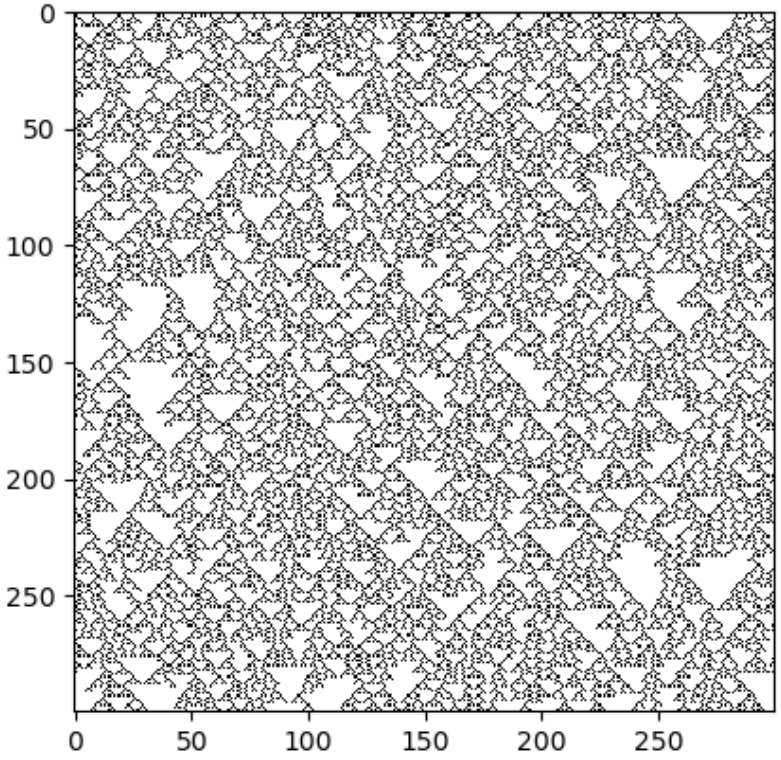
A study of a

Diploid Cellular Automata



First assignment of the course *Introduction to complex systems*

Breebaart Rik
Cristinelli Giacomo
Haoran Dai
Vedder Casper

Utrecht University
November 2021

1 Introduction

A complex system that has been studied extensively over that last 70 years is the Cellular Automata (CA). Cellular automata are models that are discrete both in time and space. Based on some simple rules, which we will discuss more properly later, a whole zoo of patterns and interesting behaviour can emerge. Therefore, since von Neumann first published about them [3] a lot of research has been done on these systems, with seminal work being done by Stephen Wolfram, coining it *a new kind of science* [4].
To make this more mathematically concrete, we need to define what a CA exactly is. A CA is defined by a quadruple $\langle d, \mathcal{A}, \mathcal{N}, f \rangle$, where d is the dimension. $\mathcal{A}$ is a finite set. $\mathcal{N}$ gives the neighbourhood, i.e. the cells that affect another cell directly. f is the ruleset, i.e. a function $f : \mathcal{A}^{\mathcal{N}} \to \mathcal{A}$ that takes for each cell the neighbourhood and then decides the fate of the cell.
A very well known example is Conways *Game of Life*, where a $2D$ grid is evolved in time. The rules are as follows, a cell stays alive (denoted by ■) if in its neighbourhood contains 2 or 3 other alive cells. A cell becomes alive if it has has three living cells in its neighbourhood [1]. A simple time evolution of this model is shown in 1.

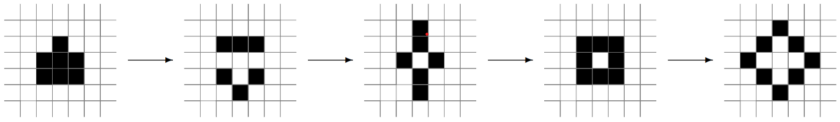


Figure 1: A time evolution in the Game of Life, taken from [1].

1.1 Elementary Cellular Automata

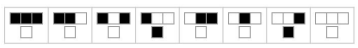
We will consider a simple form of CA, namely *elementary cellular automata* (ECA). This is a CA where the dimension is 1, where we will also only consider two options for the cells (i.e. $\{■, □\}$). These systems were studied very much in depth by Stephen Wolfram. In our previously defined language we then have:

$$ECA = \langle 1, \{■□\}, □□□, f \rangle \tag{1}$$

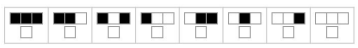
Where our rule is defined by:

$$(F(x))_i = f\left(x_{i-1}, x_i, x_{i+1}\right) \tag{2}$$

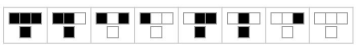
Where x_i denotes the cell we are considering. As for each cell there are 2 possibilities (black, white) and the neighbourhood consists out of 3 cells (left, middle, right) one has $2^3 = 8$ configurations possible for the neighbourhood. The rule then decides for each configuration what happens, either it returns 0 (white) or 1 (black). As it can only return 2 values we have $2^8 = 256$ possible rules. A good way to represent these rules is by using the binary expansion of a number, we can then say that returning white corresponds to 0 and returning black corresponds to 1. For example, rule 18 has as its 8 bit binary expansion 00010010. The order in which these numbers appear then correspond to the configuration by some convention. We can see this clearly in figure 2, where one can also see the convention used.



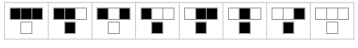
(a) Rule 18



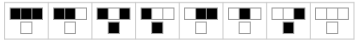
(b) Rule 0



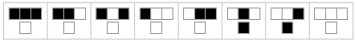
(c) Rule 204



(d) Rule 94



(e) Rule 50



(f) Rule 6

Figure 2: Example of some rules used throughout this paper, rule 0 is also known as the *null rule*, rule 204 is known as the *identity rule*

A simple example of a rule is rule 0, the *null rule*, as it sends every configuration to 0. This rule is shown in figure 2b. Another example is the *identity rule*, rule 204, which sends the configuration to the same configuration, as can be seen in figure 2c.
Although ECA are in principle very simple a lot of complexity can arise from them. As is seen in the following figure:

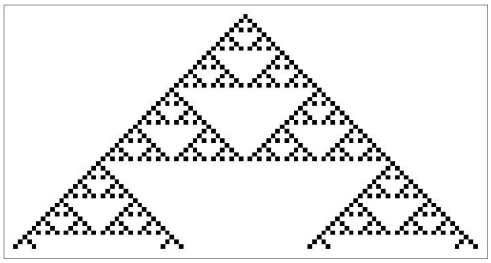


Figure 3: An *ECA* with rule 18 and as starting conditions one black cell.

A question one could ask is how these systems behave with random initial conditions. It turns out, that even though the details of these systems vary, the general behaviour does not vary a lot. This was described by Wolfram in the 80s. Based on the empirical behaviour of these random configurations he classified them in 4 classes [4].

- *Class 1:* Almost all initial configurations lead to the same uniform fixed point configuration
- *Class 2:* Almost all initial configurations lead to a periodically repeating configuration
- *Class 3:* Almost all initial configurations lead to essentially random looking behavior
- *Class 4:* Localized structures with complex interactions emerge.

It is also convenient to define Langton's parameter λ, which we will use later. This is defined as the percentage of all f which map to a non zero state, i.e:

$$\lambda := \frac{m}{2^3} \tag{3}$$

Where m is the number of f that map to a non zero state. For example, for rule 18 we have $\lambda = 2/8$ and for rule 0 it is 0.

1.2 Diploid Cellular Automata

In this context it is also possible to define a probabilistic ECA, where the ECA is Markov chain $(\xi^t)_{t\in\mathbb{N}}$ on the configarution X_n with transition matrix $p(x, y)$:

$$p(x,y) = \prod_{i\in\mathbb{L}_n} p_i\left(y_i \mid x\right) \text{ with } p_i\left(y_i \mid x\right) = y_i\phi\left(x_{i-1}, x_i, x_{i+1}\right) + \left(1 - y_i\right)\left[1 - \phi\left(x_{i-1}, x_i, x_{i+1}\right)\right] \tag{4}$$

Here $\phi\left(x_{i-1}, x_i, x_{i+1}\right)$, can be seen as a rule which determines if the cell becomes 0 or 1. However it can also be values between 0 and 1, it is then understood as the chance the cell will return 1, and $1 - \phi$ as the chance it returns 0. Now we will be considering the following model:

$$\phi = (1 - \lambda)f_1 + \lambda f_2 \tag{5}$$

Here f_1 and f_2 are selected from the 256 rules for normal ECAs. λ then gives the chance of selecting rule f_2 and $(1-\lambda)$ the chance of selecting f_1 (note that this is a different λ than the Langton's parameter). As it depends on two rules this is known as a *diploid ECA*. One can also note that for the cases $\lambda = 0$ and $\lambda = 1$ we obtain again a normal deterministic ECA. We will denote the probability P_x as the probability associated with the process started at $x \in X_n$ and $\mu_t^x(y) = P_x\left(\xi^t = y\right)$ the probability that the chain started at x will be in the configuration y at time t.

In this report we will study such systems for several rules, both theoretically and numerically. In section 2 we will theoretically study the system for a very simple case consisting of the null rule and the identity rule. This simple system allows us to do some actual computations. In section 3 we will study the ECA of rules 18/94 and the null rule in more detail and study whether the system has a phase transition for some critical λ. In our last section (section 4), we will be considering the asynchronous cellular automata, which is a probabilistic ECA that uses the identity rule and another rule.

