the dynamics of families of iterated self-maps of an interval $f_\alpha : I \to I$, where α is a real parameter and the f_α are continuous functions varying continuously with α, such as the well known $f_\alpha(x) = \alpha x(1 - x)$. For example, for some α, most initial values give stable or periodic orbit behavior, while for other α a variety of periodic and chaotic behaviors result

depending on initial values. Many cellular automata rules give stable or periodic behavior for most initial configurations, while other rules give more complex behaviors, dependent on initial configurations.

For self-maps the transitions between behaviors as α varies can be quantified, understood, and hence predicted (e.g., the Feigenbaum constant). For cellular automata, the transition from one rule to another is discrete, there is no definitive linear ordering of rules, and predicting behavior of a given rule is not always easy (but see [3, 1, 6]).

Continuously varying rules

As previous authors have observed, $[k]^{\mathbf{Z}}$ can be endowed with the product topology arising from the discrete topology on $[k]$. It is then compact and g_τ is continuous, so $([k]^{\mathbf{Z}}, g_\tau)$ is a dynamical system. However, the "continuity" here is more a topological artifact than a fact to the human eye, and this distinction persists if we are interested in varying τ "continuously."

Here we introduce a method for changing the parameter τ continuously, by allowing states to be real numbers in the interval $[0, k-1]$. The topology is the usual one on the real line, so continuity really appears so to the eye and we obtain some parallels to standard dynamical systems behavior.

The basic technique is to interpolate a continuous function

$$f : [0, k^R - 1] \to [0, k - 1]$$

to the discrete function $\tau : [k]^R \to [k]$ (recall $[k]$ denotes a discrete set) in the sense that for some bijection $\beta : [k]^R \to [k^R - 1]$ (with $\beta(0) = 0$), we have

$$f(\beta(x)) = \tau(x)$$

for all x in $[k]^R$. β itself can be interpolated by a real-valued function

$$\beta^* : [0, k-1]^R \to [0, k^R - 1]$$

Then the CA evolves using $f \circ \beta^*$ as the transition mapping. We could, instead, use directly some interpolating function $f^* : [0, k-1]^R \to [0, k-1]$ to τ, but multidimensional interpolants make for complicated formulas and awkward graphics.

The technique is now illustrated in the two most studied cases, $k = 2$, $R = 3$ and "totalistic" $k = 2$, $R = 5$. In the former case, a (discrete) transition rule is given by its action on the eight triples $000, 001, \ldots, 111$. Such a rule can therefore be denoted by an eight-element $0, 1$-vector. For example, 00110011 means $000 \to 0$, $001 \to 0$, $\ldots$, $111 \to 1$. (Other authors use the opposite notation, 11001100, but our direction works better here.) Using $\beta(x, y, z) = 4x + 2y + z$, we interpolate a continuous function $f :$ $[0, 7] \to [0, 1]$ to the discrete function $f(0) = 0$, $f(1) = 0$, $f(2) = 1$, $\ldots$, $f(7) = 1$, as illustrated in figure 1(a). We can then calculate the value of cell i in the CA at the next time step as $f(4c_{i-1} + 2c_i + c_{i+1})$, where c_j is the current value of cell j, and this will be consistent with the discrete case. For
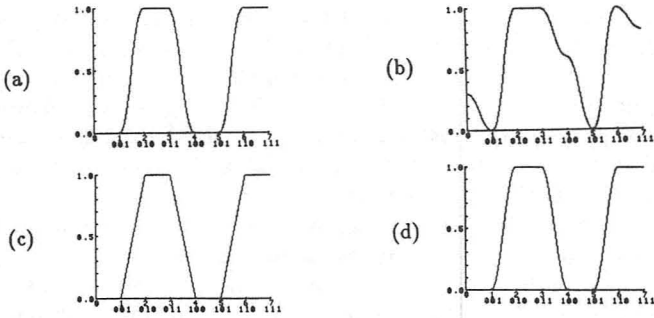
Figure 1: Interpolating continuous functions to discrete $k = 2$, $R = 3$ rules. (a) Quadratic spline for the rule 00110011. (b) Quadratic spline for the rule 0.3 0 1 1 0.6 0 1 0.82. (c) Linear spline for the rule 00110011. (d) Trigonometric spline for 00110011.

