Impact of (a)synchronism on ECA: towards a new classification[1]

Isabel Donoso-Leiva[a,c], Eric Goles[a], Martín Ríos-Wilson[a], Sylvain Sené[b,c]

[a] *Facultad de Ingeniería y Ciencias, Universidad Adolfo Ibáñez , Chile*
[b] *Université publique, Marseille, France*
[c] *Aix Marseille Univ, CNRS, LIS, Marseille, France*

Abstract

In this paper, we study the effect of (a)synchronism on the dynamics of elementary cellular automata. Within the framework of our study, we choose five distinct update schemes, selected from the family of periodic update modes: parallel, sequential, block sequential, block parallel, and local clocks. Our main measure of complexity is the maximum period of the limit cycles in the dynamics of each rule. In this context, we present a classification of the ECA rule landscape. We classified most elementary rules into three distinct regimes: constant, linear, and superpolynomial. Surprisingly, while some rules exhibit more complex behavior under a broader class of update schemes, others show similar behavior across all the considered update schemes. Although we are able to derive upper and lower bounds for the maximum period of the limit cycles in most cases, the analysis of some rules remains open. To complement the study of the 88 elementary rules, we introduce a numerical simulation framework based on two main measurements: the energy and density of the configurations. In this context, we observe that some rules exhibit significant variability depending on the update scheme, while others remain stable, confirming what was observed as a result of the classification obtained in the theoretical analysis.

Keywords: Elementary Cellular Automata, Asynchronism, Classification, Asymptotic Complexity

[1] Partial results of this work were presented at the international conference LATIN 2024 (Latin American Theoretical Informatics) [1]

1. Introduction

Cellular automata are collections of discrete state entities (the cells) arranged over a grid that interact with each other according to a local rule over discrete time. They were first introduced by Ulam and Von Neumann in the 1940s [2] following from automata networks which were defined by McCulloch and Pitts in the same decade [3]. From there, fundamental results have been obtained such as the introduction of the retroaction cycle theorem [4], the self organizing behavior [5], the Turing universality of the model itself [6, 7], the structure of the space of elementary cellular automata [8], the undecidability of all nontrivial properties of limit sets of cellular automata [9], hierarchy of certain cellular automata [10]. While CA are simple models, they are able to exhibit great complexity which ranges from biology and health sciences, as a representational model of disease spreading [11] and its impact[12], to social models [13, 14], to parallel and distributed computation [15, 16], to physics [17].

Despite major theoretical contributions having provided since the 1980s a better comprehension of these objects [18, 7] from computational and behavioral standpoints, understanding their sensitivity to (a)synchronism remains an open question on which any advance could have deep implications in computer science (around the themes of synchronous versus asynchronous computation and processing [19, 20]) and in systems biology (around the temporal organization of genetic expression [21, 22]). In this context, numerous studies have been published by considering distinct settings of the concept of synchronism/asynchronism, i.e. by defining update modes which govern the way automata update their state over time. For instance, Kauffman [23] and Thomas [24] modeled biological regulation networks with automata networks, while (a)synchronism sensitivity has been studied *per se* according to deterministic and non-deterministic semantics in [25, 26, 27, 28] for a family of automata networks and in [29, 30, 31] for cellular automata subject to stochastic semantics. In particular, Fatès proposed a classification of elementary cellular automata under stochastic fully asynchronous update modes [30].

Our Contribution

Since the aim of this paper is to increase the knowledge on asynchronism sensitivity, elementary cellular automata were chosen because they are a family of restricted and "simple" cellular automata, which has been well

studied [32, 33, 34]. By studying ECA we can put the focus on the periodic update modes (from the classical parallel update mode to a family of more general ones known as local clocks [35]) and their impact over the dynamics. Here, we approach the subject with ideas derived from [36, 37] and pay attention to the influence of update modes on the resulting asymptotic dynamical behavior, in particular in terms of the maximal period of limit cycles. Additionally, for rules whose behavior is too complex to analyze directly, we have ran computational experiments that used measures of density and energy to study their asymptotic dynamical behavior. For these experiments we have implemented a number of update modes per family, for three different orders of magnitude of ring sizes.

In this paper, we highlight formally that the choice of the update mode can have a deep influence on the dynamics of systems. In particular, two specific elementary cellular automata rules, namely rules 2 and 184 (the traffic rule) as defined by Wolfram's codification, are studied here. These rules were chosen because the results obtained require proofs which serve as example of how to obtain similar results for other rules that are mentioned in their respective section. In particular, the traffic rule also emphasizes sensitivity to (a)synchronism.

Note that both rules belong to the Wolfram's class II [32], which means that, according to computational observations, these cellular automata evolve asymptotically towards a "set of separated simple stable or periodic structures". Since our (a)synchronism sensitivity measure consists in limit cycles maximal periods, Wolfram's class II is naturally the most pertinent one in this context.

Secondly, we have run computational experiments over rules 90 and 150 (of Wolfram's class III) and rules 54 and 110 (of Wolfram's class IV) to discuss the influence of different update modes over rules whose dynamics is already known to be too chaotic (class III) or too complex (class IV) to show jumps in asymptotic complexity as a result of the influence of the different update modes. Instead, we show the changes (or lack thereof) in their dynamics by way of showing whether there are changes in the evolution of density and energy.

Structure of the Paper

In Section 2, the main definitions and notations are formalized. The emphasizing of elementary cellular automata (a)synchronism sensitivity is

presented in the first part of Section 3 through demonstrations that set an upper-bound for the limit cycle periods of rules 2 and 184 depending on distinct families of periodic update modes. In the second part, experimental results for rules 90, 150, 54 and 110 under different update modes are presented, using measures of density and energy. The paper ends with Section 4 in which we discuss some perspectives of this work.

2. Preliminaries

General notations. Let $[\![n]\!] = \{0, \ldots, n-1\}$, let $\mathbb{B} = \{0, 1\}$, and let x_i denote the i-th component of vector $x \in \mathbb{B}^n$. Given a vector $x \in \mathbb{B}^n$, we can denote it classically as $(x_0, \ldots, x_{n-1})$ or as the word $x_0 \ldots x_{n-1}$ if it eases the reading.

2.1. Cellular automata and elementary cellular automata

In broad terms, a *cellular automata* (CA) of size n is a collection of n cells that we will represent by the set $[\![n]\!]$, each of which will have a state within the alphabet Q. The cells interact with each other over discrete time according to a rule which is defined by the state of the cells in their respective neighborhood. A configuration x is an element of Q^n. Formally, a CA is a tuple $(\mathbb{Z}^d, Q, N, F)$ composed of

- $\mathbb{Z}^d$ the $d-$dimensional cellular space, also known as *grid*, which can be finite or infinite,
- Q the finite set of states that a cell can have, called the *alphabet*,
- N the neighborhood, which associates a cell of the grid with its neighbors,
- F the global rule, which is defined by local functions $f_i^{N_i} \to Q$, where f_i is the $i-$th component of F and N_i is the neighborhood of cell i.

An *elementary cellular automata* (ECA) is a kind of CA such that the grid is $1-$dimensional, the alphabet is $\mathbb{B}$, the local function f_i is the same for the entire grid and it depends on the state of the cells $i-1$, i itself, and $i+1$. From where it follows that there are $2^{2^3} = 256$ distinct ECA, of which there are 88 that are not equivalent through symmetry [8].

As previously stated, the grid can be either infinite or finite. In this paper we have chosen to work with finite ECA, which effectively mean that the grid will be viewed as a $1-$dimensional torus, that is, a ring of integers modulo

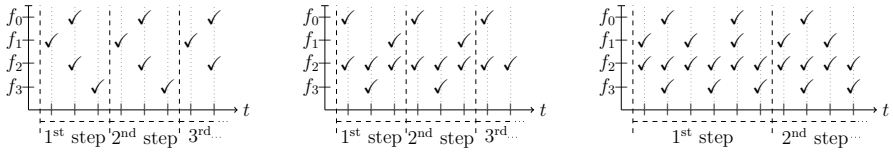

Figure 1: Illustration of the execution over time of local transition functions for a CA of size 4 according to (left) $\mu_{\text{BS}} = (\{1\}, \{0, 2\}, \{3\})$, (center) $\mu_{\text{BP}} = \{(2), (1, 0, 3)\}$, and (right) $\mu_{\text{LC}} = ((3, 2, 1, 2), (1, 0, 0, 1))$. The ✓ symbols indicate the moments at which the automata update their states; the vertical dashed lines separate periodical time steps from each other.

n: $\mathbb{Z}/n\mathbb{Z}$. As such, the neighborhood of cell 0 will be $\{n-1, 0, 1\}$ and that of cell $n-1$ will be $\{n-2, n-1, 0\}$.

Having defined ECA, we will now define the functions that govern the order in which we will execute the local functions on each cell, and the families in which we can group them.

2.2. Update modes

When we determine the order in which the cells of a CA are updated, we are defining an update mode (also called update schedule or scheme). We will work with deterministic and periodic update modes, because their structure allow for a kind analysis that stochastic update modes. In broad terms, given a CA over a grid of size n, we can define a *deterministic* (resp. *periodic*) *update mode* as an infinite (resp. finite) sequence $\mu = (B_k)_{k\in\mathbb{N}}$ (resp. $\mu = (B_0, \ldots, B_{p-1})$), where B_i is a subset of $[\![n]\!]$ for all $i \in \mathbb{N}$ (resp. for all $i \in [\![p]\!]$). An update mode μ can also be defined as a function $\mu^\star : \mathbb{N} \to \mathcal{P}([\![n]\!])$, which associates each time to a subset of $[\![n]\!]$ such that $\mu^\star(t)$ corresponds to the cells that are updated at each time t. Furthermore, when μ is periodic, there exists $p \in \mathbb{N}$ such that for all $t \in \mathbb{N}$, $\mu^\star(t+p) = \mu^\star(t)$.

We will consider three families of update modes: the block-sequential ones [18], the block-parallel ones [38, 39] and the local clocks ones [40]. Updates induced by each of them over time are depicted in Figure 1.

A *block-sequential update mode* $\mu_{\text{BS}} = (B_0, \ldots, B_{p-1})$ is an ordered partition of $[\![n]\!]$, with B_i a subset of $[\![n]\!]$ for all i in $[\![p]\!]$. Informally, μ_{BS} defines an update mode of period p separating $[\![n]\!]$ into p disjoint blocks so that all cells of a same block update their state in at the same time while the blocks are iterated in series. The other way of considering μ_{BS} is: $\forall t \in \mathbb{N}, \mu^\star_{\text{BS}}(t) = B_{t \mod p}$.

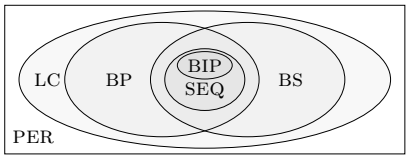

Figure 2: Order of inclusion of the defined families of periodic update modes, where PER stands for "periodic". Note that PAR is not included since it has cardinal one.

A *block-parallel update mode* $\mu_{\text{BP}} = \{S_0, \dots, S_{s-1}\}$ is a partitioned order of $[\![n]\!]$, with $S_j = (i_{j,k})_{0 \leq k \leq |S_j|-1}$ a sequence of $[\![n]\!]$ for all j in $[\![s]\!]$. Informally, μ_{BP} separates $[\![n]\!]$ into s disjoint subsequences so that all cells of a same subsequence update their state in series while the subsequences are iterated in parallel. Note that there exists a natural way to convert μ_{BP} into a sequence of blocks of period $p = \text{lcm}(|S_0|, \dots, |S_{s-1}|)$. It suffices to define function φ as: $\varphi(\mu_{\text{BP}}) = (B_\ell)_{\ell \in [\![p]\!]}$ with $B_\ell = \{i_{j,\ell \mod |S_j|} \mid j \in [\![s]\!]\}$. The other way of considering μ_{BP} is: $\forall j \in [\![s]\!], \forall k \in [\![|S_j|]\!]$, $i_{j,k} \in \mu^\star_{\text{BP}}(t) \iff k = t \mod |S_j|$.

A *local clocks update mode* $\mu_{\text{LC}} = (P, \Delta)$, with $P = (p_0, \dots, p_{n-1})$ and $\Delta = (\delta_0, \dots, \delta_{n-1})$, is an update mode such that each cell i of $[\![n]\!]$ is associated with a period $p_i \in \mathbb{N}$ and an initial shift $\delta_i \in [\![p_i]\!]$ such that $i \in \mu^\star_{\text{LC}}(t) \iff t = \delta_i \mod p_i$, with $t \in \mathbb{N}$.

We will now introduce three particular cases or subfamilies of these three latter update mode families. The *parallel update mode* $\mu_{\text{PAR}} = ([\![n]\!])$ in which every cell updates its state at each time step, such that $\forall t \in \mathbb{N}, \mu^\star_{\text{PAR}}(t) = [\![n]\!]$. A *bipartite update mode* $\mu_{\text{BIP}} = (B_0, B_1)$ is a block-sequential update mode composed of two blocks such that the cells in a same block do not act on each other (notice that if the grid is finite and the boundary condition is periodic, then this definition induces that such update modes are necessarily associated with grids of even size, and that there are exactly two bipartite update modes, depending on if the even numbered cells are updated first or second.) A *sequential update mode* $\mu_{\text{SEQ}} = (\phi([\![n]\!]))$, where $\phi([\![n]\!]) = \{i_0\}, \dots, \{i_{n-1}\}$ is a permutation of $[\![n]\!]$, makes one and only one cell updates its state at each time step so that all cells have updated their state after n time steps depending on the order induced by ϕ. All these update modes follow the order of inclusion pictured in Figure 2.

Since we are focusing on periodic update modes, we need to differentiate two kinds of time steps. A *substep* is a time step at which a subset of cells change their states. A *step* is the composition of substeps having occurred

over a period p, that is, when all blocks have been updated.

2.3. Dynamical systems

An ECA A, with $f \in [\![256]\!]$, along with an update mode μ, denoted by the pair (f, μ), define a *discrete dynamical system*. (f, M) denotes by extension any dynamical system related to A under the considered update mode families, with $M \in \{\text{PAR}, \text{BIP}, \text{SEQ}, \text{BP}, \text{BS}, \text{LC}, \text{PER}\}$.

Let f be an ECA which is applied over a grid of size n and let μ be a periodic update mode represented as a periodical sequence of subsets of $[\![n]\!]$ such that $\mu = (B_0, ..., B_{p-1})$. Let $F = (f, \mu)$ be a tuple composed of the global function f that goes from $\mathbb{B}^n$ to itself defining the dynamical system related to ECA f applied over a grid of size n and update mode μ. Let $x \in \mathbb{B}^n$ a configuration of F.
The *trajectory* of x is the infinite path $\mathscr{T}(x) \triangleq x^0 = x \to x^1 = F(x) \to \cdots \to x^t = F^t(x) \to \cdots$, where $t \in \mathbb{N}$ and

$$\begin{aligned} F(x) =& f_{B_{p-1}} \circ \cdots \circ f_{B_0}, \\ & \text{with } f_{B_k}(x)_i = \begin{cases} f_i(x) & \text{if } i \in B_k, \\ x_i & \text{otherwise} \end{cases}, \forall k \in [\![p]\!], \forall i \in [\![n]\!], \text{ and} \\ F^t(x) =& \underbrace{F \circ \cdots \circ F}_{t \text{ times}}(x). \end{aligned}$$

The *orbit* of x is the set $\mathcal{O}(x)$ composed of all the configurations which belongs to $\mathscr{T}(x)$. Since f is defined over a grid of finite size and the boundary condition is periodic, the temporal evolution of x governed by the successive applications of F leads it to eventually enter into a *limit phase*, i.e. a cyclic subpath $\mathscr{C}(x)$ of $\mathscr{T}(x)$ such that $\forall y = F^k(x) \in \mathscr{C}(x), \exists t \in \mathbb{N}, F^t(y) = y$, with $k \in \mathbb{N}$. $\mathscr{T}(x)$ is then separated into two phases, the limit phase and the *transient phase* which corresponds to the finite subpath $x \to \cdots \to x^\ell$ of length ℓ such that $\forall i \in [\![\ell + 1]\!], \nexists t \in \mathbb{N}, x^{i+t} = x^i$. The *limit set* of x is the set of configurations belonging to $\mathscr{C}(x)$. In the context of ECA, it is convenient to represent trajectories by *space-time diagrams* which give a visual aspect of the latter, as illustrated in Figure 3.

From these definitions, we derive that F can be represented as a graph $\mathscr{G}_F = (\mathbb{B}^n, T)$, where $(x, y) \in T \subseteq \mathbb{B}^n \times \mathbb{B}^n \iff y = F(x)$. In this graph, which is classically called a *transition graph*, the non-cyclic (resp. cyclic) paths represent the transient (resp. limit) phases of F. More precisely, the

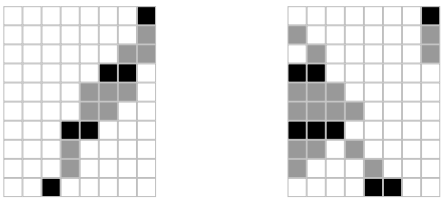

Figure 3: Space-time diagrams (time going downward) representing the 3 first (periodical) steps of the evolution of configuration $x = (0,0,0,0,0,0,0,1)$ of dynamical systems (left) $(2, \mu_{\text{BS}})$, and (right) $(184, \mu_{\text{BP}})$, where $\mu_{\text{BS}} = (\{0,4\},\{1,6\},\{2,3,5,7\})$, and $\mu_{\text{BP}} = \{(0),(3),(2,1,6),(5,4,7)\}$, The configurations obtained at each step are depicted by lines with cells at state 1 in black. Lines with cells at state 1 in gray represent the configurations obtained at substeps.

cycles of $\mathscr{G}_F$ are the *limit cycles* of F. When a limit cycle is of length 1, we call it a *fixed point*. Furthermore, if the fixed point is such that all the cells of the configuration has the same state, then we call it an *homogeneous fixed point.*

For some of the proofs, we will need to use the following specific notations. Let $x \in \mathbb{B}^n$ be a configuration and $[i,j] \subseteq [\![n]\!]$ be a subset of cells. We denote by $x_{[i,j]}$ the projection of x on $[i,j]$. Since we work on ECA, such a projection defines a sub-configuration and can be of three kinds:

- $i < j$ and $x_{[i,j]} = (x_i, x_{i+1}, \ldots, x_{j-1}, x_j)$,
- or $i = j$ and $x_{[i,j]} = (x_i)$,
- or $i > j$ and $x_{[i,j]} = (x_i, x_{i+1}, \ldots, x_n, x_0, \ldots, x_{j-1}, x_j)$.

Thus, given $x \in \mathbb{B}^n$ and $i \in [\![n]\!]$, an ECA f can be rewritten as

$$f(x) = (f(x_{[n-1,1]}), f(x_{[0,2]}), \ldots, f(x_{[i,i+2]}), \ldots, f(x_{[n-3,n-1]}), f(x_{[n-2,0]}).$$

Abusing notations, the word $u \in \mathbb{B}^k$ is called a *wall* for a given dynamical system if for all $a, b \in \mathbb{B}$, $f(aub) = u$, and in this work we will assume that walls are of size 2, i.e. $k = 2$, unless otherwise stated. Moreover, a word u is an *absolute wall* (resp. a *relative wall*) for an ECA rule if it is a wall for all update modes (resp. a strict subset of update modes). We say that a rule F can *dynamically create new walls* if there exits a time $t \in \mathbb{N}$ and an initial configuration $x^0 \in \mathbb{B}^n$ such that $x^t (= F^t(x^0))$ has a higher number of walls than x^0. Finally, we say that a configuration x is an *isle* of $1's$ (resp. an

isle of $0's$) if there exists an interval $I = [a, b] \subseteq [\![n]\!]$ such that $x_i = 1$ (resp. $x_i = 0$) for all $i \in I$ and $x_i = 0$ (resp. $x_i = 1$) otherwise.

3. Results

In this section we will present the results of our research. In the first part we show the theoretical results, where we work towards a theoretical classification for ECA. And in the second part we offer some experimental results where we compare the dynamics of energy and density under different update modes for some of the rules whose dynamics we have not yet been able to classify.

3.1. Theoretical Results

The theoretical results are based on the study of complexity through the length of the maximal limit cycle that can be reached for each pair of rule and update mode. We have named it the asymptotic complexity of the dynamics.

We have theoretical results for 67 of the 88 non-equivalent ECA. While we have been able to classify all rules belonging to class I, there are rules in class II (25, 37, 57, 58, 62, 74, 154) whose limit cycles are still unbounded. Note that while most rules belonging to class III have behaviors too chaotic to describe theoretically, we have found one, rule 18, whose limit cycles are bounded for all BS update modes.

3.1.1. $\Theta(1)$

There are 25 ECA that always reach limit cycles whose length does not depend on the length of the ring. These are divided into two groups:

- Rules that always reach fixed points, regardless of the length of the ring or the update mode under which they are applied. These rules are: 0, 4, 8, 12, 44, 72, 76, 78, 128, 132, 136, 140, 164, 200 and 204.
- Rules that can reach limit cycles of length 2, 3 or 6. The length of the cycle does not depend on the length of the ring, and changing the update mode cannot increase the maximum length of the cycle. These rules are: 5, 13, 28, 29, 32, 36, 51, 77, 160 and 232.

We have omitted most proofs for these claims, since they are the result of simple analyses of the underlying rules. Nevertheless, we will now include two of the theorems to give an idea of how the results are obtained.

Rule	111	110	101	100	011	010	001	000
72	0	1	0	0	1	0	0	0
76	0	1	0	0	1	1	0	0

Table 1: Definition of rules 72 and 76

Theorem 1. *Rules* 72 *and* 76 *always reach fixed points.*

Proof. We know that $x = 0^n$ is a fixed point, so let $x = 0^{a_1}1^{b_1}\dots 0^{a_k}1^{b_k}$, since $f_r(111) = 0$ for $r \in \{72, 76\}$, meaning that the configuration $y = 1^n$ will become $f(y) = 0^{a_1}1^{b_1}\dots 0^{a_k}1^{b_k}$.
Since $f_r(011) = f_r(110) = 1$ and $f_r(001) = f_r(100) = 0$, for all $r \in \{72, 76\}$, groups composed of exactly two 1's are walls, meaning that once the word 0110 appears it will not change.
This also means that groups of ones larger than two cannot increase in size. In any other case, groups of 1's of size at least three will decrease in size in each iteration until:

- case 1: the group is left with exactly two ones, in which case it becomes a wall and remains unchanging.
- case 2: for rule 72, $f_{72}(010) = 0$, isolated 1's will disappear, with which the resulting fixed points can be written as $0^{a'_1}110^{a'_2}\dots 110^{a'_{k'}}11$.
- case 3: for rule 76, $f_{76}(010) = 1$, isolated 1's will be preserved, meaning that the resulting fixed points can be written as $0^{a'_1}1^{b'_1}0^{a'_2}1^{b'_2}\dots 0^{a'_{k'}}1^{b'_{k'}}$, with $b_i \in \{1, 2\}, i \in \{1, \dots, k'\}$.

□

Theorem 2. *Rule* 232 *always reaches limit cycles of length at most 2.*

Proof. We know that $x = 0^n$ and $x = 1^n$ are fixed points, so let $x = 0^{a_1}1^{b_1}\dots 0^{a_k}1^{b_k}$, with $a_1 + b_1 + \dots + a_k + b_k = n$.
Since $f_{232}(010) = 0$ and $f_{232}(101) = 1$, if there are isolated 0's or 1's, then they will disappear.
If $b_i > 1$ with $i \in \{1, \dots, k\}$, those groups of ones will not decrease in size, since $f_{232}(111) = f_{232}(011) = f_{232}(110) = 1$. Similarly, if $a_i > 1$ with $i \in \{1, \dots, k\}$, those groups of zeros will not decrease in size, since $f_{232}(000) = f_{232}(001) = f_{232}(100) = 0$. This leads to a configuration that can

Rule	111	110	101	100	011	010	001	000
232	1	1	1	0	1	0	0	0

Table 2: Definition of rule 232

be written as $x' = 0^{a'_1}1^{b'_1}\dots 0^{a'_k}1^{b'_k}$, with $a_i, b_i > 1$ for all $i \in \{1, \dots, k\}$.
There is a special case for the parallel update schedule, where if $x = (01)^{\frac{n}{2}}$ and $y = (10)^{\frac{n}{2}}$, then $f_{232}(x) = y$ and $f_{232}(y) = x$, which is when we find limit cycles of length 2.
However, even if the initial configuration is $x = (01)^{\frac{n}{2}}$, we only need one cell to be updated before or after its neighbors to give rise to a domino effect where we eventually reach a fixed point.
Indeed, if $x = (01)^{\frac{n}{2}-1}11$, this means that the wall 111 has appeared, and note that one of the cells on either side of the wall must be equal to 0, and once that cell is updated, the number of consecutive 1's will increase. So after less than n iterations, the configuration will have reached the fixed point 1^n. Analogous for $x = (01)^{\frac{n}{2}-1}00$, where the configuration reaches the fixed point 0^n.
If there is more than one cell that updates before or after its neighbors, then the configuration $x = (01)^{\frac{n}{2}}$ is destroyed faster and it reaches fixed points that can be written as $x' = 0^{a'_1}1^{b'_1}\dots 0^{a'_k}1^{b'_k}$, same as before. □

3.1.2. $\mathcal{O}(n)$

We have found 33 ECA such that the longest cycle we can reach is directly proportional to the length of the ring, regardless of the update mode. These rules are: 2, 3, 6, 7, 10, 11, 14, 15, 18, 19, 23, 24, 26, 27, 33, 34, 35, 38, 40, 42, 43, 46, 50, 94, 104, 130, 134, 138, 142, 162, 168, 170 and 172.

There are three rules that behave differently to the rest. These rules are 40, 168 and 172, and they are only able to reach limit cycles of length $\mathcal{O}(n)$ when all cells are updated simultaneously, that is, with the parallel update mode. If there is just one cell that is updated before or after the rest, or that is updated more than once per iteration, the rule is no longer able to reach these limit cycles and instead reaches fixed points for all update modes. In simpler terms, rules 40, 168 and 172 always reach fixed points for all update modes *except* the parallel one.

We have chosen to show the proof for Rule 2, since most of them follow a similar logic.