2 Theoretical part

Consider the Diploid elementary cellular automata introduced in the previous section, i.e. the probabilistic ECA modelled by

$$\phi = (1-\lambda)f_1 + \lambda f_2.$$

In this section, we will analytically study the behaviour of the very simple D-ECA obtain by imposing $f_1 := 0$ (the null rule) and $f_2 := 204$ (the identity rule).

Question 1

For a single cell started at 1, calculate the probability that its state is 0 at time t. For the all chain show that $\mu_t^1(0) = (1-\lambda^t)^n$.

For a Markov chain given by a single cell $\{x^t\}_t$, we have that

$$\begin{aligned} P(x^t = 0) &= (1-\lambda)P(x^{t-1} \in \{0,1\}) + \lambda P(x^{t-1} = 0) \\ &= (1-\lambda)P(x^{t-1} = 0) + (1-\lambda)P(x^{t-1} = 1) + \lambda P(x^{t-1} = 0) \\ &= (1-\lambda)P(x^{t-1} = 1) + P(x^{t-1} = 0). \end{aligned}$$

where $(1-\lambda)$ is the probability to choose rule f_1. We used the fact that f_1 sends everything to 0 and that f_2 sends 0 to 0 and 1 to 1. Similarly

$$P(x^t = 1) = \lambda P(x^{t-1} = 1)$$

where λ is the probability to choose rule f_2. So, using that $P(x^0 = 1) = 1$ we get that

$$P(x^t = 1 | x^0 = 1) = \lambda^t.$$

All together, we obtain that

$$\begin{aligned} P(x^t = 0 | x^0 = 1) &= (1-\lambda)P(x^{t-1} = 1 | x^0 = 1) + P(x^{t-1} = 0 | x^0 = 1) \\ &= (1-\lambda)\lambda^{t-1} + P(x^{t-1} = 0 | x^0 = 1) \\ &= (1-\lambda)\sum_{i=0}^{t-1} \lambda^i = (1-\lambda)\frac{1-\lambda^t}{1-\lambda} = 1-\lambda^t \end{aligned}$$

Where we used a geometric series $\sum_{k=0}^{n} r^k = (\frac{1-r^{n+1}}{1-r})$.

For rule 0 and 204 the next step is independent of neighbours and thus for a chain of n the probability of the full chain to be zeros (starting at all 1's) is

$$\mu_t^1(0) = P(\xi = \{0\} | \xi^0 = \{1\}) = \prod_{i=0}^{n-1} P(x_i^t = 0 | x_i^0 = 1) = \prod_{i=0}^{n-1} (1-\lambda^t) = (1-\lambda^t)^n.$$

(Here we started counting the chain sites at 0).

Question 2

Try to obtain the probability of reaching at time t the configuration with all zeros if started in the configuration with all ones.

Let $\{S^t\}_t$ be the Markov chain corresponding to the number of ones in the configuration $\{\xi^t\}_t$. We want to compute $P(S^t = 0 | S^0 = n)$, i.e. the probability of having a 0 configuration at time t, starting from a configuration with only ones. Therefore, we want to estimate the probability of removing all the ones from the initial configuration in exactly t steps. Let $R = (R_1, ..., R_t)$ be the random vector with $R_i := S^{i-1} - S^i$ consisting of the number of ones removed at time i. It is not difficult to see that R follows a multinomial distribution with parameters n and $p = (p_1, ..., p_t)$ where

$$p_i = (1-\lambda)\lambda^{i-1}.$$

In fact, this p_i is exactly the probability for a single cell to change status at time i starting from 1 as initial condition. Moreover, let $k = (k_1, ..., k_t) \in \mathbb{N}_0^t := (\mathbb{N} \cup \{0\})^t$ be such that $k_1 + ... + k_t = n$. Then, by combinatorial arguments, $\binom{n}{k_1,...,k_t}$ is the number of possible ways of removing n ones with the constraint of creating exactly k_i new zeros at each step i. By independence of the cells, $p_i^{k_i}$ is the probability of loosing k_i ones at time i. Therefore, we obtain that

$$P(R = k) = \binom{n}{k_1, ..., k_t} \prod_{i=1}^{t} p_i^{k_i}.$$

which is the multinomial distribution of parameters n and p. Note also that

$$P(S^t = 0 | S^0 = n) = \sum_{k_1+...+k_t=n} P(R_1 = k_1, ..., R_t = k_t)$$

because the rightmost term sums the probability of removing ones for every possible sequence $\{k_1, ..., k_t\}$. Therefore

$$\begin{aligned} P(S^t = 0 | S^0 = n) &= \sum_{k_1+...k_t=n} P(R_1 = k_1, ..., R_t = k_t) \\ &= \sum_{k_1+...k_t=n} \binom{n}{k_1, ..., k_t} \prod_{i=1}^{t} p_i^{k_i} \\ &= \sum_{k_1+...k_t=n} \binom{n}{k_1, ..., k_t} \prod_{i=1}^{t} ((1-\lambda)\lambda^{i-1})^{k_i} \\ &= \left(\sum_{i=1}^{t} (1-\lambda)\lambda^{i-1} \right)^n \\ &= (1-\lambda)^n \left(\frac{1-\lambda^t}{1-\lambda} \right)^n = (1-\lambda^t)^n. \end{aligned}$$

Where we used the famous Multinomial Theorem

$$\left(\sum_{i=1}^{m} x_i \right)^n = \sum_{k_1+...+k_m=n} \binom{n}{k_1, ..., k_m} \prod_{t=1}^{m} x_t^{k_t}.$$

Question 3

Now, suppose to compute this probability on a time scale diverging logarithmically with n, namely, take $t = \alpha \log n$, for some $\alpha > 0$. Show that, for any $\lambda \in (0, 1)$, if $\alpha < -1/\log \lambda$ then $\mu_t^1(0)$ tends to 0 in the infinite volume limit, whereas it tends to 1 if $\alpha > -1/\log \lambda$.

We want to calculate:

$$\lim_{n\to\infty} (1 - \lambda^{\alpha \log n})^n$$

We can rewrite $\lambda^{\alpha \log n}$ as:

$$\lambda^{\alpha \log n} = e^{\log \lambda^{\alpha \log n}} = e^{\alpha \log n \log \lambda} = n^{\alpha \log \lambda} \tag{6}$$

Thus the limit we need to calculate is:

$$y = \lim_{n\to\infty} \left(1 - n^{\alpha \log \lambda}\right)^n = \lim_{n\to\infty} \left(1 - \frac{1}{n^{-a}}\right)^n$$

Where we put $a = \alpha \log \lambda$. Now instead of calculating this we calculate $x = \log y$ with y our actual limit, later we can then calculate e^x to get our answer. Thus we focus on:

$$x = \log \left[\lim_{n\to\infty} \left(1 - \frac{1}{n^{-a}}\right)^n \right] \tag{7}$$

As at our domain log is continuous we can get the limit outside the log, after this we can do some rewriting.

$$x = \lim_{n\to\infty} \log\left[\left(1 - \frac{1}{n^{-a}}\right)^n\right] \tag{8}$$

$$= \lim_{n\to\infty} n \log\left[1 - \frac{1}{n^{-a}}\right] \tag{9}$$

$$= \lim_{n\to\infty} \frac{\log\left[1 - \frac{1}{n^{-a}}\right]}{\frac{1}{n}} \tag{10}$$

Now to calculate this we use l'Hopitals rule, which we can use as both the numerator as the denumerator is differentiable on the domain we are considering, which then gives us:

$$x = \lim_{n\to\infty} \frac{an^{1+a}}{1 - n^a} = \begin{cases} -1, & \text{if } a = -1. \\ 0, & \text{if } a < -1. \\ -\infty, & \text{if } a > -1. \end{cases}$$

The first two cases are straightforward to see and the last one can be seen by applying l'Hopital another time giving $x = \lim_{n\to\infty} -(1+a)n$ indeed giving $-\infty$ when $a > -1$. The first two cases are also easy to plug in y. We will discuss the case $a < 1$ later. We can now exponentiate to get y.

$$y = \lim_{n\to\infty} \left(1 - \frac{1}{n^{-a}}\right)^n = e^x \tag{11}$$

Which can now calculate by plugging in our results:

$$y = \exp\left[\log\left[\lim_{n\to\infty}\left(1 - \frac{1}{n^{-a}}\right)^n\right]\right] = \begin{cases} e^{-1}, & \text{if } a = -1. \\ e^0 = 1, & \text{if } a < -1. \end{cases}$$

Now for the case $\alpha < -1$ we exponentiate equation 2, getting the limit outside the exponent again we get:

$$\log\left[\lim_{n\to\infty}\left(1 - \frac{1}{n^{-a}}\right)^n\right] = \lim_{n\to\infty} \exp[\frac{an^{1+a}}{1 - n^a}]$$

As the exponent tends to minus infinity, this limit goes to zero. Now piecing together our results we conclude that:

$$\lim_{n\to\infty}\left(1 - \frac{1}{n^{-\alpha\log\lambda}}\right)^n = \begin{cases} e^{-1}, & \text{if } a = \alpha\log\lambda = -1. \\ 1, & \text{if } a = \alpha\log\lambda < -1. \\ 0, & \text{if } a = \alpha\log\lambda > -1 \end{cases}$$