example, if a cell's current value is 0.8 and its left and right neighbors have values 0.7 and 0.2 respectively, then its value in the next generation will be $f(2.8 + 1.6 + 0.2) = f(4.6)$. We considered three interpolating functions: the linear, quadratic, and trigonometric splines

$$f_{\text{lin}}(x) = (f(n+1) - f(n))(x - n) + f(n)$$

$$f_{\text{quad}}(x) = \begin{cases} 2(f(n+1) - f(n))(x - n)^2 + f(n), & n \le x \le n + \frac{1}{2} \\ 2(f(n) - f(n+1))(x - n - 1)^2 \\ \quad + f(n+1), & n + \frac{1}{2} \le x \le n + 1 \end{cases}$$

and

$$f_{\text{trig}}(x) = \begin{cases} \frac{1}{2}(f(n) + f(n+1) + (f(n) - f(n+1))\cos(\pi x)), & \text{for } n \text{ even} \\ \frac{1}{2}(f(n) + f(n+1) - (f(n) - f(n+1))\cos(\pi x)), & \text{for } n \text{ odd} \end{cases}$$

where $0 \le x \le 7$, $n = [x]$, the greatest integer in x, and $f(0), \ldots, f(7)$ are the given rule values. These are illustrated in figure 1. All three interpolants give qualitatively similar results (see below; cf. uniformity of behavior for unimodal self-maps). In the examples below we use f_{quad} unless otherwise specified.

The point of introducing a continuous function for f is that we can now consider automata rules such as $f(0) = 0.3$, $f(1) = 0$, $f(2) = 1$, ..., $f(4) = 0.6$, ..., $f(7) = 0.82$, where some of the rule values are not 0 or 1. We just interpolate the rule values, as in figure 1b, and use the same $f(4c_{i-1} + 2c_i + c_{i+1})$ method as before to obtain successive generations of cell values. For example, with the rule in figure 1b, an initial configuration of one cell with value 1.0 and all other cells having value 0.0, the three nonzero cells in the next generation will have values $f(4 \cdot 0 + 2 \cdot 0 + 1) = f(1) = 0$, $f(4 \cdot 0 + 2 \cdot 1 + 0) = f(2) = 1$, and $f(4 \cdot 1 + 2 \cdot 0 + 0) = f(4) = 0.6$. In the following generation the cell with current value 0.6 will obtain the value $f(4 \cdot 1 + 2 \cdot 0.6 + 0) = f(5.2)$.

A transition rule is "totalistic" if the value $\tau(a_1, \ldots, a_R)$ depends only on the sum $a_1 + \cdots + a_R$. Thus there are 32 legal totalistic $k = 2$, $R = 5$ rules. These are of interest because some exhibit "class 4" (locally organized) behavior, not present in any of the $k = 2$, $R = 3$ rules and believed to be capable of supporting universal computation [7]. In the totalistic case, τ can be viewed as a function from $[k(R-1)]$ to $[k]$, and we naturally interpolate a continuous $f : [0, R(k-1)] \to [0, k-1]$.

In the $k = 2$, $R = 5$ totalistic case, the rules can again be notated by 0,1-vectors, this time of length 6, giving the next states for the possible sums $0, 1, 2, \ldots, 5$ of the five surrounding cells. Thus, the totalistic rule 000110 says that the state of a cell in the next generation should be 0 unless three or four of the five cells about it (including itself) are currently in state 1. To obtain a continuous automaton we again take a (continuous) interpolating function $f : [0, 5] \to [0, 1]$. We always take

$$\beta(x_1, \ldots, x_R) = x_1 + \ldots + x_R$$

in the totalistic case.

Global behavior

Fractional rule values allow us to progress continuously from one discrete rule to another. This is easiest if we consider two discrete rules that differ in just one position, for example the $k = 2$, $R = 3$ rules 00110011 and 00111011. We can vary the fifth component from 0 to 1, holding the initial configuration constant, and observe the transition in behavior between these rules (see figure 2). In this way we can see a progression revealing how one discrete behavior changes to a radically different one (a "homotopy of behaviors," if you like).

