Proving a phase transition in cellular automata under asynchronous dynamics

Damien Regnault

IBISC, Université d'Évry Val-d'Essonne, 23 boulevard de France, 91037 Evry CEDEX. `http://www.ibisc.fr/~dregnault/`

Abstract. Cellular automata are a model of parallel computing. It is well known that simple cellular automata may exhibit complex behaviors such as Turing universality [2,17]. The underlying mechanisms of these rules are now rather well understood (see Kari's survey [13] or Sarkar's survey [20]).

Less results are known about probabilistic cellular automata. The most famous ones come from Toom [21] and Gács [10,11]. They have shown that cellular automata are still able to perform reliable computation in presence of random faults even in one dimension. Recently, Fatès [6] has exhibited a family of simple one dimensional probabilistic rules which can solve the density classification problem with arbitrary precision.

Several studies have focused on a specific probabilistic dynamics: α-asynchronism where at each time step each cell has a probability α to be updated. Experimental studies [4,5] followed by mathematical analysis [1,3,8,9,18] have permitted to exhibit simple rules with interesting behaviors. Among these behaviors, most of these studies conjectured that some cellular automata exhibit a polynomial/exponential phase transition on their convergence time, *i.e.* the time to reach a stable configuration. The study of these phase transitions is crucial to understand the behaviors which appear at low synchronicity. A first analysis [19] proved the existence of the exponential phase in cellular automaton FLIP-IF-NOT-ALL-EQUAL but failed to prove the existence of the polynomial phase. In this paper, we prove the existence of a polynomial/exponential phase transition in a cellular automaton called FLIP-IF-NOT-ALL-0.

1 Introduction

Cellular automata are made of several cells which are characterized by a state. Time is discrete and at each iteration, the state of a cell evolves according to the states of its neighbors. On one hand, cellular automata are used as a model of parallel computing. Methods of programing them are rather well understood [13,20]. On the other hand, they are also a common tool to model real life phenomena [16].

Theoretical studies of cellular automata as a computation model mainly focus on the synchronous dynamics, *i. e.* at each time step all cells are updated. Only few studies focus on other dynamics. For modeling real life phenomena, the assumption of synchronicity may be too restrictive.

In this article, we will consider stochastic dynamics. At each time steps, only the cells of a randomly chosen set are updated. The introduction of randomness has different motivations. It can be used:

- to simulate faults in a system, *i.e.* as a difficulty to overcome;
- as an oracle, *i.e.* as a tool to develop more efficient algorithms;
- in the definition of a model based on cellular automata.

Faults tolerant cellular automata are cellular automata where at each time step, each cell has a constant probability to be faulty updated and to switch to any states. Toom [21] and Gács [10,11] have developed rules which are able to make reliable computation in presence of random faults. Their results interlace two rules: a rule which is turing universal for the parallel dynamics and a rule which detects and erases faults.

In algorithmic, it is well known that randomness can be useful to develop efficient algorithms [12]. Recently, Fatès [6] has shown that the density classification problem can be achieved with arbitrary precision by two states one dimensional cellular automata under probabilistic dynamics. In this problem, the cells must all choose the overall majority state of the initial configuration. Land and Belew [14] have shown that this problem cannot be solved under deterministic dynamics.

Some theoretical studies [1,3,8,9,18] have focusses on α-asynchronous cellular automata where at each time step, each cell has a probability α to be updated. The first theoretical analyzes [1,8,9] mainly relies on simple stochastic processes such as random walk and coupon collector. Later studies [3] have focus on the minority rule. The authors manage to analyze the very first and last steps of a classical evolution of Minority from a random configuration but they are not able to analyze the whole dynamics. They argue that even if Minority on a random configuration seems a "simple" process, some specific initial configuration leads to different dynamics. They conjectured that Minority can simulate some classical stochastic process like percolation or TASEP on specific initial configurations. From these works, it seems that one simple cellular automaton may simulate several stochastic processes which were independently studied by different communities. If these conjectures are true, then these works may shed a new light on stochastic process simulation.