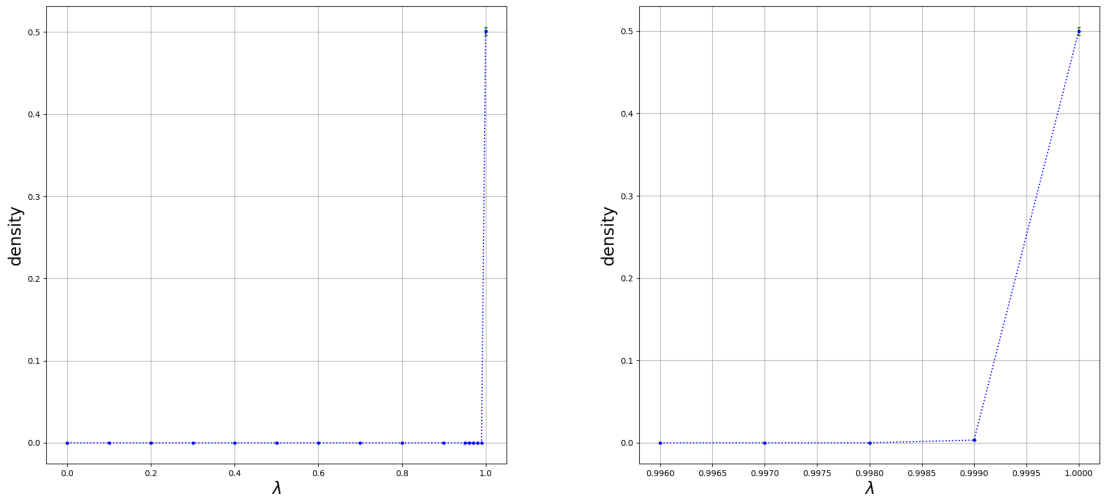
Noting that $\log\lambda$ is always negative (and thus switches the equality sign when divided by on both sides) indeed gives the equalities $\mu = 1$ when $\alpha > -1/\log\lambda$ and $\mu = 0$ when $\alpha < -1/\log\lambda$, as posed in the question.

Question 4

Despite rule 204 is not in the list $\mathcal{F}_2$, does this heuristics shed some light on what it is going on? Come back to the simulation studies you have performed. The previous observation is consistent with what you have observed?

Considering the results from Question 3, where a clear phase transition is observed dependent on the value of λ, we would expect similar behaviour in the simulations. We will further discuss simulation of the D-ECA in the next section, however, it is good to note here that we indeed observe a second order phase transitions in the density of the system when looking at the long time limit and for a system of sufficiently large size in these simulations. So, the observation of these phase transitions observed in the theory for f_2 as rule 204 is consistent with that observed in our simulations which will be discussed in the next section. While more complex rules will be harder to solve theoretically the heuristics of the problem will remain the same (some space points will go to 1 and others to 0). A difference to note however is that in question 3 the phase transition we derived is of first order, while in simulation section we see

second order phase transition. It is also noted that the results from our simulation nicely agree with the theoretical results. When our simulation evaluates the diploid ECA with rule 204, we indeed recognise a first order phase transition as is seen in figure 4. The information about the simulation is explained in the next chapter.



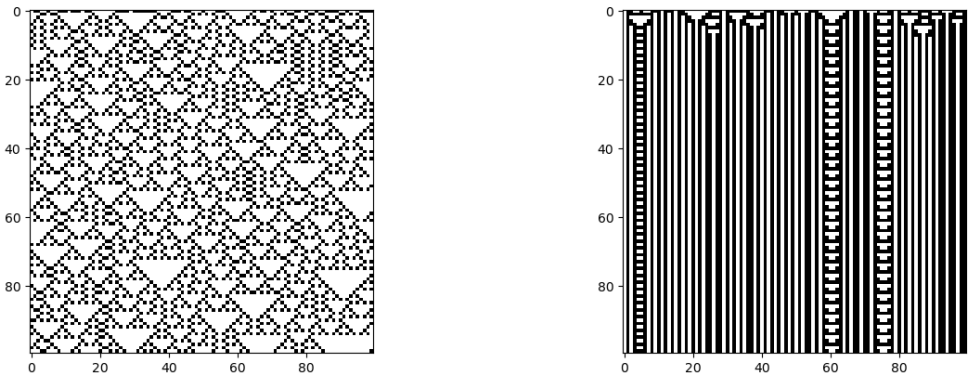
(a) λ ranging from 0 to 1

(b) λ ranging from 0.996 to 1

Figure 4: The results from the simulation for the equilibrium density as a function of λ for rule 204, here the first order phase transition becomes apparent.

3 Simulation

In the previous section, we studied a very simple D-ECA given by a combination of the zero rule and the identity rule. In this section, we will extend the construction to more general D-ECAs by making use of simulations of the model. Specifically, we will consider a couple of examples where the second rule (f_2) of Equation (5) is either rule 18 or 94. Figure 5 shows the space-time evolution of both rules with uniformly distributed initial conditions. Recall that rule 94 and 18 are of Wolfram class II and III, respectively. Therefore, it is reasonable to expect more complex behaviour for the latter. Moreover, the Langton parameter of rule 94 is 5/8 while the one of rule 18 is only 1/4. Hence we can also expect higher values of density of the former.



(a) Rule 18

(b) Rule 94

Figure 5: Space-time grid of the chosen rules.

We will estimate each result by averaging 50 values obtained by different initial configurations. Unfortunately, for computational reasons, the results for the volume limit (see Figure 8) will only rely on 20 different runs. Nevertheless, this will give us an edge on the generality of the results.

These simulation have been implemented in the programming language `python` for which we will briefly explain the implementation in the next section.

3.1 Code implementation

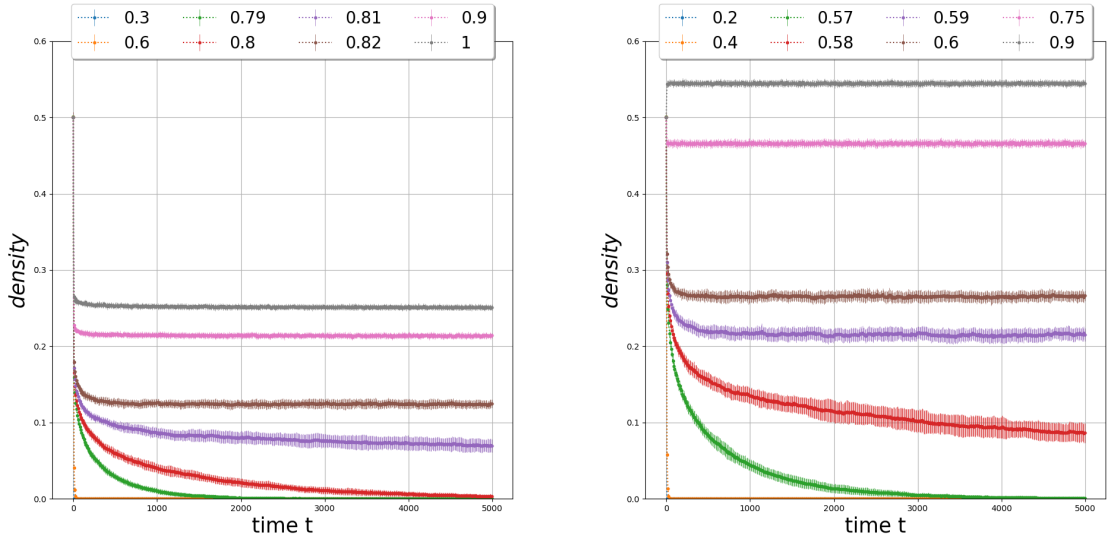
To perform the D-ECA simulation for a given λ for T steps with a grid of size N we first define an initial vector $x_{t=0}$ of N points which all have a 50% chance of being either 1 or 0. Next, to speed up the algorithm the random numbers needed to determine which of the rules would be used in the D-ECA are created before the simulation. This speeds up the computation as computing random numbers can be costly. Now for the simulation of the dynamics of the D-ECA T steps are performed. Here the next state x_{t+1} is determined based on the previous state x_t and the given rules. To determine this next state, a neighbour array is created in which a row i is of the form $(x_{t,i-1}, x_{t,i}, x_{t,i+1})$. Where the $x_{t,i}$ it the grid Boolean at time t and position i. For the elements at the boundary periodic boundaries are used. With this neighbour array, the rules can easily be checked by means of a If-statement along each row. This then outputs a new grid $\tilde{x}_{t+1}$ based on that rule. Now with a probability based on the random numbers initialized before the simulation the next state can be determined with a probability λ using the $\tilde{x}_{t+1,i}$ of rule f_1 and $(1-\lambda)$ using the $\tilde{x}_{t+1,i}$ of rule f_2 at each i. Repeating this process for T steps gives a final configuration. With this implementation the program also be used for a combination of two different rules which we will see to be useful in the case of Asynchronous Cellular Automata (A-ECA).
In the case if the simulation we will be doing we use $N = 10000$ and $T = 5000$ and repeated the simulation a number of times for a given λ to obtain a average and standard deviation of the results. Some portions of a simple version of the code are shown in appendix A.
Every plot that will be shown in the following section is obtain by the `errorbar` command of the `mathplotlib.pyplot` library. It shows the behaviour of the data in terms of the average and the standard

deviation.