Rule	111	110	101	100	011	010	001	000
2	0	0	0	0	0	0	1	0

Theorem 3. $(2, \text{BS})$ *of size n can reach limit cycles of size $\mathcal{O}(n)$.*

Proof. Since the update mode is block-sequential, we can distinguish two sets of cells depending on the order in which a cell is updated with regard to its right-hand neighbor: one where $\mu_i \leq \mu_{i+1}$ and one where $\mu_i > \mu_{i+1}$. Let $L = \{\ell_j\}_{j=1}^{N} = \{i \in \{0, \ldots, n-1\} \mid \mu_{\ell_i} \leq \mu_{\ell_i+1}\}$ the set of cells that are updated before or at the same time as their right-hand neighbor.

Let $x = 1^k 0^{n-k}$ be an initial configuration that has an isle of 1's and all the state of all other cells is equal to 0.

We can find four cases:

Case 1: The first cell of the isle updates *after* the one to its left the isle, and the last cell updates *before* the one to its right, denoted $\mu_{p-1} \leq \mu_p$ and $\mu_q \leq \mu_{q+1}$. This means that $p-1, q \in L$. Let $p-1 = \ell_j$ and $q = \ell_k$, with $k = j + s$ (the reasoning is the same if $k = j+1$).

		ℓ_{j-1}			ℓ_j	p	$\ldots$	ℓ_{k-1}		$\ldots$	ℓ_k			ℓ_{k+1}			
0	$\ldots$	0	0	$\ldots$	0	1	$\ldots$	1	1	$\ldots$	1	0	$\ldots$	0	0	$\ldots$	0

Since all cells between ℓ_{j-1} and ℓ_j must be updated from right to left, and $f_2(001) = 1$, we will gain 1's to the left until we reach ℓ_{j-1}. Analogously, because $f_2(110) = 0$ and $f_2(111) = 0$, on the right side of the isle we will lose 1's until we reach ℓ_j.

		ℓ_{j-1}			ℓ_j	p	$\ldots$	ℓ_{k-1}		$\ldots$	ℓ_k			ℓ_{k+1}			
0	$\ldots$	0	0	$\ldots$	0	1	$\ldots$	1	1	$\ldots$	1	0	$\ldots$	0	0	$\ldots$	0
0	$\ldots$	0	1	$\ldots$	1	0	$\ldots$	0	0	$\ldots$	0	0	$\ldots$	0	0	$\ldots$	0

After the first iteration is completed, the isle is again delimited by cells that belong to L, so the process repeats. It is easy to see that after $N < n$ iterations the isle will return to its position from ℓ_{j-1} to ℓ_j.

Case 2: First and last cell of the isle update *after* the cells outside the isle, denoted $\mu_{p-1} \leq \mu_p$ and $\mu_q > \mu_{q+1}$. This means that $p - 1 \in L$.

		ℓ_{j-1}			ℓ_j	p	$\ldots$	q	$q+1$	$\ldots$	ℓ_{j+1}			
0	$\ldots$	0	0	$\ldots$	0	1	$\ldots$	1	0	$\ldots$	0	0	$\ldots$	0

Since cell l_j must update before p does, and $f_2(001) = 1$,then we gains ones to the left, and since all cells between ℓ_{j-1} and ℓ_j update from right to left, we continue to gain ones until we reach ℓ_{j-1}. On the other hand, since all cells between a ℓ_j and q also update from right to left, we will lose ones at the right side of the isle, until we reach ℓ_j. En effect, the isle moves.

		ℓ_{j-1}			ℓ_j	p	$\ldots$	q	$q+1$	$\ldots$	ℓ_{j+1}			
0	$\ldots$	0	0	$\ldots$	0	1	$\ldots$	1	0	$\ldots$	0	0	$\ldots$	0
0	$\ldots$	0	1	$\ldots$	1	0	$\ldots$	0	0	$\ldots$	0	0	$\ldots$	0

Note that if there are one or more cells that belong to L inside the aisle, then we will only lose ones starting from q until the first cell in L that it reaches.

From here it develops following the analysis from Case 1.

Case 3: The first and last cells of the isle are updated *before* the one to its left and right, respectively, denoted $\mu_{p-1} > \mu_p$ and $\mu_q < \mu_{q+1}$. This means that $q \in L$, let $q = \ell_k$.

					$p-1$	p	$p+1$	$\ldots$	ℓ_{k-1}		$\ldots$	ℓ_k	$q+1$		
0	$\ldots$	0	0	$\ldots$	0	1	1	$\ldots$	1	1	$\ldots$	1	0	$\ldots$	0

We know from Case 2 that after the first iteration we will lose 1's from l_k to l_{k-1}. On the other hand, since there could be a cell between p and ℓ_k that is in L, that cell would update before its left side neighbor, meaning that it would change its state. But we know that the first cell of the isle of ones has to update before the one outside, so the neighborhood will be 011 and $f_2(011) = 0$, so we also lose 1's to both sides of the isle.

					$p-1$	p	$p+1$	$\ldots$	ℓ_{k-1}		$\ldots$	ℓ_k	$q+1$		
0	$\ldots$	0	0	$\ldots$	0	1	1	$\ldots$	1	1	$\ldots$	1	0	$\ldots$	0
0	$\ldots$	0	0	$\ldots$	0	0	0	$\ldots$	0	0	$\ldots$	0	0	$\ldots$	0

This case always leads to a fixed point.

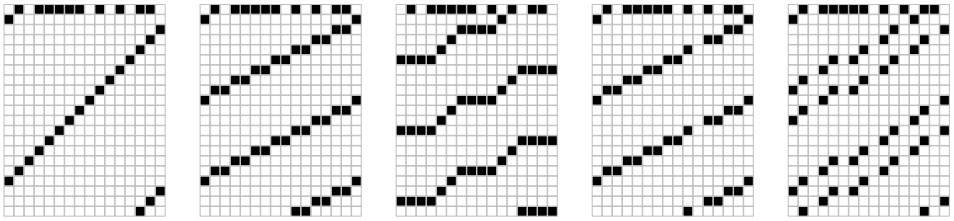

Figure 4: Space-time diagrams (time going downward) of configuration 0101111101010110 following rule 2 depending on: (a) the parallel update mode $\mu_{\text{PAR}} = (\llbracket 16 \rrbracket)$, (b) the bipartite update mode $\mu_{\text{BIP}} = (\{i \in \llbracket 16 \rrbracket \mid i \bmod 2 \equiv 0\}, \{i \in \llbracket 16 \rrbracket \mid i \bmod 2 \equiv 1\})$, (c) the block-sequential update mode $\mu_{\text{BS}} = (\{3, 9, 15\}, \{2, 4, 8, 10, 14\}, \{1, 5, 7, 11, 13\}, \{0, 6, 12\})$, (d) the block-parallel update mode $\mu_{\text{BP}} = \{(0, 1), (2, 3), (5), (6, 7), (8, 9), (10), (4, 11), (12, 13), (14, 15)\}$, and (e) the local clocks update mode $\mu_{\text{LC}} = (P = (4, 4, 2, 4, 4, 2, 4, 4, 2, 4, 4, 2, 4, 4, 2, 4), \Delta = (0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0))$.

Case 4: First cell of the isle updates *before* the one outside and last cell of the isle updates *after* the one outside of it, denoted $\mu_{p-1} > \mu_p$ and $\mu_q > \mu_{q+1}$. This means that $p, q \notin L$.

	$p-1$	p	$p+1$	$\dots$	ℓ_j	$\dots$		ℓ_k		$\dots$	q	$q+1$		
$\dots$	0	1	1	$\dots$	1	1	$\dots$	1	1	$\dots$	1	0	$\dots$	0

We know from Case 2 that we will 1's to the left because $\mu_{p-1} > \mu_p$, which means that this case always leads to Case 3 (which leads to fixed points):

	$p-1$	p	$p+1$	$\dots$	ℓ_j	$\dots$		ℓ_k		$\dots$	q	$q+1$		
$\dots$	0	1	1	$\dots$	1	1	$\dots$	1	1	$\dots$	1	0	$\dots$	0
$\dots$	0	0	0	$\dots$	0	0	$\dots$	0	0	$\dots$	0	0	$\dots$	0

Note that if there are more than one isle of 1's in the initial configuration, and since isles always move from right to left (going from ℓ_j to ℓ_{j-1}), they cannot interact with each other under Rule 2, from where we have the result. □

Corollary 1. $(2, \text{PAR})$ *and* $(2, \text{BIP})$ *of size* n *always reach limit cycles of size at most* $\mathcal{O}(n)$.

Rule	Parallel	Sequential	Block-Sequential	Block-Parallel	Local Clocks
1	$\Theta(1)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$
9	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$
56	$\mathcal{O}(n)$	1	$\mathcal{O}(n)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$
73	?	$\Theta\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$
108	$\Theta(1)$	$\Theta\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$
152	$\mathcal{O}(n)$	1	$\mathcal{O}(n)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$
156	$\Theta(1)$	$\Theta\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$
178	$\Theta(1)$	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$
184	$\mathcal{O}(n)$	1	$\mathcal{O}(n)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$	$\Omega\left(2^{(\sqrt{n\log(n)})}\right)$

Table 3: Longest cycle for each rule and update mode

Proof. Parallel and Bipartite update modes are special cases of block-sequential update modes, meaning that they inherit the result. □

Corollary 2. $(2, \text{BP})$ *and* $(2, \text{LC})$ *of size* n *can reach limit cycles of size* $\Omega(n)$.

Proof. From Theorem 3. We have shown that we can find update modes in BS that reach limit cycles of size $\mathcal{O}(n)$, and these update modes also correspond to BP and LC. But that does not mean that it is the longest limit cycle that can be reached. □

3.1.3. $\Omega\left(2^{(\sqrt{n\log(n)})}\right)$

In this section we will show rules for which we have found update modes such that limit cycles of superpolynomial length can be reached. Table 3 shows the order of longest cycle one can reach for each combination of rule and update mode, except for rule $(73, \text{PAR})$ that has limit cycles whose length we cannot describe.

Rule	111	110	101	100	011	010	001	000
56	0	0	1	1	1	0	0	0
152	1	0	0	1	1	0	0	0
184	1	0	1	1	1	0	0	0

Note that rules 108 and 156 have the earliest jump, going from limit cycles of constant length with parallel update mode to superpolynomial limit cycles with sequential update modes, unlike the rest of the rules on this list that are only able to reach superpolynomial limit cycles under BP update modes.

Rules 1 and 178 for which the increase is more gradual, going from constant, to $\mathcal{O}(n)$ to $\Omega\left(2^{(\sqrt{n\log(n)})}\right)$. Then we have rule 9, that starts with limit cycles $\mathcal{O}(n)$ and then increases to $\Omega\left(2^{(\sqrt{n\log(n)})}\right)$.

Finally, we have rules 56, 152 and 184 which are the only ones whose complexity decreases for sequential update modes and then increases for block-sequential and then increases again for block-parallel.

To show how we have proven the results displayed on Table 3, we have selected rule 184 (also known as the *traffic rule*) to serve as an example. This choice was made because of its particular behavior (along with the one for rules 56 and 152, that follow the same proof), where the complexity decreases from PAR to SEQ before increasing twice; once for BS and then again for BP. Each of these complexities is proven in a different way and the rest of the proofs for the rest of the rules follow similar lines of reasoning.

Theorem 4. $(184, \text{SEQ})$ *can only reach homogeneous fixed points.*

Proof. By definition, with sequential update mode, it is not possible for two consecutive cells to be updated simultaneously. Let us consider an initial configuration such that it consist of an isle of ones surrounded by zeros: $y = 1^k 0^{n-k}$. Since $f_{184}(000) = 0$ and $f_{184}(111) = 1$, we can focus on what happens where zeros and ones meet. Without loss of generality, let us assume that the first one is on cell 0 and the last one is on cell $k-1$. We will proceed with a case-by-case analysis.

1. Case 1. The cells inside the isle update before the ones outside: $\mu_0 > \mu_1$ and $\mu_{k-1} < \mu_k$: since $f_{184}(011) = 1$ and $f_{184}(110) = 0$ we have that $y^1 = (1)^{k-\ell}(0)^{n-k+\ell}$, where ℓ will depend on the update mode. It will be given by the first cell going from right to left that updates *before*

0	1	...	$k-\ell$	$k-\ell+1$	...	$k-2$	$k-1$	k	$k+1$	...	n-1
1	1	...	1	1	...	1	1	0	0	...	0
1	1	...	1	0	...	0	0	0	0	...	0
$\vdots$											
0	0	...	0	0	...	0	0	0	0	...	0

Table 4: Representation of Cases 1 and 3: $\mu_0 > \mu_1$ or $\mu_0 < \mu_1$ and $\mu_{k-1} < \mu_k$.

0	1	...	k	$k+1$	...	$k+r$	$k+r+1$	...	n-1
1	1	...	1	1	...	0	0	...	0
1	1	...	1	1	...	1	0	...	0
$\vdots$									
1	1	...	1	1	...	1	1	...	1

Table 5: Representation of Cases 2 and 4: $\mu_0 > \mu_1$ or $\mu_0 < \mu_1$ and $\mu_{k-1} > \mu_k$.

the one on the right, meaning that $\mu_{k-\ell} < \mu_{k-\ell+1}$.
It is easy to see that on the second iteration the isle is still on the same case, from where we can conclude that the process repeats and the number of ones decreases until we reach the homogeneous fixed point 0^n.

2. Case 2. The cells inside the isle update before the ones outside: $\mu_0 < \mu_1$ and $\mu_{k-1} > \mu_k$. Since $f_{184}(001) = 0$, the state of the cells to the left of the isle remain unchanged. But, since $f_{184}(100) = 1$ the number of ones can increase to the right: $y^1 = 1^{k+r}0^{n-k-r}$, where r will be given by the first cell going from left to right that updates *after* the one to its right, meaning $\mu_{k+r} > \mu_{k+r+1}$. Its easy to see that for the next iteration the ends of the isle follow the same order of updating as it had to begin with, from where we can conclude that the isle will continue to increase its size until it reaches the homogeneous fixed point 1^n.

3. Case 3. $\mu_0 < \mu_1$ and $\mu_{k-1} < \mu_k$. Follows the same analysis as case 1. As it has been shown, there is no increasing or decreasing the number of ones at the left side of the isle, because $f_{184}(011) = 1$ and $f_{184}(001) = 0$. Which means that it will lose ones until it reaches the homogeneous

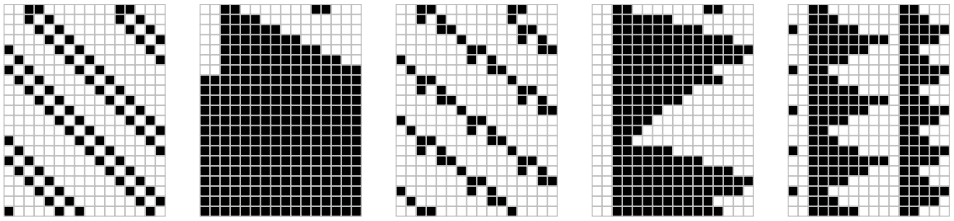

Figure 5: Space-time diagrams of configuration 0011000000011000 following rule 184 depending on: (a) the parallel update mode $\mu_{\text{PAR}} = (\llbracket 16 \rrbracket)$, (b) the bipartite update mode $\mu_{\text{BIP}} = (\{i \in \llbracket 16 \rrbracket \mid i \equiv 0 \mod 2\}, \{i \in \llbracket 16 \rrbracket \mid i \equiv 1 \mod 2\})$, (c) the block-sequential update mode $\mu_{\text{BS}} = (\{0, 1, 2, 5, 6, 7, 10, 11, 12, 15\}, \{3, 4, 8, 9, 13, 14\})$, (d) the block-parallel update mode $\mu_{\text{BP}} = \{(0, 3), (4, 5, 6, 7), (1, 8, 9, 10), (2, 14, 13, 15), (11, 12)\}$, (e) the local clocks update mode $\mu_{\text{LC}} = (P = (2, 4, 4, 2, 4, 4, 4, 4, 4, 2, 4, 4, 2, 4, 4, 4), \Delta = (0, 0, 0, 1, 0, 1, 2, 1, 3, 0, 0, 0, 1, 0, 1, 3))$.

fixed point 0^n.

4. Case 4. $\mu_0 > \mu_1$ and $\mu_{k-1} > \mu_k$. The analysis is identical to case 2, because the left side of the isle doesn't change its behavior regardless of if the cell inside the isle updates before or after the one outside. Meaning that it will gain ones until it reaches the homogeneous fixed point 1^n.

Now, if we generalize this by considering any initial configuration as isles of ones separated by zeros, we can see that every isle has to follow one of the four previous cases, from where we can conclude that isles that follow cases 1 and 3 will disappear, while the ones that follow cases 2 and 4 will grow in size.

Note that if an isle of type 2 reaches one of type 1 or 3 as it grows they will combine which will result in an isle of case 3. Similarly, if an isle of case 4 reaches one of types 1 or 3, the combination of them results in an isle of case 1.

From the previous analysis we can conclude that sequential update modes always lead to homogeneous fixed points. □

Theorem 5. $(184, \text{BS})$ *of size n has largest limit cycles of size $\mathcal{O}(n)$.*

Proof. Doing a similar analysis as the one done for theorem 4, is easy to see that $(184, \text{BS})$ can only reach (homogeneous) fixed points if there exist only one group of two consecutive cells that update on the same substep.

Let μ be a BS update mode such that there is one group of (at least) three consecutive cells that are updated simultaneously.
Let us consider an initial configuration $x = 10^{n-1}$, such that the only one is on one of the three cells that update on the same substep.
We will denote by e_1, e_2 the cells that update at the same substep than their right hand neighbor ($\mu_{e_1} = \mu_{e_2} = \mu_{e_2+1}$), the set $\{r_i\}_{i=1}^{j}$ will be the cells that are updated *after* their right hand neighbor ($\mu_{r_i} > \mu_{r_i+1}$) and the set $\{\ell_i\}_{i=1}^{k}$ will be the cells that are updated *before* their right hand neighbor ($\mu_{\ell_k} < \mu_{\ell_k+1}$).

We will proceed by cases, considering the order in which cells are updated outside our group.

First case: $r_j e_1 e_2 \ell_k$.
$\underline{t = 1}$ the only 1 moves once cell to the right because $f_{184}(100) = 1$ and $f_{184}(010) = 0$.
$\underline{t = 2}$ the 1 advances again, but since the next cell to the right is updated *before* the one on the right, we gain ones until the *first* cell such that $\mu_i > \mu_{i+1}$, which means that it is a cell in $\{r_i\}_{i=1}^{j}$. Let us call that cell r_1.
$\underline{t = 3}$ We continue to gain ones until the next cell to the right that belongs to $\{r_i\}_{i=1}^{j}$, which will be r_2. In this instance, we do not lose ones to the left, because $f_1 84(001) = 0$ and $f_{184}(011) = 1$.

		r_j	e_1	e_2	ℓ_k	...		r_1			r_2
0	...	0	1	0	0	0	...	0	0	...	0
0	...	0	0	1	0	0	...	0	0	...	0
0	...	0	0	0	1	1	...	1	0	...	0
0	...	0	0	0	1	1	...	1	1	...	1
⋮											
1	...	1	0	0	1	1	...	1	1	...	1
1	...	1	1	0	1	1	...	1	1	...	1

$\underline{t \in \{4, \ldots, j\}}$ Similar as to what occurs on $t = 3$, and because we are on this first case, the cell to the left of e_1 will have to be r_j.
$\underline{t = j + 1}$ s_1 changes its state to 1, and it is the only cell whose state can change.

Now, let us see what happens with one 0 surrounded by ones.

$\underline{t = 1}$ the only 0 moves once cell to the left.
$\underline{t = 2}$ the 0 advances again, but this time we gain 0's to the left until just *before* the *first* cell marked with an ℓ. Note that the cell marked with the ℓ has not changed yet.
$\underline{t = \{3, \ldots, k\}}$ Analogously, the number of zeros keeps increasing until it finally reaches ℓ_k, which leaves us with two surviving 1's.

		ℓ_1			r_j	s_1	s_2	ℓ_k			
1	…	1	1	…	1	1	1	0	1	…	1
1	…	1	1	…	1	1	0	1	1	…	1
1	…	1	0	…	0	0	1	1	1	…	1
⋮											
0	…	0	0	…	0	0	1	1	0	…	0
0	…	0	0	…	0	0	1	0	0	…	0

$\underline{t = k + 1}$ ℓ_k becomes 0, since $f_{110} = 0$, with which the only surviving 1 is on cell s_2, and we already know from the previous analysis what happens next.

It takes j steps to go from one 1 surrounded by zeros on the position s_2 to one 0 surrounded by ones, on that same position, and it takes j steps to return to the configuration we had on $t = 1$ on the first analysis, and because $j + k < n$, we have found limit cycles $\mathcal{O}(n)$.

Using the same reasoning, we can prove that a second case with $\ell_1 s_1 s_2 r_1$ follows the same behavior and it produces limit cycles of length $(j + 1) + (k + 1) \leq n$. Similarly, cases 3 ($r_j s_1 s_2 r_1$) and 4 ($\ell_1 s_1 s_2 \ell_k$) lead to limit cycles of length $(j + 1) + k < n$ and $j + (k + 1) < n$, resp. □

Remark 1. *Note that limit cycles in $\mathcal{O}(n)$ can also be found with block-sequential configurations with (at least) two groups of size two or more consecutive cells that update on the same substep.*

Lemma 6 is a fundamental part of all rules that reach limit cycles of length $\Omega\left(2^{\sqrt{n \log n}}\right)$: we need to be able to prove that there are walls. Once we find the walls, if we can prove that the behavior between walls is proportional to the distance between the walls, then we have proven that the length of the limit cycles is indeed $\Omega\left(2^{\sqrt{n \log n}}\right)$.

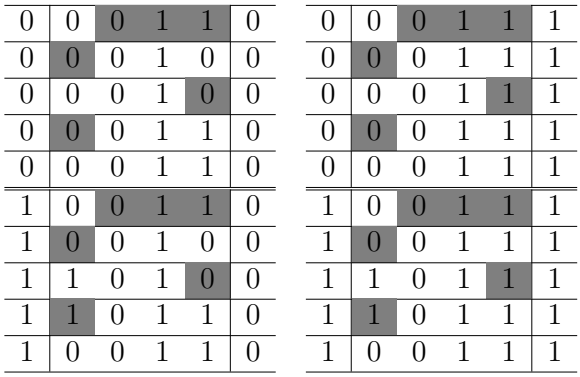

Figure 6: The four possible cases for the relative wall $w = 0011$.

Lemma 6. *$w = 0011$ is a relative wall for ECA rule 184.*

Proof. We proceed case by case as shown in Figure 6.

As long as the cells at the border of the w update twice before the ones around them are updated, this order of updating preserves the word $w = 0011$. □

Theorem 7. *$(184, \text{BP})$ of size n can reach limit cycles of size $\Omega\left(2^{\sqrt{n \log n}}\right)$.*

Proof. We need to start with an initial configuration that contains a wall $w = 0011$, and we need to prove that we can find an update mode in BP that preserves the relative wall.

The idea is to focus in what can happen between two walls, since by definition of wall the dynamics of two subconfigurations separated by a wall are independent from each other.

Let y be a subconfiguration of size $k + 8$ such that

$$y = w_1 y_4 y_5 \dots y_{k+3} w_2$$

such that for all $i \in \{i, \dots, k\}$, $y_i = 0$ (resp. $y'_i = 1$). And the block-parallel update mode

$$\begin{aligned} \mu_{\text{BP}} = \{&(y_3, y_0), (y_1, y_5, y_6, y_4), (y_2, y_7, y_8, y_9), (y_{k+5}, y_{10}, y_{11}, y_{12}), \\ &(y_{13}, y_{14}, y_{15}, y_{16}), \dots, (y_{k+6}, y_{k+1}, y_{k+2}, y_{k+3}), (y_{k+4}, y_{k+7})\} \end{aligned} \tag{1}$$

if $k \mod 4 \equiv 0$,

$$\begin{aligned} \mu_{\text{BP}} = \{&(y_3, y_0), (y_1, y_5, y_6, y_4), (y_2, y_7, y_8, y_9), (y_{k+5}, y_{10}, y_{11}, y_{12}), \\ &(y_{13}, y_{14}, y_{15}, y_{16}), \dots, (y_{k+6}, y_{k+1}, y_k, y_{k+3}), (y_{k+2}), (y_{k+4}, y_{k+7})\} \end{aligned} \tag{2}$$

if $k \mod 4 \equiv 1$,

$$\begin{aligned}\mu_{\text{BP}} =&\{(y_3, y_0), (y_1, y_5, y_6, y_4), (y_2, y_7, y_8, y_9), (y_{k+5}, y_{10}, y_{11}, y_{12}),\\ &(y_{13}, y_{14}, y_{15}, y_{16}), \ldots, (y_{k+6}, y_{k-1}, y_k, y_{k+3}), (y_{k+2}, y_{k+1}), (y_{k+4}, y_{k+7})\}\end{aligned} \tag{3}$$

if $k \mod 4 \equiv 2$,

$$\begin{aligned}\mu_{\text{BP}} =&\{(y_3, y_0), (y_1, y_5, y_6, y_4), (y_2, y_7, y_8, y_9), (y_{k+5}, y_{10}, y_{11}, y_{12}),\\ &(y_{13}, y_{14}, y_{15}, y_{16}), \ldots, (y_{k+6}, y_{k-1}, y_k, y_{k+3}),\\ &(y_{k-2}), (y_{k+2}, y_{k+1}), (y_{k+4}, y_{k+7})\}\end{aligned} \tag{4}$$

if $k \mod 4 \equiv 3$.

