

Abrupt Behaviour Changes in Cellular Automata under Asynchronous Dynamics

Damien Regnault

Complex Systems Institute
LIP (UMR CNRS ENSL UCBL INRIA 5668)
46 allée d'Italie, 69364 Lyon Cedex 07, France
damien.regnault@ens-lyon.fr

Abstract. We propose an analysis of the relaxation time of the elementary finite cellular automaton **214** (Wolfram coding) under α -asynchronous dynamics (i.e. each cell independently updates with probability $0 < \alpha \leq 1$ at each time step). While cellular automata have been intensively studied under synchronous dynamics (all cells update at each time step), much less work is available about asynchronous dynamics. In particular, the robustness to asynchronism is a feature which is far from being cleared up. [1,2] have studied double quiescent automata (DQECA) under fully and α -asynchronous dynamics. [1] did not analyse the behavior of all DQECA and left some conjectures concerning four automata, among which automaton **214** which seems to have a specific behavior under α -synchronous dynamics. Our work partially answers one of those conjectures, and both illustrates the richness of the behaviours involved by asynchronism on cellular automata and the challenge of their mathematical prediction. Far from being a marginal case study, our analysis provides a very relevant example of the way the dynamics is affected by asynchronism and of the mathematical tools which can be used to predict the asymptotic behaviour of such complex systems.

1 Introduction

The aim of this article is to analyze the asynchronous behavior of the unbounded finite cellular automaton **214**. Cellular automata are widely used to model systems involving a huge number of interacting elements such as agents in economy, particles in physics, proteins in biology, distributed systems, etc. In most of these applications, in particular in many real system models, agents are not synchronous. Depending on the transition rules, the behaviour of the system may vary widely when asynchronism increases in the dynamics. More generally one can ask how much does asynchronous in real system perturbs computation. In spite of this lack of synchronism, real living systems are very resilient over time. One might then expect the cellular automata used to model these systems to be robust to asynchronism and to other kind of failure as well (such as misreading the states of the neighbors). It turns out that the resilience to asynchronism widely varies from one automata to another (e.g., [3,4]). Only few theoretical

studies exist on the influence of asynchronism. Most of them usually focus on one specific cellular automata (e.g., [5,6,7]) and do not address the problem globally. In 2003, Gács shows in [8] that it is undecidable to determine if in a given automata, the sequences of changes of states followed by a given cell is independent of the history of the updates. Related work on the existence of stationary distribution on infinite configurations for probabilistic automata can be found in [9].

We continue here a study begun in [2] and [1] on the effects of asynchrony on the global evolution of the system given an *arbitrary* set of local rules. In particular we study how asynchronicity affects the relaxation time of such systems. In [2], the authors carried out a complete analysis of the class of one-dimensional double quiescent elementary cellular automata (DQECA), where each cell has two states, 0 and 1, which are quiescent (i.e., where each cell for which every neighboring cell is in the same state as itself remains in that state) and where each cell updates according to its state and the states of its two immediate neighbours. They study the behaviour of these automata under fully asynchronous dynamics, where only one random cell is updated at each time step. They show that one can classify the 64 DQECAs in six categories according to their relaxation times under full asynchronism (either constant, logarithmic, linear, quadratic, exponential or infinite) and furthermore that the relaxation time characterizes their behaviour, i.e., that all automata with relaxation times of the same order present the same kind of space-time diagrams. In [1], this study is extended to a continuous range of asynchronism from fully asynchronous dynamics to fully synchronous dynamics: the α -asynchronous dynamics, with $0 < \alpha \leq 1$. In this setting, each cell is updated independently with probability α at each time step. When α varies from 1 down to 0, the α -asynchronous dynamics evolves from the fully synchronous regime to a more and more asynchronous regime. As α approaches 0, the probability that the updates involve at most one cell tends to 1, and the dynamics gets closer and closer to a kind of fully asynchronous dynamics up to a time rescaling by a factor $1/\alpha$.