In this paper, we will prove the existence of a phase transition in $1D$ cellular automaton FLIP-IF-NOT-ALL-0 depending on α. This is the first proof of a phase transition in a probabilistic cellular automaton even if the existence of phase transition in α-asynchronous dynamics was conjecture and analyzed empirically [5]. Moreover, the rule $1D$ Minority is equivalent the $1D$ rule FLIP-IF-NOT-EQUAL. A first study [19] proved the existence of one part of the phase transition on this cellular automaton but failed to prove the other part. Rules FLIP-IF-NOT-ALL-0 and FLIP-IF-NOT-EQUAL only differ on one neighborhood. Only one last argument is necessary to prove the existence of a phase transition for Minority rule.

In Gács' $1D$ cellular fault tolerant cellular automaton [10], the main difficulty was to develop a process which is able to detect faulty regions, *i.e.* to save one bit of information in presence of random faults. Since cells have only a local vision of the information, the positive rates conjecture states that it is not possible to keep one bit of information safe from the random faults and thus that reliable computation were impossible. Gács provided a counter-example but the community was unsatisfied by the size and the complexity of its rule and of the following proof. The quest for a simple fault-tolerant cellular automaton with a proof of reasonable size is still open. Another interest of FLIP-IF-NOT-ALL-0 is that one phase of the transition phase leads to the emergence of a white region and the other phase of the transition phase leads to the emergence of a checkerboard pattern. Thus this paper is another step to a better understanding of the emergence of homogeneous regions in $1D$ stochastic processes.

2 Asynchronous cellular automata

2.1 Definition

We give here a formal definition of FLIP-IF-NOT-ALL-0, also called ECA 50 (Wolfram encoding). The next part presents informally its behavior.

Definition 1 (Configuration). *Consider $n \in \mathbb{N}$, we denote by $\mathbb{Z}/n\mathbb{Z}$ the set of* cells *and $Q = \{0, 1\}$ the set of* states *(0 stands for white and 1 for black in the figures), n is the size of the configuration. The* neighborhood *of a cell i consists of the cells $i-1$, i and $i+1 \bmod n$. A* configuration *c is a function $c : \mathbb{Z}/n\mathbb{Z} \to Q$; c_i is the state of the cell i in configuration c.*

We consider configurations of size $n \in \mathbb{N}$ with periodic boundary conditions thus all computation on the position of a cell are made modulo n.

Definition 2 (FLIP-IF-NOT-ALL-0). *The* rule *of a cellular automaton is a function which associates a state to a neighborhood. The rule δ of* FLIP-IF-NOT-ALL-0 *is defined as follows:*

$$\delta(c_{i-1}, c_i, c_{i+1}) = \begin{cases} c_i & \text{if } c_{i-1} = c_i = c_{i+1} = 0 \\ 1 - c_i & \text{otherwise} \end{cases}$$

Time is discrete and in the classic deterministic synchronous dynamics all the cells of a configuration are updated at each time step according to the transition rule of the cellular automaton (see figure 1). Here we consider a stochastic asynchronous dynamics where only a random subset of cells is updated at each time step.

Definition 3 (Asynchronous dynamics). *Given $0 < \alpha \leqslant 1$, we call α-*asynchronous dynamics *the following process : time is discrete and c^t denotes the random variable for the configuration at time t. The configuration c^0 is the* initial configuration. *The configuration at time $t+1$ is the random variable defined by the following process : each cell has independently a probability α to be updated according to the rule δ (we say that the cell* fires at time *t) and a probability $1 - \alpha$ to remain in its current state. A cell is said* active *if its state changes when fired.*

Figure 1 presents different space-time diagrams of ECA 50 for different values of α. By abuse of notation $\delta(c)$ is the probability distribution obtained after updating c one time with rule δ under asynchronous dynamics.