We will calculate each case. If $k \mod 4 \equiv 0$:

— for y', we have

$$\begin{aligned}
y' &= (0011)(1)^k (0011)\\
y'^1 &= (0011)(1)^{k-1}(0)^1 (0011)\\
y'^2 &= (0011)(1)^{k-2}(0)^2 (0011)\\
y'^3 &= (0011)(1)^{k-4}(0)^4 (0011)\\
y'^4 &= (0011)(1)^{k-5}(0)^5 (0011)\\
y'^5 &= (0011)(1)^{k-6}(0)^6 (0011)\\
y'^6 &= (0011)(1)^{k-8}(0)^8 (0011)\\
&\vdots\\
y'^t &= (0011)(1)^{k-t-\lfloor \frac{t}{3} \rfloor}(0)^{t+\lfloor \frac{t}{3} \rfloor} (0011)
\end{aligned}$$

we need to use $t^* = \frac{3(k-8)}{4}$

$$\begin{aligned}
y'^{t^*} &= (0011)(1)^8(0)^{k-8} (0011)\\
y'^{t^*+1} &= (0011)(1)^7(0)^{k-7} (0011)\\
y'^{t^*+2} &= (0011)(1)^5(0)^{k-5} (0011)\\
y'^{t^*+3} &= (0011)(1)^4(0)^{k-4} (0011)\\
y'^{t^*+4} &= (0011)(1)^2(0)^{k-2} (0011)\\
y'^{t^*+5} &= (0011)(0)^k (0011)\\
&= y
\end{aligned}$$

— for y, we have

$$\begin{aligned}
y &= (0011)(0)^k (0011)\\
y^1 &= (0011)(1)^1(0)^{k-1} (0011)\\
y^2 &= (0011)(1)^3(0)^{k-3} (0011)\\
y^3 &= (0011)(1)^6(0)^{k-6} (0011)\\
y^4 &= (0011)(1)^9(0)^{k-9} (0011)\\
&\vdots\\
y^{4+\tau} &= (0011)(1)^{9+4\tau}(0)^{k-(9+4\tau)} (0011)
\end{aligned}$$

we need to consider $\tau^* = \frac{k}{4} - 3$

$$\begin{aligned}
y^{4+\tau^*} &= (0011)(1)^{k-3}(0)^3 (0011)\\
y^{4+\tau^*+1} &= (0011)(1)^k (0011)\\
&= y'
\end{aligned}$$

From where $T = (t^* + 4) + (4 + \tau^* + 1) = k$. And we can conclude that the cycle within the two walls separated by $k \mod 4 \equiv 0$ is $\mathcal{O}(k)$.

If $k \mod 4 \equiv 1$, since the first cells of the configuration update in the same order in this update mode we start with the same configurations. For y, we have

$$
\begin{aligned}
y &= (0011)(0)^k(0011) \\
y^1 &= (0011)(1)^1(0)^{k-1}(0011) \\
y^2 &= (0011)(1)^3(0)^{k-3}(0011) \\
y^3 &= (0011)(1)^6(0)^{k-6}(0011) \\
y^4 &= (0011)(1)^9(0)^{k-9}(0011) \\
&\vdots \\
y^{4+\tau} &= (0011)(1)^{9+4\tau}(0)^{k-(9+4\tau)}(0011)
\end{aligned}
$$

Now, we need to consider $\tau^* = \frac{k-13}{4}$ because there is a small change in the order in which we update the cells closer to w_2 (and because k is no longer divisible by 4).

$$
\begin{aligned}
y^{\tau^*} &= (0011)(1)^{k-4}(0)^4(0011) \\
y^{\tau^*+1} &= (0011)(1)^{k-3}(0)^3(0011) \\
y^{\tau^*+2} &= (0011)(1)^{k-2}(01)(0011) \\
y^{\tau^*+3} &= (0011)(1)^{k-1}(0)^1(0011)
\end{aligned}
$$

Which proceeds exactly as the previous case, calculating from y'^1. This means we can conclude that for $k \mod 4 \equiv 1$ the the cycle will have a length of $\mathcal{O}(k)$.

If $k \mod 4 \equiv 2$

— for y', we have

$$
\begin{aligned}
y' &= (0011)(1)^k(0011) \\
y'^1 &= (0011)(1)^{k-1}(0)^1(0011) \\
y'^2 &= (0011)(1)^{k-4}(0)^4(0011) \\
y'^3 &= (0011)(1)^{k-6}(0)^6(0011) \\
y'^4 &= (0011)(1)^{k-7}(0)^7(0011) \\
y'^5 &= (0011)(1)^{k-8}(0)^8(0011) \\
y'^6 &= (0011)(1)^{k-10}(0)^{10}(0011)
\end{aligned}
$$

$$
y'^t = (0011)(1)^{k-t-\lfloor \frac{t}{3} \rfloor - 2}(0)^{t+\lfloor \frac{t}{3} \rfloor + 2}(0011)
$$

we need to use $t^* = \frac{3(k-10)}{4}$, and it takes 5 more iterations to return to y.

— for y, we have

$$y^{4+\tau} = (0011)(1)^{9+4\tau}(0)^{k-(9+4\tau)} (0011)$$

and we need to consider $\tau^* = \frac{k-14}{4}$

$$\begin{aligned}
y^{4+\tau^*} &= (0011)(1)^{k-5}(0)^5 && (0011)\\
y^{4+\tau^*+1} &= (0011)(1)^{k-2}(0)^2 && (0011)\\
y^{4+\tau^*+2} &= (0011)(1)^{k-3}(0)^3 && (0011)\\
y^{4+\tau^*+3} &= (0011)(1)^k && (0011)\\
&= y'
\end{aligned}$$

From where $T = (t^* + 4) + (4 + \tau^* + 3) = k$. And we can conclude that the cycle within the two walls separated by $k \mod 4 \equiv 2$ is $\mathcal{O}(k)$.

If $k \mod 4 \equiv 3$

— for y', we have

$$\begin{aligned}
y' &= (0011)(1)^k && (0011)\\
y'^1 &= (0011)(1)^{k-1}(0)^1 && (0011)\\
y'^2 &= (0011)(1)^{k-4}(0)^4 && (0011)\\
y'^3 &= (0011)(1)^{k-7}(01)(0)^5 && (0011)\\
y'^4 &= (0011)(1)^{k-8}(0)^8 && (0011)\\
y'^5 &= (0011)(1)^{k-9}(0)^9 && (0011)\\
y'^6 &= (0011)(1)^{k-11}(0)^{11} && (0011)
\end{aligned}$$

$$y'^t = (0011)(1)^{k-t-\lfloor\frac{t}{3}\rfloor-3}(0)^{t+\lfloor\frac{t}{3}\rfloor+3} (0011)$$

we need to use $t^* = \frac{3(k-11)}{4}$, and it takes 5 more iterations to return to y. — for y, we have

$$y^{4+\tau} = (0011)(1)^{9+4\tau}(0)^{k-(9+4\tau)} (0011)$$

and we need to consider $\tau^* = \frac{k-11}{4}$

$$\begin{aligned} y^{4+\tau^*} &= (0011)(1)^{k-2}(0)^2 \quad (0011) \\ y^{4+\tau^*+1} &= (0011)(1)^{k-3}(0)^3 \quad (0011) \\ y^{4+\tau^*+2} &= (0011)(1)^{k} \quad (0011) \\ &= y' \end{aligned}$$

From where $T = (t^* + 4) + (4 + \tau^* + 3) = k$. And we can conclude that the cycle within the two walls separated by $k \mod 4 \equiv 3$ is $\mathcal{O}(k)$.

Thus, we have proven that the behavior between walls that are at a distance k has limit cycles of length $\mathcal{O}(k)$.
And because of the independence of the dynamics between two pairs of walls, the asymptotic dynamics of a global configuration x is a limit cycle whose length is given by the least common multiple of the lengths of all limit cycles of the subconfigurations.

We can consider a configuration and block-parallel update mode such that the limit cycles are distinct primes whose sum is equal to $n - 4m$, with m the number of walls. Then, thanks to Theorem 18 from [41] we know that the maximal product of distinct primes whose sum is less than n, when n tends to infinity, can be approached by $\sqrt{n \log n}$.

Hence, we can conclude that the largest limit cycle of the family $(184, \text{BP})$ applied over a ring of size n is $\Omega\left(2^{\sqrt{n \log n}}\right)$.

□

Corollary 3. $(184, \text{LC})$ *of size n can reach limit cycles of size* $\Omega\left(2^{\sqrt{n \log n}}\right)$.

Proof. From Theorem 7, since we know that BP update modes are also LC update modes, we know that we can find a LC that can reach limit cycles of superpolynomial length. □

Corollary 4. $(152, \text{SEQ})$ *and* $(56, \text{SEQ})$ *can only reach homogeneous fixed points.*

Proof. Follows the same proof as Theorem 4. □

Corollary 5. $(152, \text{BS})$ *and* $(56, \text{BS})$ *of size n has longest limit cycles of length* $\mathcal{O}(n)$.

Proof. Follows the same proof as Theorem 5. □

Corollary 6. $(152, \text{BP})$, $(152, \text{LC})$, $(56, \text{BP})$ *and* $(54, \text{LC})$ *can reach limit cycles of size* $\Omega\left(2^{\sqrt{n \log n}}\right)$.

Proof. Follows the same reasoning as Theorem 7. □

Corollary 7. $(184, \text{PAR})$, $(152, \text{PAR})$ *and* $(56, \text{PAR})$ *of size* n *has longest limit cycles of length* $\mathcal{O}(n)$.

Proof. As a consequence of Theorem 5 and Corollary 5. □

Corollary 8. $(184, \text{BIP})$, $(152, \text{BIP})$ *and* $(56, \text{BIP})$ *can only reach homogeneous fixed points.*

Proof. As a consequence of Theorem 4 and Corollary 4. □

3.2. Experimental Results

There are around twenty rules that are too complex or too chaotic to obtain clear mathematical proofs. So in order to study them we devised experiments where we calculated the dynamics of the density and energy. This allows us to observe what happens when we change the update modes to SEQ, BS, BP and LC. Indeed, we would like to understand how these properties are affected by the different update modes. With this goal in mind we have chosen rules 90, 150, 54 and 110. These rules are well known and famous for their complexity, with the first two belonging to class III [42] and the other two to class IV of Wolfram's complexity classification [8].

Definition 8 (Density)**.** *The* density *is defined as the average number of ones in a configuration* $x \in \mathbb{B}^n$ *(introduced as magnetization in [43]), that is:*

$$d(x) = \frac{1}{n} \sum_{i=0}^{n-1} x_i,$$

with x_i *the state of cell* i *for all* $i \in \{0, \ldots, n-1\}$.

Definition 9 (Energy)**.** *The* energy *of a configuration is defined as*

$$e(x) = \sum_{i=0}^{n-1} \frac{1 - 2x_i}{2} \left((2x_{i-1} - 1) + (2x_{i+1} - 1)\right),$$

with x_i *the state of cell* i *for all* $i \in \{0, \ldots, n-1\}$.

n	s	m SEQ	m BS	m BP	m LC
8	32,128	32	32	32	32
38	32,128	32	32	32	32
138	32,128	32	32	32	32

Table 6: Summary of configuration (s) and update mode (m) sample sizes for rings of sizes $n = 8$, $n = 38$ and $n = 138$.

Informally, the energy of a configuration x is a measure of the total number of cells in a configuration that have a different state to that of their neighbors. Note that the energy can have values between $-n$ and n. For example, a homogeneous configuration will have maximum energy i.e. $-n$, and in a configuration with alternating states $1010\ldots$ the energy will be maximum, i.e. n. This concept was used in the context of the study of discrete dynamical systems for threshold networks [44, 45] and icing models [46]. In order to compare different ring sizes, we can define *normalized energy* as

$$\bar{e}(x) = \frac{1}{n}\sum_{i=0}^{n-1} \frac{1-2x_i}{2}\left((2x_{i-1}-1)+(2x_{i+1}-1)\right).$$

Protocol

We considered three parameters to start: the size of the ring (n), the size of the sample of configurations (s) and the size of the sample of update modes (m). We decided to perform the experiments through a number of samples of configurations, instead of calculating for all possible initial configurations for rings of size 20 and higher because the computational time grew too large which would not allow to experiment with different update modes.

We chose rings of sizes 8, 38 and 138, with sample sizes of 32 and 128 configurations. The configuration samples were created such that each cell had the same probability to hold state 0 or 1, thus each sample has average density equal to 0.5 and average energy equal to 0. Furthermore, we ran both configuration sample sizes for each ring size under 32 different update mode from each family. We calculated the dynamics over 1000 time steps of the average density and energy (averaging over both the configuration sample sizes and the update mode sample size). Table 6 shows a summary of the considered sample sizes.

m SEQ	m BS	m BP	m LC
32	96	32	96

Table 7: Update mode (m) sample sizes for $n = 16$

After performing these simulations, we realized that higher orders of magnitude for the size of the ring were not needed because the experiments hinted that the overall behavior for each rule with its respective update modes resulted in similar graphs for rings of sizes $n = 8$, $n = 38$ and $n = 138$. This meant that we could experiment on smaller rings in which every possible initial configuration could be tested, which lets us pay particular attention to update modes.

We chose a ring of size 16, which allowed us to perform 1000 time steps of the rule under different update modes for all possible configurations in a reasonable amount of time. As a result, we were able to perform the experiments with different number of blocks for BS and different length of period for LC. Table 7 shows a summary of the considered sample sizes.

3.2.1. Different ring and sample sizes

In this section we will show that the behaviors of density and energy for rule 110 (rules 54, 90 and 150 will be discussed in the appendix) are very similar when comparing rings of sizes 8, 38 and 138, which gives us examples in three different orders of magnitude.

Examples of the update modes for $n = 8$ are defined on Table 8.

As seen on Fig. 7, we have that for all three ring sizes with sequential update modes the average density increases until it stabilizes at around .74, while the energy increases slightly for all three. Similarly, for block-sequential update modes, on Fig. 8 we can see that kind of behavior for the density, while the energy averages close to 0 for all three ring sizes.

For the block-parallel update modes we can see on Fig. 9 that while the density always increases to very similar amounts, the energy does show different behaviors for each ring size, while for the local clocks update modes we have that the energy always decreases (Fig. 10). Note that these two families of update modes allow for single cells to be updated more than once per time step, which could be the reason for their behavior to be less predictable than that of SEQ and BS.

Update Mode	Definition
Sequential	$(2, 3, 4, 5, 7, 6, 1, 0)$
Block Sequential	$(\{1\}, \{2, 7\}, \{6, 4\}, \{5, 0, 3\})$
Block Parallel	$\{(1, 2), (5, 3), (7), (6), (4, 0)\}$
Local Clocks	$\{P = (2, 3, 3, 1, 1, 2, 3, 3), \Delta = (1, 0, 2, 0, 0, 0, 1, 0)\}$

Table 8: Update modes used for $n = 8$.

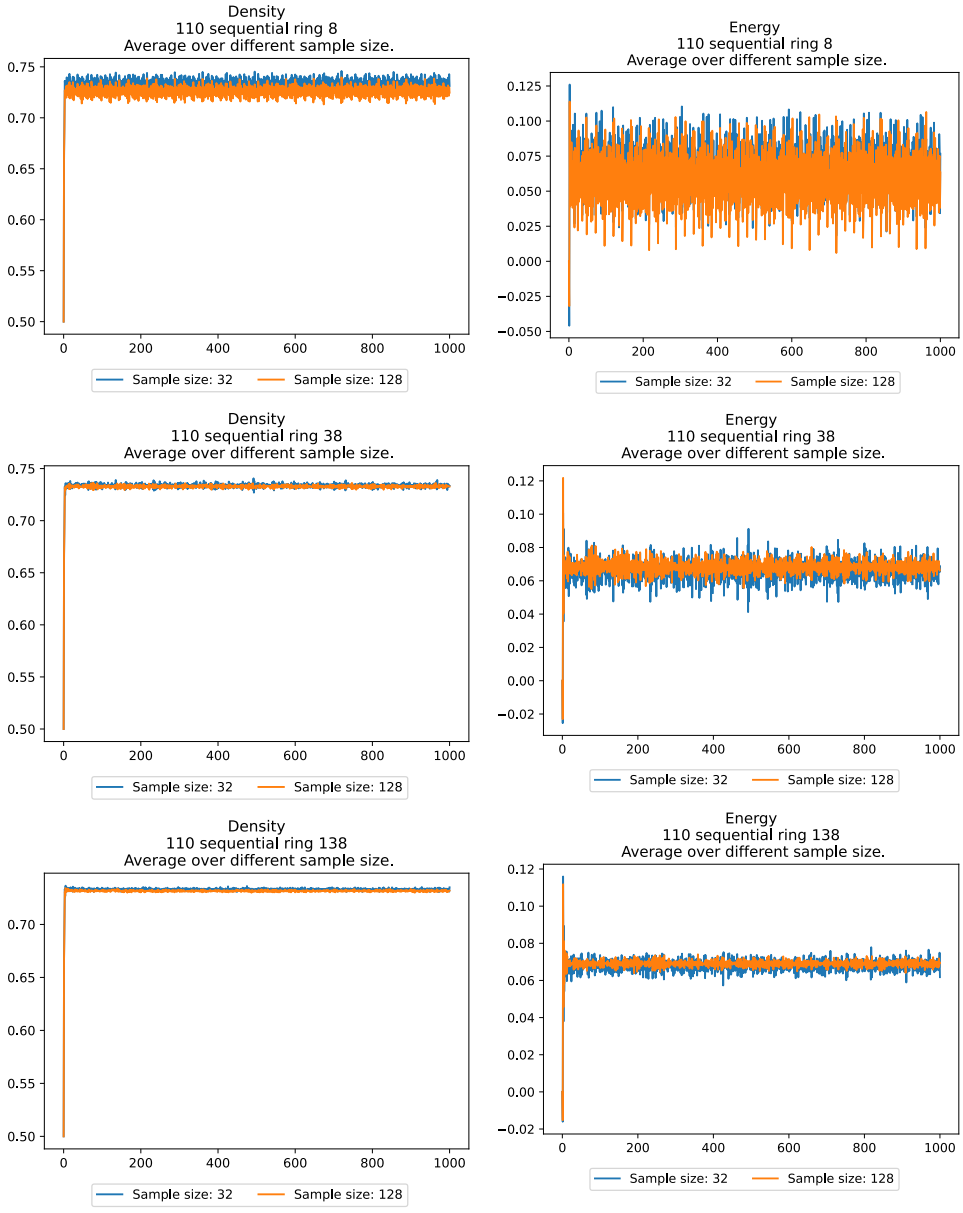

Figure 7: Density (left) and normalized energy (right) for a ring of size $n = 8$ (top), $n = 38$ (middle) and $n = 138$ (bottom) under rule 110 with sequential update mode with sample sizes of $s = 32$ and $s = 128$ initial configurations, over 1000 time steps.

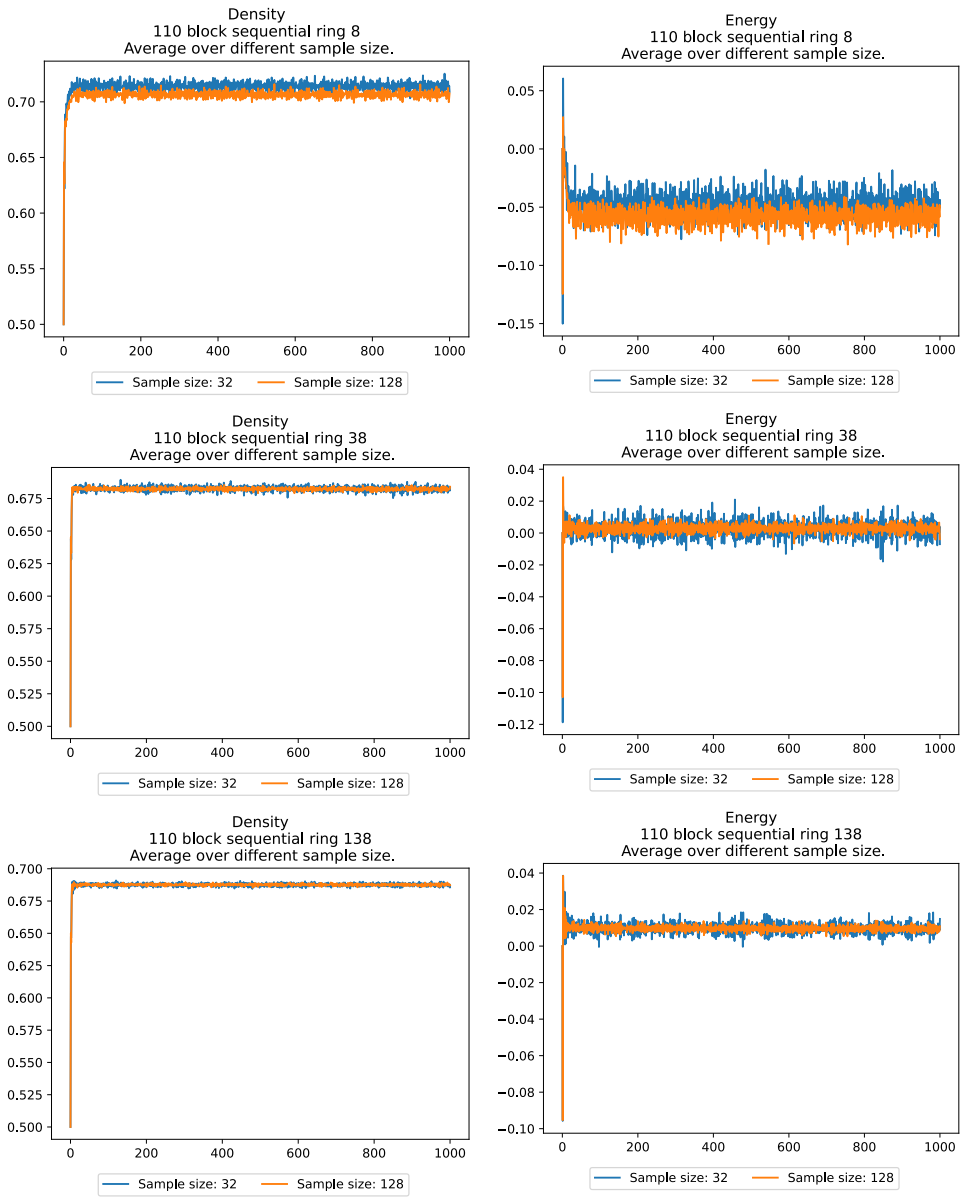

Figure 8: Density (left) and normalized energy (right) for a ring of size $n = 8$ (top), $n = 38$ (middle) and $n = 138$ (bottom) under rule 110 with block-sequential update mode with sample sizes of $s = 32$ and $s = 128$ initial configurations, over 1000 time steps.

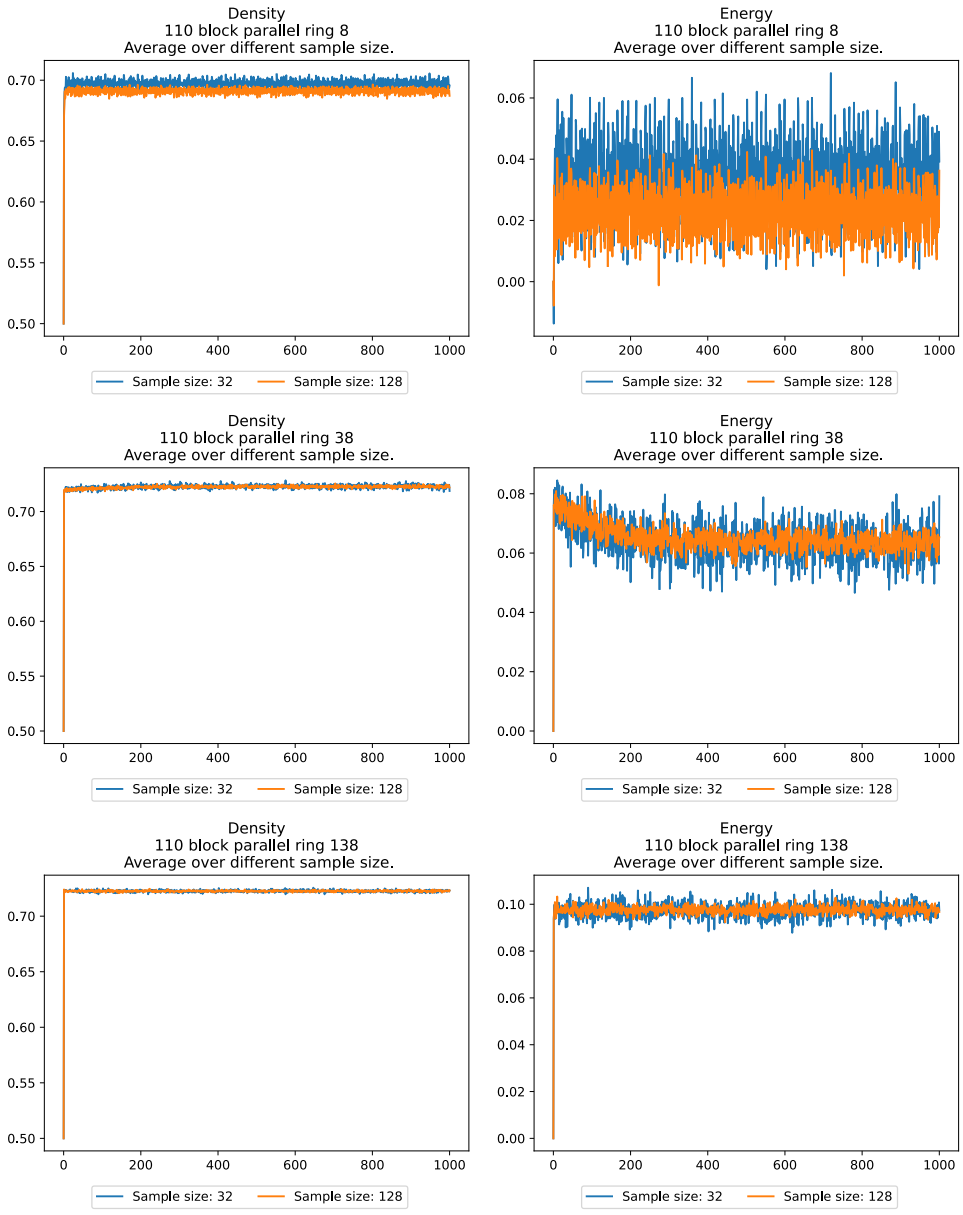

Figure 9: Density (left) and normalized energy (right) for a ring of size $n = 8$ (top), $n = 38$ (middle) and $n = 138$ (bottom) under rule 110 with block-parallel update mode with sample sizes of $s = 32$ and $s = 128$ initial configurations, over 1000 time steps.