The comparison between the fully asynchronous dynamics and the synchronous dynamics in [2] shows that most of the studied automata have drastically different behaviors. The comparison between the fully asynchronous dynamics and the α -asynchronous dynamics in [1] shows that new phenomena could appear under α -asynchronous dynamics. Nevertheless after a time rescaling, most of the studied automata seem to have the same global behavior under both dynamics. The only automata where these phenomena change drastically its behavior is automaton **194**. Its relaxation time is $O(n^3)$ under fully asynchronous dynamics, $O(\frac{n}{\alpha^2(1-\alpha)})$ under α -asynchronous dynamics and it diverges under synchronous dynamics. Thus there is a speed up from fully asynchronous to the α -asynchronous dynamics because of a so called *spawning* phenomenon (see [1]). The authors conjecture that four other automata have a specific behavior under α -asynchronous dynamics. Cellular automaton **214** studied here is one of them. It diverges (i.e., it never reaches a fixed point) under both fully asynchronous dynamics and synchronous dynamics. Nevertheless, we prove here

that the cellular automaton **214** converges to a fixed point in linear time under α -asynchronous dynamics when $\alpha > 0.9999$; we also exhibit the phenomenon accountable for this fast convergence. Now, this is the most explicit case to show the difference between α -asynchronous dynamics and the two other dynamics.

Section 2 introduces the main definitions and presents our main result. Section 3 presents the probabilistic tools developed in [1] used in our analysis. The omitted proofs can be found in the extended version of this paper available on the website of the author.

2 Definitions, Notations and Main Results

In this paper, we consider the elementary cellular automaton **214** on finite size configurations with periodic boundary conditions. We recall briefly the notations and definitions introduced in [1].

Definition 1. An Elementary Cellular Automata (ECA) is given by its transition function $\delta : \{0, 1\}^3 \rightarrow \{0, 1\}$. We denote by $Q = \{0, 1\}$ the set of states.

We denote by $U = \mathbb{Z}/n\mathbb{Z}$ the set of cells. A finite configuration with periodic boundary conditions $x \in Q^U$ is a word indexed by U with letters in Q .

Definition 2. Here is the transition function of cellular automaton **214**:

$x \ y \ z$	000	001	100	101	010	011	110	111
$\delta_{214}(x, y, z)$	0	1	1	0	1	0	1	1

We consider three kinds of dynamics for ECAs: the *synchronous dynamics*, the α -*asynchronous dynamics* and the *fully asynchronous dynamics*. The synchronous dynamics is the classic dynamics of cellular automata, where the transition function is applied at each (discrete) time step on each cell simultaneously.

Definition 3 (Synchronous Dynamics). The synchronous dynamics $S_\delta : Q^U \rightarrow Q^U$ of an ECA δ , associates deterministically to each configuration x the configuration y , such that for all $i \in U$, $y_i = \delta(x_{i-1}, x_i, x_{i+1})$.

Definition 4 (Asynchronous Dynamics). An asynchronous dynamics AS_δ of an ECA δ associates to each configuration x a random configuration y , such that $y_i = x_i$ for $i \notin S$, and $y_i = \delta(x_{i-1}, x_i, x_{i+1})$ for $i \in S$, where S is a random subset of U chosen by a daemon. We consider two types of asynchronous dynamics:

- in the α -asynchronous dynamics, the daemon selects at each time step each cell i in S independently with probability α where $0 < \alpha \leq 1$. The random function which associates the random configuration y to x according to this dynamics is denoted AS_δ^α .
- in the fully asynchronous dynamics, the daemon chooses a cell i uniformly at random and sets $S = \{i\}$. The random function which associates the random configuration y to x according to this dynamics is denoted AS_δ^F .

For a given ECA δ , we denote by x^t the random variable for the random configuration obtained after t applications of the asynchronous dynamics function AS_δ on configuration x , i.e., $x^t = (AS_\delta)^t(x)$.

Definition 5 (Fixed point). We say that a configuration x is a fixed point for δ under asynchronous dynamics if $AS_\delta(x) = x$ whatever the choice of S is (the cells to be updated). \mathfrak{F}_δ denotes the set of fixed points for δ .

Fact 1 if n is even then $\mathfrak{F}_{214} = \{0^n, 1^n, (01)^{n/2}\}$, otherwise $\mathfrak{F}_{214} = \{0^n, 1^n\}$. The configuration 0^n cannot be reached from any other configurations whatever the dynamics is.