3.2 Results

In these simulation studies, we will analyze the behaviour of the chosen D-ECAs for a number of cells equal to $N = 10^4$ and a time-evolution of $T = 5 \cdot 10^3$. In particular, we will focus on the behaviour of the *density* of the chosen D-ECA, i.e. the fraction of cells set to 1 in the final configuration (at time T). This value is massively dependent on the choice of the parameter λ and the choice of the second rule f_2. However, it seems to depend only slightly on the number of iterates T.

Time limit of the density
In Figure 6, the averaged density of different $\lambda - s$ is plotted with respect to the number of iterations.

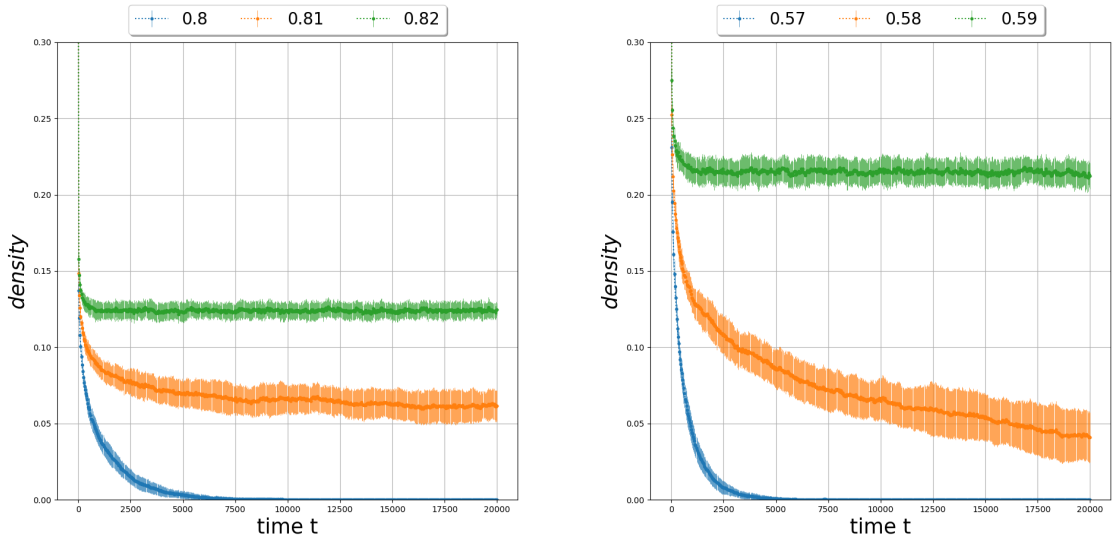


(a) rule 18

(b) rule 94

Figure 6: Behaviour of the density over time (up to T=5000) for the chosen rules and some λ−s specified in the legend. Average from 50 runs.

In both cases it appears that the density of (almost) every λ stabilizes very quickly around a small interval. In particular, we can clearly spot two different types of convergence. An asymptotic fall to zero for every λ below some certain point that depends on the rule and a stable equilibrium for bigger values. These specific points, namely $\lambda_c \sim 0.81$ in Figure 6a and $\lambda_c \sim 0.58$ in Figure 6b, are called *critical* and we will see later their importance to the system. We can already see that the standard deviation at those points tends to stay higher than the rest and the behaviour of the density there appears to be less predictable (less stable). In fact, in both cases, the density at those points seems to decrease over time. It might be plausible that, for $T \to \infty$, this downtrend brings this density to zero, as happens for lower λ-s. This would mean that the "actual" critical values are slightly bigger than those 2, respectively. We investigated more on this claim and we saw for example, in Figure 7a that, on the contrary, the density at $\lambda = 0.81$ seems to stabilize for $T > 5000$. However, in Figure 7b, it is clear that the density at $\lambda = 0.58$ keeps decreasing over time and it is reasonable to expect it to reach 0 at some point.



(a) Rule 18

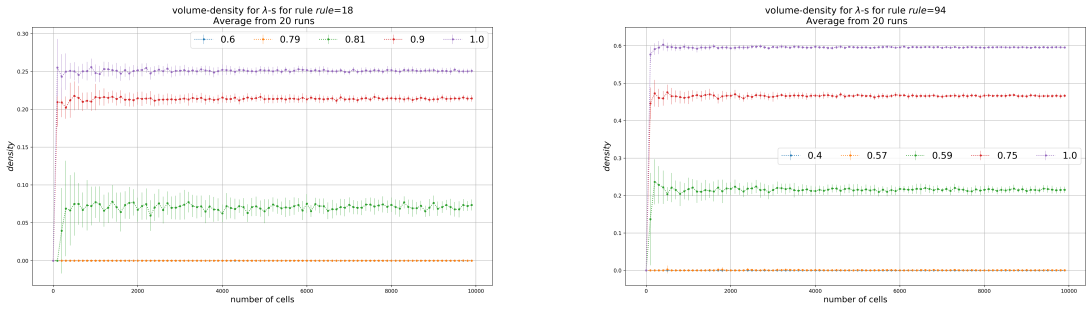
(b) Rule 94

Figure 7: Behaviour of the density over time (up to T=20000) for the chosen rules around the critical value of λ. Average from 50 runs.

In view of these results, we could say that the critical value of rule 18 is slightly smaller than $\lambda = 0.81$, and the one of rule 94 is slightly bigger than $\lambda = 0.58$. This can also be seen by the fact that the value $\lambda = 0.8$ in Figure 6a is not in equilibrium at $T = 5000$ but it rapidly settles to 0 after that, as seen in Figure 7a. On the contrary, for rule 94, the only value of λ that is not in equlibrium at $T = 5000$ is the critical value.

In conclusion, we can say that this system represent a reliable description (with some minor inaccuracies) of the behaviour of the density over time and we can expect, up to a certain level of confidence, the same convergence even for $T \to \infty$.

Volume limit of the density
The behaviour of the density is even more stable in the volume limit.



(a) rule 18

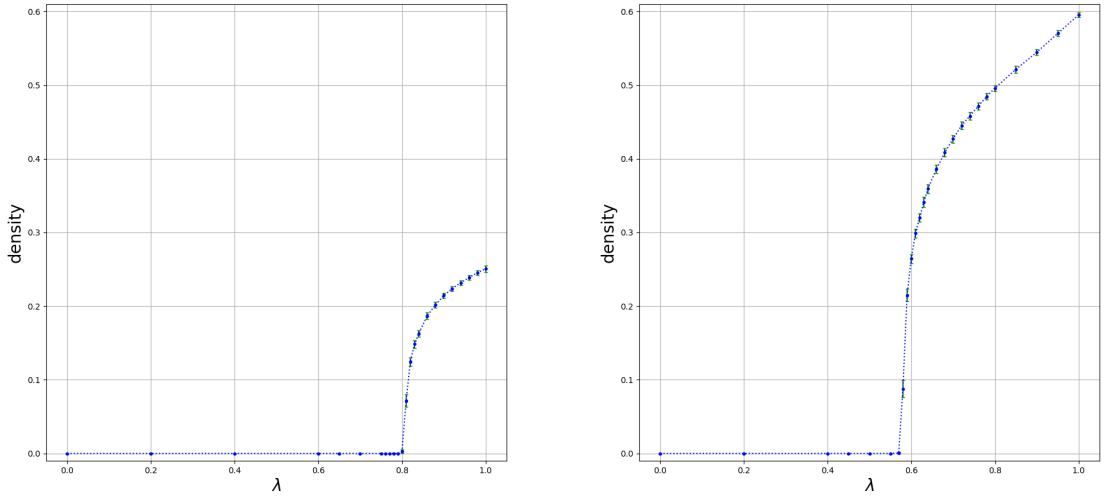
(b) rule 94

Figure 8: Behaviour of the density for the chosen rules and for some values of λ over the number of cells. Average from 20 runs.

The plots in Figure 8a and 8b are obtained by computing the average density of certain $\lambda-$s when the number of cells is increasing with steps of 100 cells from 0 to 10^4. The average density seems to stabilize very quickly for all the values of the λ and already at $n = 2000$ we have reliable estimates of the different

densities.

Generic description of the behaviour of the density:
In Equation (5), we set $(1 - \lambda)$ to be the probability at which the zero rule is applied. This means that the lower λ is, the more likely is that each cell is sent to zero. Therefore, it is reasonable to expect zero density for low values of λ. This is exactly what we encountered when we critically studied the behaviour of the density.



(a) Rule 18

(b) Rule 94

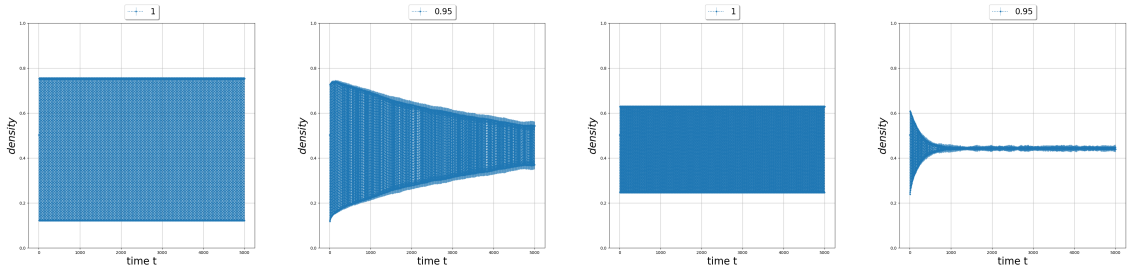
Figure 9: Behaviour of the density over the values of λ for the chosen rules. Average from 50 runs.