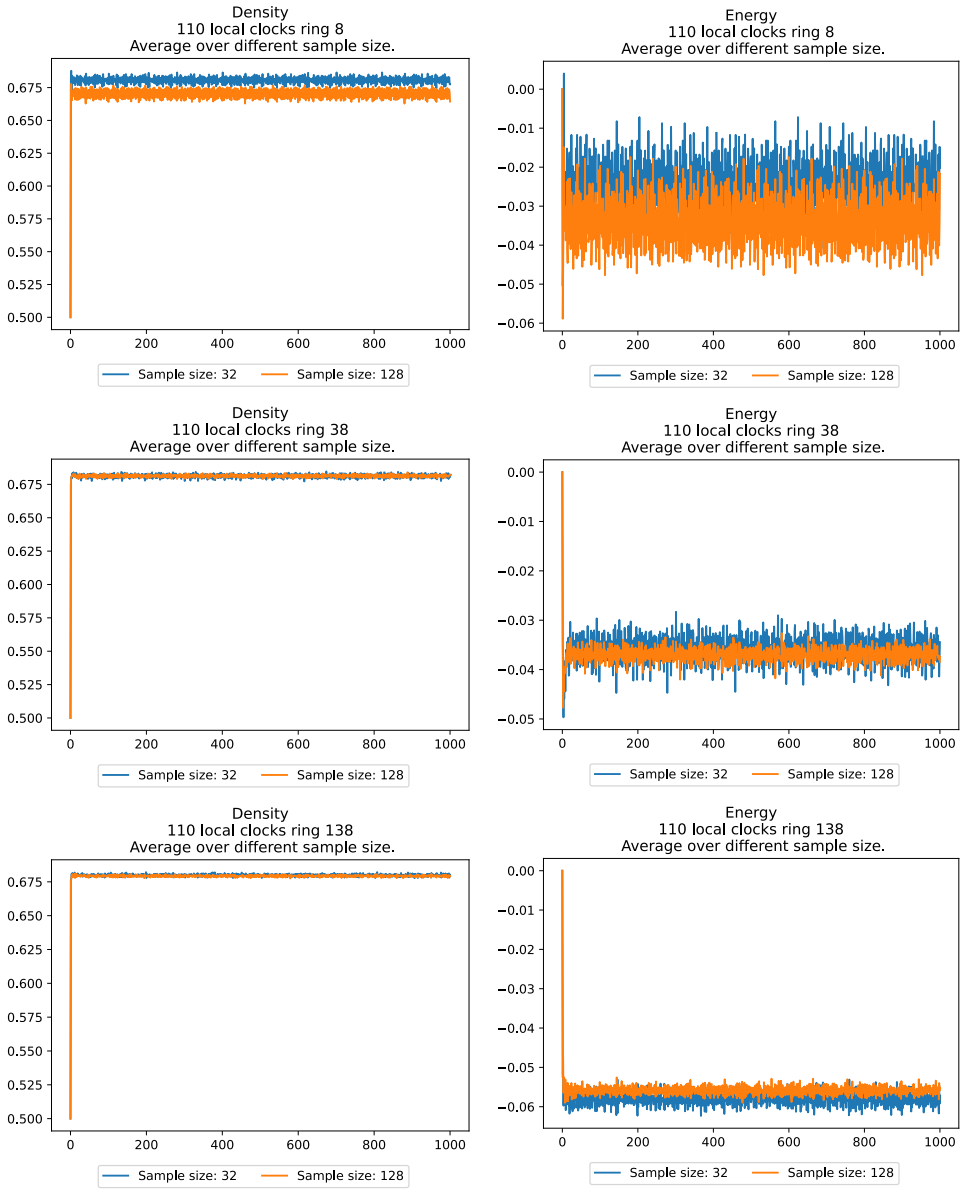

Figure 10: Density (left) and normalized energy (right) for a ring of size $n = 8$ (top), $n = 38$ (middle) and $n = 138$ (bottom) under rule 110 with local clocks update mode with sample sizes of $s = 32$ and $s = 128$ initial configurations, over 1000 time steps.

3.2.2. All configurations

In this section we will show the results for the average density and energy for all possible configurations of a ring of size $n = 16$. After the experiments from the previous section, we decided to use a ring of size $n = 16$, because it was the order of magnitude in-between, and the experiments were able to be completed for all initial configuration in a reasonable amount of time.

We calculated the dynamics for $m = 32$ block-sequential update modes with 3, 4 and 5 blocks; $m = 32$ local clocks update modes with periods 2, 4 and 5. As well as $m = 32$ sequential and $m = 32$ block-parallel update modes.

As expected, block-parallel and local clocks showed the biggest variation for all rules, which agrees with our hypothesis that BP and LC are the update modes that induce the greatest differences within the rules.

Rule 110

This rule shows a variety of behaviors depending on the update modes. First of all, note that for $(110, \text{SEQ})$ (Fig. 11) we see all update modes clustered together both for density and energy. Similarly, for $(110, \text{BS})$ (Fig. 13), the density stays close together for the different update modes belonging to this category for the different number of blocks. Note that the energy can decrease a noticeable amount before going back to hovering around values close to zero.

This is very different from what we can observe for $(110, \text{BP})$ and $(110, \text{LC})$ (Figs. 12 and 14, resp.), where the values are more spread out. We can see that the value of the density does not decrease from .5, but for the energy it can vary as much towards the positive as towards the negative.

Note that in all cases, the values of density and energy "stabilize" very quickly, which is backed by the values of variance shown on Figures A.45 through to A.48. Furthermore, Fig. A.49 shows that indeed there is a slight increase of variance between update modes of the same class for local clocks and block-parallel when compared to sequential and block-sequential.

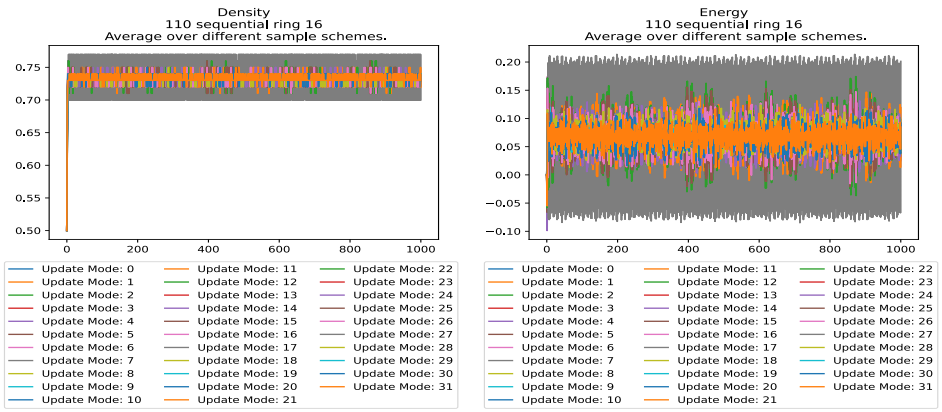

Figure 11: Density (left) and normalized energy (right) for a ring of size 16 under rule 110 average over all configurations, with different sequential update modes, over 1000 time steps.

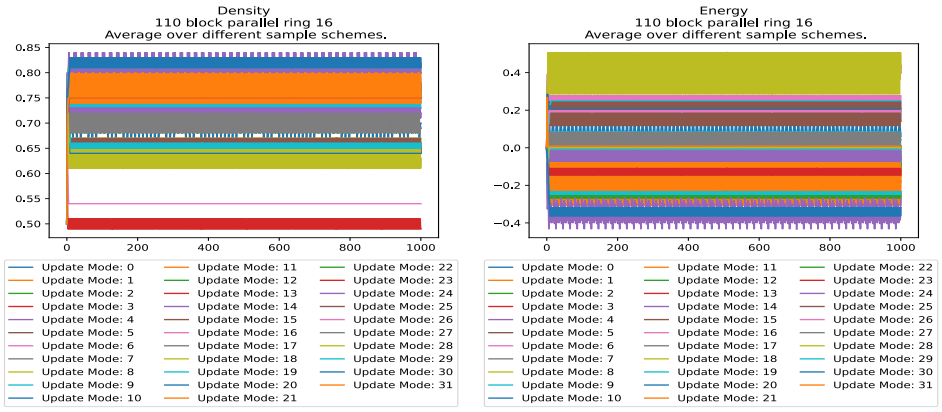

Figure 12: Density (left) and normalized energy (right) for a ring of size 16 under rule 110 average over all configurations, with different block-parallel update modes, over 1000 time steps.

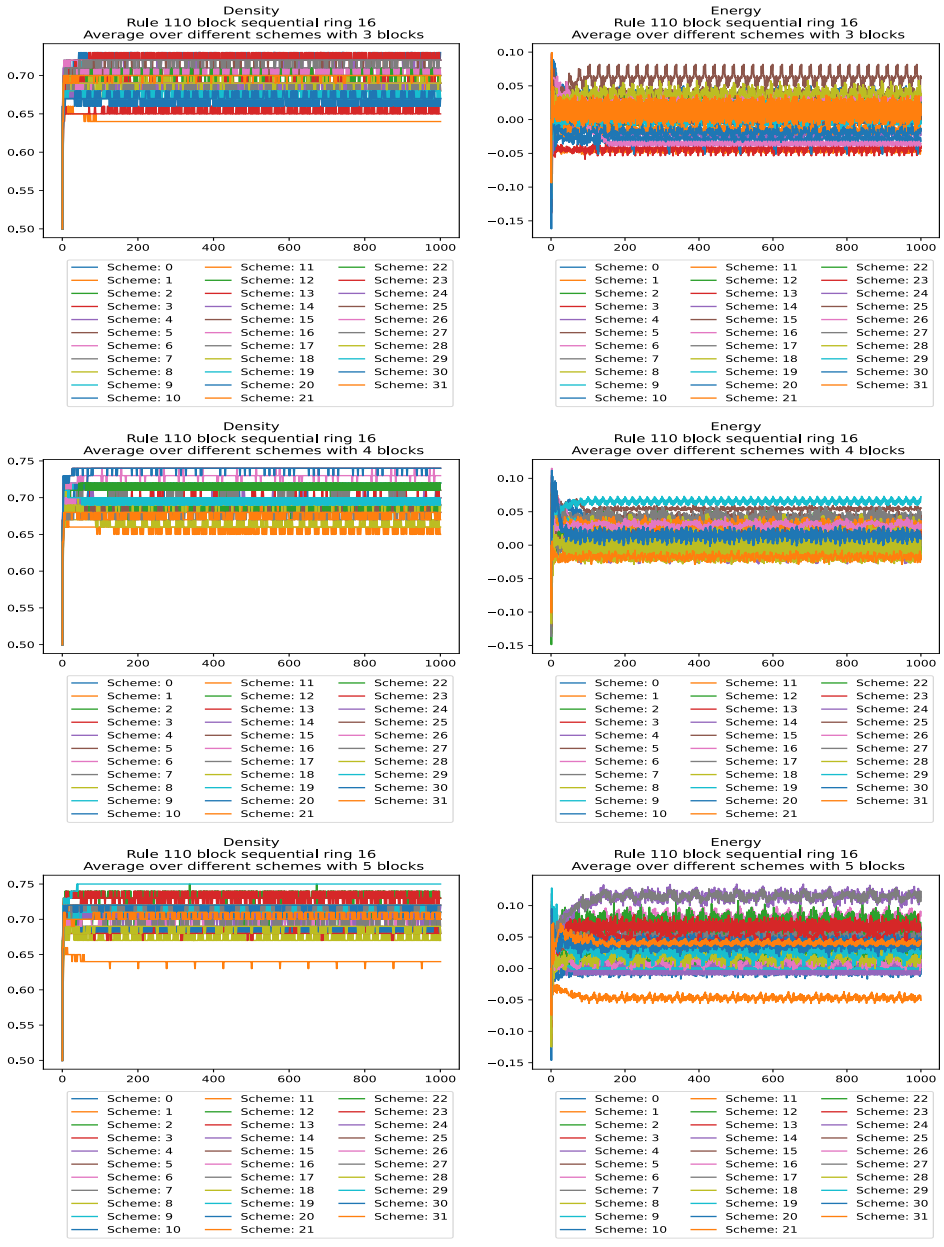

Figure 13: Density (left) and normalized energy (right) for $n = 16$ under $(110, \text{BS})$ with 3 (top), 4 (middle) and 5 (bottom) blocks, over 1000 time steps.

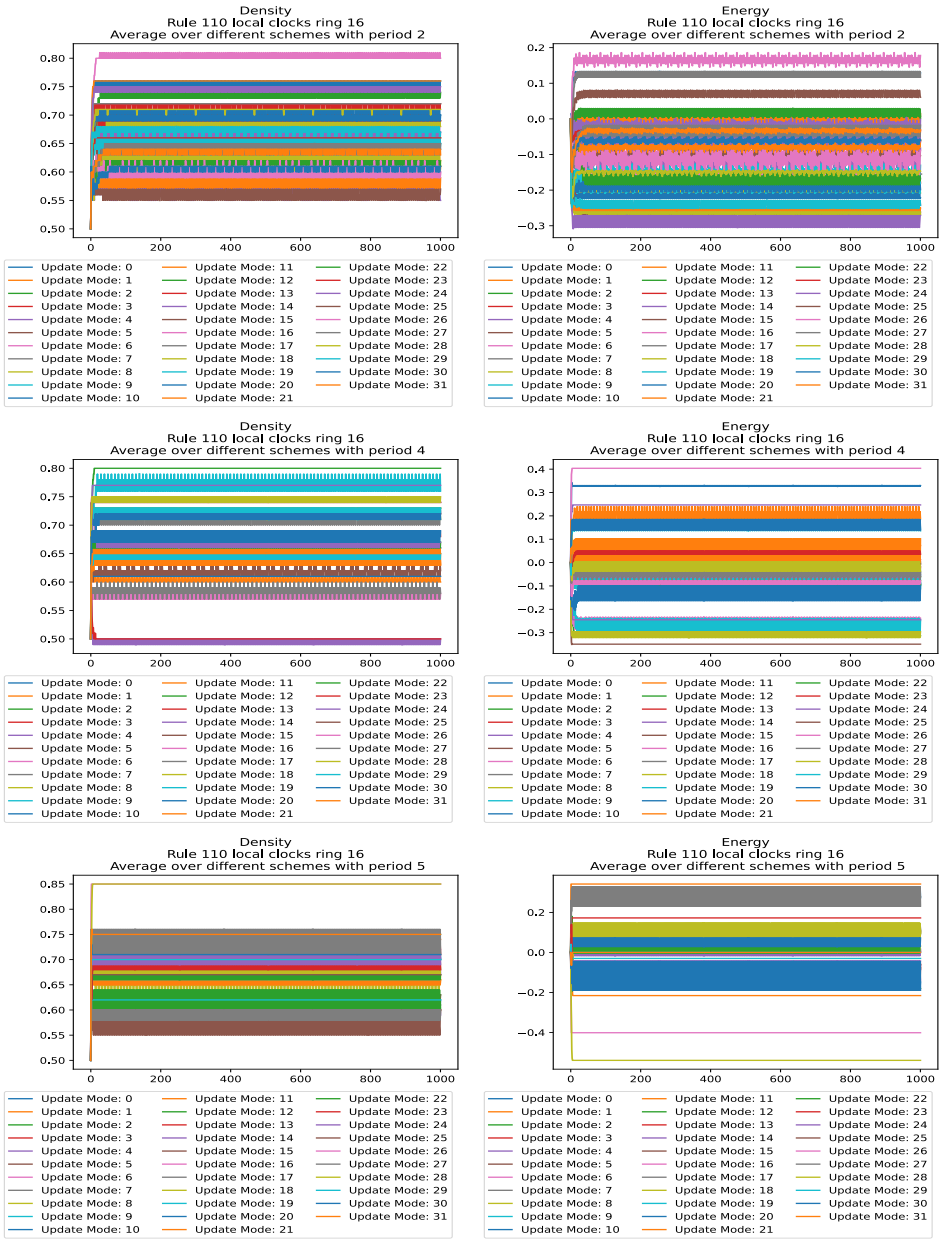

Figure 14: Density (left) and normalized energy (right) for $n = 16$ under $(110, \text{LC})$ with period 2 (top), 4 (middle) and 5 (bottom), over 1000 time steps.

Rule 150

Rule 150 is very stable for all update modes, unlike rule 110 the average density and energy change very little regardless of the update mode, which is shown on Figs. 15 and 16. More examples of this behavior are displayed in the appendix on Figs. A.40 through to A.43. Additionally, Fig. A.44 presents the variance which shows that that the variance is indeed 0 for all update modes in each class.

Similarly, rule 90 (which also belongs to class III) produces the same type of graphs, which can be found on the Appendix on Figs. A.31 through to A.34.

The fact that both rules that belong to class III, especially rule 150, have such stable average density and energy can be explained by the fact that there is no clear pattern that emerges from this rule. As a consequence we have a larger variety of possible configurations at each time step; thus the values of density and energy tend to be to 0.5 and 0, respectively.

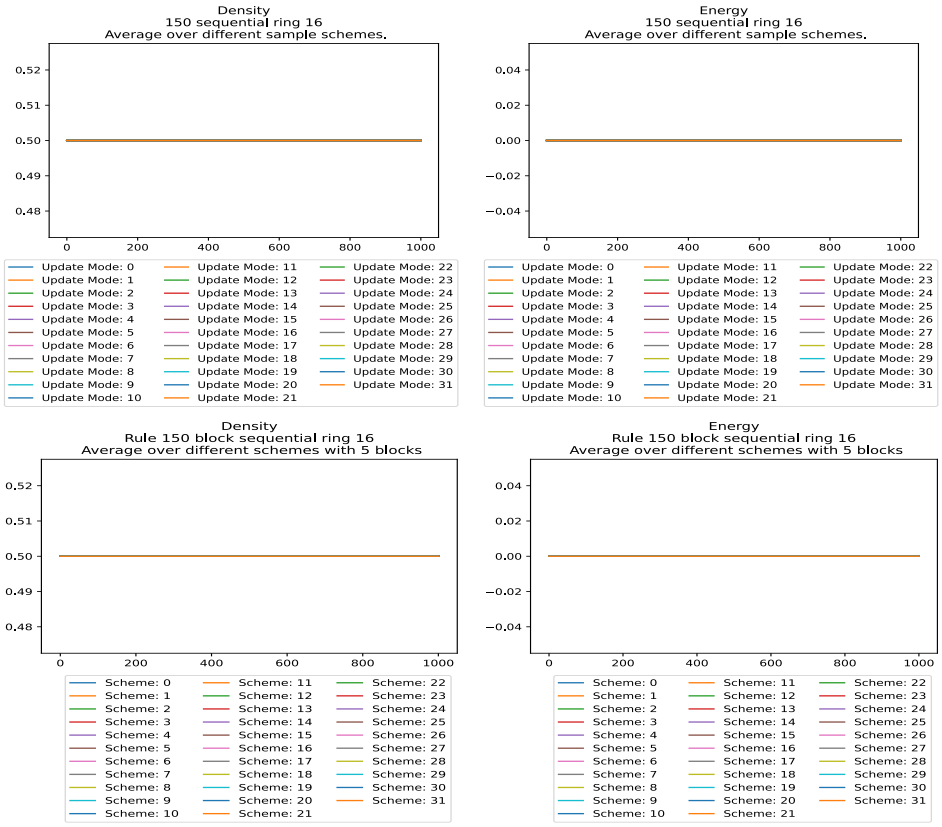

Figure 15: Density (left) and normalized energy (right) for a ring of size 16 under rule 150 average over all configurations, with different sequential (top) and block-sequential (bottom) update modes, over 1000 time steps.

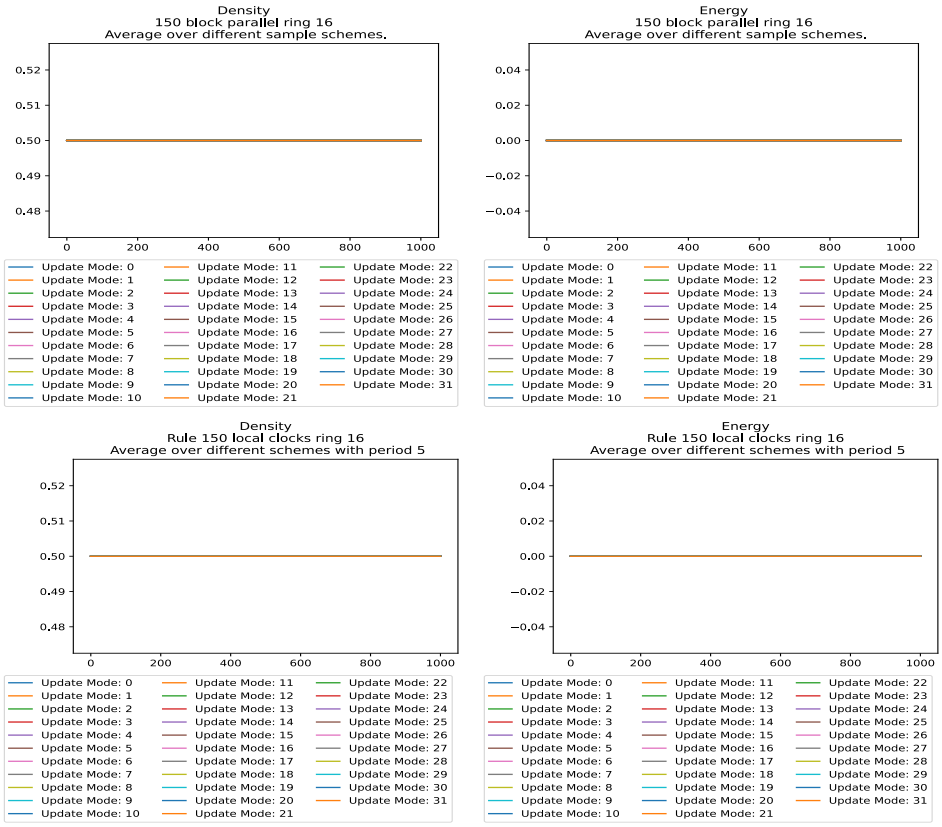

Figure 16: Density (left) and normalized energy (right) for a ring of size 16 under rule 150 average over all configurations, with different block-parallel (top) and local clocks (bottom) update modes, over 1000 time steps.

Rule 54

Rule $(54, \text{SEQ})$ shows a very stable behavior, identical to $(150, \text{SEQ})$, but for update modes under all other categories its behavior is more similar to ones seen under rule 110, as we can see on Fig. 17 using $(54, \text{LC})$ as an example. More examples with the rest of the update modes can be found in the Appendix.

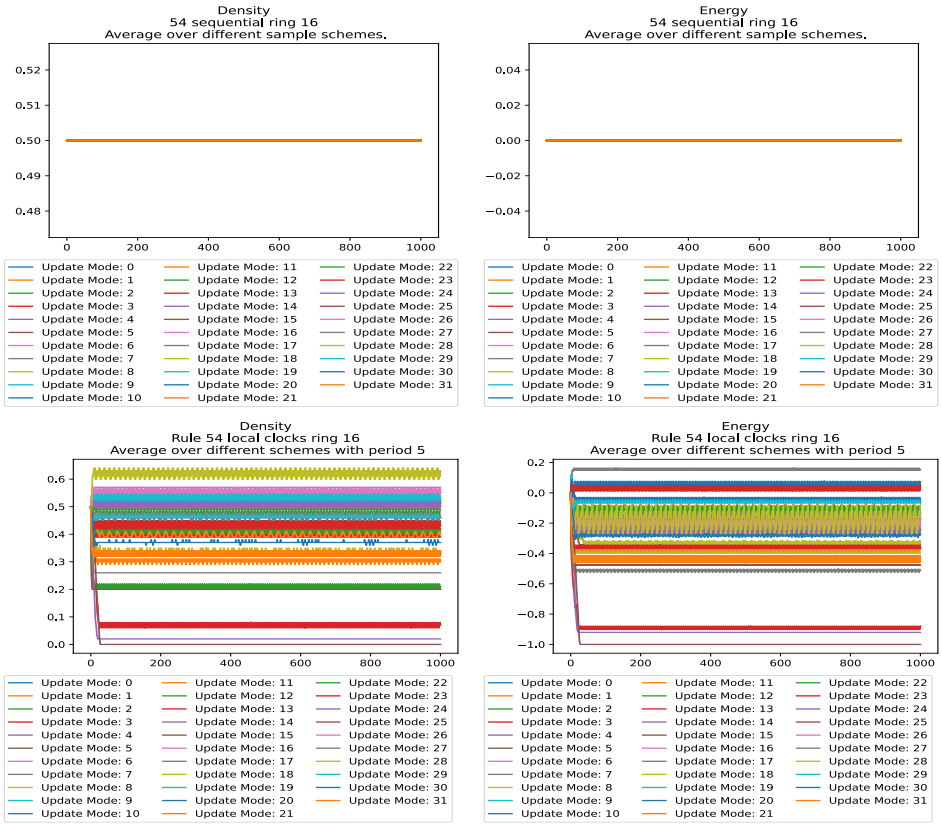

Figure 17: Density (left) and normalized energy (right) for a ring of size 16 under rule 54 average over all configurations, with different sequential (top) and local clocks with maximum period 5 (bottom) update modes, over 1000 time steps.

4. Discussion

Theoretical Results

In this paper, we have studied 67 of the 88 ECA under different update modes, proving that there is a set of rules whose longest limit cycles remain of the same length regardless of the update mode, which shows that some ECA are robust to (a)synchronism.

We have shown that there are three other behaviors, summarized in Table 3. Firstly, we have rules 108 and 156 with a notorious change in complexity, going from constant (PAR) to superpolynomial (SEQ). Secondly, we have rules 1 and 178 in which the increase of the maximum period is more "gradual". And thirdly, we have rules 56, 152 and 174, where the length of the longest limit cycle decreases from linear (PAR) to constant (SEQ), but then returns to linear (BS) and increasing again to superpolynomial (BP).

We have found that all rules capable of reaching superpolynomial cycles, are able to achieve them under block-parallel and local clocks update modes. This emphasizes that if we were able to prove that there exists a hierarchy between periodic update modes, BP and LC would be higher than the others studied in this paper. However, note that there are rules whose complexity remains constant under block-parallel and local clocks update modes.

Experimental Results

We have studied through measures of density and energy two chaotic rules (90 and 150), which appear to have little to no change in the average of these measures when the update mode is modified.

Under the light of the experiments we can see that there are no favored patterns for rule 150 under any of the analyzed update modes. Which would be coherent the chaotic nature of the rule.