Definition 4 (Stable configuration). *A configuration c is a stable if for all $i \in \mathbb{Z}/n\mathbb{Z}$, $\delta(c_{i-1}, c_i, c_{i+1}) = c_i$.*

FLIP-IF-NOT-ALL-0 admits only one stable configuration: $\bar{0} = 0^n$ where all cells are in state 0. Since any black cell is active, by firing all the black cells and no white cell, we have the following fact:

Fact 1 (One step convergence) *If $\alpha < 1$, any configuration, evolving under* FLIP-IF-NOT-ALL-0 *and α-asynchronous dynamics, can reach the stable configuration $\bar{0}$ in one step.*

Definition 5 (Worst case convergence). *We say that a random sequence of configurations $(c^t)_{t \geqslant 0}$ evolving under* FLIP-IF-NOT-ALL-0 *and α-asynchronous dynamics* converges *from an initial configuration c^0 if the random variable $T = \min\{t : c^t \text{ is stable }\}$ is finite with probability 1. We say that the convergence occurs in polynomial (resp. exponential) time on expectation if and only if $\mathbb{E}[T] \leqslant p(n, 1/\alpha)$ (resp. $\mathbb{E}[T] \geqslant b^n$) for some polynomial p (resp. constant $b > 1$) and for any initial configuration (for at least one initial configuration).*

From the definition of stable configuration, it follows that if there is t such that c^t is a stable configuration then for all $t' \geqslant t$ the configuration $c^{t'}$ is the same stable configuration. Since $(c^t)_{t \geqslant 0}$ is a finite Markow chain and since there is a path from any configuration to the stable configuration all white (see fact 1), any sequence of configuration converges with probability 1 when $\alpha < 1$.

Theorem 1 (Main result). *Consider a sequence of configurations $(c^t)_{t \geqslant 0}$ evolving under rule* FLIP-IF-NOT-ALL-0 *and α-asynchronous dynamics then if $\alpha \leqslant 0.5$ then $\mathbb{E}[T] = O(n^2\alpha^{-1})$ and if $\alpha \geqslant 1 - \epsilon$ (where $\epsilon = 0.187 \times 10^{-13} > 0$) then $\mathbb{E}[T] = \Omega(2^n)$.*

This is the first time that a phase transition is formally proved on a asynchronous cellular automata. This result shows that simple rules exhibit complex behavior and turn out to be hard to analyze. The following part exposes experimental results on the behavior of FLIP-IF-NOT-ALL-0. Section 3 is dedicated to the proof of the polynomial time convergence on expectation and section 4 is dedicated to the proof of the exponential time convergence on expectation.

2.2 Observations

In this section, we present empirical result on FLIP-IF-NOT-ALL-0 and the ideas behind theorem 1 and theorem 4. A detailed empirical study of this automata was published by Fatès [5]. We only present here the concept relevant to the rest of the article. Fatès determined a critical value $\alpha_c \approx 0.6282$. When

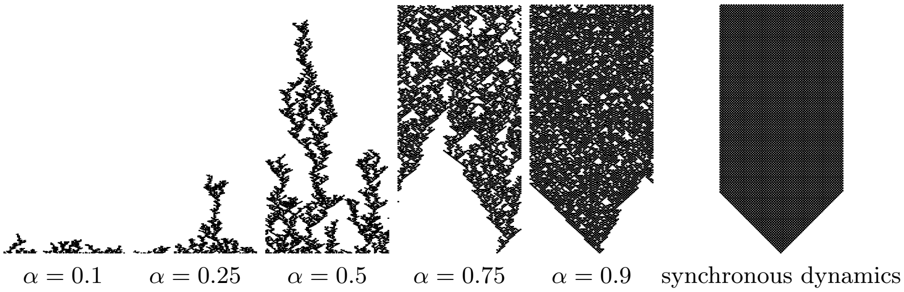


Fig. 1. FLIP-IF-NOT-ALL-0 under different dynamics. The height of the cells is scaled according to α

$\alpha < \alpha_c$ then the stochastic process converges in polynomial time and when $\alpha > \alpha_c$ the stochastic process converges in exponential time.