To display CA evolution when cells have real number values (say, in the interval $[0, 1]$), we can use a spectrum of colors, or textures in monochrome, to represent cell values. In practice only finitely many gradations are available, and for monochrome display, at least, it heightens features if only a quite limited number of different outputs are used. The diagrams below use 11 textures, representing decimal values rounded to the nearest 0.1. Figure 2 illustrates some typical transitions between rules. All figures were produced from the author's C language programs running on a Sun 3/60. The programs are available from the author.

The ultimate objective is to have a single parameter that as it varies covers the whole range of k^{k^R} rules, so that we might hope to identify parameter intervals particular to specific behaviors (cf., predictability of periodic or chaotic behavior for self-maps of an interval). Putting together 0-1 changes in single components of rules over some linear ordering of rules can be viewed as achieving such a parameterization, although it may be judged artificial.

For example, the 128 legal $k = 2$, $R = 3$ rules can be arranged in an order such that only one place changes at a time (Gray code). We first consider

the entire spectrum of 128 rules arranged in this way. The initial configuration considered for these rules was always a single live cell. Figure 3 shows "randomness" and periodicity indexes for this spectrum. The randomness index represents the percentage of predictable sites per generation after a large number of generations. This can be construed as a Kolmogorov complexity measure, that is, a measure of the shortest (Turing machine) program capable of computing later generations. Its computation is necessarily somewhat subjective and was carried out by human. The periodicity index is $1/k$ if (the predictable part of) the evolution diagram has apparent "period" k, where "period" is interpreted loosely enough to allow for increasing length of generations. For example, the second diagram in figure 2f [0110100 0.2] is 0.75 predictable and has period 2. Steps of 0.1 between rules were used. Figure 3 displays no overall pattern that would allow us to predict discrete CA behaviors. The question remains whether there exists some other ordering of rules from which such a pattern can be observed.

The behavior of transitions can be divided roughly into three types: vertical, horizontal, and mixed. The vertical case is typified by some regularity on one or both edges of an evolution diagram becoming more pronounced as the parameter approaches 0 or 1, eventually closing out an irregular middle or other-side portion. Transition from partially irregular to regular shape can appear very sudden, resulting from changes of 10^{-10} or less in the parameter value (see figure 4a). This is perhaps reminiscent of crystal formation — past a certain critical point, structural regularity forces itself upon the whole configuration. In reverse, the sudden onset of irregularity could be compared with the onset of turbulence.

In horizontal transitions, intermediate global periodic behavior is observed with varying periods (see figure 4b). In some cases the transition rule is effectively a function $f : [0,\ k\ -\ 1] \rightarrow [0,\ k\ -\ 1]$ (instead of from $[0,\ k\ -\ 1]^R$), because some positions do not affect the rule behavior (e.g., 00100000). In these cases we have a univariate self-map of an interval and the usual period doubling into chaos is to be expected. However, changing periodic behavior also appears in more complex cases that are not just reincarnations of one-dimensional dynamical systems (e.g., from 00111110 to 00111111). This is examined in the next section.