On the other hand, we saw that the values of density and energy for rules 54 (except under sequential update mode) and 110 find different points of stability for each update mode, and that once those values are reached, the values of density and energy show little variation from one time step to another. Furthermore, local clocks and block-parallel present greater variety of values of stability, which could be attributed to the fact that those update modes allow cells to be updated more than once per step.

Future Work

The results that we have obtained at this point do not allow us to differentiate between block-parallel and local clocks in terms of complexity. Thus, we would like to establish a hierarchy between them eventually, and we suspect that local clocks should be at the highest rung.

Naturally, it is of interest to continue the theoretical analysis for rules 25, 37, 57, 58, 62, 74, 154 (class II), and research what properties (if any) do these rules have that made them incompatible with the strategies developed for other rules in the same class. Conversely, what properties does rule 18 have that allowed us to use methods that did not work for the rest of the rules in class III? Furthermore, can we prove the existence of limit cycles of exponential length for some combination of rule and update mode.

Another question is about if we can expand the research about sensitivity under different update modes to one-dimensional cellular automata with different radii, whether some of the approaches found for radius 1 can be of use in those cases.

We need further experiments to check the behavior of the rest of the rules in classes III and IV, and whether their behavior resembles what we have found for the rules we have presented as examples in this paper. Similarly, for the ones that we were unable to classify belonging to class II: can we expect to find patterns in rules belonging to class II that we did not find for the other two classes?

For the most part, we know that rule 90 has similar types of graph to rule 150. However, it is worth studying the specific update modes whose density and energy become stable around values different from 0.5 and 0, respectively.

Additionally, is it possible to do a similar study of energy and density that tells us something about ECA under non-deterministic and/or stochastic update modes?

Acknowledgements. This work has been partially funded by the HORIZON-MSCA-2022-SE-01 project 101131549 "ACANCOS" project (IDL, EG, MRW, SS), the ANR-24-CE48-7504 "ALARICE" project (SS), the STIC AmSud 22-STIC-02 "CAMA" project (IDL, EG, MRW, SS), and ANID FONDECYT Postdoctorado 3220205 (MRW).

References

[1] I. Donoso Leiva, E. Goles, M. Ríos-Wilson, S. Sené, Asymptotic (a)synchronism sensitivity and complexity of elementary cellular automata, in: Proceedings of Latin American Symposium on Theoretical Informatics (LATIN), Springer, 2024, pp. 272–286. doi:10.1007/978-3-031-55601-2_18.

[2] J. von Neumann, Theory of self-reproducing automata, University of Illinois Press, 1966, edited and completed by A. W. Burks.

[3] W. S. McCulloch, W. Pitts, A logical calculus of the ideas immanent in nervous activity, Journal of Mathematical Biophysics 5 (1943) 115–133. doi:10.1007/BF02478259.

[4] F. Robert, Itérations sur des ensembles finis et automates cellulaires contractants, Linear Algebra and its Applications 29 (1980) 393–412. doi:10.1016/0024-3795(80)90251-7.

[5] S. Wolfram, Statiscal mechanics of cellular automata, Reviews of Modern Physics 55 (1983) 601–644. doi:10.1103/RevModPhys.55.601.

[6] A. R. Smith III, Simple computation-universal cellular spaces, Journal of the ACM 18 (1971) 339–353. doi:10.1145/321650.321652.

[7] E. Goles, S. Martínez, Neural and automata networks: dynamical behavior and applications, Vol. 58 of Mathematics and Its Applications, Kluwer Academic Publishers, 1990. doi:10.1007/978-94-009-0529-0.

[8] W. Li, N. Packard, et al., The structure of the elementary cellular automata rule space, Complex systems 4 (3) (1990) 281–297.
URL https://www.complex-systems.com/abstracts/v04_i03_a03/

[9] J. Kari, Rice's theorem for the limit sets of cellular automata, Theoretical Computer Science 127 (1994) 229–254. doi:10.1016/0304-3975(94)90041-8.

[10] P. Concha-Vega, E. Goles, P. Montealegre, M. Ríos-Wilson, J. Santivañez, Introducing the activity parameter for elementary cellular automata, International Journal of Modern Physics C 33 (09) (2022) 2250121. doi:10.1142/S0129183122501212.

[11] S. Hoya White, A. Martín del Rey, G. Rodríguez Sánchez, Modeling epidemics using cellular automata, Applied Mathematics and Computation 186 (1) (2007) 193–202. doi:10.1016/j.amc.2006.06.126.

[12] G. Sirakoulis, I. Karafyllidis, A. Thanailakis, A cellular automaton model for the effects of population movement and vaccination on epidemic propagation, Ecological Modelling 133 (3) (2000) 209–223. doi:https://doi.org/10.1016/S0304-3800(00)00294-5.

[13] J. M. Sakoda, The checkerboard model of social interaction, The Journal of Mathematical Sociology 1 (1) (1971) 119–132. doi:10.1080/0022250X.1971.9989791.

[14] T. C. Schelling, Dynamic models of segregation, The Journal of Mathematical Sociology 1 (2) (1971) 143–186. doi:10.1080/0022250X.1971.9989794.

[15] A. R. Smith III, Real-time language recognition by one-dimensional cellular automata, Journal of Computer and System Sciences 6 (3) (1972) 233–253. doi:https://doi.org/10.1016/S0022-0000(72)80004-7.

[16] J. Mazoyer, A six-state minimal time solution to the firing squad synchronization problem, Theoretical Computer Science 50 (2) (1987) 183–238. doi:10.1016/0304-3975(87)90124-1.

[17] G. Y. Vichniac, Simulating physics with cellular automata, Physica D: Nonlinear Phenomena 10 (1) (1984) 96–116. doi:10.1016/0167-2789(84)90253-7.

[18] F. Robert, Discrete iterations: a metric study, Vol. 6 of Springer Series in Computational Mathematics, Springer, 1986. doi:10.1007/978-3-642-61607-5.

[19] D. M. Chapiro, Globally-asynchronous locally-synchronous systems, Ph.D. thesis, Stanford University (1984).
URL https://ui.adsabs.harvard.edu/abs/1984PhDT.....50C

[20] B. Charron-Bost, F. Mattern, G. Tel, Synchronous, asynchronous, and causally ordered communication, Distributed Computing 9 (1996) 173–191. doi:10.1007/s004460050018.

[21] M. R. Hübner, D. L. Spector, Chromatin dynamics, Annual Review of Biophysics 39 (2010) 471–489. doi:10.1146/annurev.biophys.093008.131348.

[22] B. Fierz, M. G. Poirier, Biophysics of chromatin dynamics, Annual Review of Biophysics 48 (2019) 321–345. doi:10.1146/annurev-biophys-070317-032847.

[23] S. A. Kauffman, Metabolic stability and epigenesis in randomly constructed genetic nets, Journal of Theoretical Biology 22 (1969) 437–467. doi:10.1016/0022-5193(69)90015-0.

[24] R. Thomas, Boolean formalization of genetic control circuits, Journal of Theoretical Biology 42 (1973) 563–585. doi:10.1016/0022-5193(73)90247-6.

[25] E. Goles, L. Salinas, Comparison between parallel and serial dynamics of Boolean networks, Theoretical Computer Science 296 (2008) 247)–253. doi:10.1016/j.tcs.2007.09.008.

[26] J. Aracena, E. Fanchon, M. Montalva, M. Noual, Combinatorics on update digraphs in Boolean networks, Discrete Applied Mathematics 159 (2011) 401–409. doi:10.1016/j.dam.2010.10.010.

[27] M. Noual, S. Sené, Synchronism versus asynchronism in monotonic Boolean automata networks, Natural Computing 17 (2018) 393–402. doi:10.1007/s11047-016-9608-8.

[28] L. Paulevé, S. Sené, Non-deterministic updates of Boolean networks, in: Proceedings of the International Workshop on Cellular Automata and Discrete Complex Systems (AUTOMATA 2021), Vol. 90 of OASIcs, Schloss Dagstuhl, 2021, pp. 10:1–10:16. doi:10.4230/OASIcs.AUTOMATA.2021.10.

[29] T. E. Ingerson, R. L. Buvel, Structure in asynchronous cellular automata, Physica D: Nonlinear Phenomena 10 (1984) 59–68. doi:10.1016/0167-2789(84)90249-5.

[30] N. Fates, M. Morvan, An experimental study of robustness to asynchronism for elementary cellular automata, Complex Systems 16 (2005) 1–27. doi:10.25088/ComplexSystems.16.1.1.

[31] A. Dennunzio, E. Formenti, L. Manzoni, G. Mauri, m-Asynchronous cellular automata: from fairness to quasi-fairness, Natural Computing 12 (2013) 561–572. doi:10.1007/s11047-013-9386-5.

[32] S. Wolfram, Universality and complexity in cellular automata, Physica D: Nonlinear Phenomena 10 (1984) 1–35. doi:10.1016/0167-2789(84)90245-8.

[33] K. Culik II, S. Yu, Undecidability of CA classification schemes, Complex Systems 2 (1988) 177–190.
URL https://www.complex-systems.com/abstracts/v02_i02_a02

[34] P. Kůrka, Languages, equicontinuity and attractors in cellular automata, Ergodic Theory and Dynamical Systems 17 (1997) 417–433. doi:10.1017/S014338579706985X.

[35] L. Paulevé, S. Sené, Systems biology modelling and analysis: formal bioinformatics methods and tools, Wiley, 2022, Ch. Boolean networks and their dynamics: the impact of updates. doi:10.1002/9781119716600.ch6.

[36] M. Ríos-Wilson, G. Theyssier, Intrinsic universality in automata networks I: Families and simulations, Theoretical Computer Science 997 (2024) 114511. doi:10.1016/j.tcs.2024.114511.

[37] M. Ríos-Wilson, G. Theyssier, Intrinsic universality in automata networks III: On symmetry versus asynchrony, Theoretical Computer Science 1022 (2024) 114890. doi:10.1016/j.tcs.2024.114890.

[38] J. Demongeot, S. Sené, About block-parallel Boolean networks: a position paper, Natural Computing 19 (2020) 5–13. doi:10.1007/s11047-019-09779-x.

[39] K. Perrot, S. Sené, L. Tapin, Combinatorics of block-parallel automata networks, in: Proceedings of the International Conference on Current Trends in Theory and Practice of Computer Science, Springer, 2024, pp. 442–455. doi:10.1007/978-3-031-52113-3_31.

[40] M. Ríos-Wilson, On automata networks dynamics: an approach based on computational complexity theory, Ph.D. thesis, Universidad de Chile

& Aix-Marseille Université (2021).
URL https://theses.hal.science/tel-03264167/document

[41] M. Deléglise, J.-L. Nicolas, On the largest product of primes with bounded sum, Journal of Integer Sequences 18 (2015).
URL http://dml.mathdoc.fr/item/hal-02079237

[42] S. Wolfram, Computation theory of cellular automata, Communications in mathematical physics 96 (1) (1984) 15–57.

[43] J. Zabolitzky, Critical properties of rule 22 elementary cellular automata., Journal of statistical physics 50 (1988) 1255–1262. doi:10.1007/BF01019164.

[44] J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities., Proceedings of the national academy of sciences 79 (8) (1982) 2554–2558. doi:10.1073/pnas.79.8.2554.

[45] E. Goles-Chacc, F. Fogelman-Soulié, D. Pellegrin, Decreasing energy functions as a tool for studying threshold networks, Discrete Applied Mathematics 12 (3) (1985) 261–277. doi:https://doi.org/10.1016/0166-218X(85)90029-0.

[46] B. M. McCoy, J.-M. Maillard, The importance of the ising model, Progress of Theoretical Physics 127 (5) (2012) 791–817. doi:10.1143/PTP.127.791.

Appendix A. Figures

Appendix A.1. Rule 54

Different sized rings

We can see on Fig. A.18 that with bigger sample size both density and energy stabilize at around .5 and 0 resp. for sequential update modes, unlike with block-sequential, block-parallel and local clock where the average value of the density and energy decrease, as seen on Figs. A.19, A.19 and A.21. Note that for each update modes both density and energy reach similar values, especially for rings of size 38 and 138, which agrees with our assessment that the size of the ring is not as important.

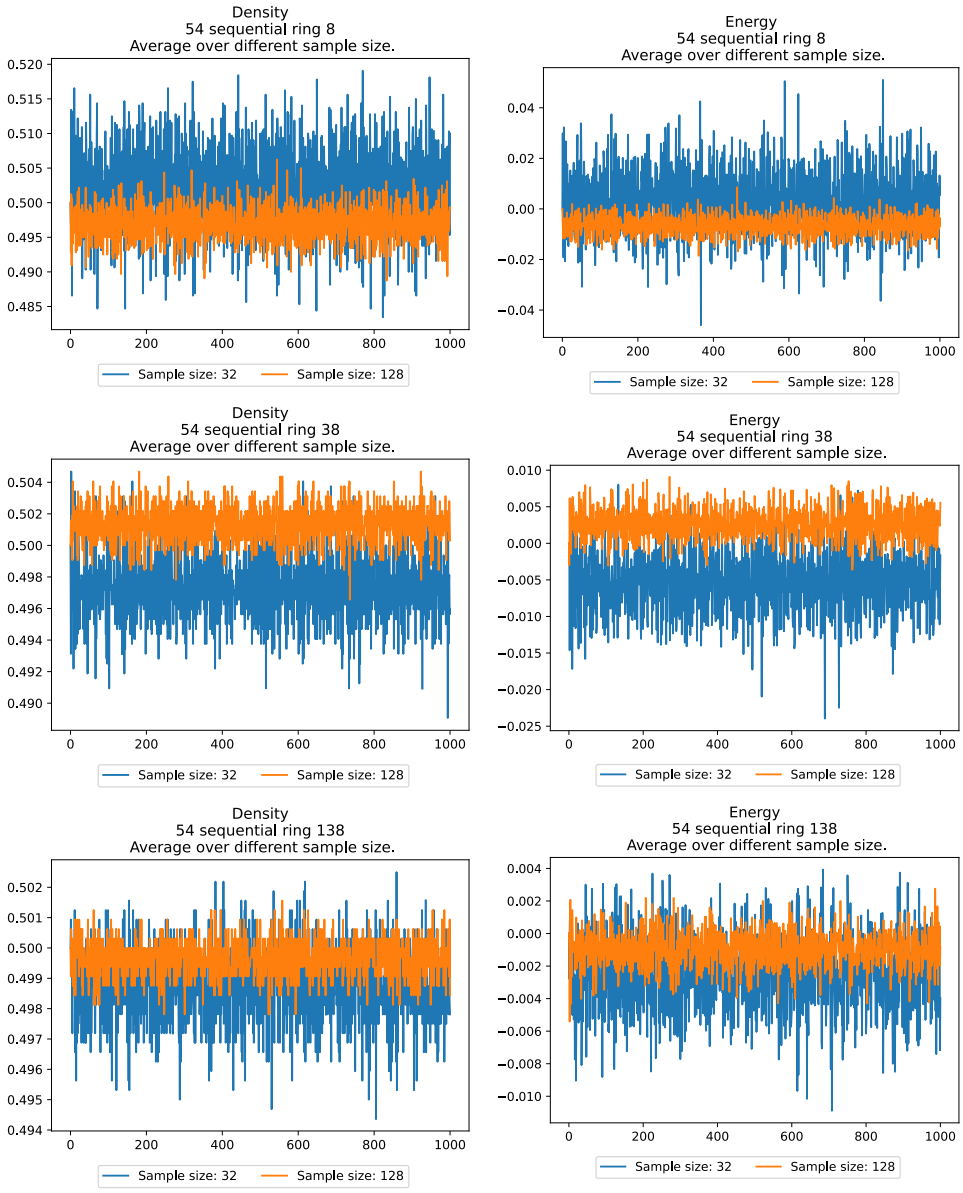

Figure A.18: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 54 with sequential update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

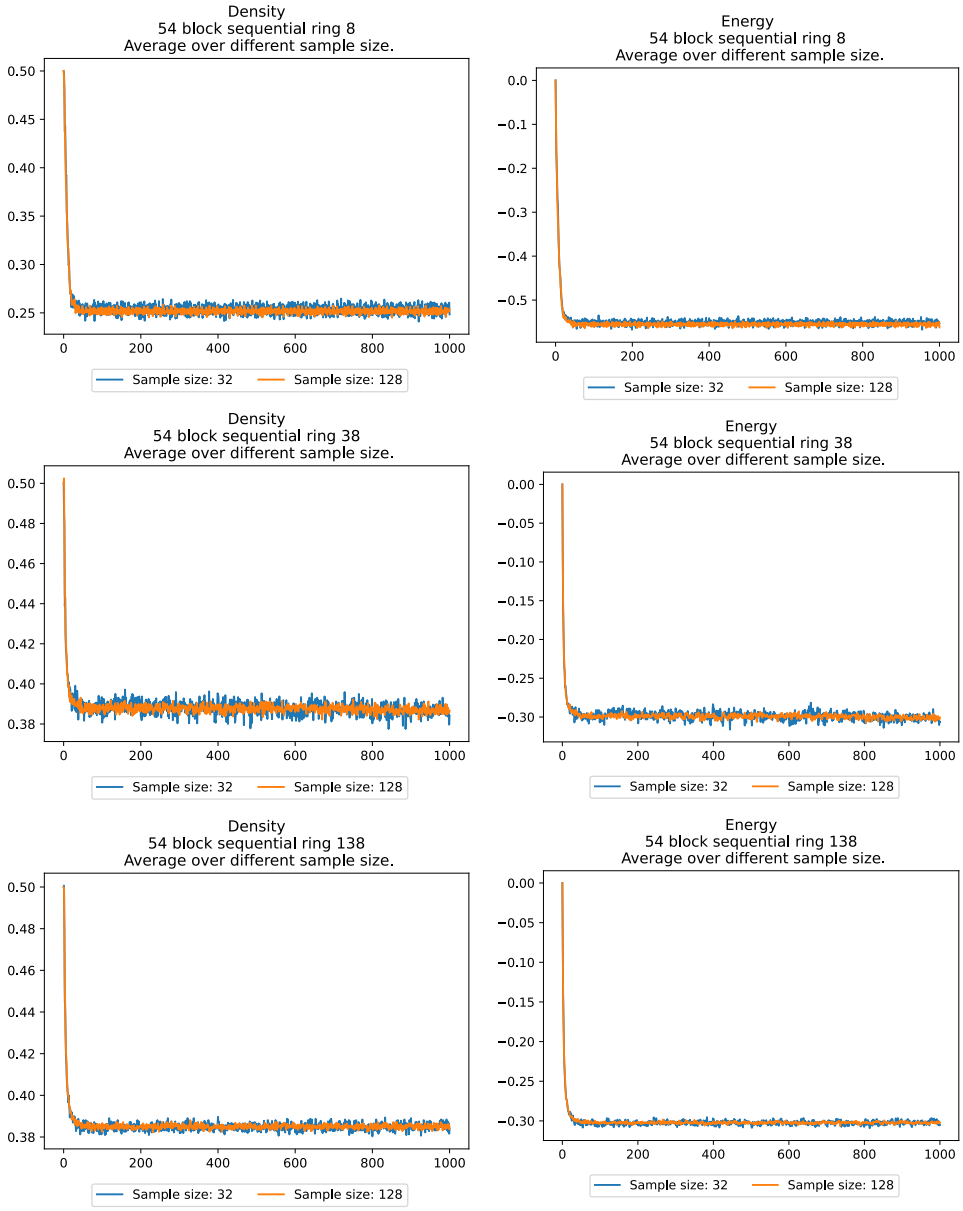

Figure A.19: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 54 with block-sequential update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

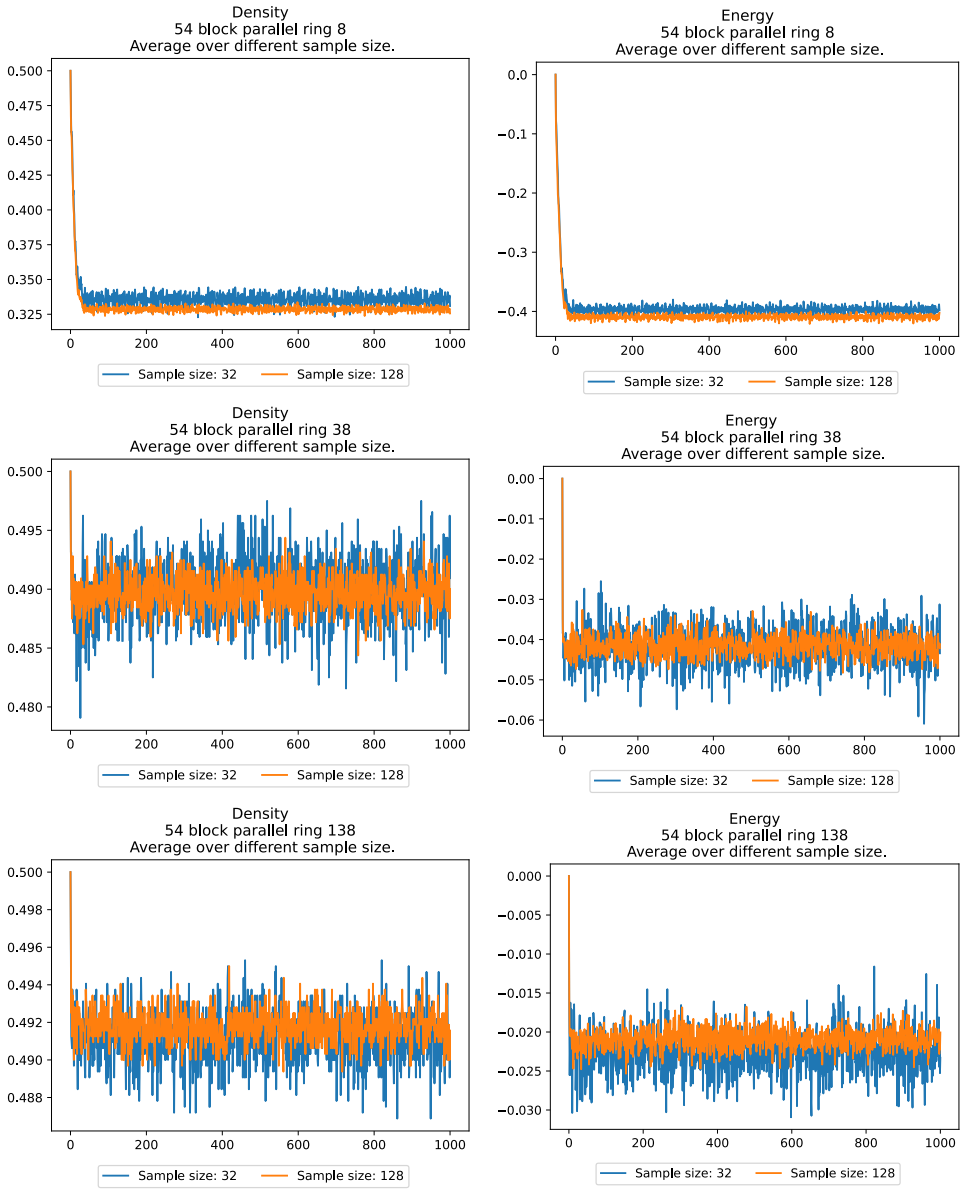

Figure A.20: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 54 with block-parallel update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

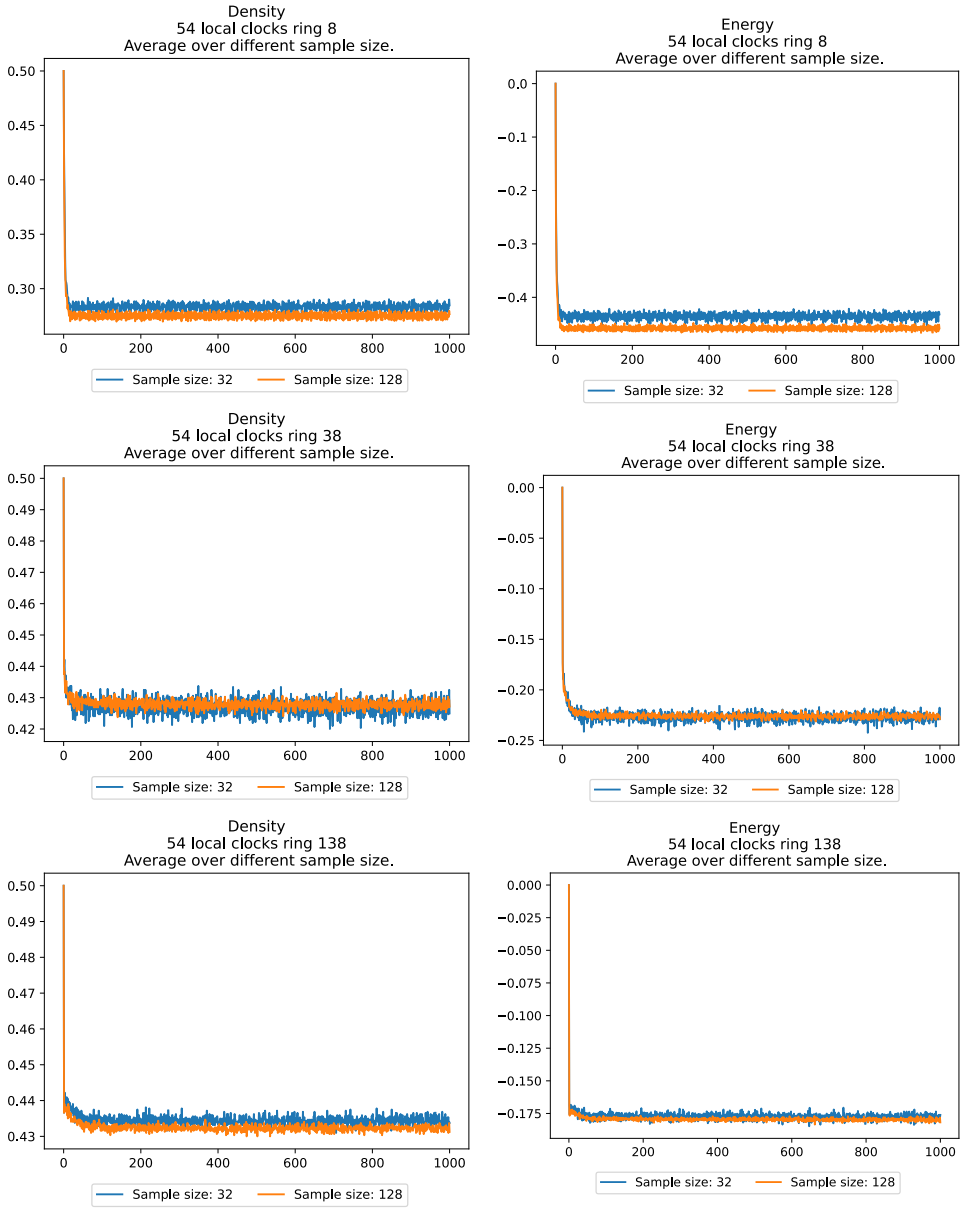

Figure A.21: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule (54, LC) with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

All configurations of ring size 16

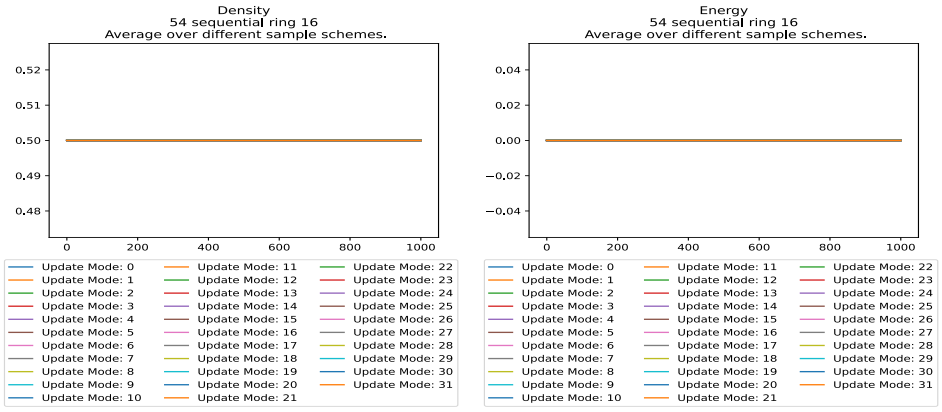

Figure A.22: Density (left) and normalized energy (right) for a ring of size 16 under rule $(54, \text{SEQ})$ average over all configurations, with different update modes, over 1000 time steps.