Polynomial time convergence. If $\alpha < \alpha_c$, only black regions crumble and white regions expand on the whole configuration. The last isolated black cells manage to survive only for a little while and the dynamics quickly reaches the stable configuration $\bar{0}$. Lot of cells are inactive during these periods. We will prove in theorem 2 that the dynamics converges in polynomial time on expectation. This proof uses a potential function, a technique already used to analyze asynchronous cellular automata [3,8,9,18]. The proof is here more technical. Until now, this technique was ineffective to analyze FLIP-IF-NOT-ALL-EQUAL when $\alpha \leqslant \alpha'_c$. The analysis of FLIP-IF-NOT-ALL-0 is easier and we able are to conclude. The difference between the two automata will be discussed in the open question of section 5.

The idea of the result presented here is the following: consider a bi-infinite configuration with one semi-infinite white region on the left and one semi-infinite black region on the right. The border of the configuration corresponds to the position of the first black cell. Consider the limit case $\alpha = 0.5$: with probability $\frac{1}{2}$ the border move 1 cell to left, with probability $\frac{1}{4}$ the border move to the right by at least 1 cell, with probability $\frac{1}{8}$ the border move to the right by at least 2 cells, $\ldots$, with probability $\frac{1}{2^i}$ the border moves to the right by at least $i-1$ cells. Thus, on expectation the movement of the border is 0 and it behaves as a non-biased random walk. For finite configurations, this means that the size of a white configuration behaves as a non-biased random walk in the worst case and thus reaches a size of n in quadratic time. The next section formalizes this idea and proves the convergence of the dynamics in polynomial time on expectation for $\alpha \leqslant 0.5$.

Exponential time convergence. If $\alpha > \alpha_c$ then white and black regions crumble and a checkerboard pattern invades the space-time diagram. Almost all the cells are active and flip their states at each time step. We will prove in theorem 4 that the dynamics converges in exponential time on expectation by using a coupling with oriented percolation. This method is similar to what have been done in [19] to analyze FLIP-IF-NOT-ALL-EQUAL.

3 Polynomial convergence time when $\alpha \leqslant 0.5$

In this section, we prove that the convergence time is polynomial on expectation when $\alpha \leqslant 0.5$. Thus from now on, we assume that $\alpha \leqslant 0.5$. We will define a function F which associates to each configuration c a potential $F(c) \in \{0, \dots, 2n+2\}$ such that:

- the expected variation of potential is negative at each time step and
- stable configuration $\bar{0}$ is the only configuration of zero potential.

We will conclude by using the following lemma which is folklore (a proof can be found in [7]). Consider $\epsilon > 0$, an integer $m > 0$ and a sequence $(X^t)_{t \geqslant 0}$ of random variables with values in set $\{0, \dots, m\}$ and a suitable filtration $(\mathcal{F}^t)_{t \in \mathbb{N}}$:

Lemma 1. *Suppose that :*

- *if* $0 < X^t < m$ *then* $\mathbb{E}[X^{t+1} - X^t | \mathcal{F}^t] \leqslant 0$ *and* $\Pr\{|X^{t+1} - X^t| \geqslant 1 | \mathcal{F}^t\} \geqslant \epsilon$*;*
- *if* $X^t = m$ *then* $\mathbb{E}[X^{t+1} - X^t | \mathcal{F}^t] \leqslant -\epsilon$*.*