In Figure 9, we plotted the average of the $\lambda-$density of the chosen rules. In both cases, the density is 0 for any $\lambda \leq 0.5$. Then, abrupt changes of behaviour occur at $\lambda = 0.81$ in the case of rule 18 (Figure 9a) and $\lambda = 0.58$ in the case of rule 94 (Figure 9b). At these values, both systems seem to undergo a second order phase transition, and, in turn, the standard deviation reaches a peak, which is around 0.008 in the first case and 0.011 in the second one. After these critical points, the $\lambda-$density approaches the Langton's parameter of the relative rule, which is 1/4 for rule 18 and 5/8 for rule 94. The standard deviation at each point in the both curves is very low and fluctuates between 0.003 and 0.007.
This trend is coherent with what we saw in Figure 6a and 6b where there were two type of behaviors; one converging to zero for values smaller than the critical values and one converging to a interval for bigger ones.

Heuristic description of D-ECA with odd rules
In the case of D-ECAs with odd rules as f_2 and the zero rule as f_1, we shall expect a very different behaviour of the density. This is mainly a result of the fact that odd rules send the zero neighbourhood to 1. Therefore, we could roughly say that odd rules and the zero rule compensate each other when coupled in a D-ECA. In particular, we won't have a stable equilibrium at 0 of the density for every value of λ below a "critical" one. Instead, the behaviour of the density will result much more linear than before; the higher λ is, the higher is the probability to send the zero neighbourhood to 1 and bring the density to a positive value.
In the simplest case of all, when f_2 is rule 255, the behaviour of the density is exactly linear in λ. This is not surprising since rule zero sends everything to 0 with probability $(1 - \lambda)$ and rule 255 sends everything to 1 with probability λ. In this example, the new configuration at time t doesn't even depend on the previous one since neither of the rules differentiate the neighbourhoods. Hence, there won't be any standard deviation deriving from different initial configurations.

Another difference with even rules is the following: for some odd rules, we see a periodic behavior of the

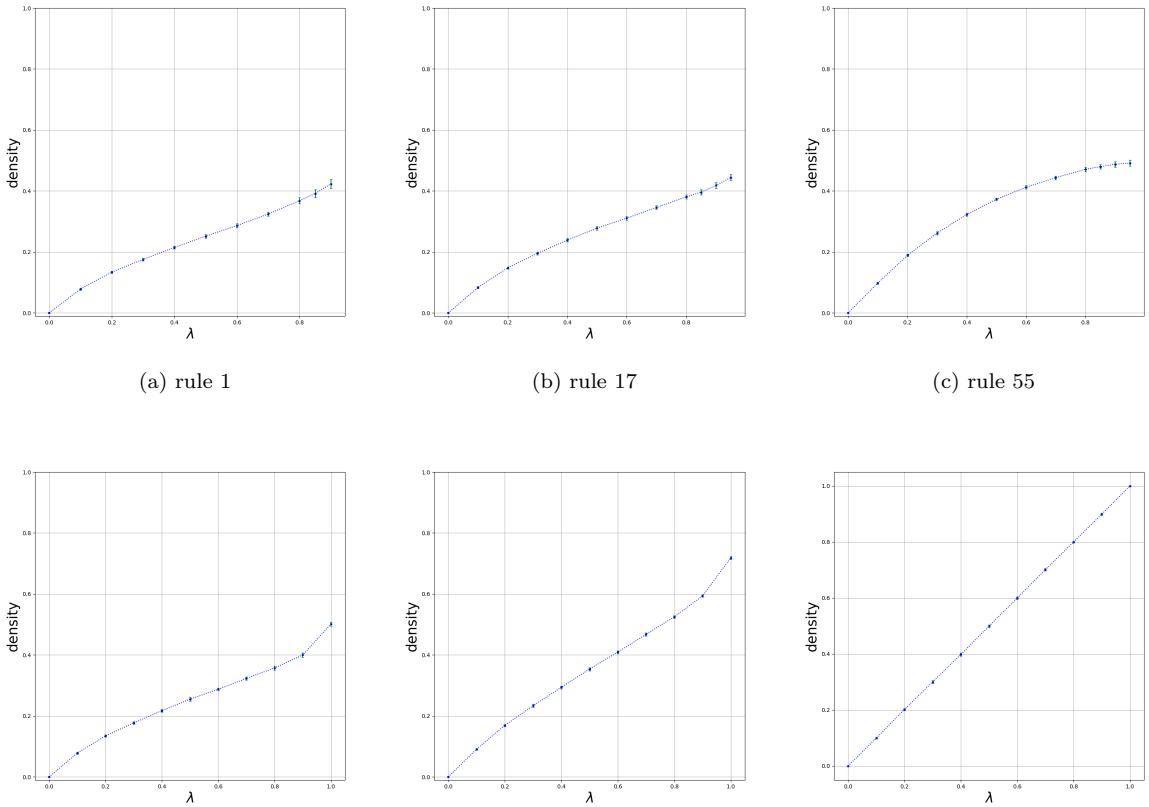
density at $\lambda = 1$. For example, in Figure 10a and 10c, we plotted the average density over time of rules 1 and 17. The density of those rules keeps oscillating between two values.



(a) rule 1 & $\lambda = 1$ (b) rule 1 & $\lambda = 0.95$ (c) rule 17 & $\lambda = 1$ (d) rule 17 & $\lambda = 0.95$

Figure 10: Oscillation and convergence of the density of some odd rules for $\lambda = 1$ and $\lambda = 0.95$ with respect to time (up to $T = 5000$). Average from 20 runs.

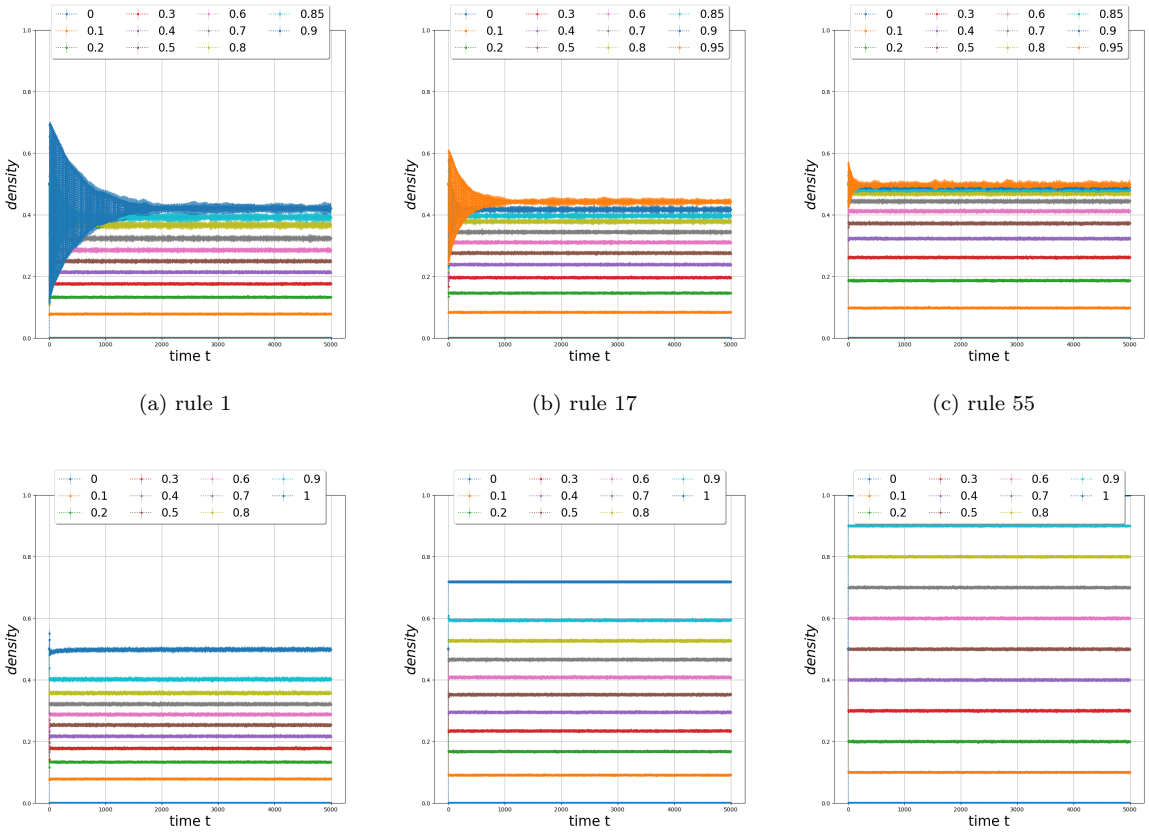
However, this periodic behaviour is something that happens only in the deterministic setting. For $\lambda < 1$, the density seems to converge to a stable equilibrium. In Figure 10b and 10d, we see that the average density of rules 1 and 17 at $\lambda = 0.95$ is already converging into a small interval over time.
For consistency to the main topic of this report, we will therefore focus our attention to values of λ whose density is stable at $T = 5000$.