As mentioned in the Results, Figs. A.23, A.24 and A.25 show that indeed these rule 54 under these update modes behaves similarly to how rule 110 did. Furthermore, note that local clocks with maximum period 5 and block-parallel have the highest variance when comparing different update modes of that class, as seen of Fig A.26

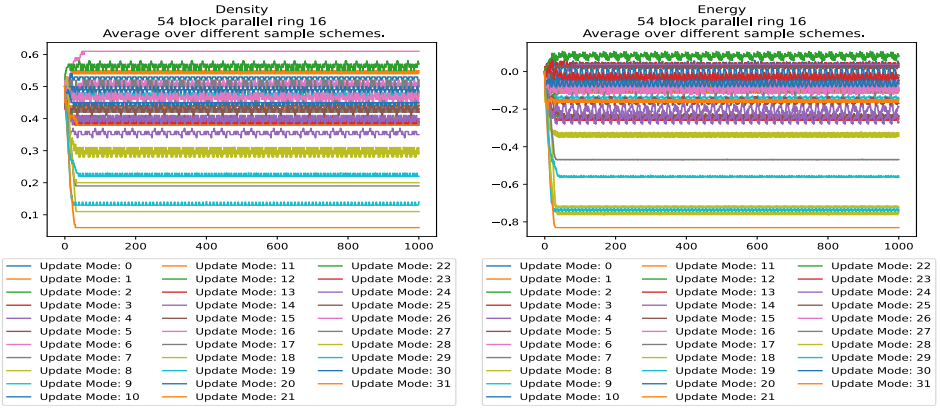

Figure A.23: Density (left) and normalized energy (right) for a ring of size 16 under rule (54, BP) average over all configurations, with different update modes, over 1000 time steps.

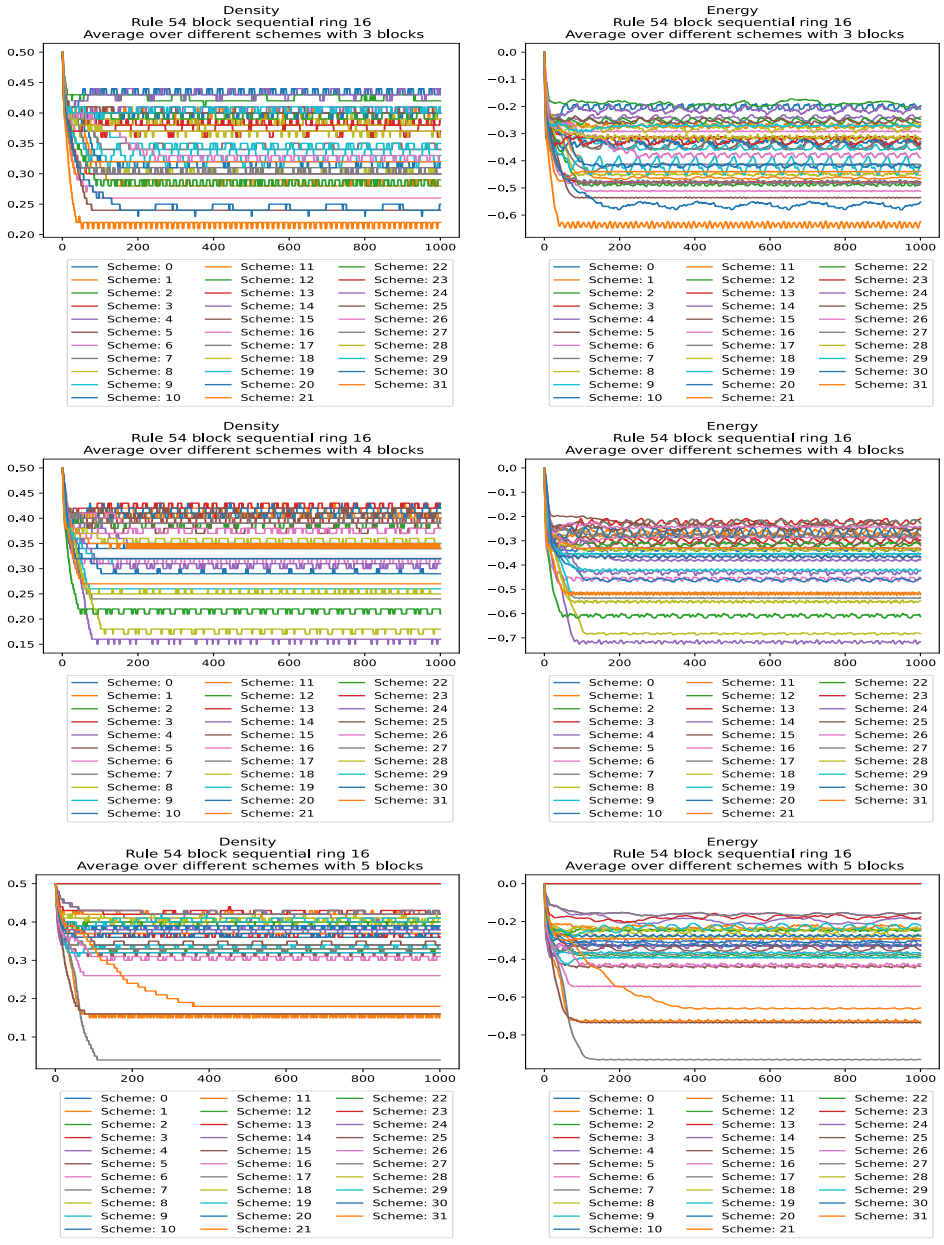

Figure A.24: Average over all configurations of Density (left) and normalized energy (right) for $n = 16$ under $(54, \text{BS})$ with 3 (top), 4 (middle) and 5 (bottom) blocks, over 1000 time steps.

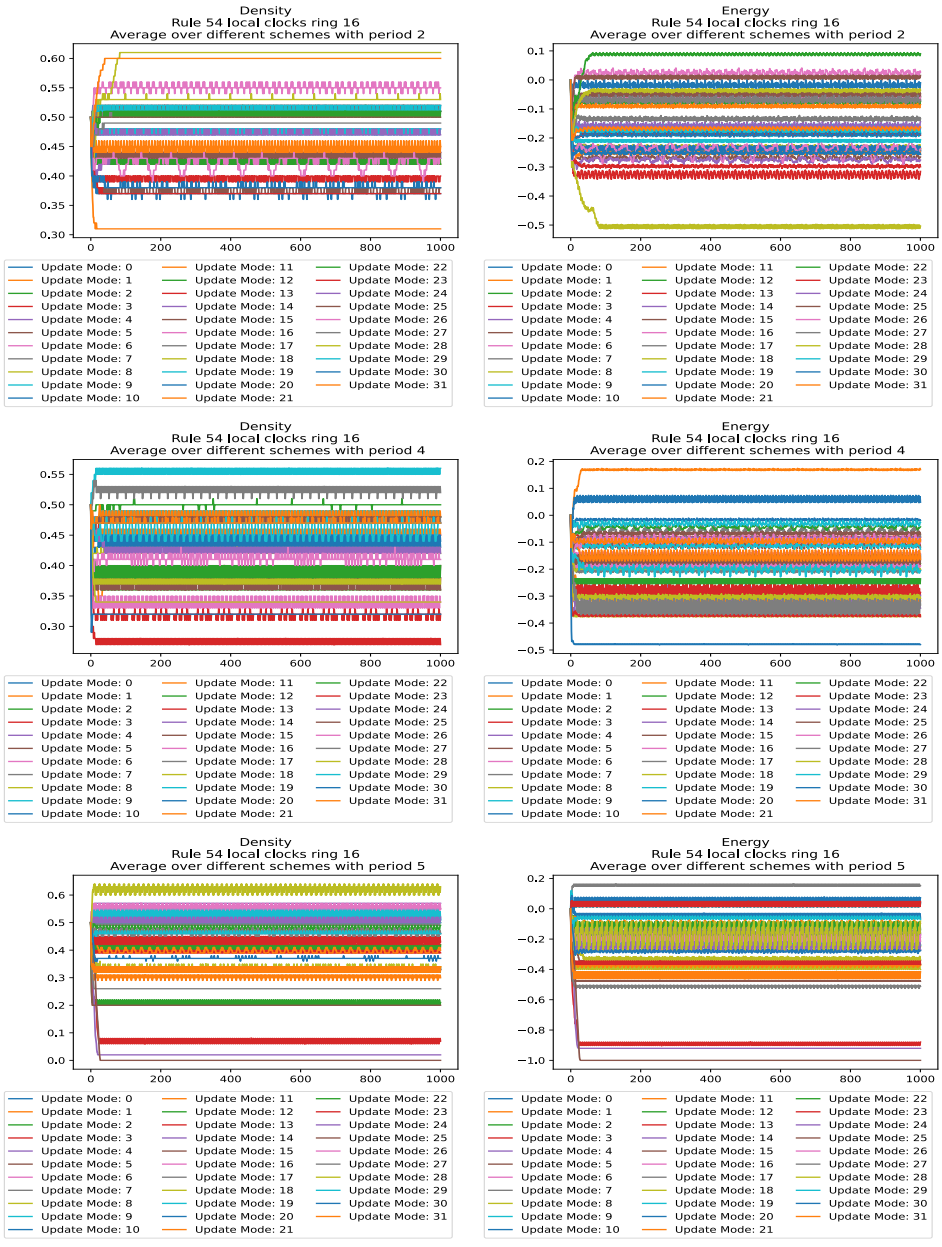

Figure A.25: Density (left) and normalized energy (right) for $n = 16$ under $(54, \text{LC})$ with period 2 (top), 4 (middle) and 5 (bottom), over 1000 time steps.

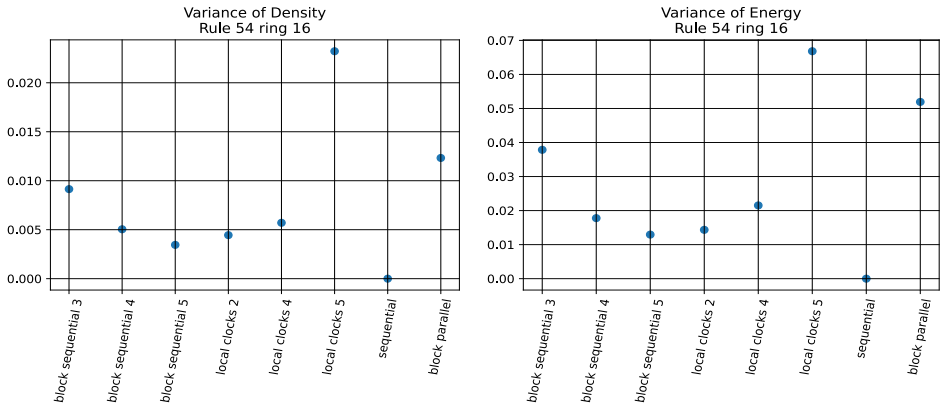

Figure A.26: Variance of Density (left) and normalized energy (right) for $n = 16$ for rule 54.

Appendix A.2. Rule 90

Different sized rings

In the case of Rule 90, it appears that the density and energy are not affected, regardless of the update mode, with the density averaging at around .5 and the energy around 0.

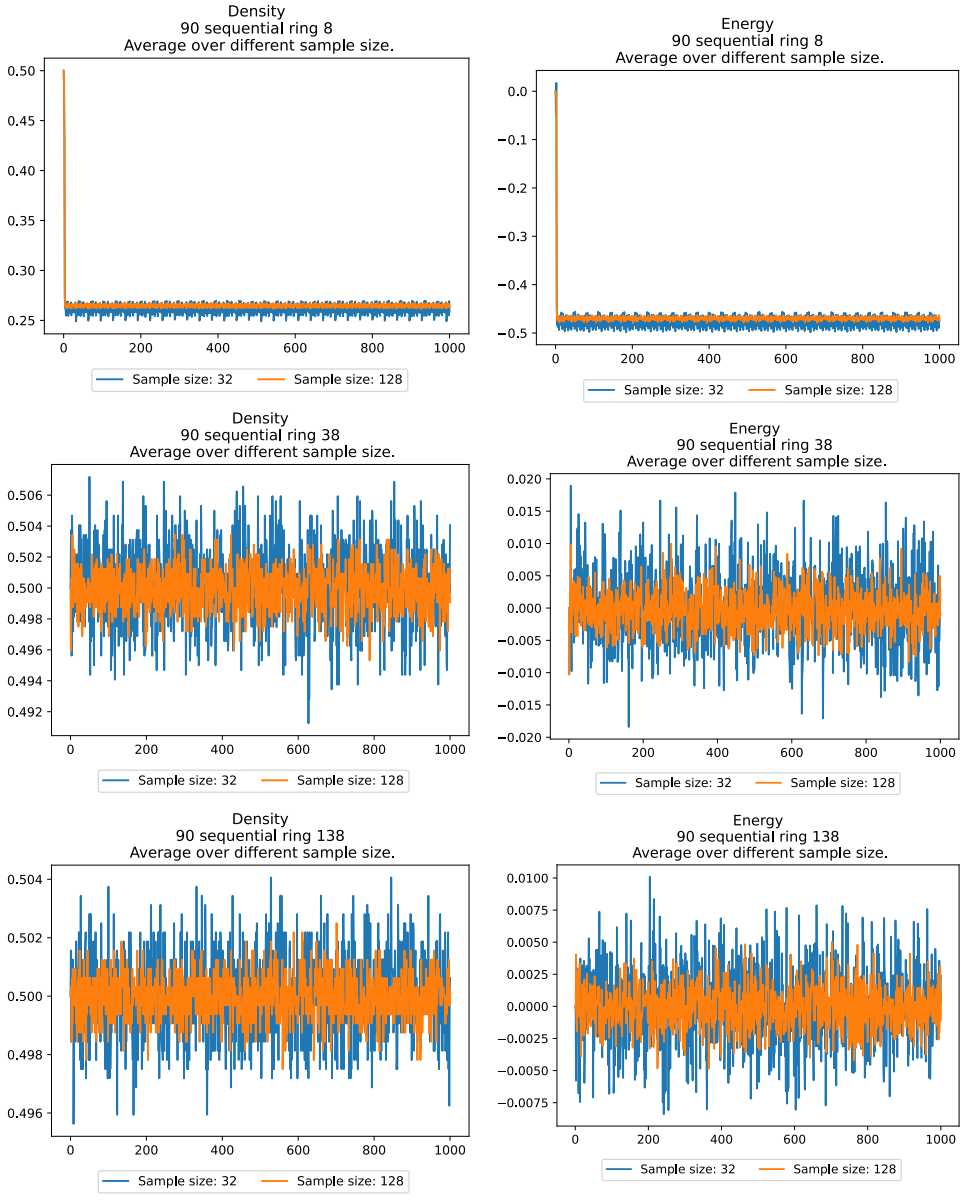

Figure A.27: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 90 with sequential update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

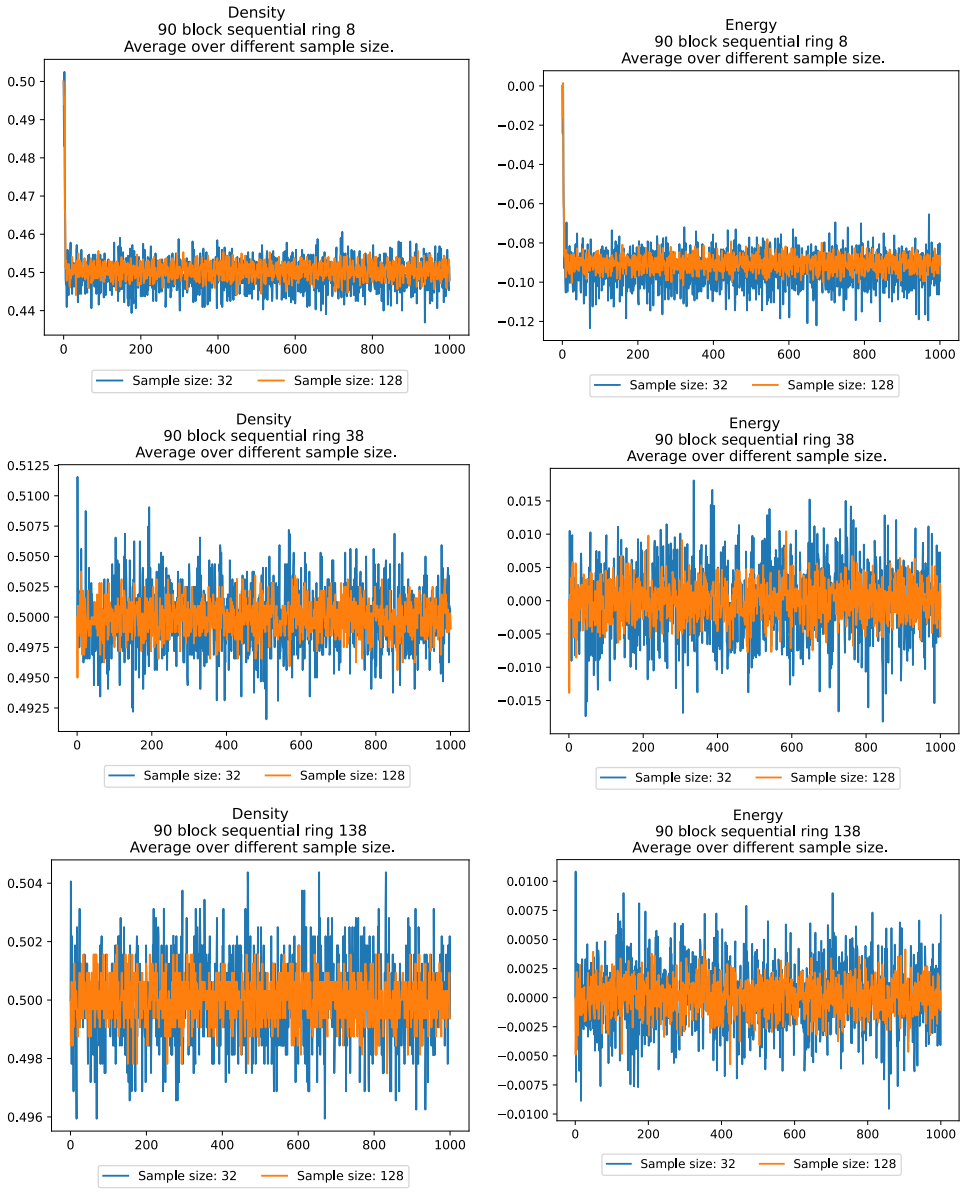

Figure A.28: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 90 with block-sequential update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

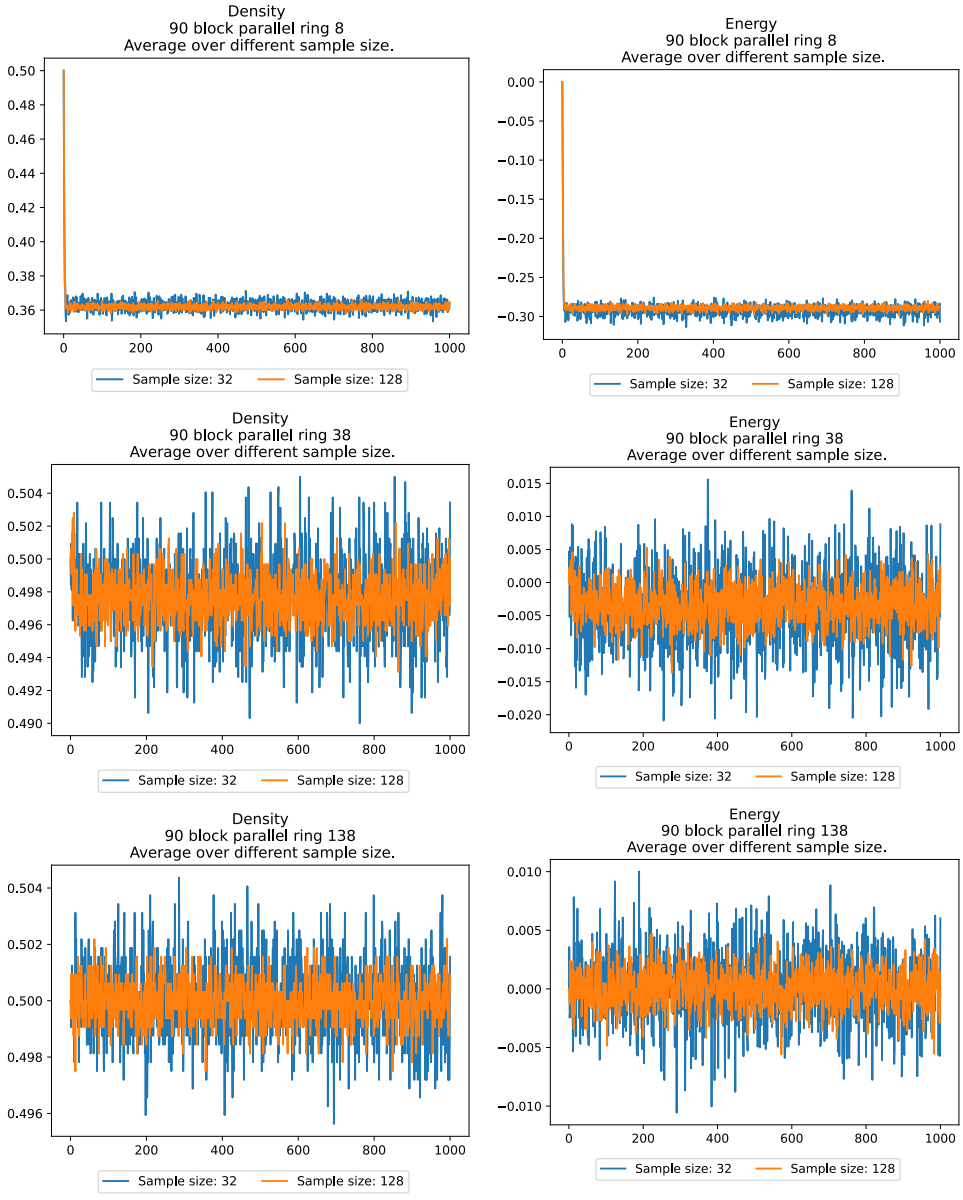

Figure A.29: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 90 with block-parallel update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

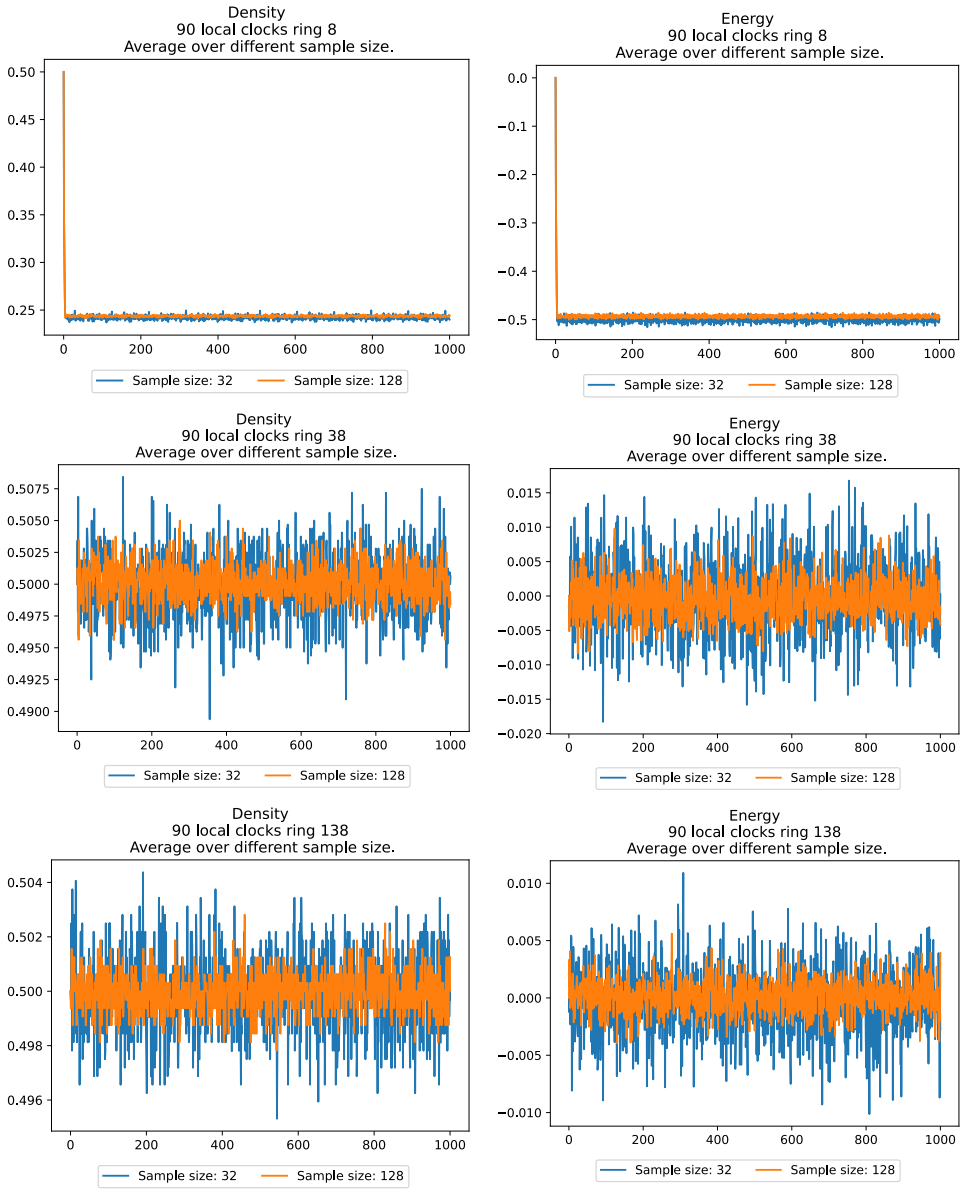

Figure A.30: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 90 with local clocks update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

All configurations of ring size 16

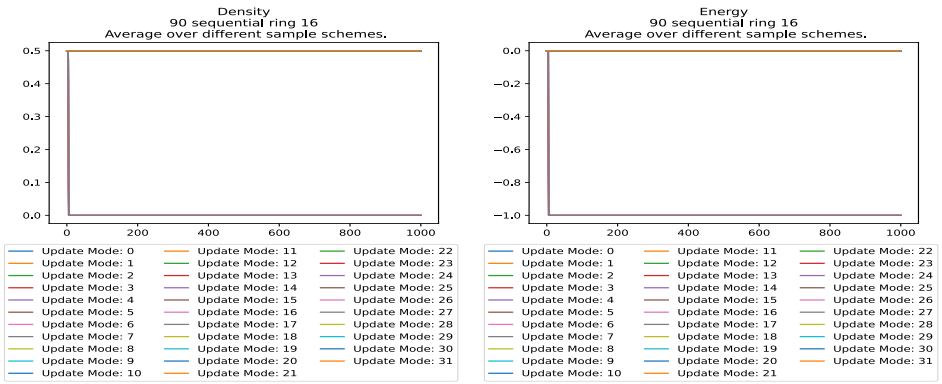

Figure A.31: Density (left) and normalized energy (right) for a ring of size 16 under rule $(90, \text{SEQ})$ average over all configurations, with different update modes, over 1000 time steps.