Let $T = \min\{t \in \mathbb{N} : X^t = 0\}$ *and* $x_0 = \mathbb{E}[X^0]$*. Then:*

$$\mathbb{E}[T] \leqslant \frac{x_0(2m + 1 - x_0)}{2\epsilon}$$

Now, we formalize the observations made in section 2.2. Consider a configuration c, a *white region* of c is a maximal set of consecutive cells which are all in the state 0. The value $W(c)$ is defined as the size of the largest white region of c: $W(c) = \max\{|W| : W \text{ is a white region of c}\}$. We can now introduce the following potential function:

Definition 6 (Potential function). *We consider the function* $F : Q^{\mathbb{Z}/n\mathbb{Z}} \to \mathbb{N}$ *defined as follows:*

$$F(c) = \begin{cases} 0 & \text{if } c = \bar{0} \\ 2n + 2 - W(c) & \text{otherwise} \end{cases}$$

Note that for all configuration c, $F(c) \in \{0, \dots, 2n + 2\}$. Moreover, $\bar{0}$ is the only configuration of potential 0 and the configuration of potential $2n + 2$ has all its cells in state 1. We denote by $\mathbb{E}[\Delta(F(c))] = \mathbb{E}[F(\delta(c)) - F(c)]$, *i.e.* the variation of potential on expectation of c after one update of FLIP-IF-NOT-ALL-0 under α-asynchronous dynamics. We introduce $p^{\bar{0}}(c) = \Pr\{\delta(c) = \bar{0}\}$ the probability that the dynamics converges in one step and for each cell i, we introduce $p_i^0(c) = \Pr\{\delta(c)_i = 0\}$ the probability that the cell is in state 0 at the next time step. The variation of potential on expectation can be expressed as follows:

$$\mathbb{E}[\Delta(F(c))] = -\mathbb{E}[\Delta(W(c))] - (n + 2)p^{\bar{0}}(c).$$

Lemma 2. *Consider a configuration c and a cell $i \in \mathbb{Z}/n\mathbb{Z}$, if $\alpha \leqslant 0.5$ then:*

$$p_i^0(c) = \begin{cases} 1 & \text{if cell } i \text{ is inactive in } c \\ \geqslant \alpha & \text{otherwise} \end{cases}$$

Proof. An inactive cell is in state 0 and will stay in state 0 with probability 1. An active cell in state 1 will switch to state 0 with probability α. An active cell in state 0 will stay in state 0 with probability $1-\alpha$. Since $0 < \alpha \leqslant 0.5$, $\alpha \leqslant 1-\alpha$. The probability that an active cell will be in state 0 at the next time step is at least α.

Fact 2 *Consider a configuration c, if $\alpha \geqslant 0.5$, then $p^{\bar{0}}(c) \geqslant \alpha^{n-W(c)+2}$.*

Our aim is to apply lemma 1 on the potential function F. We start by analyzing some special configurations c where $W(c) \leqslant 1$.

Lemma 3. *Consider a configuration c such that $W(c) \leqslant 1$, then $\mathbb{E}[\Delta(F(c))] \leqslant -\alpha(1-\alpha)^2$.*

Proof. If $W(c) = 0$ then all cells are in state 1 and $F(c)$ is maximum. Firing any cell leads to the creation of a white region of site at least 1 and thus $\mathbb{E}[\Delta(F(c))] \leqslant -\alpha$.

If $W(c) = 1$ then consider a cell i of c such that $c_i = 0$. We have $c_{i-1} = c_{i+1} = 1$. If $W(\delta(c)) = 0$ then cells $i-1$ and $i+1$ are not fired while cell i fires, this event occurs with probability $\alpha^2(1-\alpha)$. If at least one of cells $i-1$ or $i+1$ fires while cell i is inactive then $W(\delta(c)) \geqslant 2$, this event occurs with probability $(1-(1-\alpha)^2)(1-\alpha)$. Thus,

$$\mathbb{E}[\Delta(F(c))] \leqslant \alpha^2(1-\alpha) - (1-(1-\alpha)^2)(1-\alpha)$$
$$\mathbb{E}[\Delta(F(c))] \leqslant -2\alpha(1-\alpha)^2.$$

Now, the difficulty lies in showing that $\mathbb{E}[\Delta(F(c))] \leqslant 0$ for all non stable configurations c of potential $F(c) \leqslant 2n$.