Figure 2 contains examples of transitions between each class of behavior that occurs in the Gray code ordering for $k = 2$, $R = 3$ rules: stable (periodic) to stable, stable to fractal, fractal to fractal, fractal to "random" class 3. There are several examples of each type of transition over the whole range of rules. However, no clear categorization of transition behaviors based on type difference was observed. The totalistic $k = 2$, $R = 5$ case is of interest for displaying class 4 (locally organized) behavior (e.g., the rule 001011). Because totalistic rules give symmetric evolution from symmetric configurations, a random initial configuration (same for all) was used. Class 4 behavior appears to persist over relatively large intervals in transitions involving a discrete class 4 behavior rule. Also, behavior approaching class 4 sometimes arises in transition from class 1 to class 3 discrete rules, for example in
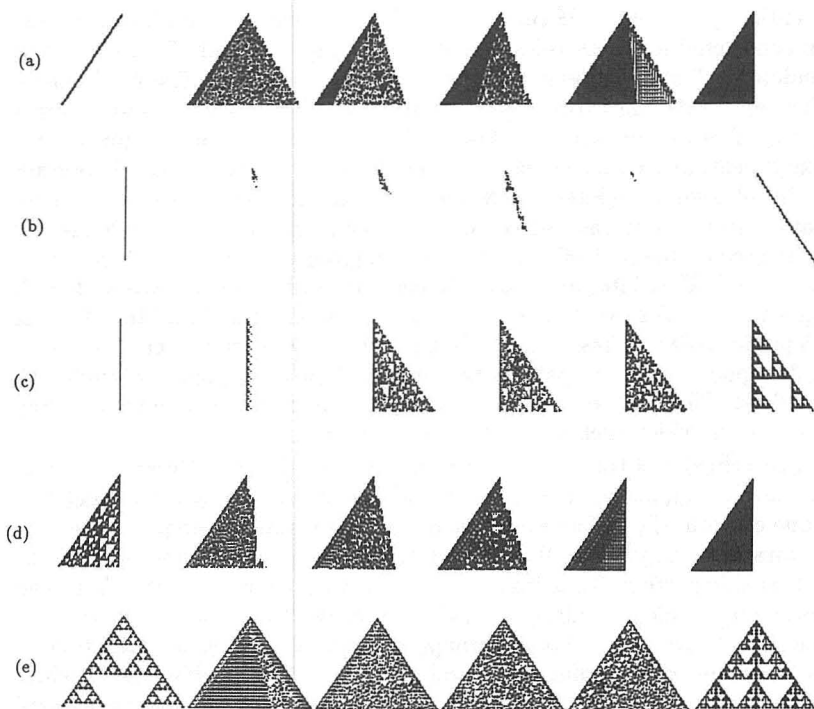
Figure 2: Transitions between discrete rules. Left and right ends are discrete rules; interpolated rules lie inbetween. Fifty generations of evolution are displayed for each rule. Triangular diagrams are $k = 2$, $R = 3$ rules, with an initial configuration of one live cell; square diagrams are totalistic $k = 2$, $R = 5$ rules with a random initial configuration (only the central 50 cells of 250 are displayed). The values of cells are real numbers, displayed using textures as explained in the text. The rules are (a) $011101x1$, (b) $0010x100$, (c) $0011x100$, (d) $0111011x$, (e) $0110100x$, (f) $0110101x$, (g) $011x1000$, (h) $00011x$, (i) $00x100$, (j) $00101x$, (k) $01x100$, with x varying from 0 to 1 (left to right) in steps of 0.2.

$00x100$ for values of x close to 0.62. Another complex intermediate behavior is observed around $x = 0.65$ in $0111x1$.

Local behavior

Some understanding of the transition process can be derived by considering a fixed cell in a given generation. As the CA transition rule changes, how does the value of the cell in that generation vary? Figure 5 shows that the variations may be quite complex even after a relatively small number of generations. Such complexity may be surprising at first, but reflection
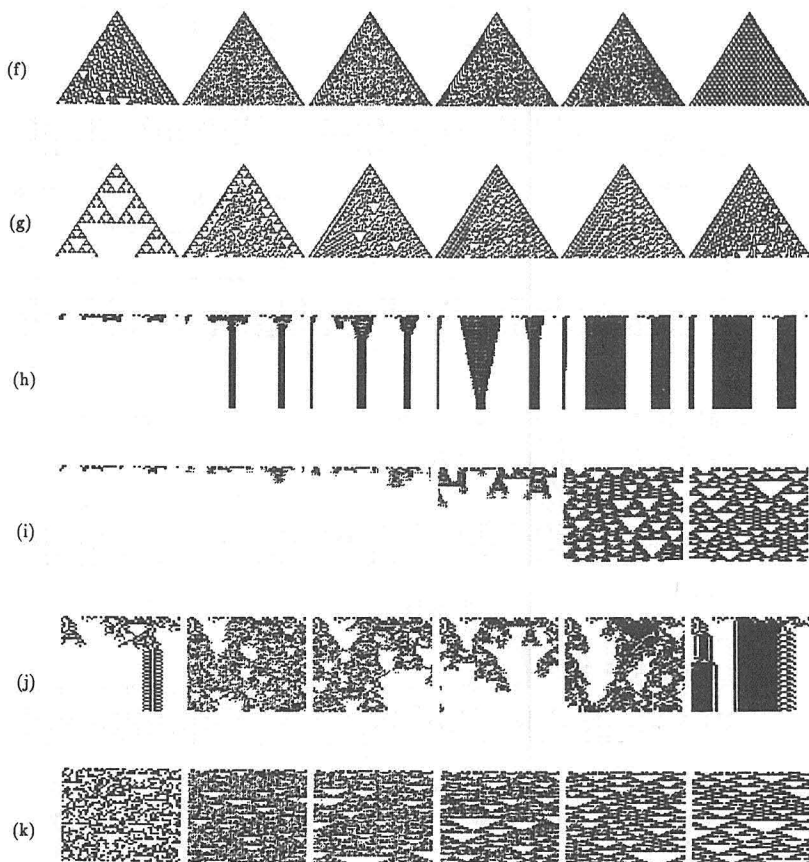
Figure 2: Cont'd.