(a) rule 1 (b) rule 17 (c) rule 55

(d) rule 129 (e) rule 155 (f) rule 255

Figure 11: λ-density plots for some odd rules. Average from 20 runs.



(a) rule 1 (b) rule 17 (c) rule 55

(d) rule 129 (e) rule 155 (f) rule 255

Figure 12: Time-densities for some odd rules and some λ specified in the legend. Average from 20 runs.

It is clear from Figure 11 that most of the $D-ECA$s where the second rule is odd are linear over λ. Although some rules, such as 1, 17 and 55, have non stable behaviour at $\lambda = 1$, the density at any other any other λ converge to a stable equilibrium for $T \to \infty$ that preserves linearity, as shown in Figure 12.

4 Asynchronous cellular automata

As we mentioned earlier, Cellular Automata (CA) are usually simulated under synchronous and deterministic conditions, which is theoretical safe but not realistic. Since the existence of noise is unavoidable in the real world, scientists introduced the Asynchronous CA (ACA) as a tool. Namely, in a conventional CA, the state of every cell in the model is updated together, before any of the new states influences other cells. In contrast, an ACA is able to update individual cells independently, in such a way that the new state of a cell affects the calculation of states in neighbouring cells.

After the proof of ACA's capability to emulate Synchronous CA (SCA) was given [2], scientists followed immediately from results on SCA that ACA are capable of emulating, e.g., Conway's Game of Life, of universal computation, and of self-replication (e.g., as in a Von Neumann universal constructor). Moreover, the general construction and the proof also applies to the more general class of synchronous automata networks (inhomogeneous networks of automata over directed graphs, allowing external inputs, which includes cellular automata as a special case), showing constructively how their behaviour may be asynchronously realized by a corresponding asynchronous automata network.

As the background knowledge for the $1D-ACA$, the one-dimensional ACA with two neighbors, we currently focus on, some research shows that the characteristics of patterns generated by several $1D-ACA$ are drastically changed from uniform patterns to patterns with multiple or chaotic phases when only low level of 'noise' is added.

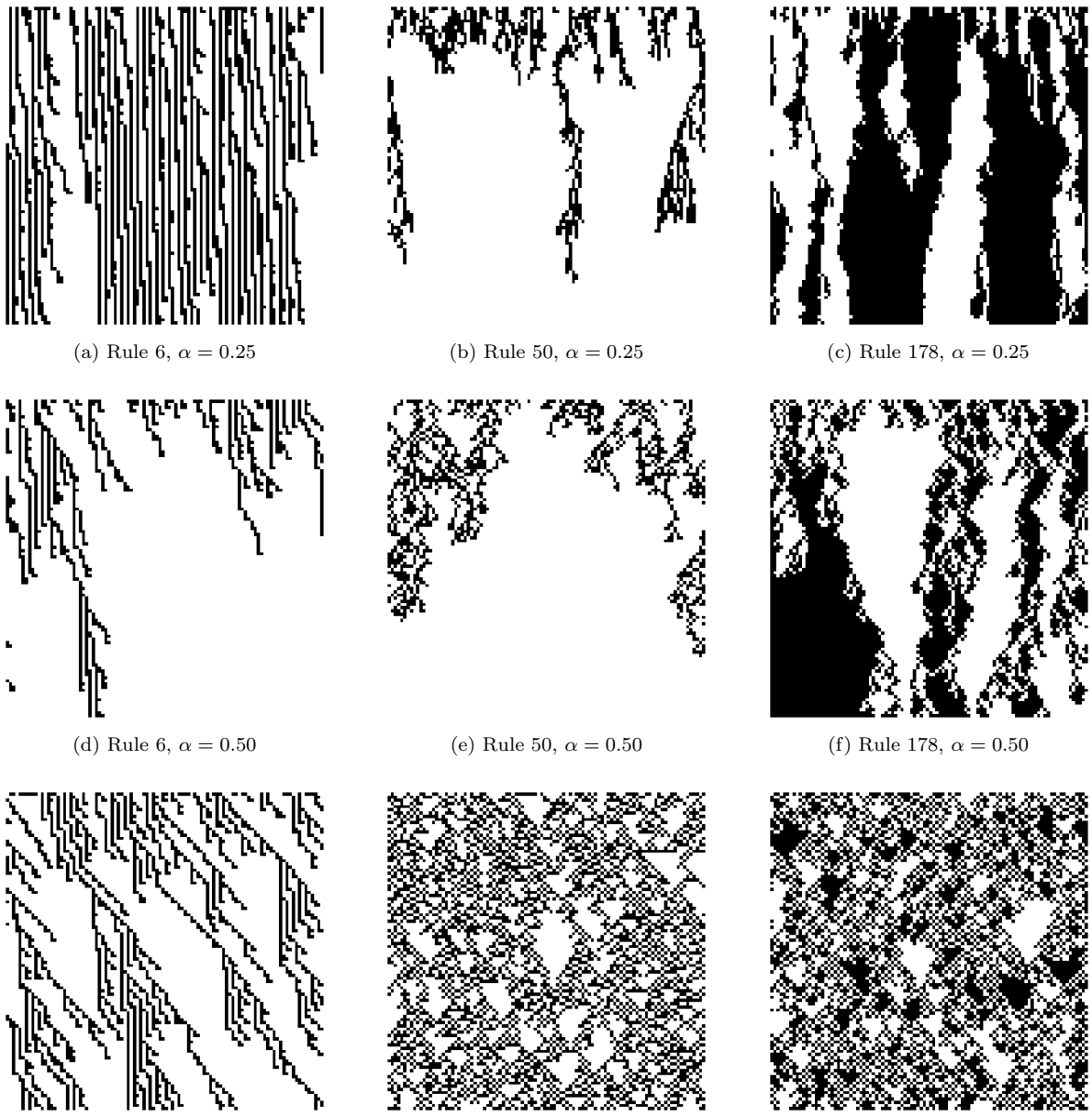
If we treat a $1D-ACA$ model within the Stochastic perspective, it can be taken as a diploid ECA model:

$$\phi = \alpha f_1 + (1-\alpha) f_2 \tag{12}$$

where f_2 is rule 204 (the identity rule). Due to this formulation, we can directly use our $D-ECA$ model with the changed rule f_2. Comparing this with equation 5 we can match our previous model to this one by interchanging f_1 and f_2 and seeing that then $\lambda = \alpha$.

4.1 Results

Space-time diagrams:
Following the same procedure as in the Simulation part, we could also draw the space-time diagrams for ACA. These diagrams are shown in fig.13 for the ACA rule 1 (f_1) being 6,50 or 178 at $\alpha = 0.25, 0.50$ and 0.75. Here again the initial condition was used of 50% likelihood of being either 0 or 1. Looking at the different behaviours observed in the diagrams, we see that the value of α greatly influences the density of the pattern observed. But that the overall structure and evolution of the patterns remain roughly similar between the different values of α.
For rule 6, we can observe a significant drop in the overall density for $\alpha = 0.5$ with an increase in the density of the region observed for $\alpha = 0.75$, however, looking at long time limit (T=5000) both cases go to a density of 0. This will be shown when discussing the final density (fig.15). The density decrease as a function of time is not observed for $\alpha = 0.25$ where the pattern seems to continue. Where for $\alpha = 0.75$ we do observe a decay in density as a function of time. This increase in time steps needed before the space time diagram of rule 6 decays to a zero density state at $\alpha = 0.75$ might explain the final increase at α approaching 1.
For Rule 50, we see that small branches are created which have long strands which eventually die out. But in contrast to rule 6 we observe that rule 50 has a quick decay of the density for $\alpha = 0.25$ instead of at higher α. For rule 50 we observe that the density does not seem to decrease as a function of time for $\alpha = 0.75$ and we would expect this system to have a phase transition between $\alpha \subset [0.5, 0.75]$ whereas the expected phase transition for rule 6 would be between $\alpha \subset [0.25, 0.5]$.
Now looking at rule 178, we see critically different behaviour in the patterns observed compared to rule 6 or 50. The patterns seem to have large regions of complex structures which collectively continue. There also seems to be no clear decay in density throughout the time evolution of the system. We do observe a change in behaviour for $\alpha = 0.75$ where smaller structures are created in the system. For this rule the density remains roughly that of the initial condition (around 50% equal to one).



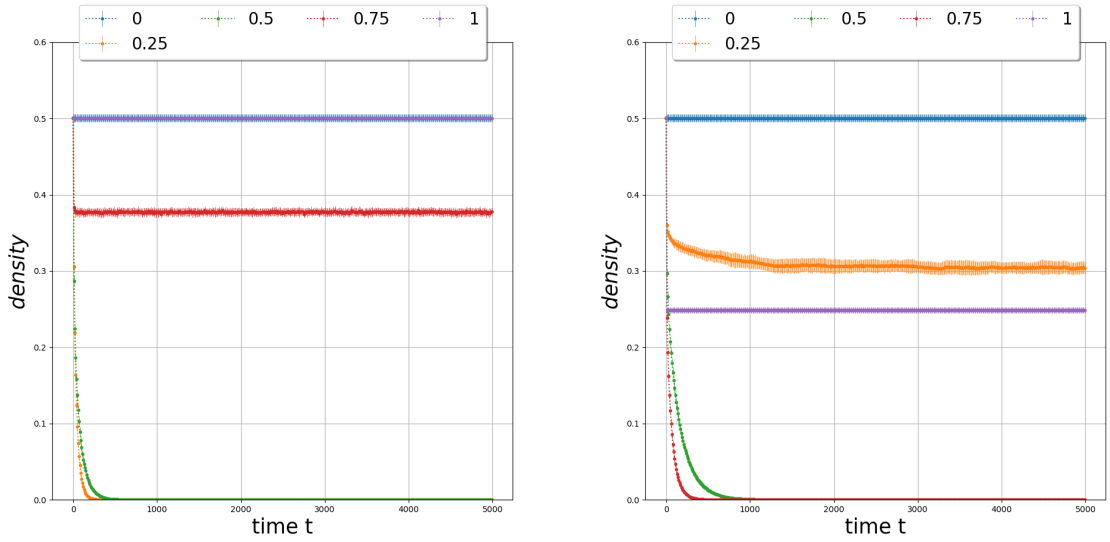
(a) Rule 6, $\alpha = 0.25$ (b) Rule 50, $\alpha = 0.25$ (c) Rule 178, $\alpha = 0.25$

(d) Rule 6, $\alpha = 0.50$ (e) Rule 50, $\alpha = 0.50$ (f) Rule 178, $\alpha = 0.50$

(g) Rule 6, $\alpha = 0.75$ (h) Rule 50, $\alpha = 0.75$ (i) Rule 178, $\alpha = 0.75$

Figure 13: Space time diagrams (Time on y-axes and space on x-axes) of the ACA where rule 2=204 and rule 1=(6,50,178) at $\alpha = 0.25, 0.5, 0.75$. The diagrams are a zoomed in region of 100 by 100 starting at $T = 0$ and $N = 5000$ taken from the full space time grid.

Time and volume limit:
Analogous to the previous section we can study the density of this model. We will study two different ACAs, namely the one with rule 50 and the one with rule 6. We can use our previously described model to do this. First we investigate whether is is safe to assume that the system is in equilibrium at $T = 5000$. We explore this by telescoping the time evolution for some judiciously chosen α's that span the whole system. This is shown in Figure 14a and 14b. Also, similar to the discussion in section 3, the system settles in equilibrium very quickly as a function of the volume.



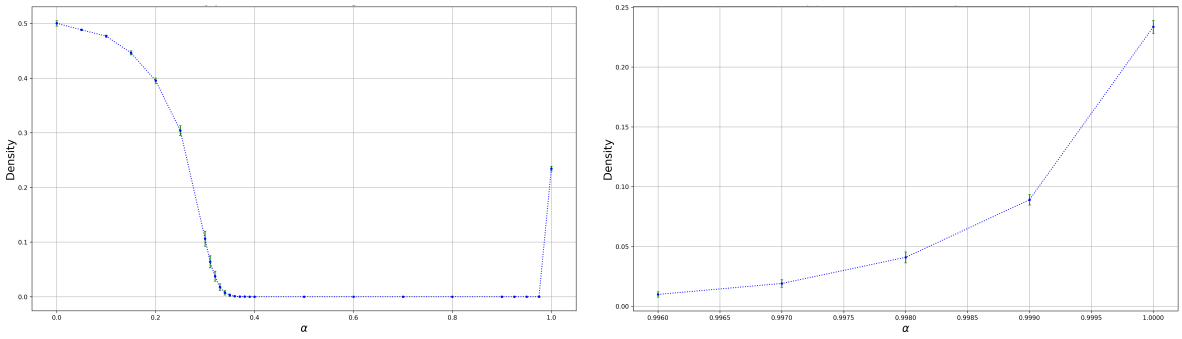
(a) Rule 50

(b) Rule 6

Figure 14: Time evolution of the density of the ACA with rule 50 and rule 6

We find that the system quickly settles in an equilibrium position for these values of α, the values that go to zero do that long before $T = 5000$. The cases also agree with Figure 13. Thus for now we assume the system to be in equilibrium at $T = 5000$, we will revisit this question later for values of α around the phase transition that there turns out to be.

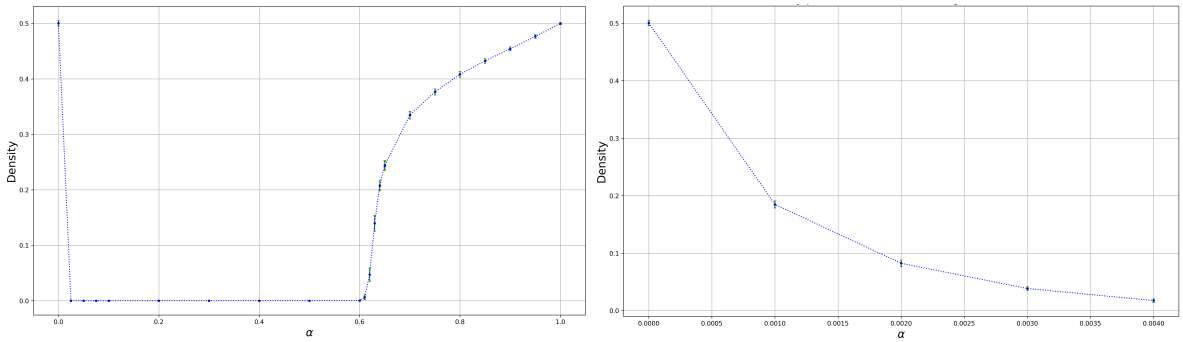
Equilibrium density:
We now proceed to calculate the density at $T = 5000$, this is shown in Figure 15a and 16a for the corresponding rules.



(a) α ranging from 0 to 1

(b) α ranging from 0.996 to 1

Figure 15: The density of the ACA with rule 6 at $T = 5000$



(a) α ranging from 0 to 1

(b) α ranging from 0 to 0.004

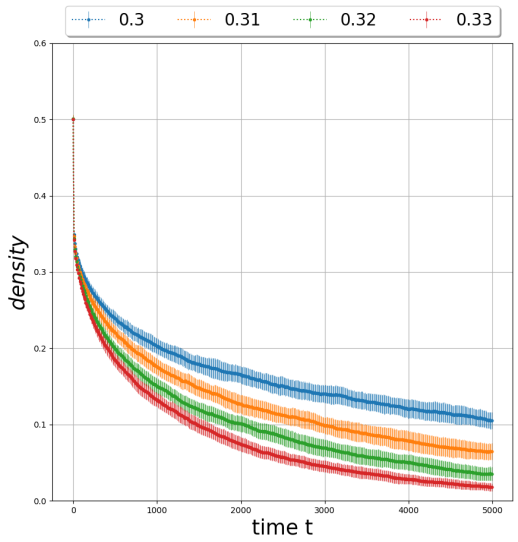
Figure 16: The density of the ACA with rule 50 at $T = 5000$

First we consider rule 6. We can see a second order phase transition taking place around $\alpha = 0.34$ and a second order phase transition at $\alpha = 1$. At first glance one might think that the phase transition at $\alpha = 1$ is first order, judging from Figure 15a, however when zooming in one sees that indeed the phase transition seems to be continuous[1] (see Figure 15b) . The behaviour can heuristically be understood as follows. The system continuously starts at a density of 0.5 as this corresponds with the identity rule ($\lambda = 0$). As the system moves away from the identity rule (i.e. λ increases) the interplay between both rules becomes important. The interplay between the rule 204 (identity) and rule 6 is one that one intuitively expects to lead to a lower density as it breaks the order from both rules. Both rule 6 and 204 fall in the Wolfram category 2, giving periodic behaviour at some point. However in this ACA this periodicity will compete with the random interplay between rule 6 and the identity rule. If this interplay is big enough, this will make the ACAs density go to zero. Thus the interplay between the two rules leads in general to a lesser density than one would naively expect from considering the densities of the sole rules. This again becomes clear as λ approaches 1 and the density rises again as we recover the density of rule 6. Concretely in this example we see that if we start with the identity ($\lambda = 0$), the density decreases slowly as rule 6 becomes more important (increasing λ), indicating a second order phase transition. However if we start with rule 6 ($\lambda = 1$), we see that if the identity rule becomes only slightly important (λ decreases a bit) the density directly goes to zero (albeit still continuously). Thus while rule 6 is only slightly destructive to the density of the identity rule (rule 204), the identity rule is very much destructive to the the density of rule 6.
We can also note that this result matches with the pictures shown in Figure 13. Another sanity check is the case $\alpha = 0$, which gives a density of 0.5, this is expected as when $\alpha = 0$ we just have the identity rule.
Our analysis of rule 50 is very much similar to rule 6. Again we can note that the random interactions between the identity (rule 204) and rule 50 is destructive to the density. However the way they interact is different. A dominant rule 6 with an increasing rule 204 gave a slow second order transition and a dominant rule 204 with an increasing rule 6 a more abrupt second order transition (figure 16b). This is the other way around for rule 50. A dominant rule 204 with increasing rule 50 is a very abrupt second order[2] transition with critical α=0, as can be seen by Figure 16b. While a dominant rule 50 with an increasing rule 204 is a very slow second order transition with critical $\alpha = 0.61$ (reading Figure 16a from the right). Meaning rule 50 is very destructive to the density coming from rule 204 while rule 204 is slightly destructive to the density of rule 50. This is in some sense thus the opposite of the ACA with rule 6, as can be seen by the figures being very rough mirror images of eachother.