Definition 6 (Relaxation Time). Given an ECA δ and a configuration x , we denote by $T_\delta(x)$ the random variable for the time elapsed until a fixed point is reached from configuration x under an asynchronous dynamics, i.e., $T_\delta(x) = \min\{t : x^t \in \mathfrak{F}_\delta\}$. The relaxation time of ECA δ is $\max_{x \in Q^U} \mathbb{E}[T_\delta(x)]$.

We can now state our main theorem.

Theorem 2 (Main result). Under α -asynchronous dynamics when $0.9999 < \alpha < 1$, the relaxation time T_{214} of cellular automaton **214** is $O(\frac{n}{1-\alpha})$.

3 Lyapunov functions based on local neighbourhoods

The reader may find more detailed definitions in [1].

Definition 7 (Mask). A mask \dot{m} is a word on $\{0, 1, \dot{0}, \dot{1}\}$ containing exactly one dotted letter in $\{\dot{0}, \dot{1}\}$. We say that the cell i in configuration x matches the mask $\dot{m} = m_{-k} \dots m_{-1} \dot{m}_0 m_1 \dots m_l$ if $x_{i-k} \dots x_i \dots x_{i+l} = m_{-k} \dots m_0 \dots m_l$.

Definition 8 (Masks basis). A masks basis \mathcal{B} is a finite set of masks such that for any configuration x and any cell i , there exists a unique $\dot{m} \in \mathcal{B}$ that matches cell i .

Masks bases will be used to define Lyapunov weight functions from local patterns. It provides an efficient tool to validate exhaustive case analysis.

Definition 9 (Local weight function). A local weight function f is a function from a masks basis \mathcal{B} to \mathbb{Z} . The local weight of the cell i in configuration x given by f is $F(x, i) = f(\dot{m})$ where \dot{m} is the unique mask in \mathcal{B} matching cell i . The weight of a configuration x given by f is defined as $F(x) = \sum_i F(x, i)$.

Notation 1 For a given random sequence of configurations $(x^t)_{t \in \mathbb{N}}$ and a weight function F on the configurations, we denote by $(\Delta F(x^t))_{t \in \mathbb{N}}$ the random sequence $\Delta F(x^t) = F(x^{t+1}) - F(x^t)$.

The next lemma provides upper bounds on stopping times for the markovian sequence of configurations $(x^t)_{t \in \mathbb{N}}$ subject to a weight function F whose average is a non increasing function of time (a *Lyapunov function*). Its proof can be found in [2].

Lemma 1. Let $m \in \mathbb{Z}_+$ and $\epsilon > 0$. Consider (x^t) a random sequence of configurations, and F a weight function such that $(\forall x) F(x) \in \{0, \dots, m\}$. Assume that if $F(x^t) > 0$, then $\mathbb{E}[\Delta F(x^t) | x^t] \leq -\epsilon$. Let $T = \min\{t : F(x^t) = 0\}$ denote the random variable for the first time t where $F(x^t) = 0$. Then, $\mathbb{E}[T] \leq \frac{m + F(x^0)}{\epsilon}$.

4 Informal description of 214's behaviour

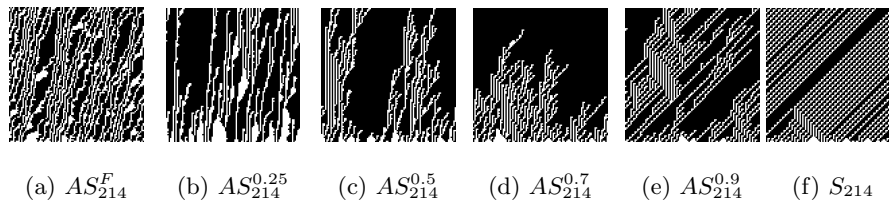


Fig. 1. BCF under different dynamics

Under fully asynchronous dynamics (see figure 1a) a configuration cannot reach a fixed point from a non fixed point configuration. In fact, the number of regions (which is also $|x^t|_{10}$ or $|x^t|_{01}$) cannot increase or decrease. So the number of regions is constant.