shows that iteration of a multimodal function (multiple local extrema on the interior of an interval) such as our interpolating function can give a function with a number of extrema that is exponential in the number of iterations. We must also bear in mind that we are not doing straightforward iteration, but using the β conversion function and the automaton rule as well. Since we no longer have a univariate map, we cannot necessarily expect an orderly progression of periodic and chaotic behaviors à la Šarkovskii. Indeed, the order and distribution of periods is much more complex than Šarkovskii's sequence (see figure 4b). The extremely complex behavior of single cells even relatively early in the evolutionary process shows us that any "smooth" transitions between global behaviors that may appear to occur (such as in figure 2c) should be carefully scrutinized. A sampling interval such as 0.1 is large compared to the scale at which local behavior can change (which is apparently of the order of 2^{-n} after n generations). This helps us to
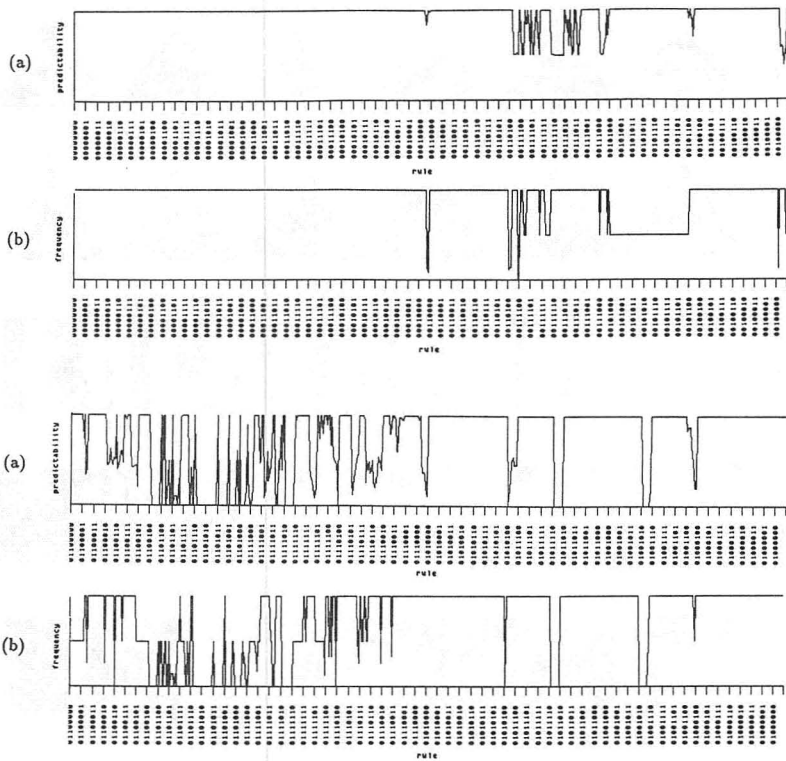
Figure 3: Randomness and periodicity of evolution in $k = 2$, $R = 3$ rules. (a) Proportion of sites with a predictable value. (b) Frequency (1/period) of predictable portion. A zero frequency means no periodicity. A predictable, zero-period pattern is probably a fractal.

understand the suddenness of some global behavior changes, such as the "crystalline" transition in figure 4a.