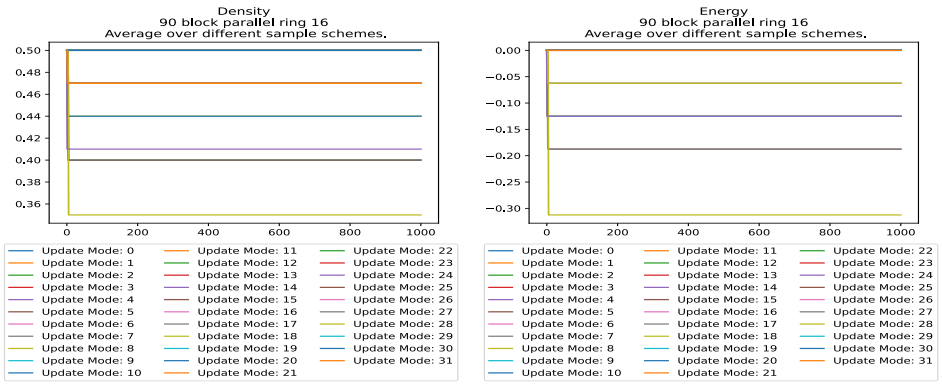

Figure A.32: Density (left) and normalized energy (right) for a ring of size 16 under rule $(90, \text{BP})$ average over all configurations, with different update modes, over 1000 time steps.

Appendix A.3. Rule 150

Different sized rings

In the case of Rule 150, even more notably than with rule 90, we can see that the energy and density are very stable, regardless of the update mode.

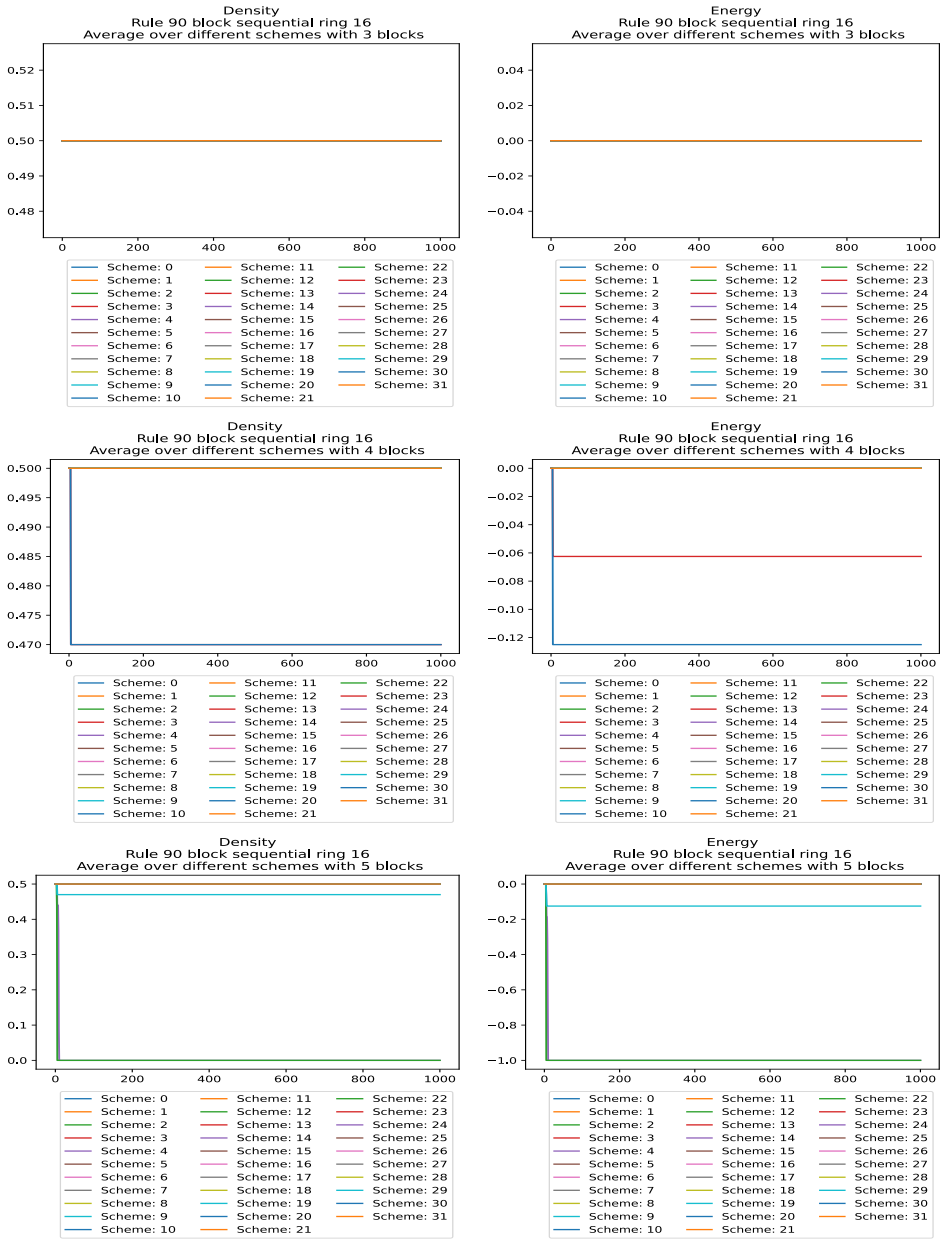

Figure A.33: Density (left) and normalized energy (right) for $n = 16$ under $(90, \text{BS})$ with 3 (top), 4 (middle) and 5 (bottom) blocks, over 1000 time steps.

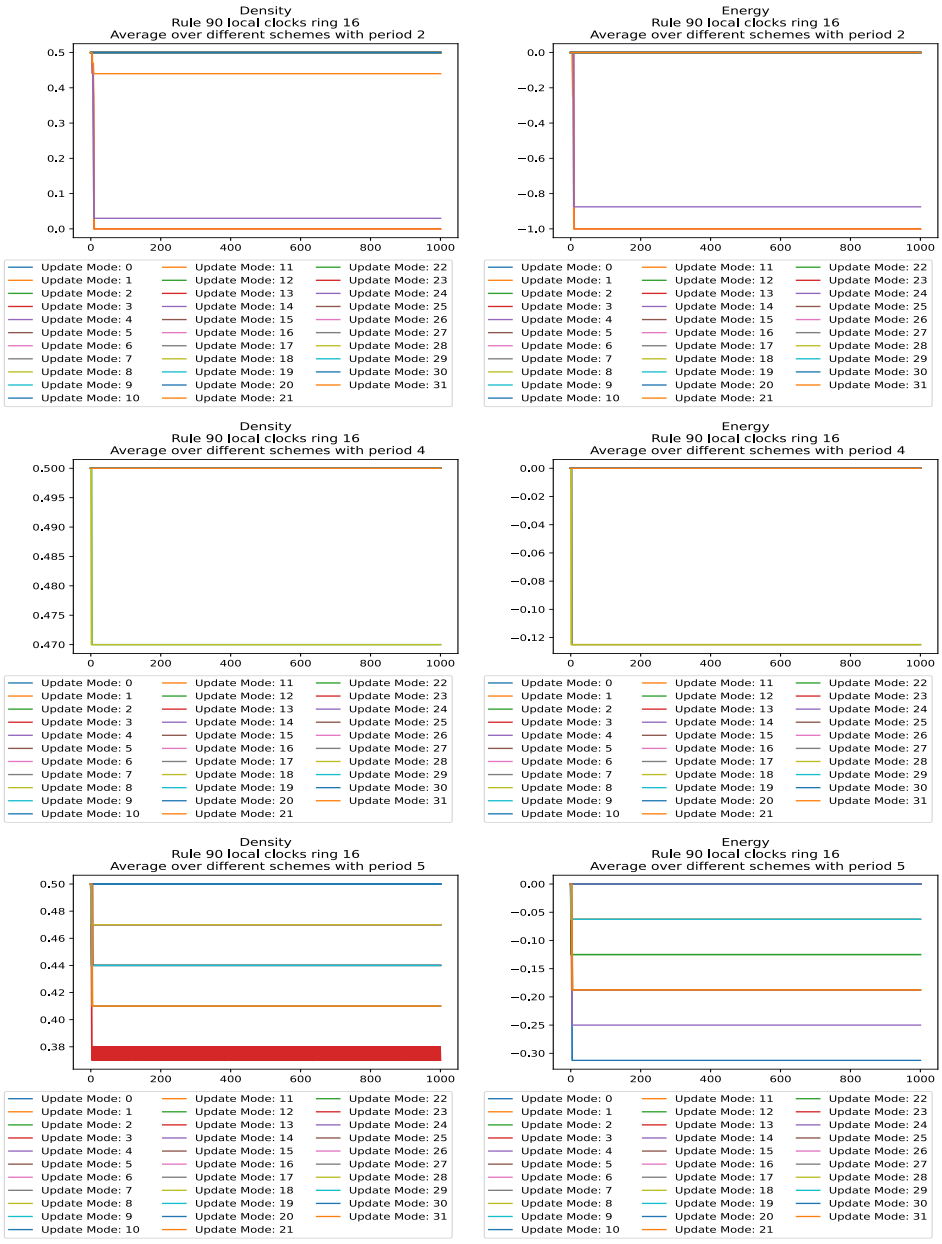

Figure A.34: Density (left) and normalized energy (right) for $n = 16$ under $(90, \text{LC})$ with period 2 (top), 4 (middle) and 5 (bottom), over 1000 time steps.

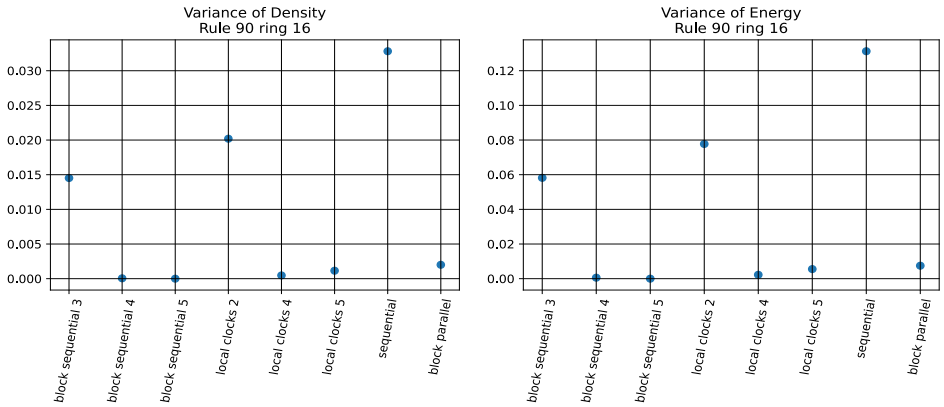

Figure A.35: Variance of Density (left) and normalized energy (right) for $n = 16$ for rule 90.

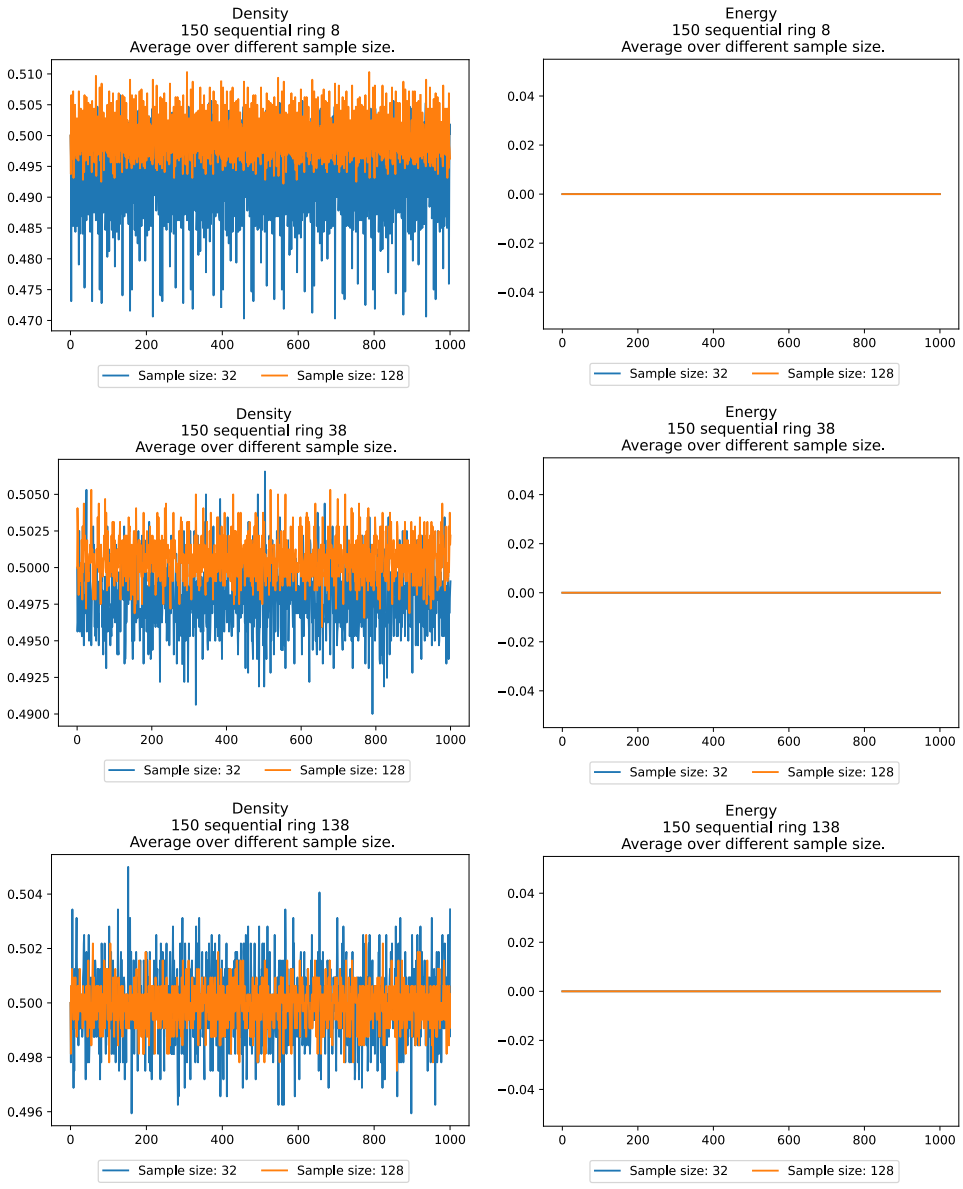

Figure A.36: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 150 with sequential update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

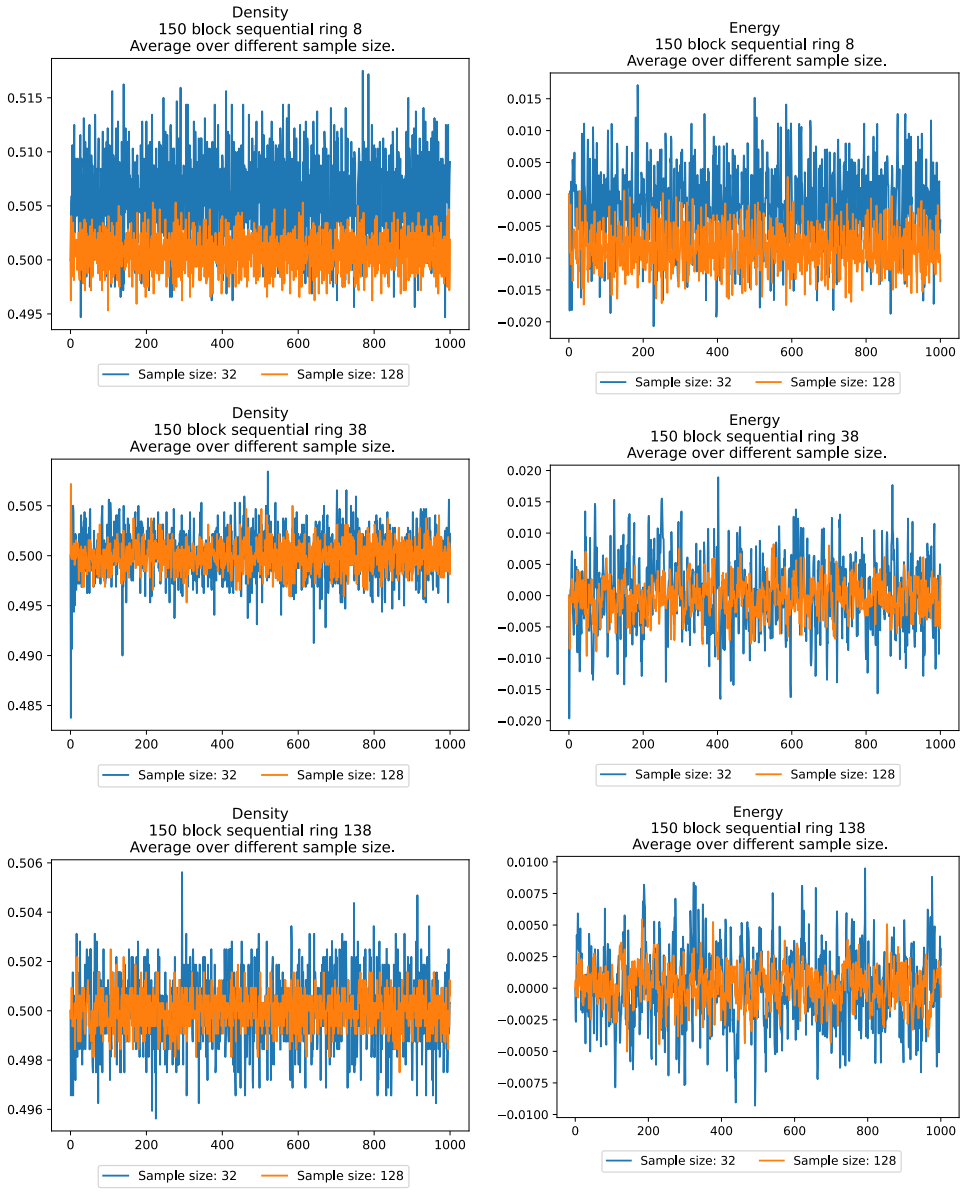

Figure A.37: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 150 with block-sequential update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

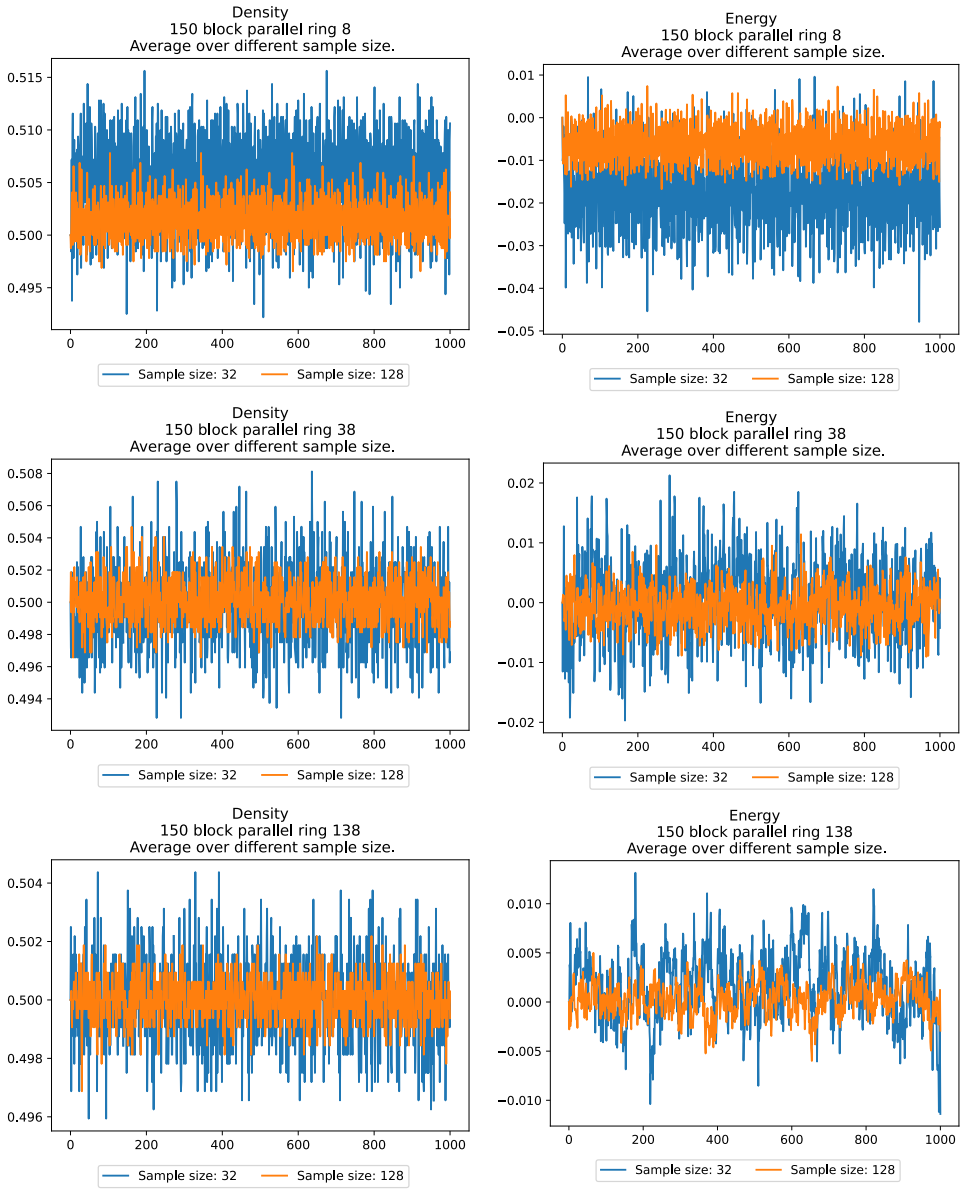

Figure A.38: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 150 with block-parallel update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

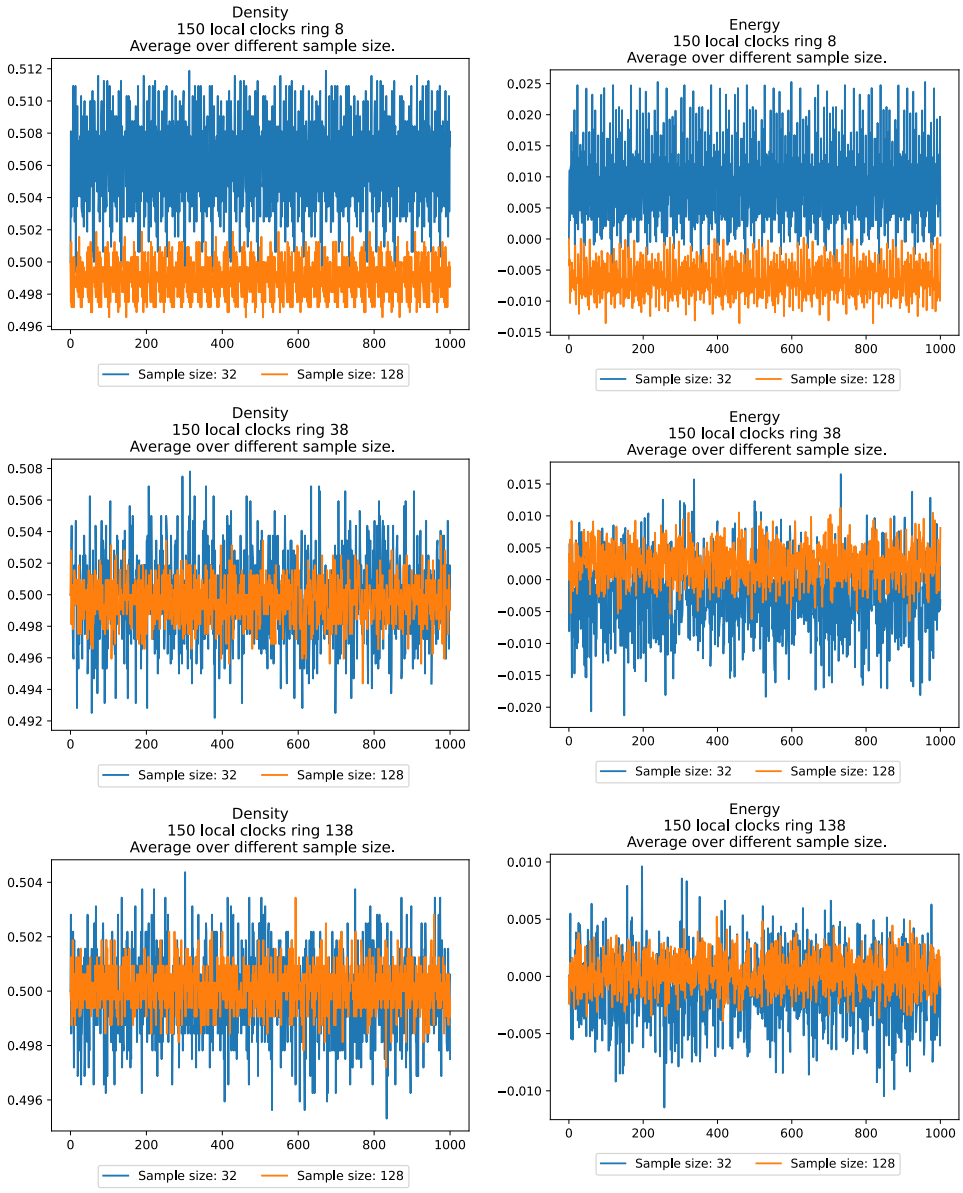

Figure A.39: Density (left) and normalized energy (right) for a ring of size 8 (top), 38 (middle) and 138 (bottom) under rule 150 with local clocks update mode with sample sizes of 32 and 128 initial configurations, over 1000 time steps.