Under α -asynchronous, as shown in figure 1(d) the automaton may converge to the fixed point 0^n . According to simulations, the relaxation time appears to be linear of the size of n and is conjectured to be $O(\frac{n}{\alpha^2(1-\alpha)})$. The number of regions can increase or decrease because of two new phenomena that have already been observed in [1]: the *spawning* phenomenon and the *annihilation* phenomenon (see fig. 2). Indeed a pattern 1001 may evolve to 1111 (the number of regions decrease) and a pattern 0011 may evolve to 0101 (the number of regions increase). Thus the only way to decrease the number of regions is the annihilation phenomenon. So the key pattern is 1001.

The difficulty of the proof is that a 0-region in the pattern 10011 has the same probability, $\alpha^2(1-\alpha)$, to spawn a new 0-region or to be annihilated and it could also evolve with probability $\alpha(1-\alpha)^2$ to 10001 (a pattern where the annihilation phenomenon is no more possible). We have to deal with two problems with the pattern 10011: the evolution towards 10001 and the fact that the probability to increase or decrease by one the number of regions is the same. For the first one, we do not have an answer yet. Now, we consider $\alpha > 0.9999$ so that this phenomenon is negligible. The bound is not tight and could be improved by tuning further the constants. Our aim is to propose an answer to the second problem. Considering figure 2, one can notice that the 0-regions are close to each other. In a pattern 10010 an annihilation phenomenon could occur but not a spawning phenomenon. This is the key phenomenon:

Definition 10. *We say that there is a collision when the first 0-region in a pattern 10010 disappears because of an annihilation phenomenon.*

More importantly, if a pattern 10011 evolves to 10101, the two 0-regions are very close. So the probability that they collide does not seem to be negligible. If we can prove this then we can find local weights such that the variation of the local weights for a pattern 10011 is negative.



Fig. 2. explanation of the convergence

So we are interested in the evolution of a pattern 10101 and we would like to show that the probability that two 0-regions collide is not negligible. We have chosen α so we can consider that the pattern 10101 evolves almost under synchronous dynamics, and that sometimes one cell doesn't update. Figure 2 shows the evolution of a pattern 10101 when there is no collision with other 0-regions. The black arrows show the most likely evolution (all cells update) and the dotted arrows show the evolution when one cell doesn't update. From these observations, we manage to deduce a weight function which expected variation is negative at time step. This function and the proof can be found in the extended version of this article.

Acknowledgements: Thanks for Nicolas Schabanel and Éric Thierry for their useful remarks that helped to improve the writing of the paper.

References

1. Fatès, N., Regnault, D., Schabanel, N., Thierry, E.: Asynchronous behaviour of double-quiescent elementary cellular automata. In: LNCS 3887 Proc. of the 7th LATIN. (2006)
2. Fatès, N., Morvan, M., Schabanel, N., Thierry, E.: Asynchronous behaviour of double-quiescent elementary cellular automata. In: LNCS 3618 Proc. of the 30th MFCS. (2005)
3. Bersini, H., Detours, V.: Asynchrony induces stability in cellular automata based models. In: Proc. of the 4th Artificial Life. (1994) 382–387
4. Fatès, N., Morvan, M.: An experimental study of robustness to asynchronism for elementary cellular automata. Submitted, [arxiv:nlin.CG/0402016](https://arxiv.org/abs/nlin.CG/0402016) (2004)
5. Fukás, H.: Probabilistic cellular automata with conserved quantities. [arxiv:nlin.CG/0305051](https://arxiv.org/abs/nlin.CG/0305051) (2005)
6. Fukás, H.: Non-deterministic density classification with diffusive probabilistic cellular automata. *Phys. Rev. E* **66** (2002)
7. Schönfisch, B., de Roos, A.: Synchronous and asynchronous updating in cellular automata. *BioSystems* **51** (1999) 123–143
8. Gács, P.: Deterministic computations whose history is independent of the order of asynchronous updating. <http://arXiv.org/abs/cs/0101026> (2003)
9. Louis, P.Y.: Automates Cellulaires Probabilistes : mesures stationnaires, mesures de Gibbs associées et ergodicité. PhD thesis, Université de Lille I (2002)
10. Fatès, N.: Fiatlux CA simulator in Java. Sources and experimental data available from <http://perso.ens-lyon.fr/nazim.fates> (2004)