This view of single cell behavior is perhaps the most revealing observation in our investigation. The great complexity arising from iterated functional composition could be expected to occur in any attempt at continuous progression between discrete rules, since some non-monotone function must be used to move between them. Thus, the discrete behaviors are just points in a potentially extremely erratic spectrum, at least at the local level. In fact, with this view of individual cells having the potential for so many changes in behavior as we progress from one discrete rule to another, the real mystery becomes why we see any "smooth" transitions at all. The tendency for adjacent cells to conspire to synchronous behavior is partially responsible for what periodic behavior we do observe, but a full explanation of this local–global interaction does not appear to be at hand.
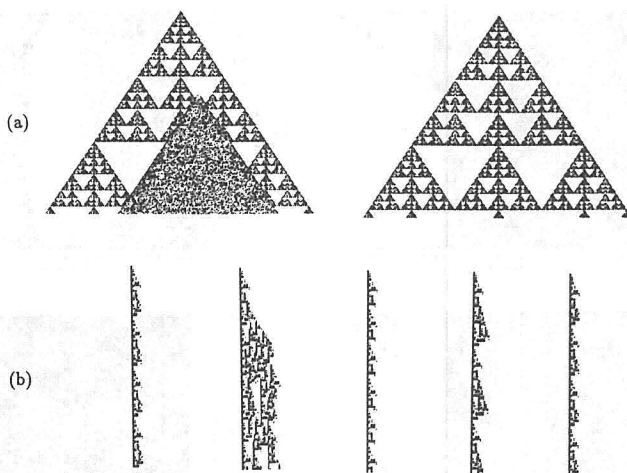
Figure 4: Sensitivity to rule changes. (a) Sudden disappearance of irregularity in $k = 2$, $R = 3$ rule 0110100x with x = 0.9712651660 (left) and x = 0.9712651661 (right). (b) Variety of periodic behaviors in 0011x100. Diagrams from left to right are x = 0.13520735 (period 25), 0.13520736 (infinite period), 0.13520737 (period 17), 0.13520738 (period 36), 0.135207385 (period 25 again).
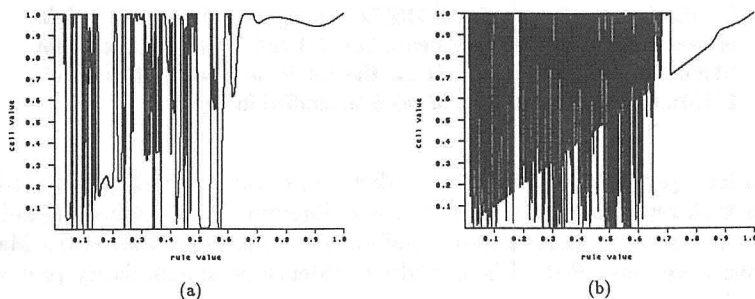

Figure 5: Reaction of a single cell in a given generation to parameter variation. (a) Cell 1 left of center, generation 9, changing from rule 01110110 to 01110111; (b) same cell, generation 20.

Another view of local transition behavior is given by a "phase portrait" diagram displaying the values of a single cell over many generations as the automaton rule changes. The analogous diagram for one-dimensional dynamical systems shows the characteristic bifurcations, and gaps showing regions of low periodicity sandwiched between chaotic regions. Diagrams for selected cells in some of the transitions of figure 2 are given in figure 6. They are seen to share qualitative characteristics with one-dimensional dynamical systems (bifurcations, gaps), but there are noticeable differences. There is apparently
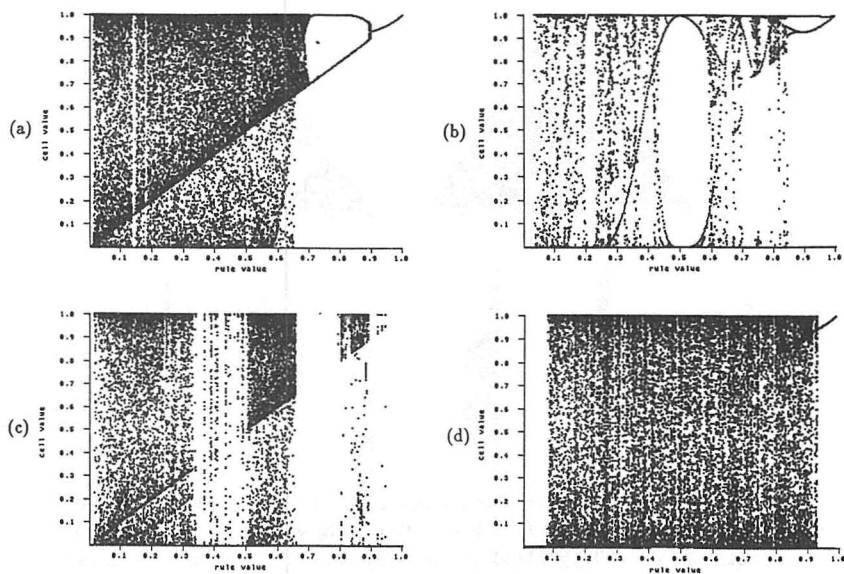
Figure 6: Evolutionary behavior of a cell as a function of automaton rule. For each rule, the values of a cell are plotted for generations 50 to 150. Regions of periodic behavior show up clearly. The diagrams show (a) cell 1 left of center (initial configuration of one live cell) for the $k = 2$, $R = 3$ rules 0111011x (see figure 2d); (b) cell 2 right of center for 0011x100 (see figure 2c); (c) cell 1 right of center for 0110011x; and (d) center cell for the totalistic $k = 2$, $R = 5$ rule 00101x. The indicated area of (a) is magnified in figure 8.

no orderly progression of periods — they can come in seemingly any order, often with repetitions. As a result, these diagrams have not been found to have the fractal properties that standard bifurcation diagrams have. Many regions were magnified without finding evidence of self-similarity (see also figure 8).

Convergence

With the complex behaviors demonstrated, some apparently sensitive to initial conditions, and the extensive real-valued computations required for even fifty generations of evolution, there must be considerable concern for numerical accuracy. Perhaps some of the pictures do not really look anything like the computed pictures at some parameter values? In fact this is almost surely true, for the reasons cited above. Nevertheless, we can believe that parameter values very close to the given ones give pictures very similar to those computed. This has been proved rigorously in the dynamical systems
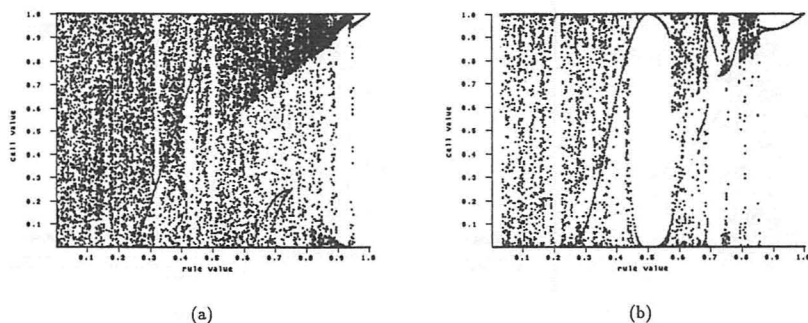
(a) (b)

Figure 7: Effect of different interpolating functions. (a) Phase protrait for cell 2 right of center in the $k = 2$, $R = 3$ rule 0011x100 under linear interpolation. (b) Same for trigonometric interpolation. Compare with figure 6b.

case [2], and the similar conclusion in our case is supported by the visual evidence.

As concerns dependence on the interpolating function, figure 7 shows the behavior of the same cells under the linear and trigonometric interpolating functions given above, compared with the quadratic interpolant used so far. We see that the trigonometric interpolant gives nearly identical behavior of the cell, and the linear one gives qualitatively similar behavior. This holds for all cells in all generations. Concerns for numerical accuracy are somewhat overshadowed by the arbitrariness of the interpolating functions, but since the qualitative behavior of the model remains intact even for nondifferentiable interpolants such as f_{lin}, our concern need not be too great after all. There is one aspect where slow convergence does appear to show, that is, in particular regions of some "bifurcation" diagrams. For example, figure 8 shows how a region of one diagram changes if later generations are added. It is not clear from the current investigations how widespread such behavior is.

Conclusions

This study was initiated to help understand the variety of global CA behaviors, but it can be argued that it has not yielded much in that direction. The difficulty is partly in arranging rules in a linear order to regard them as a continuously varying parameter and partly in the complex behavior inevitably produced by iteration of three-fold meshing of even a simple function. Figure 5 gives a good appreciation of the latter, and perhaps provides some insight into the often choppy nature of transitions between discrete rule behaviors under the paradigm examined here. We do also observe some smooth global behavior transitions on relatively large scales — cases where erratic local behaviors conspire to synchrony in producing, for example, period doubling effects. This effect is somewhat mysterious and bears further examination.

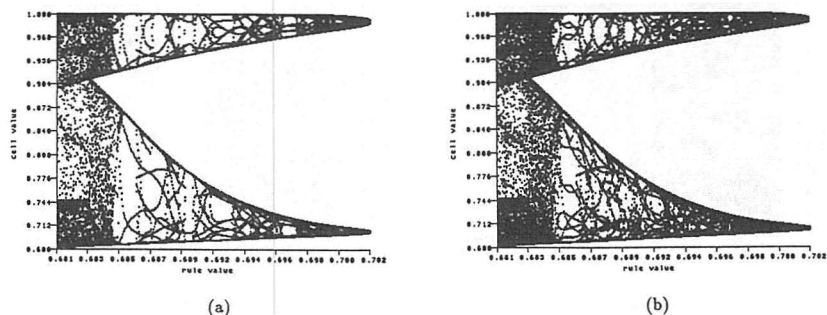(a)   (b)

Figure 8: Lack of convergence in the parameter range 0.682–0.702 for
the $k = 2$, $R = 3$ rule 0111011x. This is a magnification of part of
figure 6a. (a) Generations 100 to 200. (b) Generations 100 to 300.

One way to proceed is to conduct a mathematical investigation of the
abstract formulation, namely a function $f : [0,1]^R \to [0,1]$ and its iterated
behavior on initial configurations. An initial configuration is just a mapping
from $\mathbf{Z}$ to the set of states, i.e., a doubly infinite sequence of states. The
evolution of a cell in a one-dimensional CA can be expressed in terms of shift
and subsequence operators applied to such sequences. Using the reduction of
discrete one-dimensional CA to arithmetic recurrences given in [5] is another
possible approach.

Although we did not accomplish the original objective of bringing order
to the variety of discrete rule behaviors, some of the results of the continu-
ous case may be of interest in their own right. The technique given here of
producing continuous variation between cellular automaton rules clearly can
be applied to any k and R, and also to two- and higher-dimensional cellular
automata. We have not conducted higher-dimensional investigations. How-
ever, as others have observed, the one-dimensional case appears already to
have all the complexity of higher dimensional systems.

Acknowledgments

References

[1] P. Grassberger, "Some more exact enumeration results for 1D cellular au-
    tomata," J. Phys. A., **20** (1987) 4039–4046.

[2] S. Hammel, J. Yorke, and C. Grebogi, "Numerical orbits of chaotic processes
    represent true orbits," Bull. Amer. Math. Soc., **19** (1988) 465–469.

[3] E. Jen, "Global properties of cellular automata," J. Stat. Phys., **43** (1986)
    219–242.

[4] A. R. Smith III, "Simple computation universal cellular spaces," *J. Assoc. Computing Machinery*, **18** (1971) 339–353.

[5] J. Urías, "One-dimensional cellular automata as arithmetic recursions," *Physica D*, **36** (1989) 109–110.

[6] S. Willson, "The equality of fractional dimensions for certain cellular automata," *Physica D*, **24** (1987) 179–189.

[7] S. Wolfram, "Universality and complexity in cellular automata," *Physica D*, **10** (1984) 1–35.

[8] S. Wolfram (ed.), *Theory and Applications of Cellular Automata* (World Scientific, Singapore, 1986).