All configurations of ring size 16

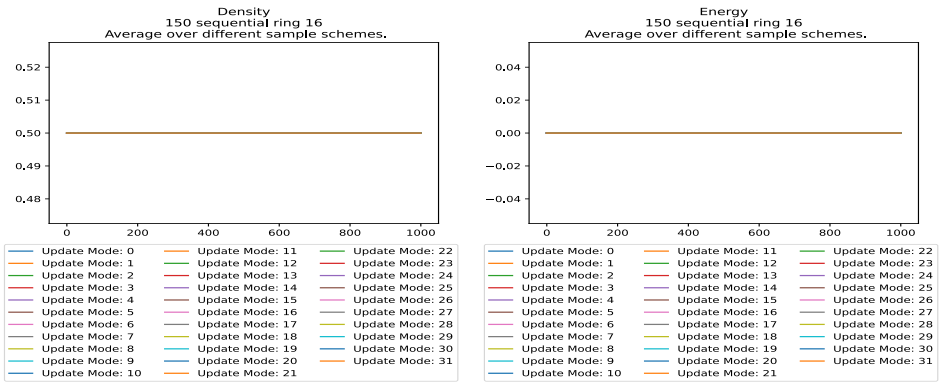

Figure A.40: Density (left) and normalized energy (right) for a ring of size 16 under rule $(150, \text{SEQ})$ average over all configurations, with different update modes, over 1000 time steps.

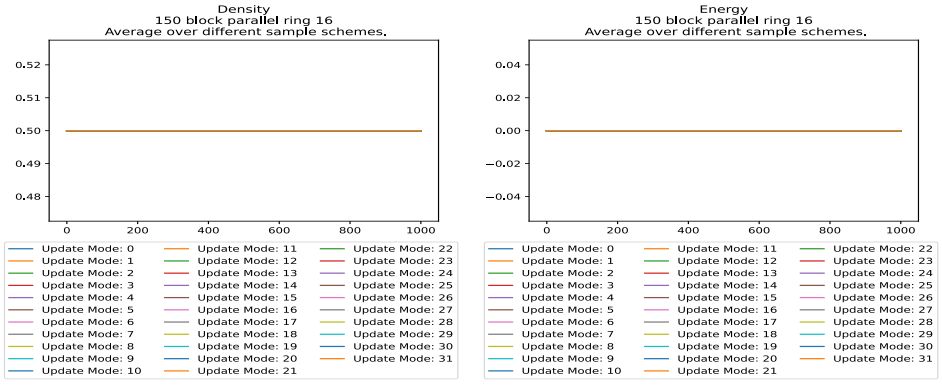

Figure A.41: Density (left) and normalized energy (right) for a ring of size 16 under rule $(150, \text{BP})$ average over all configurations, with different update modes, over 1000 time steps.

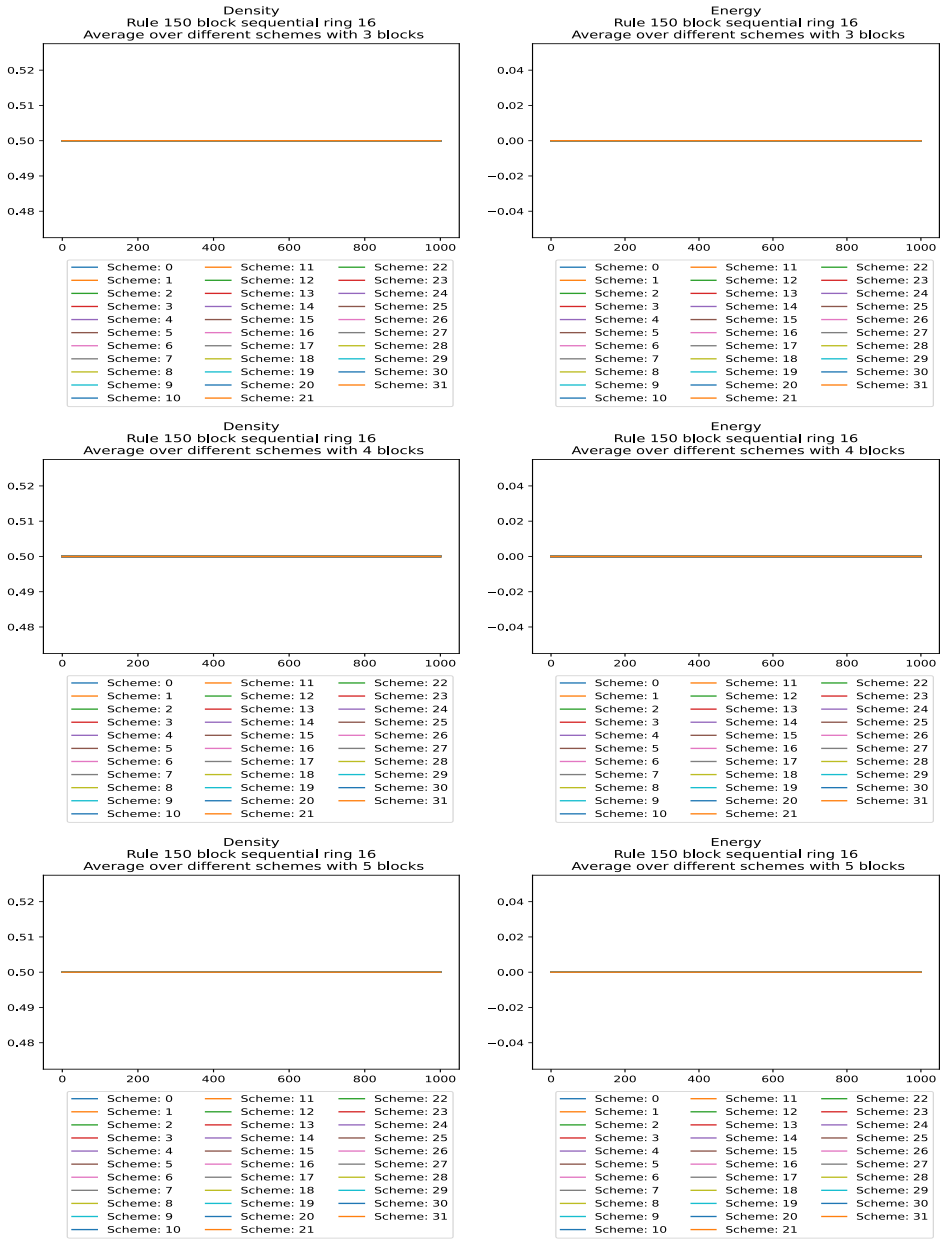

Figure A.42: Density (left) and normalized energy (right) for $n = 16$ under $(150, \text{BS})$ with 3 (top), 4 (middle) and 5 (bottom) blocks, over 1000 time steps.

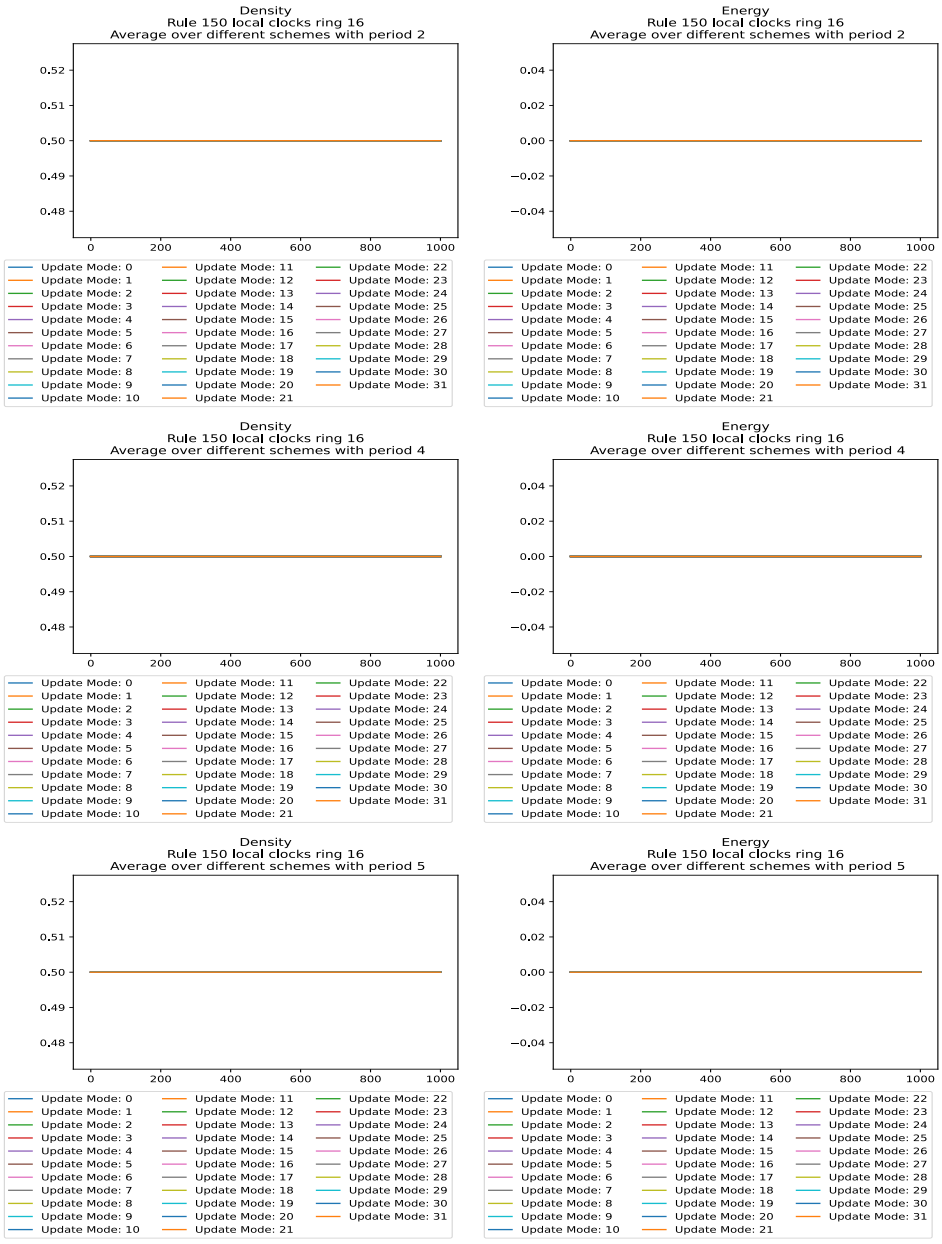

Figure A.43: Density (left) and normalized energy (right) for $n = 16$ under $(150, \text{LC})$ with period 2 (top), 4 (middle) and 5 (bottom), over 1000 time steps.

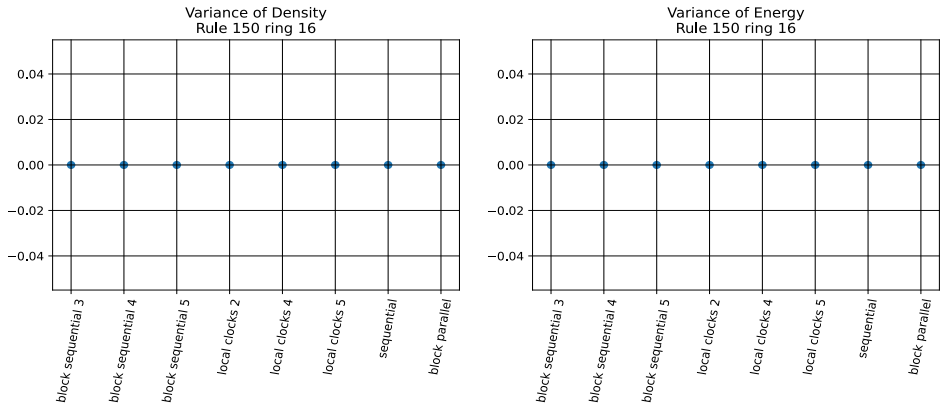

Figure A.44: Variance of Density (left) and normalized energy (right) for $n = 16$ for rule 150.

Rule 110

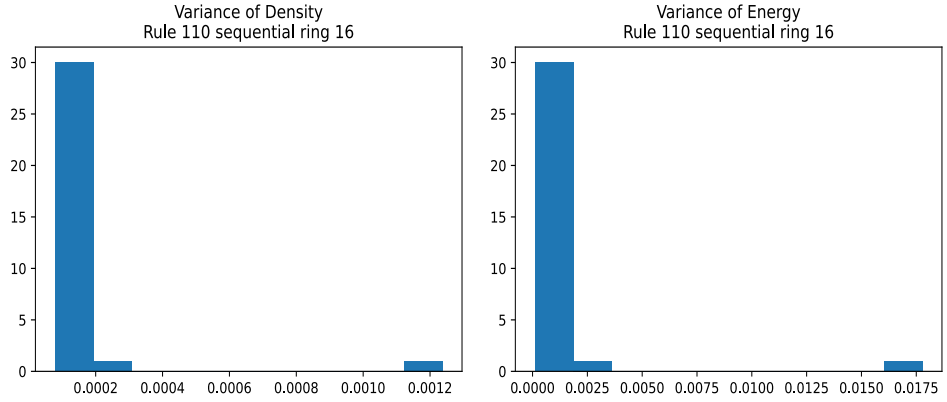

Figure A.45: Histogram of variance of Density (left) and normalized energy (right) for $n = 16$ under $(110, \text{SEQ})$

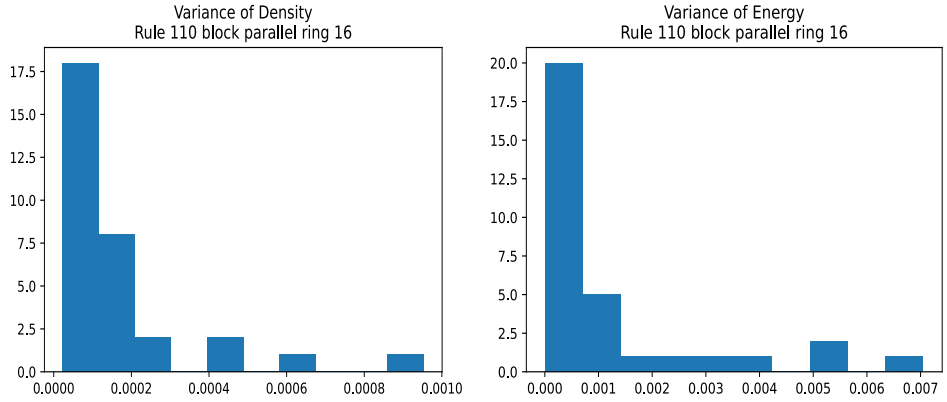

Figure A.46: Histogram of variance of Density (left) and normalized energy (right) for $n = 16$ under $(110, \text{BP})$

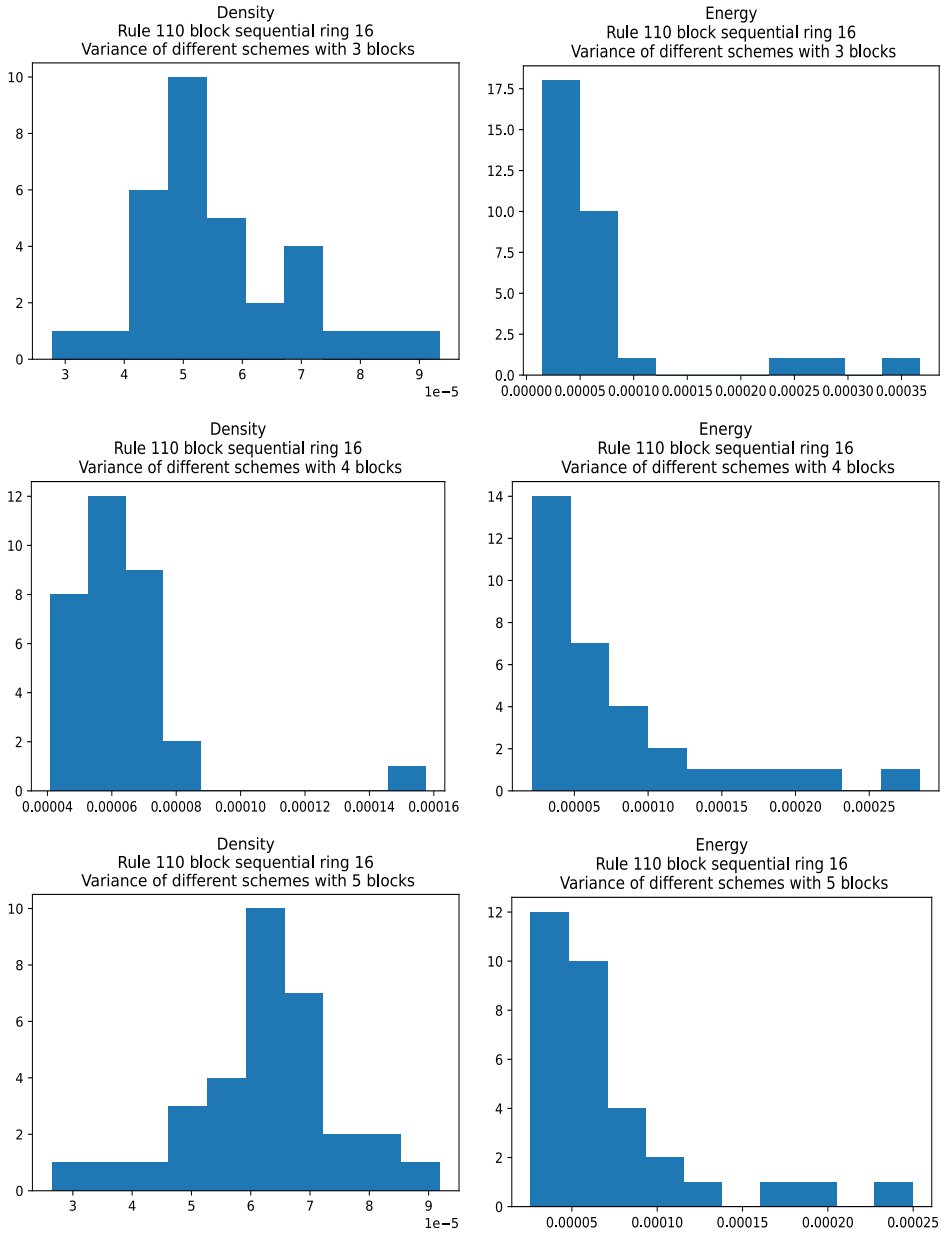

Figure A.47: Histogram of variance of Density (left) and normalized energy (right) for $n = 16$ under $(110, \text{BS})$ with 3 (top), 4 (middle) and 5 (bottom) blocks.

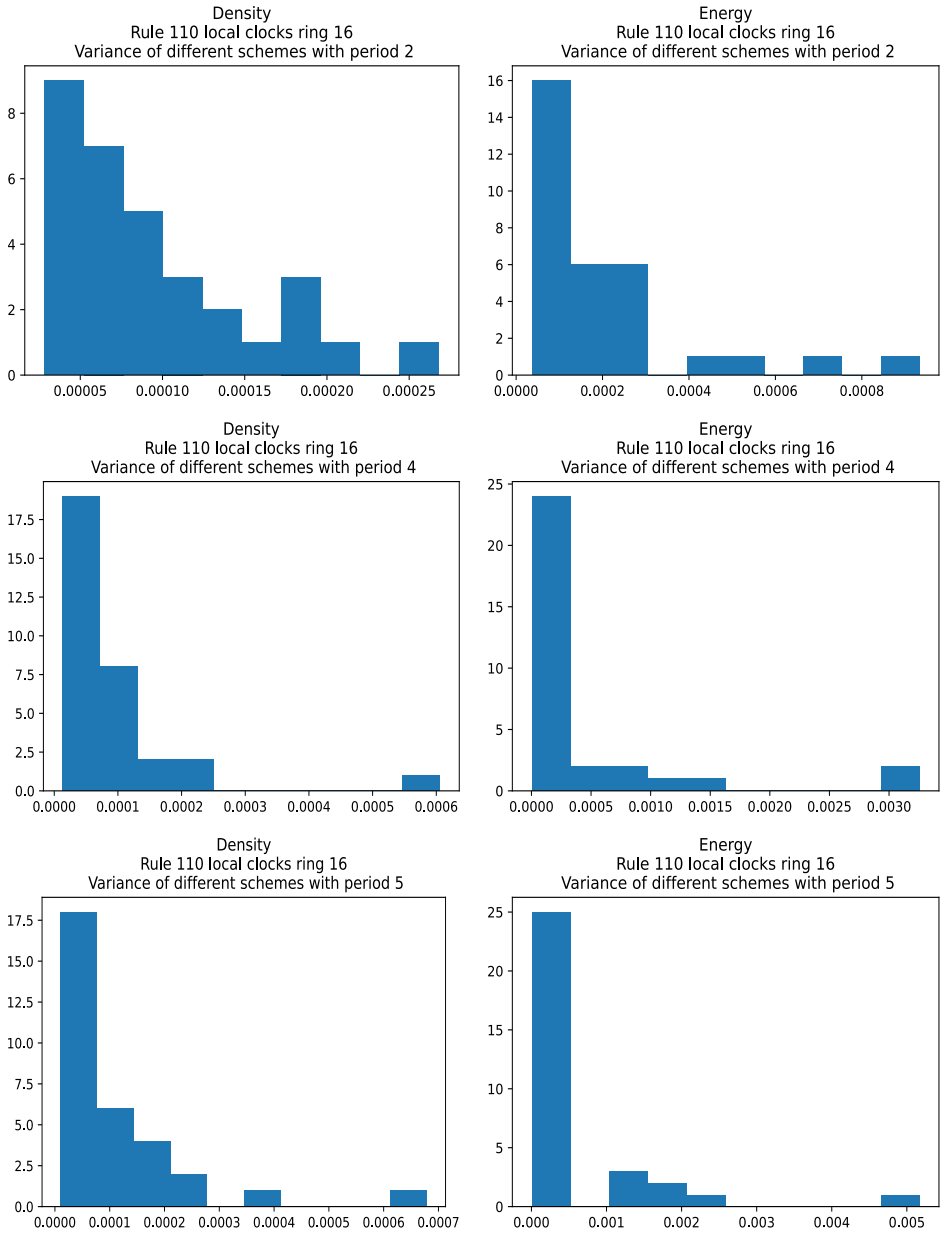

Figure A.48: Histogram of variance of Density (left) and normalized energy (right) for $n = 16$ under $(110, \text{LC})$ with period 2 (top), 4 (middle) and 5 (bottom).

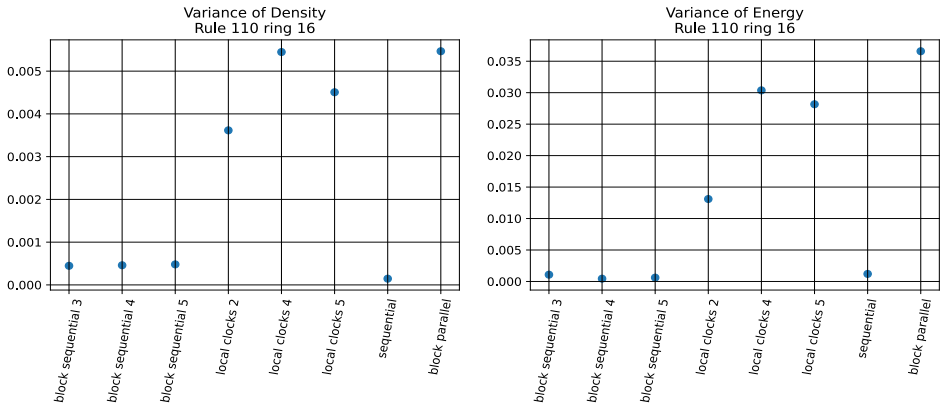

Figure A.49: Variance of Density (left) and normalized energy (right) for $n = 16$ for rule 110.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: