On Patterns and Dynamics of Rule 22 Cellular Automaton

Genaro J. Martínez
Laboratorio de Ciencias de la Computación
Escuela Superior de Cómputo
Instituto Politécnico Nacional, México
genaro.martinez@uwe.ac.uk

Andrew Adamatzky
Unconventional Computing Lab
Department of Computer Science and Creative Technologies
University of the West of England
Bristol, United Kingdom
andrew.adamatzky@uwe.ac.uk

Rolf Hoffmann
Technische Universität Darmstadt
Darmstadt, Hessen, Deutschland
hoffmann@informatik.tu-darmstadt.de

Dominique Désérable
Institut National des Sciences Appliquées
Rennes, France
domidese@gmail.com

Ivan Zelinka
Fakulta Elektrotechniky a Informatiky
Technická Univerzita Ostrava, Czechia
ivan.zelinka@vsb.cz

Rule 22 elementary cellular automaton (ECA) has a three-cell neighborhood, binary cell state, where a cell takes state "1" if there is exactly one neighbor, including the cell itself, in state 1. In Boolean terms, the cell state transition is an XOR function of three cell states. In physico–chemical terms, the rule might be seen as describing propagation of self-inhibiting quantities/species. Spacetime dynamics of rule 22 demonstrate nontrivial patterns and quasi-chaotic behavior. We characterize the phenomena observed in this rule using mean field theory, attractors, de Bruijn diagrams, subset diagrams, filters, fractals and memory.

Keywords: elementary cellular automata; rule 22; chaos and complex dynamics

https://doi.org/10.25088/ComplexSystems.28.2.125

1. Introduction

1.1 Rule 22: History

Elementary cellular automata (ECAs) [1, 2] are one-dimensional arrays of finite-state machines, or cells, that take states "0" or "1" and update their state depending on their own current state and the state of their two immediate neighbors. Rule 22 ECA has a simple cell–state transition function: a cell takes state 1 if exactly one of its neighbors, including the cell itself, is in state 1; otherwise, the cell takes state 0. When perturbed at a single site, the automaton exhibits something similar to recurrent wavefronts in excitable media, which develop into fractal structures of the Sierpiński gasket [3, 4]. Due to countless generation and annihilation of wavefronts, the dynamics of rule 22 are sometimes characterized as chaotic [2], which reflects its unpredictability rather than any relation to noise or bifurcations. Most results of studies about rule 22 were in algebraic properties or statistical approximations of the automaton dynamics. Thus, Zabolitzky [5] reported results of an extended probabilistic analysis estimating nontrivial behavior on very large arrays perturbed by configurations with low densities of state 1. He discovered critical properties that cannot be reproduced when the automaton is perturbed by a random configuration. McIntosh provided a systematic analysis of small configurations emerging in rule 22 [6]; he proposed similarities with configurations observable in Conway's Game of Life (GoL). A topological analysis linked to chaotic behavior of the rule can be found in [7].

1.2 Rule 22: Definition

A one-dimensional cellular automaton $\mathrm{CA}(k, r)$ is an array of cells x_i where $i \in \mathbb{Z}$. Each cell takes on a value from an alphabet $S = \{0, 1, \ldots, k-1\}$ with k symbols. A chain of cells $\{x_i\}$ of finite length n represents a string or global configuration c on Σ. The set of finite configurations is represented as Σ^n. An evolution is a sequence of configurations $\{c_i\}$ given by the mapping $\Phi : \Sigma^n \to \Sigma^n$, and their global relation is provided by $\Phi(c^t) \to c^{t+1}$, where t is a discrete time and every global state of c is a sequence of cell states. Cells of each configuration c^t are updated to the next configuration c^{t+1} simultaneously by a local transition function $\varphi : S^{2r+1} \to S$ as

$$\varphi(x_{i-r}^t, \ldots, x_i^t, \ldots, x_{i+r}^t) \to x_i^{t+1}$$

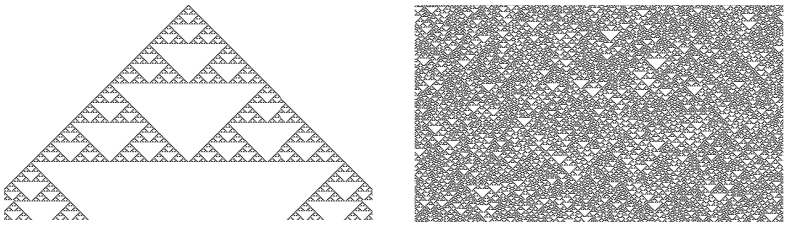
acting on a neighborhood of x_i of length $2r+1$. For (elementary) $\mathrm{ECA}(2, 1)$, $\varphi : S^3 \to S$ becomes

$$\varphi(x_{i-1}^t, x_i^t, x_{i+1}^t) \to x_i^{t+1} \tag{1}$$

and for rule 22, its local cell–state transition is given by:

$$\varphi_{\mathrm{R22}} = \begin{cases} 1 & \text{if} \quad 100, 010, 001 \\ 0 & \text{if} \quad 111, 110, 101, 011, 000. \end{cases} \tag{2}$$

Rule 22 displays a typical chaotic global behavior from random initial conditions. Figure 1(a) shows the evolution with an initial condition starting with a single cell in state 1. A pattern growing is a fractal, similar to a Sierpiński gasket. Figure 1(b) shows a development from a random initial configuration with a density 0.5 of cells in state 1.



(a) (b)

Figure 1. Exemplar dynamics in ECA rule 22. (a) Development from a single cell in state 1. (b) Development from a random configuration with density of 1-cells 0.5. Both spacetime diagrams evolve on a ring of 600 cells for 350 generations. Time evolves from top to bottom.

1.3 On Patterns and Dynamics in Rule 22

Rule 22 is considered as chaotic because:

1. Future configuration of the automaton is completely determined from its initial state because of the deterministic rule and synchronous updating.
2. Development of the automaton is sensitive to initial conditions (tiny perturbation might lead to dramatic events).
3. Global transition graph has dense periodic orbits (attractors).
4. Configurations evolved can be characterized as random.

We undertake an extensive and systematic analysis of rule 22 using different approximations aimed at discovering an emergence of novel nontrivial patterns, periodic patterns and Garden of Eden configurations. These configurations are discovered with the help of encoding initial conditions into regular expressions, de Bruijn diagrams, subset diagrams, cycle diagrams, fractals and jump graphs. We also show an effect of memory upon the dynamics of rule 22.

https://doi.org/10.25088/ComplexSystems.28.2.125

2. Mean Field Theory

Mean field theory allows us to describe statistical properties of cellular automata (CAs) without analyzing evolution spaces of individual rules [8, 9]. This approximation assumes that elements of a set of states Σ are *independent* and not correlated with each other in the rule's evolution space. One can study probabilities of states in the neighborhood in terms of probability of a single state (the state in which the neighborhood evolves), thus a probability of the neighborhood state is a product of the probabilities of each cell state in the neighborhood. A polynomial on the probabilities is derived and its curve can be used to classify the rules, as proposed by McIntosh in [9].

2.1 Mean Field in the Game of Life

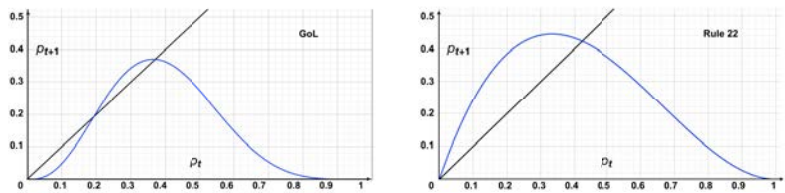
Using this approach, we can construct a mean field polynomial for a two-dimensional CA with a semi-totalistic evolution rule:

$$p_{t+1} = \sum_{v=S_{\min}}^{S_{\max}} \binom{n-1}{v} p_t^{v+1} q_t^{n-v-1} + \sum_{v=B_{\min}}^{B_{\max}} \binom{n-1}{v} p_t^{v} q_t^{n-v} \tag{3}$$

where n represents the number of cells in Moore's neighborhood, v (resp. $n-v$) the number of occurrences of state 1 (resp. 0), p_t (resp. q_t) the probability of a cell being in state 1 (resp. 0) and with $q_t = 1 - p_t$. B and S are the minimum and maximum of an interval for born and survival conditions in Conway's GoL, respectively. The GoL's polynomial is the following:

$$p_{t+1} = 84p_t^3 q_t^6 + 56p_t^4 q_t^5. \tag{4}$$

The mean field curve $\mathcal{F}$ of equation (4) displayed in Figure 2(a) shows three fixed points $p_{t+1} = p_t$ when crossing the identity. The first stable fixed point at the origin guarantees its stable state; the second



(a) (b)

Figure 2. Mean field curves for (a) GoL and (b) rule 22.

unstable point $\mathcal{F} = 0.1986$ relates to areas of densities where the spacetime dynamic is unknown. The last stable point in $\mathcal{F} = 0.37$ indicates that GoL will converge almost surely to configurations with small densities of 1.

2.2 Mean Field in Elementary Cellular Automaton Rule 22

For one dimension, we adjust equation (3) to a full local rule and not only a semi-totalistic one. All states of the neighborhood must be considered, thus p, q, n and v have the same representation as stated previously. But now the product will be with the value of each neighborhood, whence, for a one-dimensional CA(k, r) the mean field polynomial

$$p_{t+1} = \sum_{j=0}^{k^{2r+1}-1} \varphi_j(X) p_t^v q_t^{n-v}$$

which gives

$$p_{t+1} = \sum_{j=0}^{7} \varphi_j(X) p_t^v q_t^{3-v} \tag{5}$$

for ECA$(2, 1)$ and where $\varphi_j(X)$ denotes the j^{th} transition of S^3 in equation (1). Finally, the mean field polynomial for rule 22

$$p_{t+1} = 3p_t q_t^2 = 3p_t(1 - p_t)^2 \tag{6}$$

is deduced from equation (2).

In rule 22, state 1 appears with probability $3 \,/\, 8 = 0.375$ (which is close to the fixed stable point 0.37 of GoL). The mean field curve f of equation (6) displayed in Figure 2(b) shows a slope $f'(0) = 3$ at the origin. Density is maximal at $f(1 \,/\, 3) = 4 \,/\, 9 \approx 0.444$ before reaching the stable fixed point $p_{t+1} = p_t$ when crossing the identity at $p_t = 1 - \sqrt{3} \,/\, 3 \approx 0.423$. It then crosses the inflection point $f(2 \,/\, 3) = 1 \,/\, 2 \cdot f(1 \,/\, 3)$ with tangential slope $f'(2 \,/\, 3) = -1$ and decreases until $f(1) = f'(1) = 0$. Based on the mean field curves classification, rule 22 is a chaotic ECA (Figure 3).

https://doi.org/10.25088/ComplexSystems.28.2.125

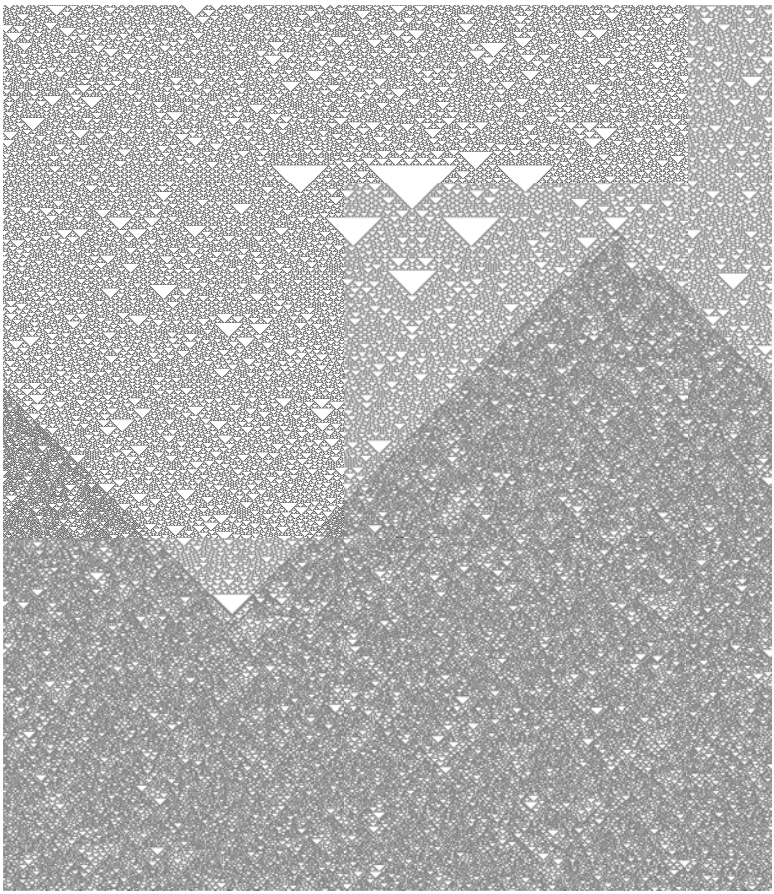


Figure 3. Irreversible phase transition in ECA rule 22 from a specific initial condition (a regular expression), which after 20 000 generations increases significantly the density of 1, related to the fixed points calculated by mean field theory in equation (6). This phase transition merges possible complex dynamics to chaos in ECA rule 22.

2.3 Mean Field Behavior of Rule 22

Various scenarios of evolution are displayed in Figure 4:

- (a_1) From initial density $d_0 = 1/3$ reaching spontaneously the maximum before evolving rapidly toward the fixed point with density $d_F = 0.423$. Transition from d_0 to d_F is not perceptible. (a_2) From initial density $d_0 = d_F$ reaching spontaneously the fixed point.
- (b_1) From initial density $d_0 = 0.8$ evolving toward the fixed point after an early phase transition. (b_2) From initial density $d_0 = 0.95$ evolving toward the fixed point after a later phase transition delimited by a polygonal broken line. (Phase transitions and critical exponents for rule

22 leading to nontrivial long-range effects were reported in [10, 11]. Asymptotic properties were described in [12].) There exists an interval between two thresholds $d'_0 \approx 0.01$ and $d''_0 \approx 0.92$ such that any (pseudo-)random initial distribution with density $d_0 < d_0 < d''_0$ converges almost surely toward fixed point d_F.

- (c_1) From initial density $d_0 = 0.97$ evolving toward a *dense* pattern with observable "backbones" (also observable in the "*Exactly 1*" ECA [13, Figure 1(d)] after crossing the phase transition polygon; two backbones arise from polygon vertices and their patterns are symmetric from either side. (c_2) From the same density $d_0 = 0.97$ and another initial distribution yielding a *sparse* pattern with backbones wherein no phase transition line does appear.

Outside interval $]d'_0, d''_0[$ sensitivity to initial conditions is high, with a positive Lyapunov exponent and chaotic behavior. Depending on small perturbations from initial configuration, four other evolutions are also possible: (*i*) disordered sparse fractals; (*ii*) convergence toward fixed point d_F but after a long period (Figure 5); (*iii*) initial configuration vanishing at first step; and (*iv*) rare events of periodic patterns (Figure 6). That is, six types of evolution altogether. Their estimations of occurrence in interval $\left[d''_0, 1\right]$ are displayed in Table 1.

Density	**ERG**	DSF	**DBB**	SBB	**VAN**	**RPP**
0.920	100	0	0	0	0	0
0.930	94	5	1	0	0	0
0.940	93	1	3	1	1	1
0.950	85	2	7	4	2	0
0.960	47	4	13	24	12	0
0.970	21	2	26	26	25	0
0.980	7	0	18	21	54	0
0.990	1	0	5	8	86	0
0.995	0	0	0	0	100	0

Table 1. Statistical estimations of evolutions (%) in interval $\left[d''_0, 1\right]$ from samples of 100 initial configurations for each density: ergodicity (ERG), disordered sparse fractals (DSF), dense backbones (DBB), sparse backbones (SBB), vanishing (VAN), rare periodic patterns (RPP). Items in bold reflect an irreversible state.

Beyond the phase transition polygon in case of convergence toward fixed point d_F, the evolution becomes ergodic, in the sense that the system has the same behavior either averaged over time or averaged over space [14]. The process is stationary and homogeneous at mesoscopic scale. In other words, there exists a smallest macrocell C of size $\xi \times \xi$, where ξ is the correlation length, as a representative (or statistical)

volume element such that density d_C in the macrocell is close to the mean density averaged within the whole system [15, 16]. Thus $d_C \approx 3/8 = 0.375$, that is, the exact ratio of 1 filling φ_{R22}.