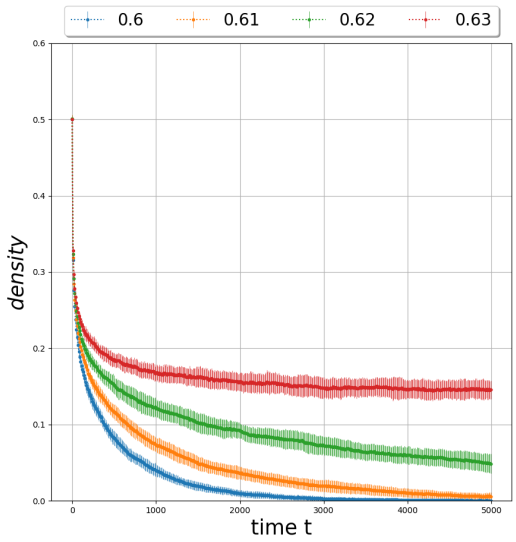
Some final remarks on the time limit:
In this analysis we assumed the system to be in equilibrium. While this is largely true, around the phase transition the density is not quite settled yet. This is seen in the Figure 17, however we expect it to be very close to equilibrium and therefore a very good approximation of it.
As around the critical points the system has not quite settled yet, we do not rule out the possibility that the abrupt phase transition (with the critical α being 0 or 1) that seem to be second order are actually first order, as the system has not yet really settled so close to the critical point. This becomes clear

[1] However, this might still be first order, see the final remarks on the time limit.
[2] See footnote 1

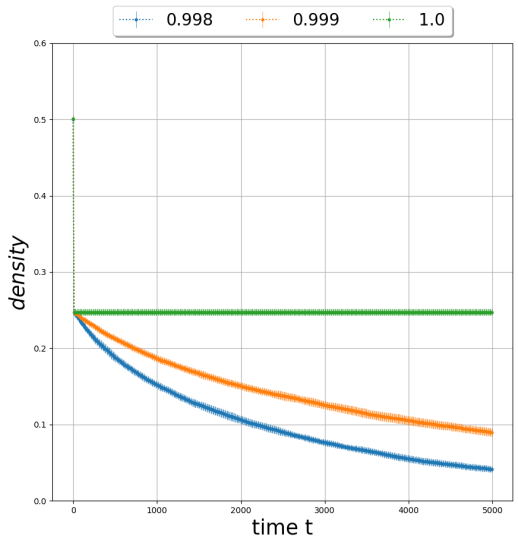
when considering figures 17c and 17d where one clearly sees that the near 0 and 1 the points are moving towards 0.



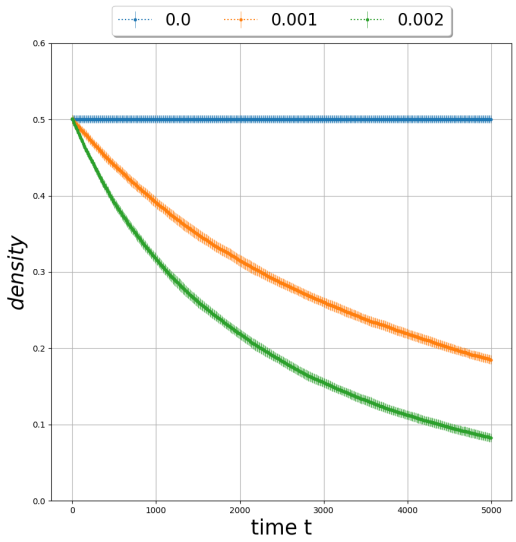
(a) Rule 6, near critical $\alpha = 0.34$



(b) Rule 50, near critical $\alpha = 0.61$



(c) Rule 6, near critical $\alpha = 1$



(d) Rule 50, near critical $\alpha = 0$

Figure 17: Rules with α's taken around the phase transition, with the α's indicated in the legend

To conclude this section. We have studied two ACAs, one with rule 6 and one with rule 50. These systems have some interesting behaviour, as separately they have non zero equilibrium densities, but the interplay between the two leads to the equilibrium density becoming 0 in a large part of the α range. The systems both exhibit two critical α's where the equilibrium density becomes non zero. For the ACA with rule 6 there is a slow second order phase transition when rule 204 becomes less important and an abrupt second order phase transition when rule 6 takes fully over. For the ACA with rule 50 there is a abrupt second order phase transition when going from rule 50 to rule 204 and a slow second order transition when going from rule 204 to rule 50.

5 Summary

In this report we studied a diploid cellular automata. A diploid cellular automata is a stochastic ECA dependent on two rules. In section 2 we have performed a theoretical study of the diploid cellular automata depending on the identity rule (rule 204) and the null rule (rule 0). The simplicity of those systems enables us to study them analytically and provides us with some insights. As the result, we calculate the probability for a single cell stays at zero status at time-step t and compute μ under the condition of $n \to \infty$ and a time-scale logarithmically diverging with n. We noted that for the analytical system a phase transition was observed which is also visible for other even rules and can be checked through numerical studies. After that, in section 3, we perform a numerical study on diploid ECA with the null rule and some more complicated rules (rule 18 and 94). After carefully studying the behaviour of these systems as a function of the time-steps, volume and λ, we find that in general the values used ($T = 5000, N = 10000$) describe the equilibrium values of the system well. However, around the critical value for phase transition, the system seems to be still slightly out of equilibrium. Using $T = 20000$ instead would solve this problem. We also find that both rule 204 and rule 94 contain a second order phase transition at some critical λ. After this we give a short heuristic argument about our expectations for a diploid ECA with the null rule and an odd rule. In the last section, we consider an asynchronous cellular automata (ACA), which is a diploid ECA using the identity rule and another rule. First we deliver space-time diagrams for the ACA with rules 6,50 and 178 to gain some intuition about these systems followed by a similar analysis to section 3. Again we conclude that our values accurately describe a system in equilibrium, except for the values around the phase transition. We find that the rules we consider have two second order phase transitions for some α, where one is very abrupt and the another relatively moderate one. In general the equilibrium density in these systems is smaller than the equilibrium density of the two separate rules, meaning the the introduction of some disorder due to randomly using the identity rule in general leads to a lower equilibrium density.

References

[1] Jarkko Kari. Cellular automata. 2011.

[2] Chrystopher L Nehaniv. Asynchronous automata networks can emulate any synchronous automata network. *International Journal of Algebra and Computation*, 14(05n06):719–739, 2004.

[3] John Von Neumann and Arthur W. Burks. *Theory of Self-Reproducing Automata*. University of Illinois Press, USA, 1966.

[4] Stephen Wolfram. *A New Kind of Science*. Wolfram Media, 2002.

A Code

In this appendix section a simple version of the code is given, which runs through the different λ values for a single run. Here we see the `ruleFunction` which was used to compute the next grid values using a neighbour array as discussed in the code implementation. The code shown bellow does not include imported packages or the creation of the array of `Lambdas` for which the densities are determined.

```
T=5000 #number of iteration
n=10000 #number of cells
np.random.seed(123)

rule1 = 0  # f1 rule number
rule2 = 94 # f2 rule number
rule1_bin = [int(x) for x in np.binary_repr(rule1, width=8)]
# binary representation of f2 rule
rule2_bin = [int(x) for x in np.binary_repr(rule2, width=8)]
# binary representation of f2 rule
rule_set = np.array([[1,1,1],[1,1,0],[1,0,1],[1,0,0],
                     [0,1,1],[0,1,0],[0,0,1],[0,0,0]])

def ruleFunction(x,rule_bin,rule_set):
    Outx = np.zeros(np.shape(x))
    if np.sum(rule_bin)!=0:
        # input a np array of size(n,1)
        neighArray = np.zeros((len(x),3))
        neighArray[:,0] =  np.roll(x,-1) #neighbour to the left
        neighArray[:,1] = x
        neighArray[:,2] = np.roll(x,1) #neighbour to the right
        bins = np.where(rule_bin)
        for i in range(np.sum(rule_bin)):
          Outx = Outx +
          (neighArray[:] == rule_set[bins[0][i]]).all(axis=1)
        return Outx

#Make the space-time grid
for j,Lambda in enumerate(lambdas):
    grid = np.zeros((T,n))
    # T x n grid (matrix) of zeroions (white cells)
    grid[0, :]=(np.random.rand(n)<0.5)
    # random first row with 50% chance of either 0 or 1

    ArrayRand = (np.random.rand(T-1,n)<Lambda)
    # create array of binaries (0,1) according to probability Lambda
    for t in range(1,T):
      grid[t] = np.abs(ArrayRand[t-1]-1)*ruleFunction(grid[t-1],rule1_bin,rule_set)
      + ArrayRand[t-1]*ruleFunction(grid[t-1],rule2_bin,rule_set)
    densities[j]=np.sum(grid[T-1,:])/n
    #determine the denisity at the final timestep
```