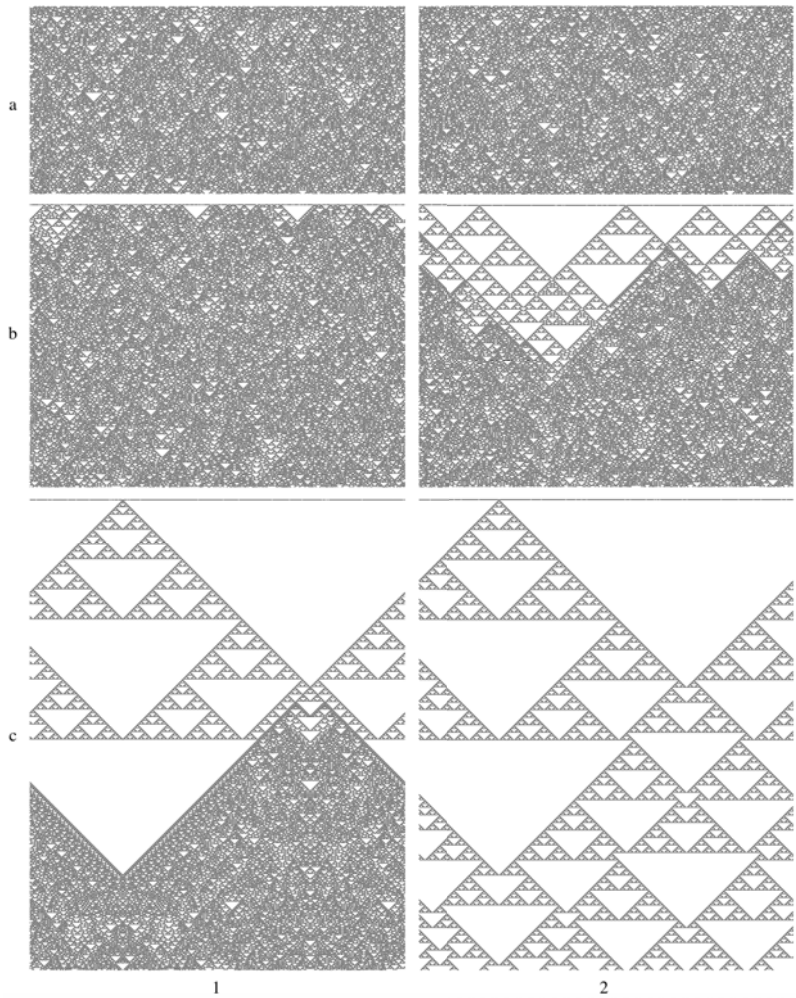


Figure 4. Evolutions in a ring of 800 cells: (a) 400 generations (a_1) from initial density $d_0 = 1/3$ (a_2) from density $d_0 = d_F$ reaching spontaneously the fixed point; (b) 600 generations (b_1) from density $d_0 = 0.8$ evolving toward d_F after an early phase transition (b_2) from density $d_0 = 0.95$ evolving toward d_F after a later phase transition; and (c) 1000 generations from den sity $d_0 = 0.97$ evolving toward (c_1) a dense pattern with thin backbones (c_2) a sparse pattern with large backbones.

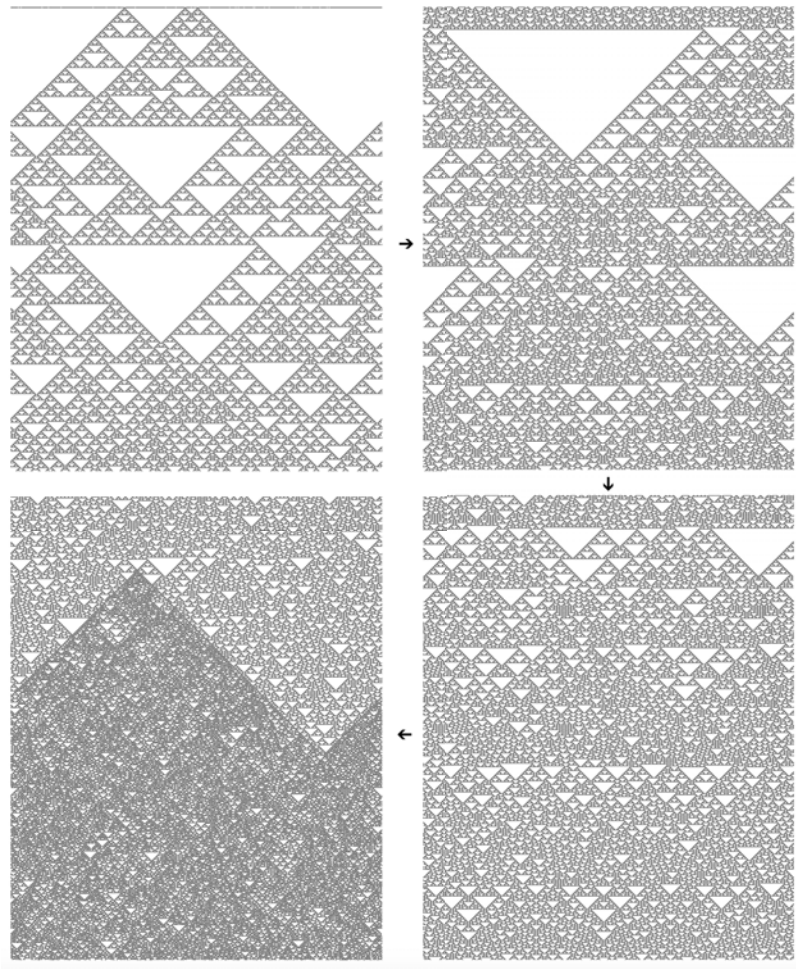


Figure 5. Disordered sparse fractal (DSF) evolving from initial density $d_0 = 0.93$ toward fixed point d_F after a long period: $(0 - 1000) \rightarrow (2000 - 3000) \downarrow (3000 - 4000) \leftarrow (27\,000 - 28\,000)$. DSF pattern appears as a long transient state before ergodicity. Compare with landscape in Figure 3 derived from a regular expression.

It should be observed that a disordered sparse fractal pattern (DSF) may evolve toward ergodicity (ERG) as in Figure 5, but sometimes after a long, unpredictable time. We denote as "DSF" such evolution remaining in this state at least within a time window of arbitrary length 10^3. In the same way, sparse backbone patterns (SBB) may evolve toward a dense backbone landscape (DBB).

https://doi.org/10.25088/ComplexSystems.28.2.125

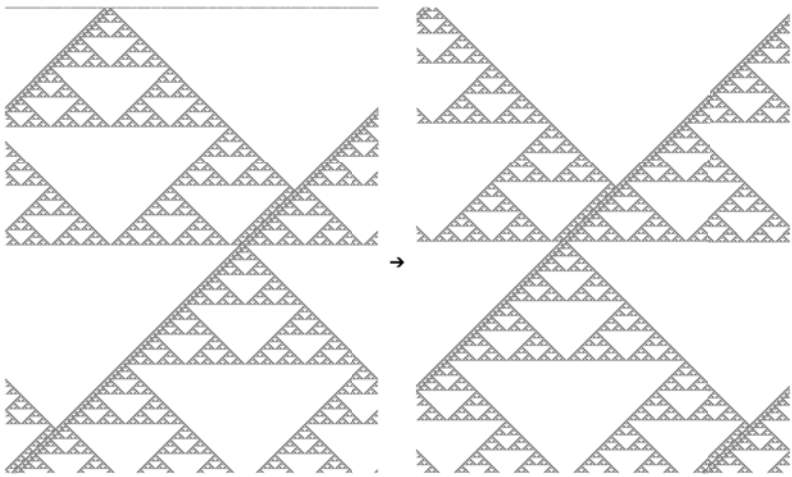


Figure 6. Rare periodic event occurring from initial density $d_0 = 0.94$ with a complete pattern moving leftward like a glider: $(0 - 1000) \to (200\,000 - 201\,000)$. Compare with the patchwork of periodic patterns in Figure 14. The probability of occurrence of such a pattern from a random initial distribution is about 10^{-3}.

A simple way to check (not to prove) whether unstable evolutions become eventually ergodic or to get a more global overview upon evolution is to skip some time steps with skip time lengths Δt. This transformation yields a projective view upon the (x, t) landscape with angle $\arctan 1 \,/\, \Delta t$. Various skipped scenarios of evolution from initial critical density $d_0 = 0.97$ in a ring of 800 cells and within a time window of length $10^3 \cdot \Delta t$ are displayed in Figure 7:

- (a) Phase transitions (a_1) DSF $\to$ ERG with $\Delta t = 32$ from disordered sparse fractal to ergodicity—up to 32 000 generations; transition occurs after about 8000 time steps (a_2) SBB $\to$ DBB with $\Delta t = 31$ from sparse to dense backbones—up to 31 000 generations; transition occurs after about 4000 time steps. Note that apparent discrepancies between densities in (a_1) and (a_2) before phase transition are no more than a side effect resulting from even or odd skip length parity.
- (b) Stratified landscapes SBB with $\Delta t = 64$ (b_1) up to 64 000 generations with observable backbones... and sub-backbones evolving like a Cantor dust (b_2) up to $4 \cdot 10^6$ generations with perpetual phase transitions.

Failing to prove the existence or not of ergodic evolution, this skipping approach nevertheless emphasizes several chaotic behaviors with long-range correlations. As well as in statistical physics, renormalization methods [15] overcome the weakness of mean field approximations that may fail or, at least, produce insufficient information.

Mean field theory is a rough approximation that assumes *independence* between neighboring sites. On the contrary, other deterministic approaches like de Bruijn diagrams assume *dependence*. They will be discussed later.

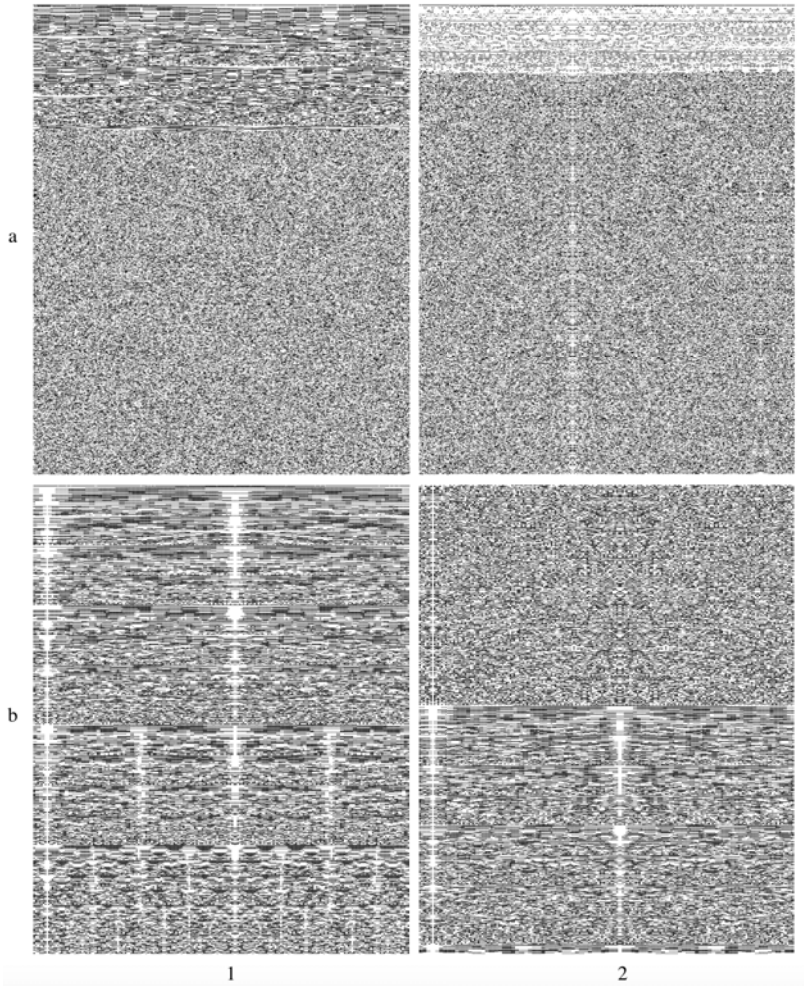


Figure 7. Skipped scenarios in a ring of 800 cells with $d_0 = 0.97$ within a time window of length $10^3 \cdot \Delta t$. (*a*) Phase transitions (a_1) with $\Delta t = 32$, DSF → ERG from disordered sparse fractal to ergodicity (a_2) with $\Delta t = 31$, SBB → DBB from sparse to dense backbones. (*b*) Stratified landscapes SBB with $\Delta t = 64$ (b_1) up to 64 000 generations with observable backbones... and sub-backbones evolving like a Cantor dust (b_2) up to $4 \cdot 10^6$ generations with perpetual phase transitions: a transition line separates a dense backbone (DBB) regime from a sparse backbone (SBB) regime; evolution remains still unstable.

https://doi.org/10.25088/ComplexSystems.28.2.125

3. Attractors

3.1 Basins of Attraction

Basins of attraction have been studied by Wuensche in the framework of ECAs and random Boolean networks [17–19]. A string of cell states x_i^t is a configuration c. An evolution is represented by a sequence of configurations $\{c_0, c_1, c_2, \ldots, c_{m-1}\}$, such that $\Phi : \Sigma^n \to \Sigma^n$, and the global transition can be represented as $\Phi(c^t) \to c^{t+1}$. A number of all global states of c is determined by the length of a string m^n (where n is the length and m the number of symbols). The structure of an attractor (Figure 8) is given in three parts. Leaves represent Garden of Eden, that is, unreachable in the evolution but only as initial global states (these states have no ancestors). Branches are configurations that have at least one ancestor and just one successor. The height of branches determines the number of generations to reach the attractor. An attractor is the final state of a string of length n. Numbers labeling vertices represent the decimal values of the strings.

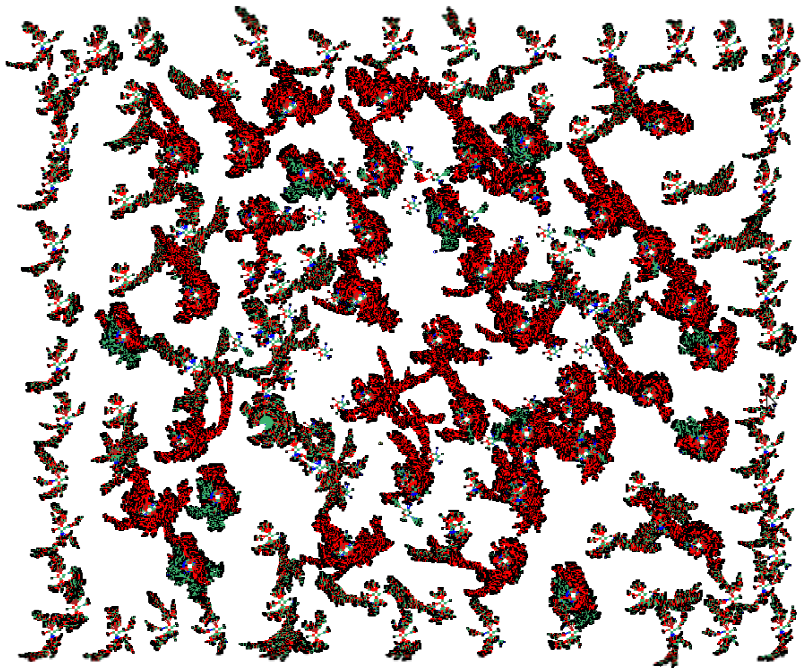


Figure 8. Basin of attractors in ECA rule 22 for rings of size 20. The number of attractors is 108 with 12 non-equivalent types. Based in the attractors' characterization, rule 22 displays chaotic behavior with highly dense, not long transients and several symmetric trees.

Wuensche [17] proposed that Wolfram's classes can be represented as a *basin classification*. In this classification, complex behavior is characterized by moderate transients, moderate-length periodic attractors, moderate in-degree and small density of leaves. This way, Figure 9 displays a type of nontrivial behavior thousands of generations later, starting with a concatenation of one of these strings calculated by one attractor of length 20:

00 000 001 000 100 000 000 → 00 000 011 101 110 000 000
→ 00 000 100 000 001 000 000 → 00 001 110 000 011 100 000
→ 00 010 001 000 100 010 000 → 00 111 011 101 110 111 000
→ 01 000 000 000 000 000 100 → 11 100 000 000 000 001 110
→ 00 010 000 000 000 010 000 → 00 111 000 000 000 111 000
→ 01 000 100 000 001 000 100 → 11 101 110 000 011 101 110.

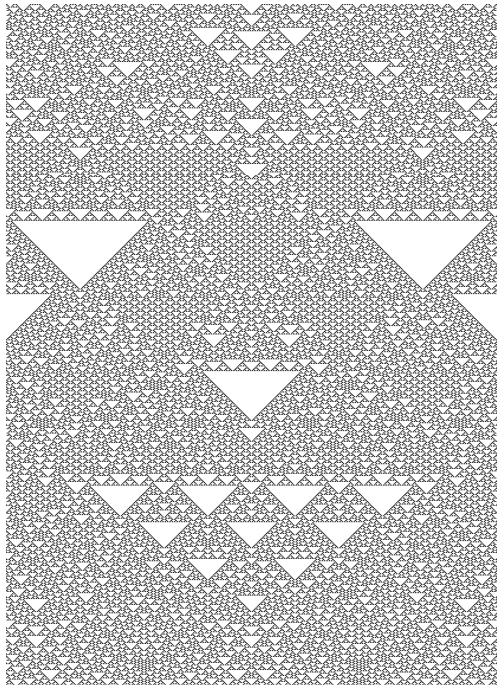


Figure 9. Discovering nontrivial patterns emerging in ECA rule 22 displaying a family of tilings of different sizes from a string of a basin of length 20 (Figure 8). A lot of these patterns can be reached with concatenation of the string 00000001000100000000.

https://doi.org/10.25088/ComplexSystems.28.2.125

Particularly, the average density for this evolution space is $4/15$. This density lives exactly between the density of the rule itself (equation (2)) and the stable fixed point in mean field theory (equation (6) and Figure 2(b)). We will note that this value is not reachable from the statistical analysis done for ECA rule 22. Also, it is the average where we report nontrivial behavior in ECA rule 22. Of course, when an evolution is evolving to this value and later switches to the density of the stable fixed point, the phase transition is irreversible (see Figure 3).

By calculating large attractors, we can discover landscapes of complexity in basins featured with nonsymmetric, high and dense ramifications: these kinds of ramifications are indicators of "unpredictable" behavior on most large configurations. No rarely chaotic rules tend to have symmetric basins. Basins of attraction can be connected into a meta-network, called the *jump graph* [18]. Jump graphs determine the next level of CA complexity by showing a probability to jump from one attractor to another attractor given a mutation on the same domain of strings [20].

Let us consider a one-bit value mutation $\Psi(\Phi(c_i)) \to \Phi(c_j)$ [18]. A configuration c_i expressed as a string $w_i = a_0 a_1 \ldots a_{n-1}$, such that it can jump into another configuration c_j expressed as a string $w_j = b_0 b_1 \ldots b_{n-1}$. Hence a_i can mutate to one b_i, where each configuration c belongs to the same field of attractors Ψ. Also, the mutation represents a loop in the same basin if $a_i = b_i$. Figure 10 shows a jump graph from the basins of attraction of length 20 (Figure 8). This way, a chaotic system presents a high density of connectivity with all the attractors in the jump graph.

3.2 Longest Paths and Representative Cycles

In this section, we will use the term *state* as an alias for global state c_i, configuration, or one-dimensional string. CA(n) denotes an ECA rule 22 with n cells and cyclic boundary. The aim of this section is to study:

- What is the length of the longest path until the zero state (00...0) is reached, and what does a related initial Garden of Eden state look like?
- How many cycles exist, how long are they, and what is a representative state for each cycle? Cycles that belong to the same class (cycles' states are equivalent under shift and mirroring) will be listed only once as a *representative cycle*, the kind of cycle we are interested in here.
- Are there similar states (cyclically shifted) that appear periodically within a cycle?
- What is the length of the longest path until a certain cycle is reached, and what does a related initial state look like?

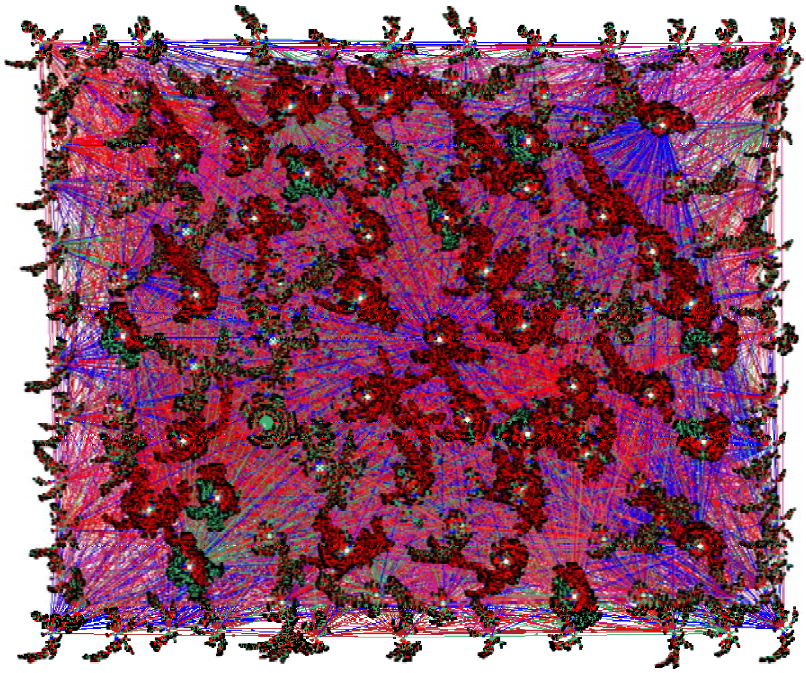


Figure 10. Jump graph in ECA rule 22 constructed with a base of attractors of length 20. The connection is determined by mutation of one bit in the strings. The chaotic behavior from jump graphs is characterized to the high connectivity between all attractors (for details see [20]).

The following terms and functions are used here:

- path(A, C) is a sequence of states (from state A to state C).
- length(path) gives the number of states of a path or cycle.
- prefix(C) is a path(A, B) where B is a direct predecessor of C.
- maxprefix(C) is a prefix(C) of maximum length.
- α = length(maxprefix(0)) where (0) is the zero state (00...0), the longest prefix(0).
- cycle denotes a periodic attractor, a cyclic path.
- k-cycle is a cycle of length k.
- ω(cycle) gives the length of a cycle.
- similar(S) is a state that can be derived from state S by cyclic shift and optional mirroring.
- ϵ is called *intra-cycle-period*.

https://doi.org/10.25088/ComplexSystems.28.2.125

- In some cycles, similar states appear again after ϵ time steps.
- k/e-cycle is a k-cycle where $e = \epsilon(k\text{-cycle})$. We may call k/e-cycles *strong* if $k = e$, and *weak* if $k > e$.
- *cycle prefix* is a prefix(D) where D belongs to a cycle.
- λ(cycle) gives the length of the longest cycle prefix.

First method and results. The CA(n), $n = 3\ldots20$, were simulated for all possible initial states and then analyzed by special programs and manual inspection. In order to avoid unnecessary simulations, similar initial states were excluded. Similar states are states that are equivalent under cyclic shift and mirroring. For instance, the number of different cases to be simulated for $n = 18$ is only 7685, which is significantly lower than 2^{18}. It should be noted that the cycles for n up to 34 were already computed by McIntosh [21].

Table 2 shows α (the length of the maxprefix(0)) and the related *normalized* initial state. A normalized state is a representative of all states that are equivalent under cyclic shift and mirroring. It is found by selecting the state with the smallest binary number among all equivalents. For example, normalize$(100\,011) = 000\,111$ and normalize$(110\,110\,100) = 001\,011\,011$ (by mirror and shift). The α values are much smaller than 2^n and not monotonically increasing with n.

n	α: Length max prefix(0)	Normalized Initial State
3	2	001
4	2	0001
5	6	01011
6	5	010111
7	6	0001111
8	8	00101011
9	2	000100101
10	7	0010111101
11	6	00101010101
12	23	001001100111
13	20	0001001100111
14	24	00010101100111
15	32	001001010100111
16	41	0000000101010011
17	53	00101010101110011
18	8	000001011000010011
19	17	0000101010111010101
20	18	00001000101010111101

Table 2. Longest path to the zero state.

Table 3 shows the results obtained by analyzing all the simulations. The operator shlP(c) means shift c to the left by P positions, and shlPm(c) means that first the mirror operator is applied before shifting.

n	ω	Representative Cycle State (Normalized)	ϵ	Repetition within Cycle	λ	Initial State of Longest Cycle Prefix
3, 5, 6		no cycle $\omega > 1$				
4	2	(0011)∗	1	c^{t+1}=shl2(c^t)	0	no prefix
7	7	(0001011)∗	1	c^{t+1}=shl4(c^t)	0	no prefix
	2	(0011)∗	1	c^{t+1}=shl2(c^t)	0	no prefix
8	4	00000101	2	c^{t+2}=shl4(c^t)	2	00100111
	6	00000011	3	c^{t+3}=shl4(c^t)	2	00101101
9	4	000000101			9	000100111
	4	0000001001	2	c^{t+2}=shl5(c^t)	7	0011010101
10	4	0000000101			5	0010110011
	6	0000010011	3	c^{t+3}=shl3m(c^t)	1	0000010011
	4	00000001001			7	00000100111
11	5	00000001111			7	00101010011
	11	00001001111	1	c^{t+1}=shl7(c^t)	9	00010110011
12	2	(0011)∗	1	c^{t+1}=shl2(c^t)	0	no prefix
= 3×4	4	000000010001	2	c^{t+2}=shl6(c^t)	6	000100110011
	5	000000001111			9	001010101101
13	5	0000000001111			13	0000101001101
14	7	(0001011)∗	1	c^{t+1}=shl3(c^t)	0	no prefix
= 2×7	12	00000010001111	6	c^{t+6}=shl9m(c^t)	4	00000101010011
15	20	000000010000011	4	c^{t+4}=shl4(c^t)	20	001010100101111
	2	(0011)∗	1	c^{t+1}=shl2(c^t)	0	no prefix
16	4	(00000101)∗	2	c^{t+2}=shl4(c^t)	2	(00100111)∗
= 2×8	6	(00000011)∗	3	c^{t+3}=shl4(c^t)	2	(00101101)∗
= 4×4	7	000000000000011			11	0000000010010101
	12	000000000000101	6	c^{t+6}=shl8(c^t)	29	0000001001010111
	12	000000000100001			10	0010101010101011
	4	000010100000101			2	00100101100100111
17	12	000000000000101			21	00101010101101011
	26	000000000010011	13	c^{t+13}=shl15m(c^t)	26	00101010010110011
	4	000100100000101	2	c^{t+2}=shl4(c^t)	2	00101010100101101
	4	(000000101)∗			45	00111001010101111
18	4	000010100000101			5	00101010011001111
= 2×9	12	000000000001001	6	c^{t+6}=shl9(c^t)	35	00000001000011101
	12	000000000000101			55	00010010100110011
	18	000001101001011	9	c^{t+9}=shl9(c^t)	7	01111001100110011

Table 3. (*continues*)

n	ω	Representative Cycle State (Normalized)	ϵ	Repetition within Cycle	λ	Initial State of Longest Cycle Prefix
	4	000100100000101			21	00001011100010111
	4	000101000000101			6	10100101010101011
19	4	001001000000101			1	00110111000110111
	12	000000000001001			78	00001100010010111
	28	000001000011101	14	c^{t+14}=shl4m(c^t)	9	01001100110011101
	2	(0011)*	1	c^{t+1}=shl2(c^t)	0	no prefix
	4	(0000001001)*	2	c^{t+2}=shl5(c^t)	44	00101100011010011
	4	001000100000101	2	c^{t+2}=shl4(c^t)	5	00110110011001000
	4	(0000000101)*			12	11010001110101111
20	4	001001000001001			62	01111001010101111
$=2\times10$	4	001001000000101			10	00010000100001111
$=4\times5$	6	100110000101111	3	c^{t+3}=shl7m(c^t)	1	01011010100101111
	6	(0000010011)*	3	c^{t+3}=shl7m(c^t)	8	00001010100010111
	8	000000001000001	4	c^{t+4}=shl10(c^t)	42	00000100011110101
	12	000000000010001	6	c^{t+6}=shl10(c^t)	24	00100101100100011
	24	000000000100001			98	01010101110101101

Table 3. Representative cycles for $n = 3...20$. ω: cycle length, ϵ: intra-cycle period, λ: length of longest cycle prefix. Not all leading zeros are displayed.

We find always the trivial $(00..0) \to (00..0)$ cycle of length 1. For even n there always exists the fixed point $(01)^* \to (01)^*$, a lonely 1-cycle with no prefix. We will not further mention or pay special attention to these basic 1-cycles.

For $n = 4$ and multiples of 4, we get the 2/1-cycle $(0011)^* \leftrightarrow (1100)^*$. The two strings are similar under shift of two positions, so the inherent pattern is the same.

For $n = 7$ and multiples of 7, we get a 7-cycle. In order to characterize this cycle, one representative is chosen; it is the one with the smallest normalized value, that is, $(0\,001\,011)^*$.

For $n = 8$, we get three cycles with length $\omega / \epsilon = 2/1, 4/2, 6/3$.

For $n = 10$, there exists a 6/3 cycle. After every three time steps, the same string appears in mirrored form and shifted three positions to the left.

For $n = 12$, the cycles of CA(4) form a subset (to be included if not detected) that is the 2/1 cycle $(001100110011) \leftrightarrow (11011001100)$.

For $n = 14$, the cycles of CA(7) form a subset that is shown as the 7/1 cycle.

Figure 8 shows all possible cycles where many of them are similar (equivalent under shift and mirroring). From that figure we may anticipate a very complex attractor structure, but we should realize that the number of representative cycles in CA(20) is only 11.

In general, because of the cyclic boundary, if k is a factor of n, then the CA(k) cycles are a subset of the CA(n) cycles. For example, the cycles of CA$(k = 4, 5, 10)$ form a subset of the CA$(n = 20)$ cycles. However there is a difference: the strings of CA(n) are cyclic repetitions of the CA(k) strings, and the original intra-cycle period ϵ may not appear in CA(n).

Second method and results. For larger n, the first method cannot further be applied due to extensive computational costs. Therefore, now only a relatively small random subset of all possible 2^n initial states is used in order to find a subset of all cycles and paths that are not necessarily the longest ones.

Ten thousand random initial states were generated for $n = 25, 30, 35, \ldots, 60$. For 5000 of the states, the probability 0.125 was used for each cell to generate a cell state 1. For the other 5000 of the initial states, at first a probability p between 0 and 1 (in steps of $1/1000$) was randomly selected. Then p was used for each cell to generate a cell state 1, otherwise 0. This technique of randomizing gave better results in experiments for rule 22 compared to the usage of a fixed probability of 0.5. CAs were simulated and cycles and path lengths were computed and processed in a semiautomatic mode. In addition, a genetic algorithm was used to find near-optimal α values. The results are presented in Table 4. Note that because of the statistical approach, the listed cycles are not complete and the true maximum path lengths could be longer; for example, for CA(60), all

n	ω Length of Cycles Detected	$\omega + \lambda$ Longest Path Ending in a Cycle	α Longest Prefix(0)
25	4, 5, 26, 28, 50, 55, 150	150 + 57 = 207	152
30	1, 4, 6, 14, 20, 40, 70, 86, 120, 240, 1070	1070 + 153 = 1223	419
35	4, 5, 12, 28, 64, 1015	1015 + 302 = 1317	179
40	1, 4, 8, 12, 16, 24, 52, 80, 124, 206, 320	124 + 2551 = 2675	303
45	4, 8, 16, 19, 2295	4 + 4815 = 4819	540
50	1, 12, 28, 31, 55, 56, 100, 117, 150, 252, 700, 3150	252 + 13 956 = 14 208	750
55	12, 28, 30, 56, 60, 330, 440, 660, 990, 4620, 36 190, 148 225	148 225 + 7124 = 155 349	13904
60	1, 12, 60, 120, 138, 395, 476, 480, 22 740, 40 980	40 980 + 1004 = 41 984	35579

Table 4. Representative cycles for $n = 25 \ldots 60$. ω: cycle length, λ: length of longest cycle prefix, $\omega + \lambda$: length of longest path detected. The values were obtained by simulation of 10 000 random initial states.

https://doi.org/10.25088/ComplexSystems.28.2.125

cycles already found for the factors CA(4, 5, 10, 12, 15, 20, 30) have to be included.

We can summarize that for $n \leq 60$, the longest paths are much smaller than $2^n - 1$ (which is achievable with other ECA rules [22]). Further work remains to find a general formula or at least boundaries for the longest paths and the cycle distribution, depending on the number of cells.

4. De Bruijn Diagrams

De Bruijn diagrams [23] were originally proposed in shift–register theory [24]. For a one-dimensional CA(k, r), the de Bruijn diagram is defined as a directed graph with k^{2r} vertices and k^{2r+1} edges. Vertices are labeled with the elements of symbols of length $2r$. An edge is directed from vertex i to vertex j, if and only if the $2r - 1$ final symbols of i are the same as the $2r - 1$ initial ones in j, forming a neighborhood of $2r + 1$ states represented by $i \diamond j$. In this case, the edge connecting i to j is labeled with $\varphi(i \diamond j)$ (the value of the neighborhood defined by the local function) [25], as shown in Figure 11 for ECA(2, 1) and for rule 22 in Figures 12 and 13.

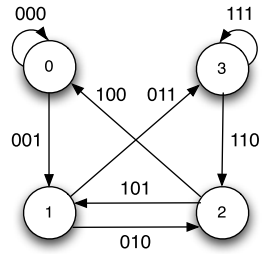


Figure 11. Generic de Bruijn diagram for ECA(2, 1).

$$M_{R\,22} = \begin{bmatrix} 0 & 1 & . & . \\ . & . & 1 & 0 \\ 1 & 0 & . & . \\ . & . & 0 & 0 \end{bmatrix}$$

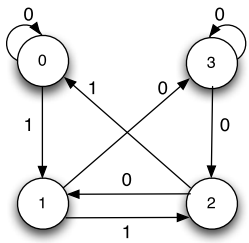


Figure 12. Connection matrix and de Bruijn diagram for ECA rule 22.

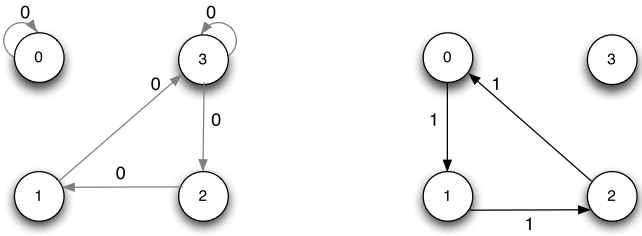


Figure 13. De Bruijn subdiagrams showing unreachable states.

4.1 Basic de Bruijn Diagram

The connection matrix M corresponding to the de Bruijn diagram [26] is as follows:

$$M_{i,j} = \begin{cases} 1 & \text{if} j \in \{ki,\, ki+1,\, \ldots ki+k-1 (\bmod k^{2\cdot r})\} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

wherein module $k^{2\cdot r}$ represents the number of vertices and j takes on values in $\{ki,\, ki+1,\, \ldots,\, ki+k-1 (\bmod k^{2\cdot r})\}$. Hence for ECA$(2,\, 1)$

$$M_{i,j} = \begin{cases} 1 & \text{if } j \in \{2i,\, 2i+1 (\bmod 4)\} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

and vertices are labeled by fractions of the overlapping of neighborhoods originated by 00, 01, 10 and 11, and the overlaps of the full neighborhood determine each connection:

$$\begin{array}{llllllll}
(0,\mathbf{0}) \diamond (\mathbf{0},0) \to 0 & \mathbf{0} & 0 & (1,\mathbf{0}) \diamond (\mathbf{0},0) \to 1 & \mathbf{0} & 0 \\
(0,\mathbf{0}) \diamond (\mathbf{0},1) \to 0 & \mathbf{0} & 1 & (1,\mathbf{0}) \diamond (\mathbf{0},1) \to 1 & \mathbf{0} & 1 \\
(0,\mathbf{1}) \diamond (\mathbf{1},0) \to 0 & \mathbf{1} & 0 & (1,\mathbf{1}) \diamond (\mathbf{1},0) \to 1 & \mathbf{1} & 0 \\
(0,\mathbf{1}) \diamond (\mathbf{1},1) \to 0 & \mathbf{1} & 1 & (1,1) \diamond (1,1) \to 1 & 1 & 1.
\end{array}$$

They are the edges of the generic de Bruijn diagram in Figure 11. The de Bruijn diagram has four vertices that can be renamed as $\{0, 1, 2, 3\}$, corresponding to four partial neighborhoods of two cells $\{00, 01, 10, 11\}$, and eight edges representing neighborhoods of size $2r+1$. The de Bruijn diagram for rule 22 is derived from the generic one (Figure 11) and it is calculated in Figure 12, where the edges are labeled by the next state.

Paths in the de Bruijn diagram may represent chains, configurations or classes of configurations in the evolution space. Vertices are sequences of symbols in the set of states and the strings are sequences of vertices in the diagram. The edges represent overlapping of the

sequences. Different intersection degrees evoke different de Bruijn diagrams (Figure 13). Thus, the connection takes place between an initial symbol, the overlapping symbols and a terminal one. For practical reasons, we can use colors, thus the color of an edge represents the next state to which each neighborhood, as shown in Figure 11, evolves.

4.2 Extended de Bruijn Diagram

An extended de Bruijn diagram [6, 26] takes into account wide overlapping of neighborhoods. We represent $M_{\mathrm{R22}}^{(2)}$ by indices $i = j = 2rn$, where $n \in \mathbb{Z}^+$, $M_{\mathrm{R22}}^{(3)}$ and $i = j = 3rn$, $M_{\mathrm{R22}}^{(4)}$ and $i = j = 4rn$, and so up to $M_{\mathrm{R22}}^{(m)}$ with $i = j = mrn$; consequently, a basic de Bruijn diagram is obtained when $m = 1$. The regular expressions derived from the de Bruijn diagram for rule 22 (Table 5) can be linked to spacetime dynamics phenomena exhibited by the rules. These include symmetric complex behavior, chaos and stable periodic behavior. Figure 14 shows in detail every periodic pattern yielded from extended de Bruijn diagrams. To read the diagram, we use notation (i, j), where i is a displacement (left or right) and j is a number of generations. Thus the pattern in position $(0, 0)$ (upper center) displays a periodic pattern without both displacement and period; the expression reproducing this pattern is $(01)^*$ (de Bruijn subdiagram in Figure 14 and equation 2 in Table 5). In this way, a lot of nontrivial patterns can be extracted. Let us consider a few examples.

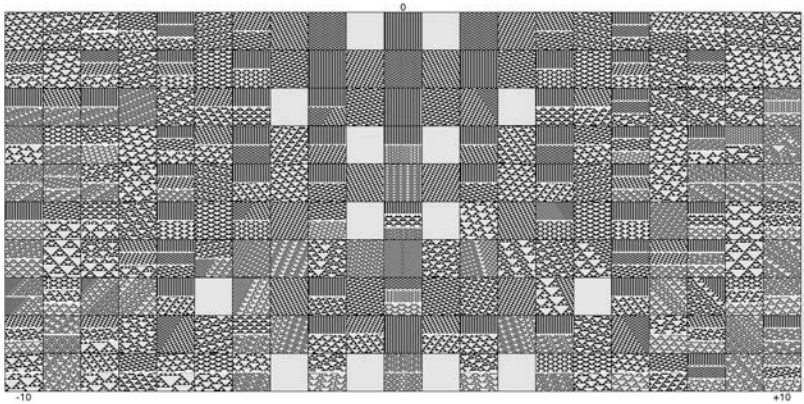


Figure 14. The whole set of periodic patterns yielded from extended de Bruijn diagrams to 10 generations with positive and negative shifts to 10 cells.

- (0, 7). The graph is characterized by two cycles, the small cycle yielding simple still-life patterns and the large cycle representing configurations emitting traveling localizations, or particles. These configurations

cannot evolve naturally, that is, from an initial random condition, because they are destroyed when a certain limit of their size is reached (Figure 15). The configurations are expressed for the following cycle ways:

4515 → 9080 → 167 → 3354 → 6708 → 13 416 → 10 449 → 4515 →
9031 → 1679 → 3358 → 6716 → 13 432 → 10 481 → 4578 →
9157 → 1931 → 3862 → 7724 → 15 448 → 14 513 → 12 642 →
8901 → 1419 → 2838 → 5676 → 11 352 → 6321 → 12 642.

The regular expression to reproduce the same pattern is calculated as equation 8 in Table 5.

- (4, 8). The graph has several paths between different cycles. Some of these cycles calculate trivial patterns. Other cycles represent configurations developed from a "fusion" of two periodic regions competing for the space (Figure 16). Other fused configurations can be found in coordinates (6, 9), (−4, 6), (−4, 8), (−5, 7), (−6, 9), (−8, 7) and (−10, 8).

Equation	Expression	Evolution
1	$(0 + 1)^*$	stable state
2	$(01)^*$	stable periodic
3	$(001)^*$	stable state
4	$11(01)^*00$	still life & symmetric complex behavior
5	$(01)^*(0 + 1)$	stable periodic & symmetric complex behavior
6	$(000\,111)^*$	stable state
7	$(0 + 0(01)^*00)^*$	stable periodic, chaos & big gaps
8	$(((01)^* + 0)00^*1)^*$	chaos, complex behavior
9	$(0 + 1) + 11(01)^*(0 + 1)$	complex behavior
10	$((01)^*00)(0 + 0(01)^*00)^*$	chaos, stable periodic, chaos & big gaps
11	$(11(01)^*00)(0 + 0(01)^*00)^*$	chaos, still life & symmetric complex behavior
12	$(0 + 0(01)^*00)^*0(01)^*(0 + 1)$	stable state, stable periodic, chaos & big gaps
13	$(0 + 1(01)^*00)(0 + 0(01)^*00)^*$	chaos, stable periodic, chaos & big gaps
14	$((0 + 1) + 11(01)^*1) + (11(01)^*00)$ $(0 + 0(01)^*00)^*(0(01)^*(0 + 1))$	–
15	$(0^*10^*)(10^*10 + (10^* + 10^*10)0^*)^*$	–

Table 5. Regular expressions derived in ECA rule 22. The set of equations is calculated using the recursive function $R^k_{i,j}$ (equation 9) to recognize k paths between nodes i to j in the de Bruijn diagram (Figure 12).

https://doi.org/10.25088/ComplexSystems.28.2.125

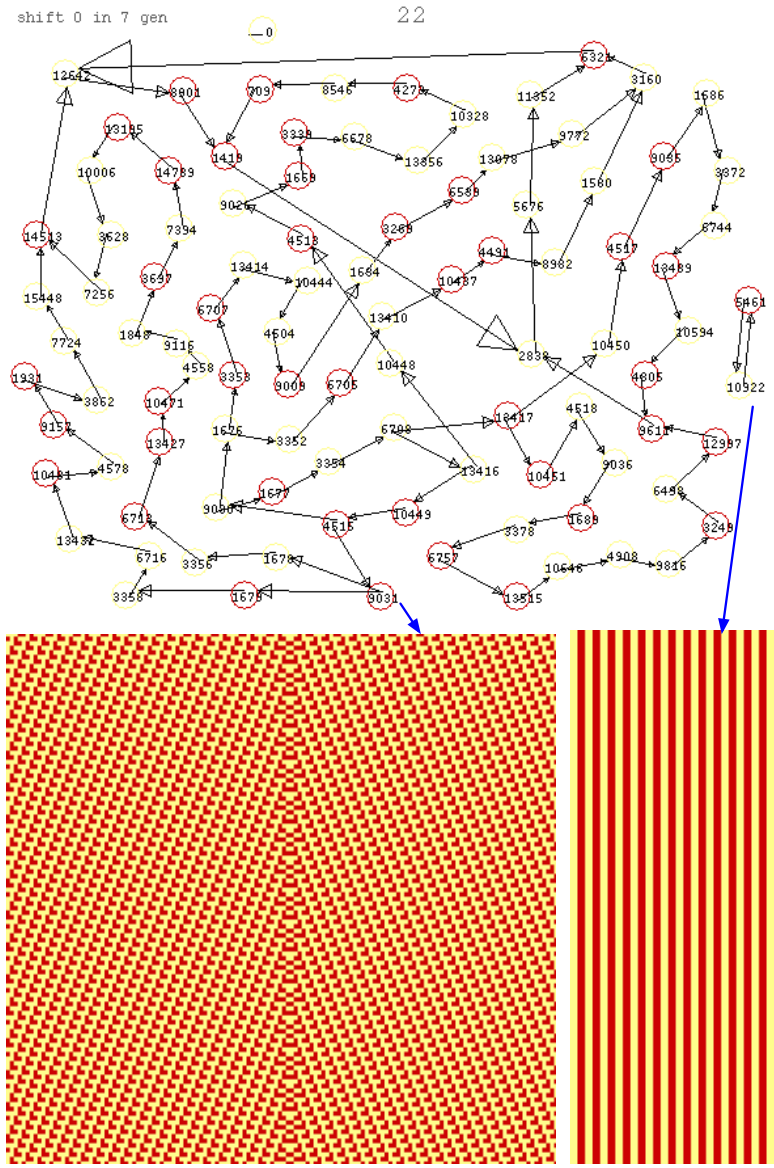


Figure 15. The extended de Bruijn diagram (0,7) calculating a pattern emitting mobile self-localizations. To reproduce this pattern, we concatenate the expressions $(1\,010\,001)^n$-111000101100010-$(1\,100\,010)^n$, where $n > 0$ is the number of copies. The small cycle represents still-life patterns.

- (10, 2). Here we observe composed triangular polygons. The polygons exist in a periodic mobile background. The background features small particles crossing the space (Figure 17). This complex behavior emerges

with probability between 1/7 and 3/7. This complex behavior was discovered with the help of expressions derived from the basins of attraction (see Figure 9). One more example of interaction of large triangular domains evolving in the periodic background is shown in Figure 18.

- (0, 4). The periodic background is stationary. Several types of triangular domains and several families of small tiles emerge (Figure 19). Other coordinates leading to similar configurations are (0, 8), (−6, 2), (−6, 6), (6, 2) and (6, 6).

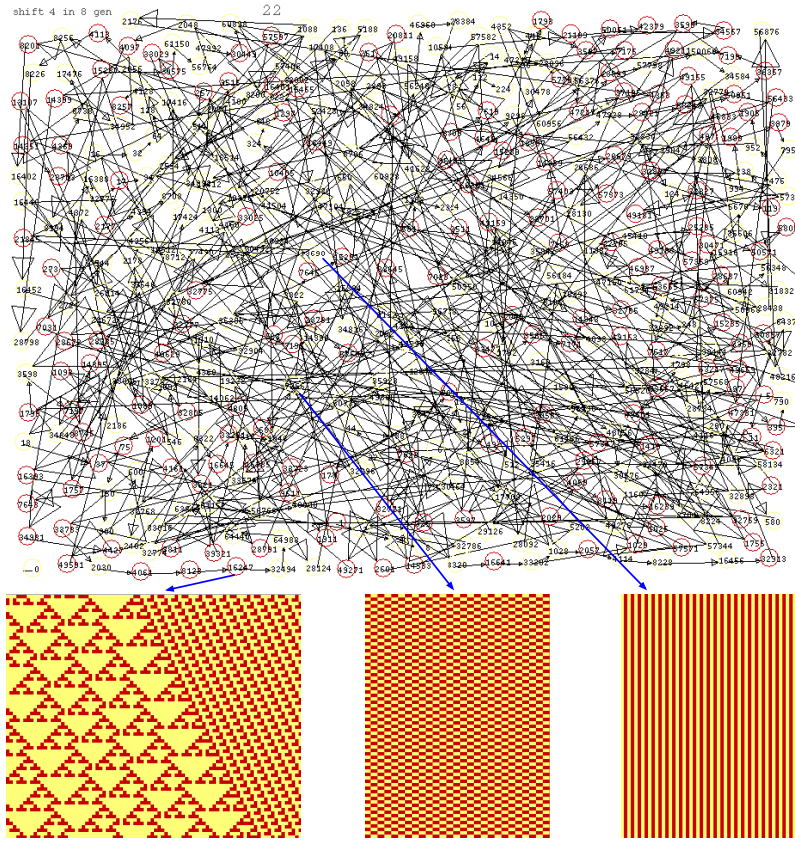


Figure 16. The extended de Bruijn diagram (4,8) calculating mobile self-localizations, small tilings and meshes. The first large cycle calculates a configuration known as fuse [6] because two periodic patterns with different densities can evolve together without perturbing each other's boundaries.

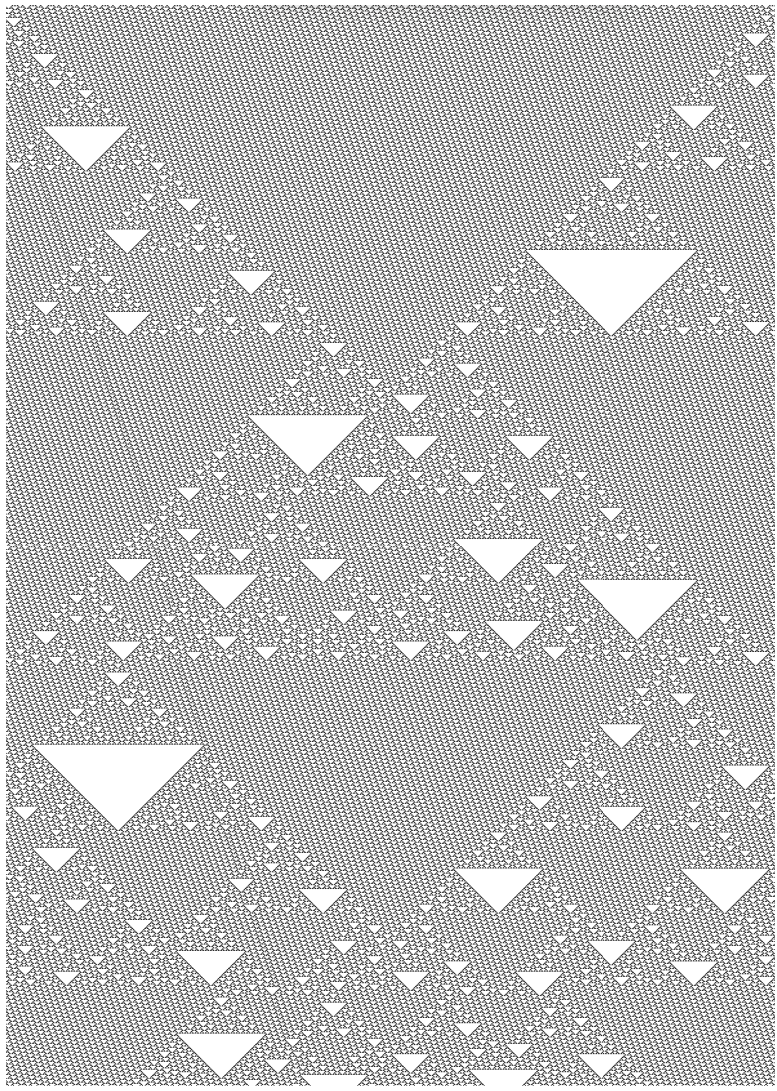


Figure 17. Nontrivial dynamics emerging in ECA rule 22 on a ring of 1198 cells during 1679 generations. Mobile localizations emerge as triangular polygons traveling in a mobile periodic background. The localizations conserve their shape when colliding with each other. This dynamic was discovered with the extended de Bruijn diagram order (10,2); see Figure 12. These configurations cannot be reached from a random initial condition.

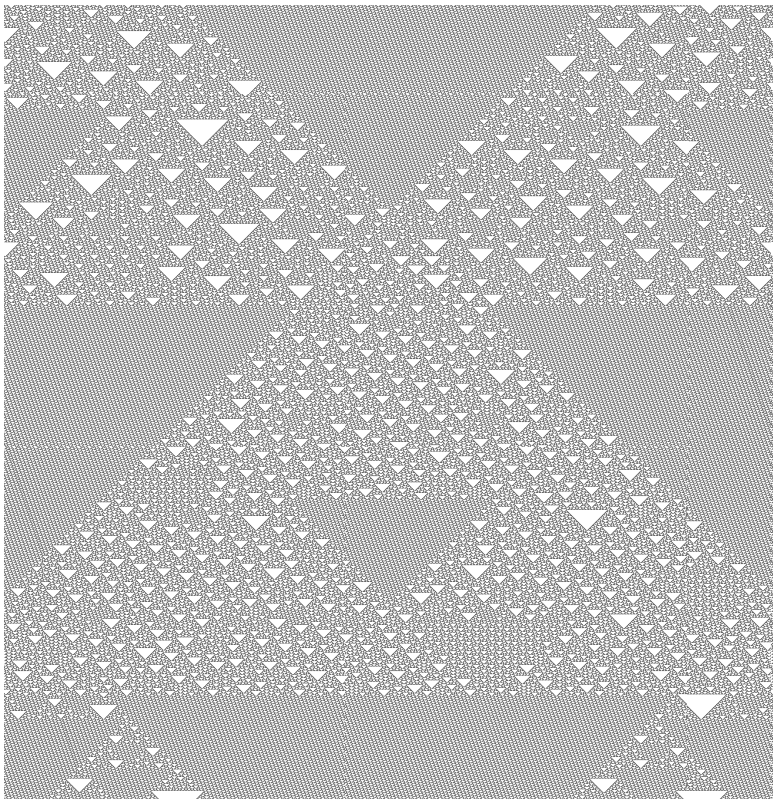


Figure 18. Nontrivial dynamics emerging in ECA rule 22 with the mobile periodic background (10,2) evolving with a high density of small tilings. Different large triangular polygons can be constructed from the interactions of other polygons.

During the analysis of configurations derived with the de Bruijn diagrams and basin of attractors, we have referred to some specific regular expressions. Regular expressions can be calculated recursively by following paths on a graph with [27]:

$$R_{i,j}^{k} = R_{i,j}^{k-1} + R_{i,k}^{k-1}\left(R_{k,k}^{k-1}\right)^{*} R_{k,j}^{k-1} \tag{9}$$

where i is the initial state and j the final state. The base case when $k = 0$ is the direct path to every node. This way, by using the basic de Bruijn diagram in Figure 12, we have calculated the whole set of regular expressions, summarized in Table 5. The first column shows an equation number, the second column the regular expression and the third column the kind of behavior that emerges when we codify configurations by these regular expressions. This way, we could evaluate these equations and explore an unlimited number of configurations.

https://doi.org/10.25088/ComplexSystems.28.2.125

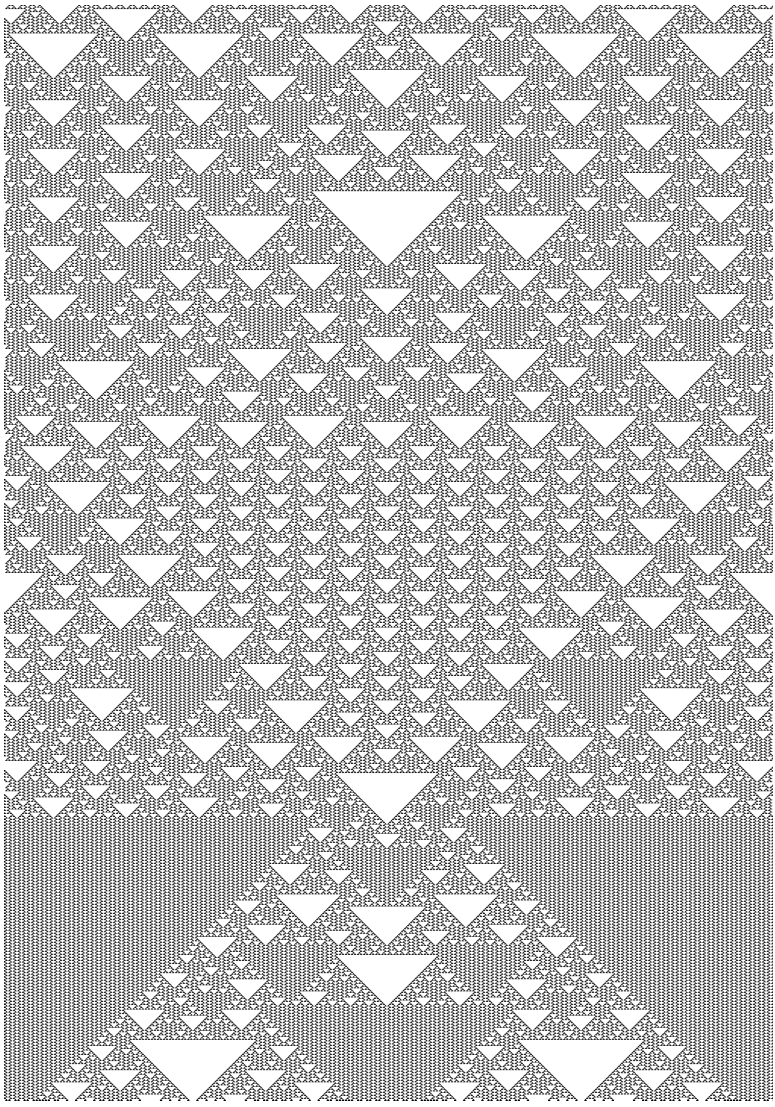


Figure 19. Nontrivial behavior emerging in a fixed periodic background; that is, this background does not move. In this periodic background, complex, large, triangular polygons can emerge as well, including several types of small tiles. This fixed periodic background can be calculated from the extended de Bruijn diagram (0,4); see Figure 12.

5. Garden of Eden

The question, Does a complex CA contain a universal constructor? is a classic problem appearing in the CA literature since von Neumann

[28]. A configuration of a universal constructor in the GoL CA was proposed by Goucher in 2010 [29]. In this context, our aim is to know if ECA rule 22 is able to construct any string. Previously this problem was studied by McIntosh [6], who found that rule 22 has a global injective relation and therefore configurations without ancestors exist.

We use a subset diagram to calculate Garden of Eden configurations [6], that is, configurations without ancestors [30]. A subset diagram has $2^{k^{2r}}$ vertices with k states and r neighbors. If all the configurations of a certain length have ancestors, then all the configurations with extensions both to the left and to the right with the same equivalence must have ancestors. But if this is not the case, then the vertices represent Garden of Eden configurations.

We can define the subset diagram as the power set of $2^{k^{2r}}$, such that each subset $S \in U_S$ (where U_S is a power set) and one symbol $a \in \Sigma$:

$$\alpha(S, a) = \bigcup_{q_i \in S} \varphi(q_i, a). \tag{10}$$

Vertices of the subset diagram are formed by the combination of each subset formed from the states of the de Bruijn diagram. Symbolic de Bruijn matrices $M_{k,s}$ or M_s are characterized by k states and s number of states in the partial neighborhood. Thus, for rule 22 we can obtain symbolic matrices, derived from the de Bruijn subdiagrams shown in Figure 13. For any ECA we have four sequences of states in the Bruijn diagram, enumerated as 0, 1, 2 and 3 (see Figure 11).

Union between subsets is represented by the state in which each sequence evolves and is assigned to the states (subsets that form it) as governed by equation (10). Relations between subsets for rule 22 are constructed in Table 6. Figure 20 shows the full scalar subset diagram. Each class of edges defines a function on Σ_0 or Σ_1. The subset diagram describes the union $\Sigma_0 \bigcup \Sigma_1$, which by itself is not functional [26].

We must distinguish four types of subsets, where it is possible to make a transition between its four unit classes. Also, we should observe that a residual of the de Bruijn diagram can be found in the subset diagram. This is because a unit class is precisely defined by the nodes of the original diagram. At first instance, we can see some relations are more frequent than others. Also there are nodes without inputs, or nodes with most connections including loops. Most important are cycles of different lengths. They are used to infer words, or sequences, that a CA could recognize. Thus, the subset diagram can be used as a general machine to recognize the universe of words in which a CA could evolve.

https://doi.org/10.25088/ComplexSystems.28.2.125

S	Label	0	1
ϕ	0	0	0
{0}	1	1	2
{1}	2	0	12
{2}	4	1	2
{3}	8	8	4
{0,1}	3	1	14
{0,2}	5	1	2
{0,3}	9	9	6
{1,2}	6	1	14
{1,3}	10	8	12
{2,3}	12	9	6
{0,1,2}	7	1	14
{0,1,3}	11	9	6
{0,2,3}	13	9	6
{1,2,3}	14	9	14
{0,1,2,3}	15	9	14

Table 6. Relations between states of the subset diagram in rule 22.

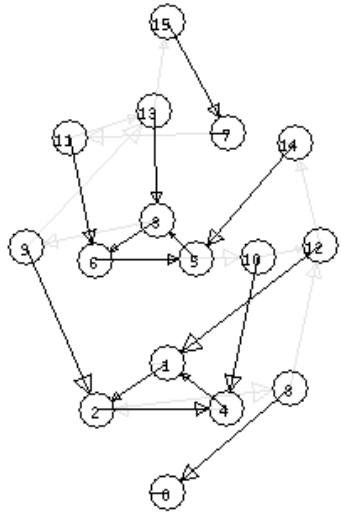


Figure 20. The simplified subset diagram for ECA rule 22.

By analyzing the full diagram, we can derive a small subset diagram that is deduced from the original diagram (de Bruijn diagram). This diagram includes only vertices with cycles, the universal and empty set and the subset with one element, yielding a new diagram that will be more practical for us. The reduction gives yet a smaller diagram to quickly read strings belonging to Garden of Eden configu-

rations. The reduction is also useful to calculate the degree of Welch indices for reversible CAs [31]. The expressions that determine Garden of Eden configurations in ECA rule 22 are listed below:

- 10110
- $01\,111^*01\,101^*$
- $11^*(011\,111)^*1^*0110$

6. Fractals

6.1 Iterated Functions in Rule 22

A fractal is constructed recursively from a self-replication of a pattern [32]. Chaotic systems often bear properties of fractals.

ECA rule 22 produces a fractal pattern, known as a Sierpiński triangle, starting from a single cell in state 1 (Figure 1). Figure 21 shows a triangle constructed with three small tiles derived in rule 22; this triangle grows in a power of two with respect to the number of cells. The main triangle (Figure 21(a)) has three replicas in the next iteration (Figure 21(b)), and the following iteration produces nine base

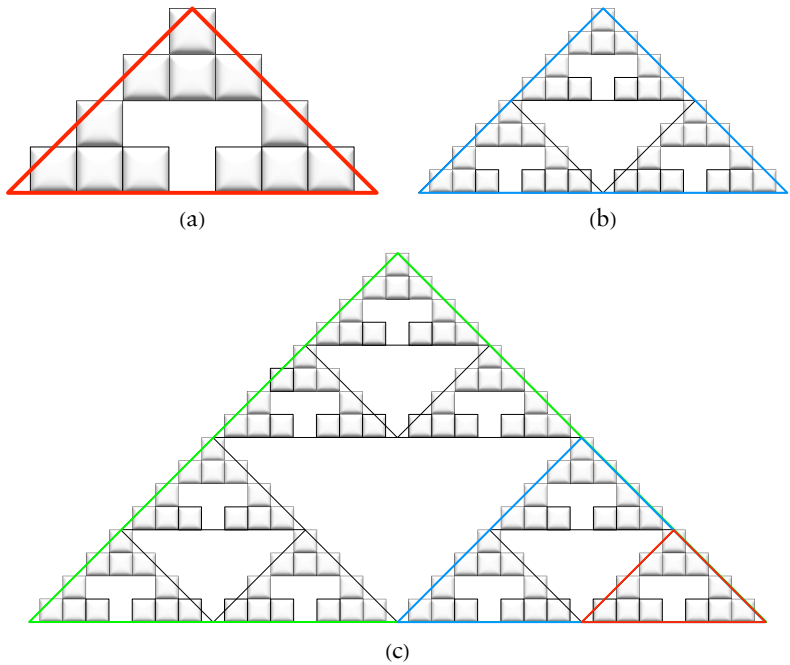


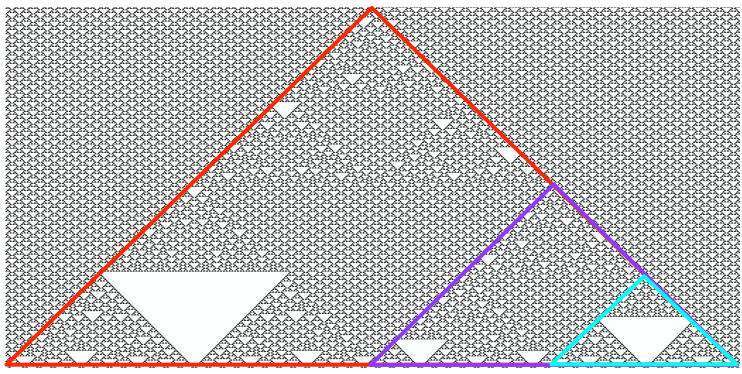
Figure 21. Iterated function determines a fractal defining a Sierpiński triangle in ECA rule 22 from a composition of three tiles starting with a 1.

replicas (Figure 21(c)). The fractal dimension D can be calculated given the number of replicates N and the scaling factor m [33]. The fractal dimension of the patterns generated by rule 22 is the following:

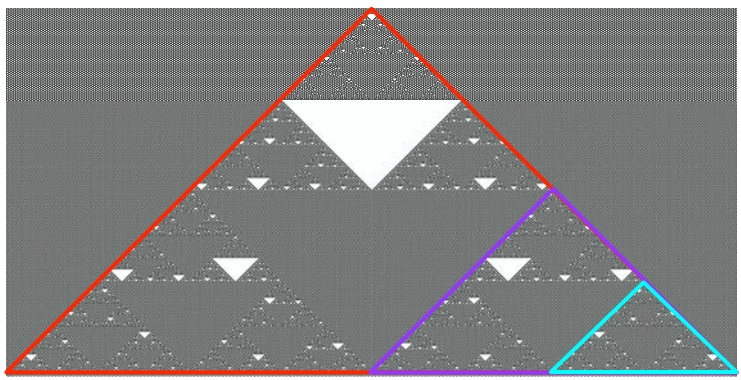
$$D = \frac{\log(N)}{\log(m)} = \frac{\log(3)}{\log(2)} = 1.5849625. \tag{11}$$

Also, ECA rule 22 displays nontrivial behavior via fractals where they emerge in different stages during the evolution. These fractals are combined with other fractals constructed from rule 22 over thousands of generations.

Using regular expressions, we found two different periodic backgrounds emerging in ECA rule 22, as discussed in Sections 5–7. Let us illustrate two fractals growing in intervals of other fractals with different compositions of tiles. Figure 22(a) shows the initial state of



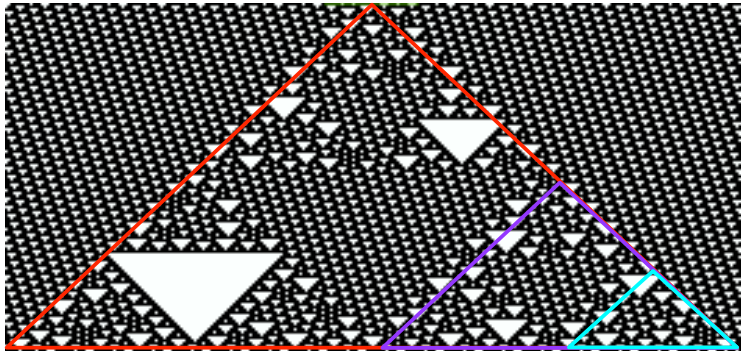
(a)



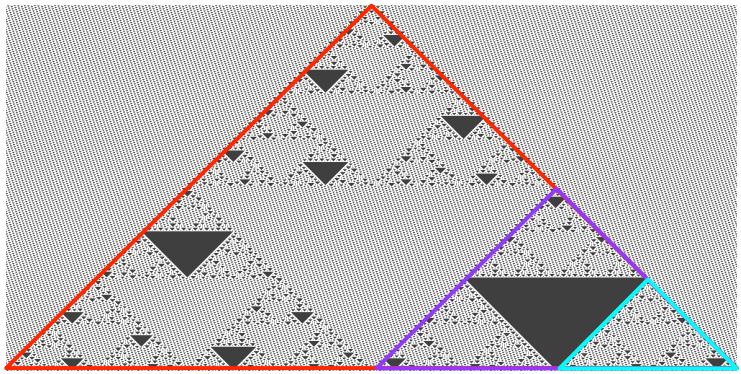
(b)

Figure 22. Composition of nontrivial fractals emerging in ECA rule 22 after thousands of generations. The iterated function preserves its fractal dimension. These fractals evolve on a periodic background without displacement.

fractals growing in a periodic background without displacement, conserving the same fractal dimension. Figure 22(b) shows the same iterated function over thousands of generations. The same behavior is tested on a periodic background with displacement in Figure 23. Composed fractals emerging in periodic backgrounds with or without displacement are disjoint.



(a)



(b)

Figure 23. Composition of nontrivial fractals emerging in ECA rule 22 after thousands of generations. The iterated function preserves its fractal dimension. These fractals evolve on a periodic background without displacement.

6.2 Rule 18, Mutations, Gaskets and Seashells

Rules 18 and 22 are complex rules widely reported in ECA literature [11, 13, 21, 34, 35]. Let us redefine them as follows: a cell takes state 1 if exactly one

- (R18): of its neighbors is in state 1: $(100, 001) \to 1$
- (R22): in its neighborhood is in state 1: $(100, 010, 001) \to 1$

https://doi.org/10.25088/ComplexSystems.28.2.125

and consider the subset of the 256 ECAs

$$\Psi_{\mathrm{R\hat{1}8}} = \begin{cases} 1 & \text{if} \quad 100, 001 \\ 0 & \text{if} \quad 111, 000 \end{cases} \tag{12}$$

that defines the 16 rules displayed in Table 7. Referring to the genotype paradigm in [36] with a rule defined by the sequence $(b_7 b_6 \ldots b_1 b_0)$, a rule R *mutates* into rule R' through bit (or *gene*) b_i, or $(R \mid b_i \mapsto R')$ with exactly a 1-bit change, which yields the mutation (or inheritance) tree in Figure 24.

Rule	111	110	101	100	011	010	001	000	Mutation
18	**0**	0	0	**1**	0	0	**1**	**0**	
22	**0**	0	0	**1**	0	1	**1**	**0**	$18 \mid b_2 \mapsto 22$
26	**0**	0	0	**1**	1	0	**1**	**0**	$18 \mid b_3 \mapsto 26$
30	**0**	0	0	**1**	1	1	**1**	**0**	$22 \mid b_3 \mapsto 30$
50	**0**	0	1	**1**	0	0	**1**	**0**	$18 \mid b_5 \mapsto 50$
54	**0**	0	1	**1**	0	1	**1**	**0**	$22 \mid b_5 \mapsto 54$
58	**0**	0	1	**1**	1	0	**1**	**0**	$26 \mid b_5 \mapsto 58$
62	**0**	0	1	**1**	1	1	**1**	**0**	$30 \mid b_5 \mapsto 62$
82	**0**	1	0	**1**	0	0	**1**	**0**	$18 \mid b_6 \mapsto 82$
86	**0**	1	0	**1**	0	1	**1**	**0**	$22 \mid b_6 \mapsto 86$
90	**0**	1	0	**1**	1	0	**1**	**0**	$26 \mid b_6 \mapsto 90$
94	**0**	1	0	**1**	1	1	**1**	**0**	$30 \mid b_6 \mapsto 94$
114	**0**	1	1	**1**	0	0	**1**	**0**	$50 \mid b_6 \mapsto 114$
118	**0**	1	1	**1**	0	1	**1**	**0**	$54 \mid b_6 \mapsto 118$
122	**0**	1	1	**1**	1	0	**1**	**0**	$58 \mid b_6 \mapsto 122$
126	**0**	1	1	**1**	1	1	**1**	**0**	$62 \mid b_6 \mapsto 126$

Table 7. Mutation table from rule 18 in the ECA subset $(b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0) = (\mathbf{0} b_6 b_5 \mathbf{1} b_3 b_2 \mathbf{1} \times \mathbf{0})$. A rule R *mutates* into rule R' through bit b_i $(R \mid b_i \mapsto R')$ with exactly a 1-bit change.

This tree contains a set of rules with complex behavior inherited from rule 18 and where Sierpiński patterns often appear. Rules 18, 22, 122 and 126 reveal a similar behavior toward ergodic regime and unstable areas, as well as rule 90, but that displays a multilayered pattern, whereas rule 30 and, by reflection, rule 86 exhibit small striped patterns near the polygonal border. Rule 26 and, by reflection, rule 82 exhibit large striped patterns at equilibrium, which may evolve toward sparse backbones at low or high densities, or periodic patterns similar to those in Figure 6. Rule 94 displays a Sierpiński

gasket, but evanescent and quickly entering a uniform polygon with vertical stripes. Finally it should be emphasized that rules 50, 54, 58 and, by reflection, 114, 62 and, by reflection, 118 display patterns somewhat far from the sieve, but nevertheless all of them enter the phase transition polygon. Rules 30, 54, 90 and 126 are examined elsewhere [37–40].

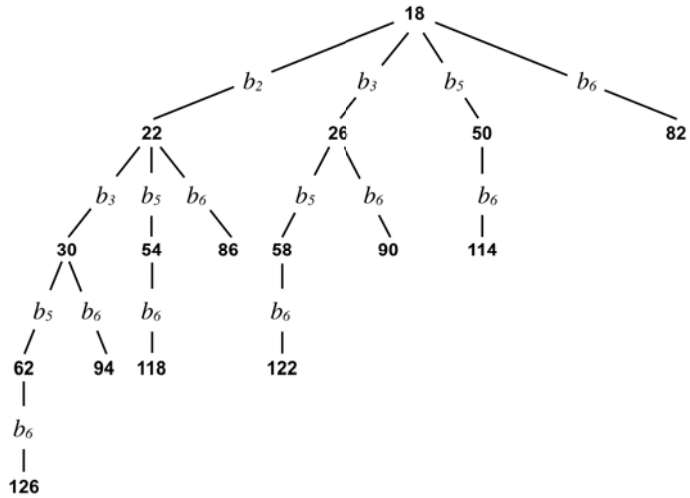


Figure 24. Mutation tree from rule 18 derived from Table 7. A branch from rule R to rule R' $(R \mid b_i \mapsto R')$ bears exactly the 1-bit genetic change b_i and $d_H(R, R') = 1$ where d_H is the Hamming distance.

This tree does not contain the subset of rules associated either by conjugation or by conjugation-reflection. Beginning from rule 129—conjugate with rule 126—this subset would display a large collection of Sierpiński-like patterns, but this time in white on a black background.

Even in the synchronous "1nCA" [41] with a minimal neighborhood of two cells—the cell itself and the either left or right adjacent cell alternating at either odd or even time steps—Sierpiński patterns appear in rules 6Δ and 9Δ, namely for the only symmetric rules with an equal number of black and white cells, and for the only rules fulfilling the maximal "sensitivity parameter."

Six rules (18, 22, 30, 90, 122, 126) evolving in their ergodic regime are displayed in Figure 25. All patterns have the Hausdorff dimension $\log_2(3)$ of equation (11). They only differ in the average density d_C of their mesoscopic minimal macrocell C.

https://doi.org/10.25088/ComplexSystems.28.2.125

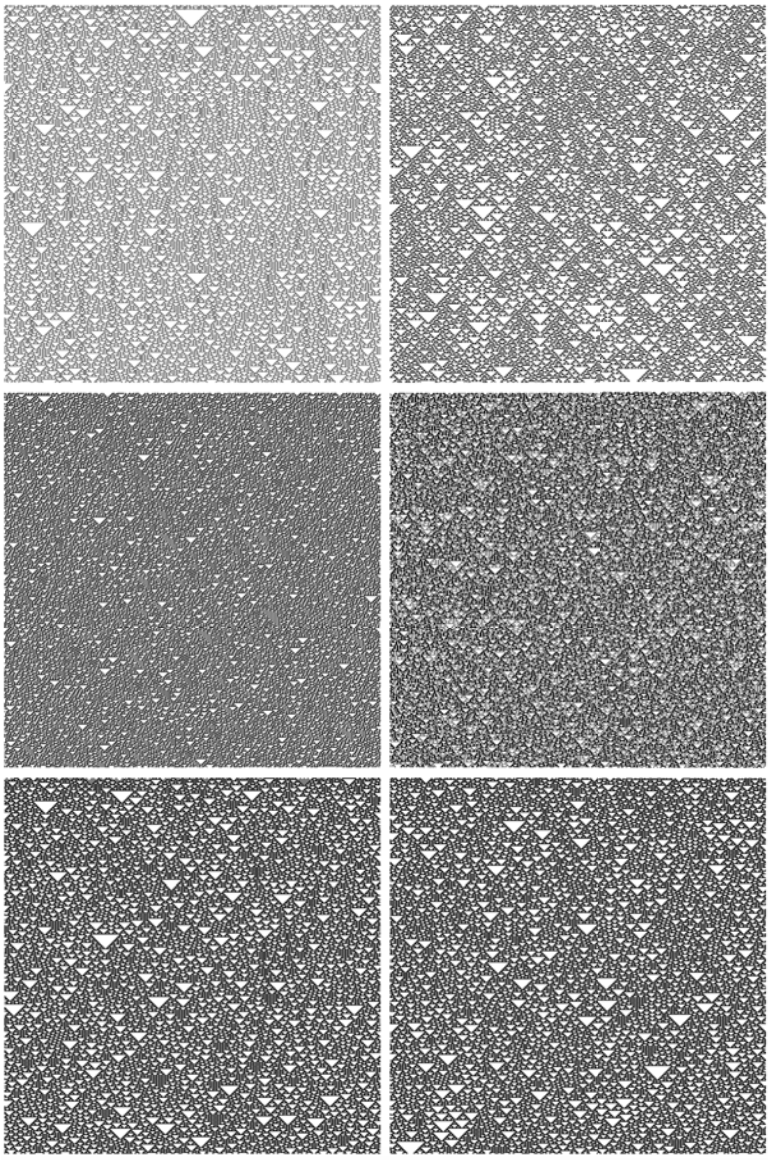


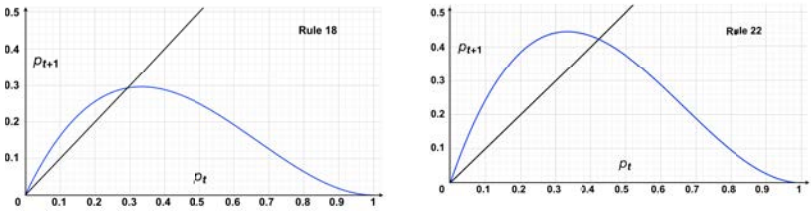
Figure 25. Evolutions in a ring of 400 cells, 400 generations from initial density $d_0 = 0.50$. Left to right, top to bottom: rules 18, 22, 30, 90, 122, 126 in ergodic regime. All patterns have the Hausdorff dimension $\log_2(3)$ of equation (11).

Referring again to the mean field curves in Section 2 now displayed for rules 18 and 22 in Figure 26, we observe that the rule 18 curve reaches its stable fixed point $p_{t+1} = p_t$ when crossing the identity at

$p_t \approx 0.293$, whereas for rule 22, the fixed point is achieved at $p_t \approx 0.423$, whence the discrepancies between densities in ergodic regime, observable in Figure 25. Moreover, the rule 18 curve shows a slope $f'_{\mathrm{R18}}(0) = 2$, whereas rule 22 shows a slope $f'_{\mathrm{R22}}(0) = 3$ at the origin. That comes from the fact that these rules induce from a single source their following evolution:

- (R18): $0^*10^* \to 0^*(101)0^*$
- (R22): $0^*10^* \to 0^*(111)0^*$

and that their density ratio at time steps $2^p - 1$ ($p > 0$) remains $2 \big/ 3$.



(a) (b)

Figure 26. (a) Rule 18 ($p_{t+1} = 2p_t q_t^2$) and (b) rule 22 ($p_{t+1} = 3p_t q_t^2$) with their mean field curves.

The Sierpiński gasket [42] appears in a wide variety of situations [43]. The binomial coefficients can be arranged to form Pascal's triangle and Pascal's triangle turns into a Sierpiński gasket with coefficients modulo two. It may turn into something like a natural tree by some diffeomorphism. This tree is embeddable into the two-dimensional diffusion graphs embedded into the triangulate lattice [44, 45]: its vertex dust forms the Sierpiński gasket patterns. The Sierpiński gasket is often known as a Banach fixed point from some contractive affine transformation into three elements.

But the most fascinating is the formation of patterns from random initial distributions of pigmentations on certain varieties of seashells [34, 46]. This phenomenon can be explained from the Gierer–Meinhardt reaction-diffusion model of the activator–inhibitor type, arising in various situations of pattern formation in morphogenesis [47, 48].

7. Discussion

7.1 Chaos or Determinism?

Despite the fact that rule 22 is based on simple and determined interactions, the behavior generated by such a system is visibly complex

and seems to be nondeterministic. The rule's behavior test chaos 0–1 [49] (classifier returning value near 1 if series is chaotic and near to 0 if deterministic) was used to determine whether behavior is chaotic/random or deterministic. It is applied directly to the time series data and does not require phase space reconstruction. These two kinds of behavior are significantly different. Physically, randomness has a stochastic nature, while deterministic chaos is generated by even a simple system that does not contain any source of randomness, as commonly understood by the public. If executed on a PC, then algorithms simulating such behavior generate only pseudorandom/chaotic behavior, and series generated in such a way are essentially deterministic and periodic but with very long period. The period is usually long enough to simulate randomness/chaos.

The term *chaos* covers a rather broad class of phenomena whose behavior may seem erratic, chaotic at first glance. Chaos has been observed in many systems (including evolutionary ones) and, in the last few years, it has been also used to replace pseudorandom number generators (PRNGs) in evolutionary algorithms (EAs). Let us mention, for example, research papers like [50] (a comprehensive overview of mutual intersection between EAs and chaos is discussed in this paper), one of the first uses of chaos inside EAs [51–53] discussing the using of deterministic chaos inside a particle swarm algorithm instead of PRNGs, [54, 55] investigating relations between chaos and randomness, or the latest ones [56] and [57–59], using chaos with EAs in applications, among others.

Additional research joining deterministic chaos and PRNGs has been done, for example, in [54]. The possibility of generation of random or pseudorandom numbers by use of the ultra-weak multidimensional coupling of one-dimensional dynamical systems is discussed there. Another paper [60] deeply investigates a logistic map as a possible PRNG and compares it with contemporary PRNGs. A comparison of logistic map results is made with conventional methods of generating pseudorandom numbers. The approach is used to determine the number, delay and period of the orbits of the logistic map at varying degrees of precision (3 to 23 bits). The logistic map we are using here was also used in [61], like a chaos-based, true random number generator embedded in reconfigurable switched-capacitor hardware. Another paper [62] proposed an algorithm for generating a PRNG, which is called a couple map lattice based on discrete chaotic iteration and combined the couple map lattice and chaotic iteration.

In [63] the authors exploit interesting properties of chaotic systems to design a random bit generator, called CCCBG, in which two chaotic systems are cross-coupled with each other. For evaluation of the bit streams generated by the CCCBG, the four basic tests are performed: monobit test, serial test, autocorrelation and Poker test. Also

the most stringent tests of randomness: the NIST suite tests have been used. A new binary stream–cipher algorithm based on dual one-dimensional chaotic maps is proposed in [55] with statistical properties showing that the sequence is of high randomness. Similar studies are also done in [51, 64–66]. For a long time, various PRNGs were used inside evolutionary algorithms. During the past few years, deterministic chaos systems (DCHS) have been used instead of PRNGs. As was demonstrated in [52, 53], very often the performance of EAs using DCHS is better or fully comparable with EAs using PRNGs. See, for example, [52].

The chaos test 0–1 [49] has already been successfully used on various tasks—for example, on experimental data from a bipolar motor [67], behavior of the cutting process [68], real experimental time series of laser droplet generation process [69]—and validated by applying it to typical nonlinear dynamic systems, including a fractional-order dynamic system [70], among others.

The same approach was used here. Test 0–1 was used on different series in order to verify and test the nature of rule 22. Figure 27 visualizes the results of our experiments. For evaluation, 2000 rule 22 behavior strings of length 2000 have been used. The same was repeated for the Mersenne twister random number generator (MTPRNG) [71] (Figure 27(c)), chaos generated by logistic equation (LE) (Figure 27(b)), and periodic series (PS): the sinus function generated from the randomly selected position (Figure 27(d)). As clearly visible, test 0–1 has classified rule 22, MTPRNG and LE as a nondeterministic series, while series based on periodic pattern are classified as deterministic. The random series were not distinguished from chaotic ones. This was probably caused by insufficient length (2000 is likely not enough) of series; however, this was not a focus of this experiment.

7.2 Memory Leads to Complexity

In this section, we show that ECA rule 22 with memory is *strongly chaotic* [72].

Conventional CAs are ahistoric (memoryless): that is, the new state of a cell depends on the neighborhood configuration solely at the preceding time step of φ. Thus, CA with *memory* (CAM) can be considered as an extension of the standard framework of CA where every cell x_i is allowed to remember some period of its previous evolution. A memory is based on the state and history of the system; thus, we design a memory function ϕ, as follows:

$$\phi\left(x_i^{t-\tau}, \ldots x_i^{t-1}, x_i^t\right) \to s_i,$$

such that $\tau < t$ determines the backward degree of memory and each cell $s_i \in \Sigma$ is a function of the series of states in cell x_i up to time step

https://doi.org/10.25088/ComplexSystems.28.2.125

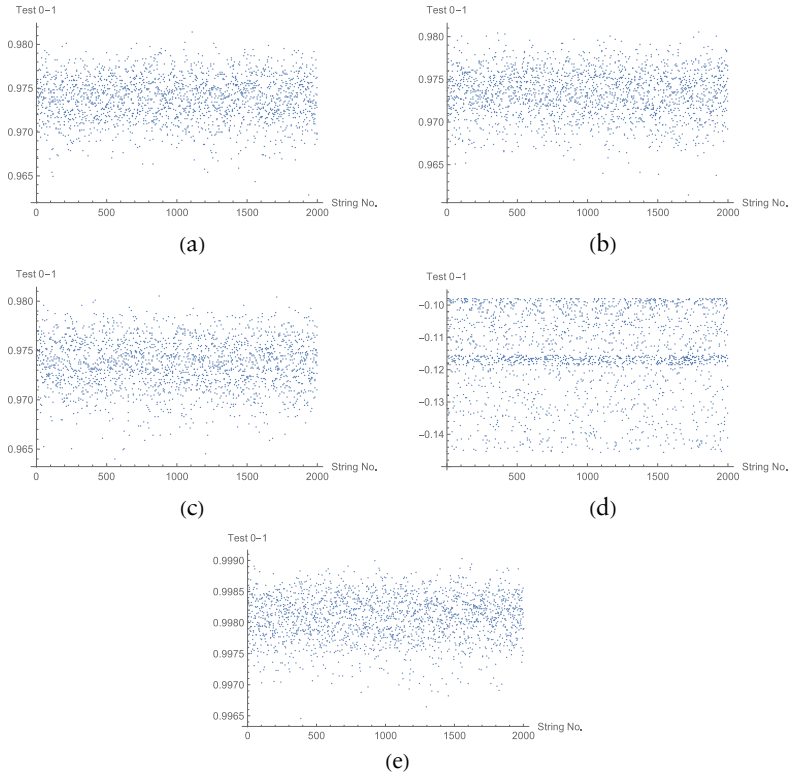


Figure 27. Test chaos 0–1 of (a) logistic equation, (b) Mersenne twister generator, (c) standard PRNG, (d) deterministic series (returned values are all around 0; i.e., evaluated process is deterministic), (e) rule 22. All results are concentrated around 1. Chaos based on test chaos 0–1 was proved.

$t - \tau$. To execute the evolution, we apply the original rule as follows:

$$\varphi\left(\ldots s_{i-1}^{t},\, s_{i}^{t},\, s_{i+1}^{t},\, \ldots\right) \to x_{i}^{t+1}.$$

In CAM, while the mapping φ remains unaltered, a historic memory of past iterations is retained by featuring each cell as a summary of its previous states; therefore cells *canalize* memory to the map φ. As an example, we can take the memory function ϕ as a *majority memory*: $\phi_{\text{maj}} \to s_i$, where in case of a tie given by $\Sigma_1 = \Sigma_0$ in ϕ, we shall take the last value x_i. So ϕ_{maj} represents the classic majority function for three variables on cells $\left(x_i^{t-\tau}, \ldots, x_i^{t-1}, x_i^{t}\right)$ and defines a temporal ring before calculating the next global configuration c. In case of a tie, the mapping is allowed to break a tie in favor of zero if $x_{\tau-1} = 0$, or in favor of one if $x_{\tau-1} = 1$.

The representation of an elementary cellular automaton with memory (ECAM) is given as follows:

$$\phi_{\mathrm{CAR}m:\tau} \tag{13}$$

where CAR represents the decimal notation of a particular ECA rule and m the kind of memory given with a specific value of τ.

Note that memory is as simple as any CA, and that the global behavior produced by the local rule is rather unpredictable; it can lead to emergent properties and so can be classed as complex. Memory functions were developed and extensively studied by Alonso-Sanz in [73]. Memory in ECAs has been studied, showing its potential to produce complex behavior from chaotic systems and beyond in [37, 72], and recently in [74] authors have included hybrid versions. Thus, we can conjecture that a memory function can produce complex behavior [72] as follows:

$$\phi(\varphi_{\mathrm{chaos}}) \rightarrow \text{complex}. \tag{14}$$

Eppstein [75] demonstrates that a CA class IV is a system where mobile self-localizations emerge. We can relate types of classes with memory functions in ECA rule 22 as:

$$\phi_{\mathrm{R22maj:3}} \Rightarrow \text{ chaos} \rightarrow \text{chaos} \tag{15}$$

$$\phi_{\mathrm{R22maj:4}} \Rightarrow \text{chaos} \rightarrow \text{complexity} \tag{16}$$

$$\phi_{\mathrm{R22maj:7}} \Rightarrow \text{chaos} \rightarrow \text{complexity} \tag{17}$$

$$\phi_{\mathrm{R22maj:8}} \Rightarrow \text{chaos} \rightarrow \text{complexity}. \tag{18}$$

ECA rule 22 with a memory function reveals complex behavior (Figure 28). This works by composing the original evolution rule with a second function affected with memory; this way the configuration with memory is evaluated with the original function.

7.3 Rareness and Unpredictability

The state transition function

$$\varphi\left(x_{i-r}^t, \ldots, x_i^t, \ldots x_{i+r}^t\right) \rightarrow x_i^{t+1}$$

can be rewritten as a Boolean formula with two XOR operations:

$$x_{i+r}^t = x_{i-r}^t \oplus x_i^t \oplus x_{i+r}^t.$$

The XOR gate is the most rare, most hard to find in natural nonlinear systems, Boolean gate. If gates g_1 and g_2 have occurrence frequencies $f(g_1)$ and $f(g_2)$, we say that gate g_1 is easier to develop or evolve than gate g_2: $g_1 \rhd g_2$ if $f(g_1) > f(g_2)$. The hierarchies of gates obtained using evolutionary techniques in liquid crystals [76], light-sensitive

https://doi.org/10.25088/ComplexSystems.28.2.125

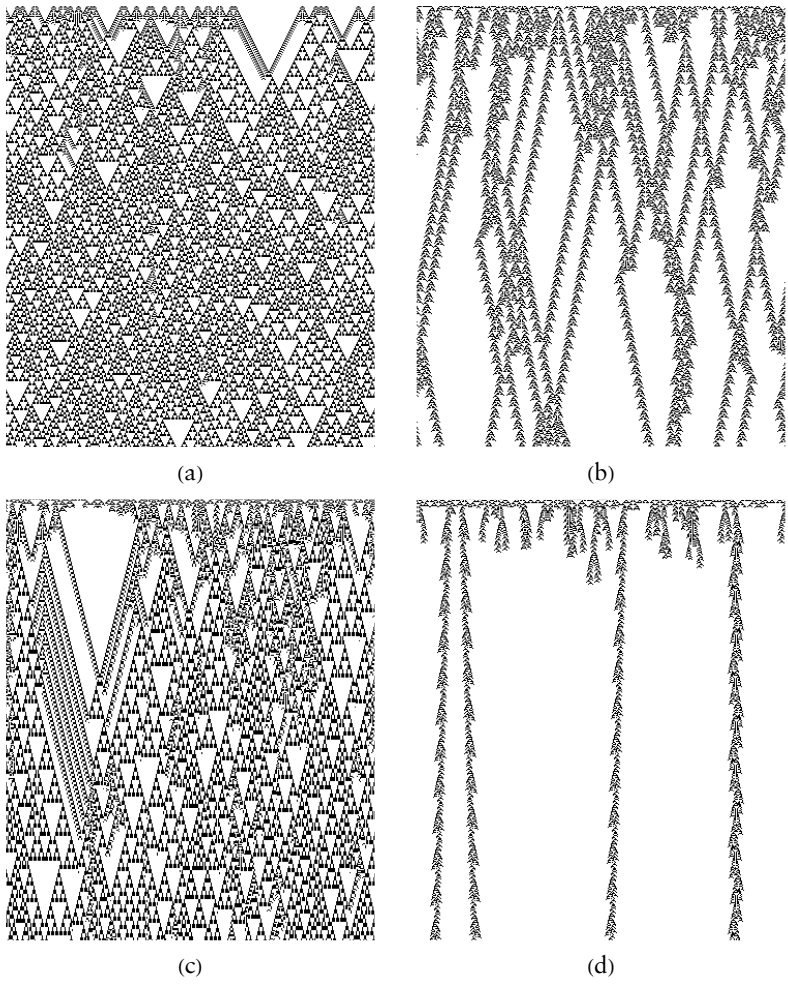


Figure 28. ECA rule 22 with a memory function reveals complex behavior. (a) Evolution of the function $\phi_{\text{R22maj:3}}$. (b) The function $\phi_{\text{R22maj:4}}$ (recently proven to be logically universal by simulating the Fredkin gate in [77, 78]). (c) The function $\phi_{\text{R22maj:7}}$ (a glider gun was discovered in this rule [72]). (d) The function $\phi_{\text{R22maj:8}}$ (particles with long period).

modification of Belousov–Zhabotinsky system [79], slime mold *Physarum polycephalum* [80] and protein molecules [81, 82]:

- Gates in liquid crystals: {OR, NOR} ▷ AND ▷ NOT ▷ NAND ▷ XOR
- Gates in Belousov–Zhabotinsky medium: AND ▷ NAND ▷ XOR
- Gates in cellular automata [83]: OR ▷ NOR ▷ AND ▷ NAND ▷ XOR

- Gates in *Physarum*: AND ▷ OR ▷ NAND ▷ NOR ▷ XOR ▷ XNOR
- Gates in protein molecules verotoxin and actin: AND ▷ OR ▷ AND-NOT ▷ XOR

The XOR gate is hard to find and spacetime dynamics of rule 22 automata are hard to predict. A strong link between the computational difficulty of a problem and its randomness was established by Yao [84]. His famous lemma, rephrased by Impagliazzo and Wigderson [85], can be seen in the framework of predictability of rule 22 ECA behavior:

> Fix a nonuniform model of computation (with certain closure properties) and a Boolean function $f:\{0, 1\}^n \to \{0, 1\}$. Assume that any algorithm in the model of a certain complexity has a significant probability of failure when predicting f on a randomly chosen instance x. Then any algorithm (of a slightly smaller complexity) that tries to guess the XOR $f(x_1) \oplus f(x_2) \oplus \cdots \oplus f(x_k)$ of k random instances $x_1, \ldots x_k$ won't do significantly better than a random coin toss.

Potential associations between dynamics in rule 22 ECA and the role of XOR functions in communication complexity [86, 87] could be explored in the future.

References

[1] S. Wolfram, "Statistical Mechanics of Cellular Automata," *Reviews of Modern Physics*, **55**(3), 1983 pp. 601–644. doi:10.1103/RevModPhys.55.601.

[2] S. Wolfram, *Cellular Automata and Complexity: Collected Papers*, Reading, MA: Addison-Wesley, 1994.

[3] S. Wolfram, "Twenty Problems in the Theory of Cellular Automata," *Physica Scripta*, **T9**, 1985 pp. 170–183. doi:10.1088/0031-8949/1985/t9/029.

[4] M. Redeker, A. Adamatzky and G. J. Martínez, "Expressiveness of Elementary Cellular Automata," *International Journal of Modern Physics C*, **24**(3), 2013 1350010. doi:10.1142/S0129183113500101.

[5] J. G. Zabolitzky, "Critical Properties of Rule 22 Elementary Cellular Automata," *Journal of Statistical Physics*, **50**(5–6), 1988 pp. 1255–1262. doi:10.1007/BF01019164.

[6] H. V. McIntosh, *One Dimensional Cellular Automata*, Frome, UK: Luniver Press, 2009.

[7] W. Jin, F. Chen and C. Yang, "Topological Chaos of Cellular Automata Rules," *2009 International Workshop on Chaos–Fractals Theories and Applications*, Shenyang, China, Piscataway, NJ: IEEE, 2009 pp. 216–220. doi:10.1109/IWCFTA.2009.52.

[8] H. A. Gutowitz and J. D. Victor, "Local Structure Theory in More than One Dimension," *Complex Systems*, **1**(1), 1987 pp. 57–68 complex-systems.com/pdf/01-1-5.pdf.

[9] H. V. McIntosh, "Wolfram's Class IV and a Good Life," *Physica D: Nonlinear Phenomena*, **45**(1–3), 1990 pp. 105–121. doi:10.1016/0167-2789(90)90177-Q.

[10] P. Grassberger, "Long-Range Effects in an Elementary Cellular Automaton," *Journal of Statistical Physics*, **45**(1–2), 1986 pp. 27–39. doi:10.1007/BF01033074.

[11] S. Sinha, "Phase Transitions in the Computational Complexity of 'Elementary' Cellular Automata," *Unifying Themes in Complex Systems* (A. A. Minai and Y. Bar-Yam, eds.), Springer, Berlin, Heidelberg: Springer, 2011 pp. 337–348. doi:10.1007/978-3-540-35866-4_33.

[12] H. Zenil and E. Villarreal-Zapata, "Asymptotic Behavior and Ratios of Complexity in Cellular Automata," *International Journal of Bifurcation and Chaos*, **23**(9), 2013 1350159. doi:10.1142/S0218127413501599.

[13] J. Gravner and D. Griffeath, "The One-Dimensional *Exactly 1* Cellular Automaton: Replication, Periodicity, and Chaos from Finite Seeds," *Journal of Statistical Physics*, **142**(1), 2011 pp. 168–200. doi:10.1007/s10955-010-0103-9.

[14] P. Halmos, *Lectures on Ergodic Theory*, Tokyo: Mathematical Society of Japan, 1956.

[15] A. Lesne, *Renormalization Methods: Critical Phenomena, Chaos, Fractal Structures* (L. Schneps, trans.), Chichester, UK: J. Wiley, 1998.

[16] D. Désérable, "Embedding Kadanoff's Scaling Picture into the Triangular Lattice," *Acta Physica Polonica B, Proceedings Supplement*, **4**(2), 2011 pp. 249–265. doi:10.5506/APhysPolBSupp.4.249.

[17] A. Wuensche and M. Lesser, *The Global Dynamics of Cellular Automata: An Atlas of Basin of Attraction Fields of One-Dimensional Cellular Automata*, Reading, MA: Addison-Wesley, 1992.

[18] A. Wuensche, *Exploring Discrete Dynamics*, 2nd ed., Bristol, UK: Luniver Press, 2016.

[19] H. V. McIntosh. "Ancestors: Commentaries on *The Global Dynamics of Cellular Automata* by A. Wuensche and M. Lesser." (Apr 29, 2019) delta.cs.cinvestav.mx/~mcintosh/cellularautomata/Working_Papers_files/global.pdf.

[20] G. J. Martínez, A. Adamatzky, B. Chen, F. Chen and J. C. Seck-Tuoh-Mora, "Simple Networks on Complex Cellular Automata: From de Bruijn Diagrams to Jump-Graphs," *Evolutionary Algorithms, Swarm Dynamics and Complex Networks* (I. Zelinka and G. Chen, eds.), Berlin, Heidelberg: Springer, 2017 pp. 241–264. doi:10.1007/978-3-662-55663-4_ 12.

[21] H. V. McIntosh, "Linear Cellular Automata." (Apr 29, 2019) matematicas.reduaz.mx/~cellularautomata/cellularautomata/Papers_files/lcau.pdf.

[22] S. Adak, S. Mukherjee and S. Das, "Do There Exist Non-linear Maximal Length Cellular Automata? A Study," in *Cellular Automata: 13th International Conference on Cellular Automata for Research and Industry (ACRI 2018)* (G. Mauri, S. El Yacoubi, A. Dennunzio, K. Nishinari and L. Manzoni, eds.), Como, Italy, Champ: Springer, 2018 pp. 289–297. doi:10.1007/978-3-319-99813-8_ 26.

[23] N. G. de Bruijn, "A Combinatorial Problem," *Proceedings of the Section of Sciences of the Koninklijke Nederlandse Akademie van Wetenschappen te Amsterdam*, **49**(7), 1946 pp. 758–764. pure.tue.nl/ws/files/4442708/597473.pdf.

[24] S. W. Golomb, *Shift Register Sequences* (portions coauthored by L. R. Welch, R. M. Goldstein and A. W. Hales), San Francisco: Holden-Day, 1967.

[25] B. H. Voorhees, *Computational Analysis of One-Dimensional Cellular Automata*, River Edge, NJ: World Scientific, 1996.

[26] H. V. McIntosh. "Linear Cellular Automata via de Bruijn Diagrams." (Apr 29, 2019) delta.cs.cinvestav.mx/~mcintosh/cellularautomata/Papers_files/debruijn.pdf.

[27] J. E. Hopcroft and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, Reading, MA: Addison-Wesley, 1979.

[28] J. von Neumann, *Theory of Self-Reproducing Automata* (A. W. Burks, ed.), Urbana, IL: University of Illinois Press, 1966.

[29] A. P. Goucher, "Universal Computation and Construction in GoL Cellular Automata," *Game of Life Cellular Automata* (A. Adamatzky, ed.) Springer, 2010 pp. 505–517. doi:10.1007/978-1-84996-217-9_ 25.

[30] S. Amoroso and G. Cooper, "The Garden-of-Eden Theorem for Finite Configurations," *Proceedings of the American Mathematical Society*, **26**(1), 1970 pp. 158–164. doi:10.1090/S0002-9939-1970-0276007-5.

[31] J. C. Seck-Tuoh-Mora, J. Medina-Marín, N. Hernández-Romero, G. J. Martínez and I. Barragán-Vite, "Welch Sets for Random Generation and Representation of Reversible One-Dimensional Cellular Automata," *Information Sciences*, **382–383**, 2017 pp. 81–95. doi:10.1016/j.ins.2016.12.009.

[32] D. Eppstein. "Fractals." *The Geometry Junkyard* (blog). (Apr 29, 2019) www.ics.uci.edu/~eppstein/junkyard/fractal.html.

[33] G. Chen and X. Dong, *From Chaos to Order: Methodologies, Perspectives, and Applications*, River Edge, NJ: World Scientific, 1998.

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

[35] S. Wolfram, "Computation Theory of Cellular Automata," *Communications in Mathematical Physics*, **96**(1), 1984 pp. 15–57. doi:10.1007/BF01217347.

[36] W. Li and N. H. Packard, "The Structure of the Elementary Cellular Automata Rule Space," *Complex Systems*, **4**(3), 1990 pp. 281–297. complex-systems.com/pdf/04-3-3.pdf.

[37] G. J. Martínez, A. Adamatzky, R. Alonso–Sanz and J. C. Seck Tuoh Mora, "Complex Dynamics Emerging in Rule 30 with Majority Memory," *Complex Systems*, **18**(3), 2009 pp. 345–365. wpmedia.wolfram.com/uploads/sites/13/2019/03/18-3-5.pdf.

[38] G. J. Martínez, A. Adamatzky and H. V. McIntosh, "Complete Characterization of Structure of Rule 54," *Complex Systems*, **23**(3), 2014 pp. 259–293. wpmedia.wolfram.com/uploads/sites/13/2018/12/23-3-4.pdf.

[39] O. Martin, A. M. Odlyzko, S. Wolfram, "Algebraic Properties of Cellular Automata," *Communications in Mathematical Physics*, **93**(2), 1984 pp. 219–258. doi:10.1007/BF01223745.

[40] G. J. Martínez, A. Adamatzky, J. C. Seck Tuoh Mora and R. Alonso-Sanz, "How to Make Dull Cellular Automata Complex by Adding Memory: Rule 126 Case Study," *Complexity*, **15**(6), 2010 pp. 34–49. doi:10.1002/cplx.20311.

[41] S. Bandini, A. Bonomi and G. Vizzari, "An Analysis of Different Types and Effects of Asynchronicity in Cellular Automata Update Schemes," *Natural Computing*, **11**(2), 2012 pp. 277–287. doi:10.1007/s11047-012-9310-4.

[42] W. Sierpiński, "O krzywej, której każdy punkt jest punktem rozgalezienia," *Prace Mat.–Fiz.*, **27**(1), 1916 pp. 77–86.

[43] M. F. Barnsley, *Fractals Everywhere*, Boston: Academic Press, 1988.

[44] D. Désérable, "A Family of Cayley Graphs on the Hexavalent Grid," *Discrete Applied Mathematics*, **93**(2–3), 1999 pp. 169–189. doi:10.1016/S0166-218X(99)00106-7.

[45] D. Désérable, "Systolic Dissemination in the Arrowhead Family," *Cellular Automata: 11th International Conference on Cellular Automata for Research and Industry (ACRI 2014)* (J. Was, G. Ch. Sirakoulis, and S. Bandini, eds.), Krakow, Poland, New York: Springer, 2014 pp. 75–86. doi:10.1007/978-3-319-11520-7_ 9.

[46] H. Meinhardt, *The Algorithmic Beauty of Seashells*, Berlin, Heidelberg: Springer, 2003.

[47] A. Gierer and H. Meinhardt, "A Theory of Biological Pattern Formation," *Kybernetik*, **12**(1), 1972 pp. 30–39. doi:10.1007/BF00289234.

[48] A. M. Turing, "The Chemical Basis of Morphogenesis," *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, **237**(641), 1952 pp. 37–72. doi:10.1098/rstb.1952.0012.

[49] G. A. Gottwald and I. Melbourne, "On the Implementation of the 0–1 Test for Chaos," *SIAM Journal on Applied Dynamical Systems*, **8**(1), 2009 pp. 129–145. doi:10.1137/080718851.

[50] I. Zelinka, S. Celikovsky, H. Richter and G. Chen (eds.), *Evolutionary Algorithms and Chaotic Systems*, Berlin: Springer, 2010.

[51] R. Caponetto, L. Fortuna, S. Fazzino and M. G. Xibilia, "Chaotic Sequences to Improve the Performance of Evolutionary Algorithms," *IEEE Transactions on Evolutionary Computation*, **7**(3), 2003 pp. 289–304. doi:10.1109/TEVC.2003.810069.

[52] M. Pluhacek, R. Senkerik, D. Davendra, Z. K. Oplatkova and I. Zelinka, "On the Behavior and Performance of Chaos Driven PSO Algorithm with Inertia Weight," *Computers & Mathematics with Applications*, **66**(2), 2013 pp. 122–134. doi:10.1016/j.camwa.2013.01.016.

[53] M. Pluháček, V. Budíková, R. Senkerik, Z. Oplatková and I. Zelinka, "On the Performance of Enhanced PSO Algorithm with Lozi Chaotic Map: An Initial Study," in *MENDEL 2012*. Brno: Brno University of Technology, 2012 pp. 40–45. publikace.k.utb.cz/handle/10563/1004708.

[54] R. Lozi, "Emergence of Randomness from Chaos," *International Journal of Bifurcation and Chaos*, **22**(2), 2012 1250021. doi:10.1142/S0218127412500216.

[55] X.-Y. Wang and L. Yang, "Design of Pseudo-Random Bit Generator Based on Chaotic Maps," *International Journal of Modern Physics B*, **26**(32), 2012 1250208. doi:10.1142/S0217979212502086.

[56] Y. Sun, L. Zhang and X. Gu, "A Hybrid Co-evolutionary Cultural Algorithm Based on Particle Swarm Optimization for Solving Global Optimization Problems," *Neurocomputing*, **98**, 2012 pp. 76–89. doi:10.1016/j.neucom.2011.08.043.

[57] W-Ch. Hong, Y. Dong, W. Y. Zhang, L.-Y. Chen and B. K. Panigrahi, "Cyclic Electric Load Forecasting by Seasonal SVR with Chaotic Genetic Algorithm," *International Journal of Electrical Power & Energy Systems*, **44**(1) 2013 pp. 604–614. doi:10.1016/j.ijepes.2012.08.010.

[58] R. Senkerik, D. Davendra, I. Zelinka, Z. Oplatkova and M. Pluhacek, "Optimization of the Batch Reactor by Means of Chaos Driven Differential Evolution," *Soft Computing Models in Industrial and Environmental Applications*, (V. Snášel, A. Abraham and E. Corchado, eds.), Berlin, Heidelberg: Springer, 2013 pp. 93–102. doi:10.1007/978-3-642-32922-7_10.

[59] D. Davendra, I. Zelinka and R. Senkerik, "Chaos Driven Evolutionary Algorithms for the Task of PID Control," *Computers & Mathematics with Applications*, **60**(4), 2010 pp. 1088–1104. doi:10.1016/j.camwa.2010.03.066.

[60] K. J. Persohn and R. J. Povinelli, "Analyzing Logistic Map Pseudorandom Number Generators for Periodicity Induced by Finite Precision Floating-Point Representation," *Chaos, Solitons & Fractals*, **45**(3), 2012 pp. 238–245. doi:10.1016/j.chaos.2011.12.006.

[61] M. Drutarovsky and P. Galajda, "A Robust Chaos-Based True Random Number Generator Embedded in Reconfigurable Switched-Capacitor Hardware," *2007 17th International Conference Radioelektronika*, Brno, Czech Republic, Piscataway, NJ: IEEE, 2007 pp. 29–34. doi:10.1109/RADIOELEK.2007.371423.

[62] X.-Y. Wang and X. Qin, "A New Pseudo-random Number Generator Based on CML and Chaotic Iteration," *Nonlinear Dynamics*, **70**(2), 2012 pp. 1589–1592. doi:10.1007/s11071-012-0558-0.

[63] N. K. Pareek, V. Patidar and K. K. Sud, "A Random Bit Generator Using Chaotic Maps," *International Journal of Network Security*, **10**(1), 2010 pp. 32–38. doi:10.6633/IJNS.201001.10%281%29.05.

[64] M. Bucolo, R. Caponetto, L. Fortuna, M. Frasca and A. Rizzo, "Does Chaos Work Better than Noise?," *IEEE Circuits and Systems Magazine*, **2**(3), 2002 pp. 4–19. doi:10.1109/MCAS.2002.1167624.

[65] H. P. Hu, L. F. Liu and N. D. Ding, "Pseudorandom Sequence Generator Based on the Chen Chaotic System," *Computer Physics Communications*, **184**(3), 2013 pp. 765–768. doi:10.1016/j.cpc.2012.11.017.

[66] A. Pluchino, A. Rapisarda and C. Tsallis, "Noise, Synchrony, and Correlations at the Edge of Chaos," *Physical Review E*, **87**(2), 2013 022910. doi:10.1103/PhysRevE.87.022910.

[67] I. Falconer, G. A. Gottwald, I. Melbourne and K. Wormnes, "Application of the 0–1 Test for Chaos to Experimental Data," *SIAM Journal on Applied Dynamical Systems*, **6**(2), 2007 pp. 395–402. doi:10.1137/060672571.

[68] G. Litak, A. Syta and M. Wiercigroch, "Identification of Chaos in a Cutting Process by the 0–1 Test," *Chaos, Solitons & Fractals*, **40**(5), 2009 pp. 2095–2101. doi:10.1016/j.chaos.2007.09.093.

[69] B. Krese and E. Govekar, "Nonlinear Analysis of Laser Droplet Generation by Means of 0–1 Test for Chaos," *Nonlinear Dynamics*, **67**(3), 2012 pp. 2101–2109. doi:10.1007/s11071-011-0132-1.

[70] S. Ke-Hui, L. Xuan and Z. Cong-Xu, "The 0–1 Test Algorithm for Chaos and Its Applications," *Chinese Physics B*, **19**(11), 2010 110510. doi:10.1088/1674-1056/19/11/110510.

[71] M. Matsumoto and T. Nishimura, "Mersenne Twister: A 623-dimensionally Equidistributed Uniform Pseudo-random Number Generator," *ACM Transactions on Modeling and Computer Simulation*, **8**(1), 1998 pp. 3–30. doi:10.1145/272991.272995.

[72] G. J. Martínez, A. Adamatzky and R. Alonso-Sanz, "Designing Complex Dynamics in Cellular Automata with Memory," *International Journal of Bifurcation and Chaos*, **23**(10), 2013 1330035. doi:10.1142/S0218127413300358.

[73] R. Alonso-Sanz, *Cellular Automata with Memory*, Philadelphia: Old City Publishing, 2008.

[74] B. Chen, F. Chen and G. J. Martínez, "Glider Collisions in Hybrid Cellular Automaton with Memory Rule (43,74)," *International Journal of Bifurcation and Chaos*, **27**(6), 2017 1750082. doi:10.1142/S0218127417500821.

[75] D. Eppstein. "Wolfram's Classification of Cellular Automata." (May 1, 2019) www.ics.uci.edu/~eppstein/ca/wolfram.html.

[76] S. L. Harding and J. F. Miller, "Evolution In Materio: Evolving Logic Gates in Liquid Crystal," *International Journal of Unconventional Computing*, 3(4), 2007 pp. 243–257. www.oldcitypublishing.com/journals/ijuc-home/ijuc-issue-contents/ijuc-volume-3-number-4-2007/ijuc-3-4-p-243-257.

[77] G. J. Martínez and K. Morita, "Conservative Computing in a One-Dimensional Cellular Automaton with Memory," *Journal of Cellular Automata*, **13**(4), 2018 pp. 325–346. www.oldcitypublishing.com/journals/jca-home/jca-issue-contents/jca-volume-13-number-4-2018/jca-13-4-p-325-346.

[78] G. J. Martínez, A. Adamatzky and K. Morita, "Logical Gates via Gliders Collisions," *Reversibility and Universality* (A. Adamatzky, ed.), Cham: Springer, 2018 pp. 199–220. link.springer.com/chapter/10.1007/978-3-319-73216-9_ 9.

[79] R. Toth, C. Stone, A. Adamatzky, B. de Lacey Costello and L. Bull, "Dynamic Control and Information Processing in the Belousov–Zhabotinsky Reaction Using a Coevolutionary Algorithm," *The Journal of Chemical Physics*, **129**(18), 2008 184708. doi:10.1063/1.2932252.

[80] S. Harding, J. Koutnik, J. Schmidhuber and A. Adamatzky, "Discovering Boolean Gates in Slime Mould," *Inspired by Nature* (S. Stepney and A. Adamatzky, eds.), Cham: Springer, 2018 pp. 323–337. doi:10.1007/978-3-319-67997-6_15.

[81] A. Adamatzky, "Computing in Verotoxin," *ChemPhysChem*, **18**(13), 2017 pp. 1822–1830. doi:10.1002/cphc.201700477.

[82] A. Adamatzky, "Logical Gates in Actin Monomer," *Scientific Reports*, **7**(1), 2017 11755. doi:10.1038/s41598-017-11333-7.

[83] A. Adamatzky and L. Bull, "Are Complex Systems Hard to Evolve?," *Complexity*, **14**(6), 2009 pp. 15–20. doi:10.1002/cplx.20269.

[84] A. C. Yao, "Theory and Application of Trapdoor Functions," *23rd Annual Symposium on Foundations of Computer Science (SFCS 1982)*, Piscataway, NJ: IEEE, 1982 pp. 80–91. doi:10.1109/SFCS.1982.45.

[85] R. Impagliazzo and A. Wigderson, "P = BPP if E Requires Exponential Circuits: Derandomizing the XOR Lemma," *Proceedings of the 29th Annual Symposium on Theory of Computing*, El Paso, TX, New York: ACM, 1997 pp. 220–229. doi:10.1145/258533.258590.

[86] Z. Zhang and Y. Shi, "Communication Complexities of Symmetric XOR Functions," *Quantum Information & Computation*, **9**(3), 2009 pp. 255–263.

[87] A. Montanaro and T. Osborne, "On the Communication Complexity of XOR Functions," arxiv.org/abs/0909.3392.