Lemma 4. *Consider a configuration c such that $2 \leqslant W(c) \leqslant n-1$ then $\mathbb{E}[\Delta(F(c))] \leqslant 0$. Moreover, $\Pr\{|\Delta(F(c))| \geqslant 1\} \geqslant \alpha(1-\alpha)^2$.*

Proof. Since a cellular automaton is shift invariant, we will now consider that $c_0 = 0$ and $\forall i \in \{n-W(c)+1, \ldots, n-1\}, c_i = 0$. Cells 0 and $n-W(c)+1$ are white active cells and cells of $\{n-W(c)+2, \ldots, n-1\}$ are white inactive cells. We consider the random variable M for the white region of $\delta(c)$ containing cell 0 if $\delta(c)_0 = 0$, cell $n-1$ if $\delta(c)_{n-1} = 0$ and maximum for the inclusion. If $\delta(c)_0 = 1$ and $\delta(c)_{n-1} = 1$ then $M = \emptyset$ (this case may only occurs when $W(c) = 2$). For $i \in \{0, \ldots, n-1\}$, cell i belongs to M if and only if:

- case A: $n-W(c)+2 < i \leqslant n-1$ or

- case B_1: $\forall j \in \{0, \dots, i\}$, $\delta(c)_j = 0$ or
- case B_2: $\forall j \in \{i, \dots, n - W(c) + 1\}$, $\delta(c)_j = 0$.

For $j \in \{1, 2\}$ we also call $B_j(i)$ the event: case B_i is true for cell j (and thus j belongs to M).

$$\begin{aligned}
\mathbb{E}[|M|] = \mathbb{E}\left[\sum_{0 \leqslant i \leqslant n-1} \mathbf{1}_{i \in M}\right] &= \sum_{0 \leqslant i \leqslant n-1} \mathbb{E}[\mathbf{1}_{i \in M}] = \sum_{0 \leqslant i \leqslant n-1} \Pr(i \in M). \\
&= W(c) - 2 + \sum_{0 \leqslant i \leqslant n-W(c)+1} \Pr(i \in M) \\
&= W(c) - 2 + \sum_{0 \leqslant i \leqslant n-W(c)+1} [\Pr(B_1(i)) + \Pr(B_2(i)) - \Pr(B_1(i) \cap B_2(i))].
\end{aligned}$$

Since $c_0 = 0$ and by lemma 2, we have $\Pr(B_1(i)) \geqslant (1 - \alpha)\alpha^i$. Then,

$$\begin{aligned}
\sum_{0 \leqslant i \leqslant n-W(c)+1} \Pr(B_1(i)) &\geqslant \sum_{0 \leqslant i \leqslant n-W(c)+1} (1 - \alpha)\alpha^i \\
&\geqslant 1 - \alpha^{n-W(c)+2}
\end{aligned}$$

Similarly, we have $\Pr(B_2) \geqslant 1 - \alpha^{n-W(c)+2}$. Note that the event $B_1(i) \cap B_2(i)$ implies that $\delta(c) = \bar{0}$. Then, $\Pr(B_1(i) \cap B_2(i)) = p^{\bar{0}}(c)$. We have:

$$\mathbb{E}[|M|] \geqslant W(c) - 2\alpha^{n-W(c)+2} - (n - W(c) + 2)p^{\bar{0}}(c)$$

Then,

$$\begin{aligned}
\mathbb{E}[\Delta(F(c))] &\leqslant W(c) - \mathbb{E}[|M|] - (n + 2)p^{\bar{0}}(c) \\
&\leqslant 2\alpha^{n-W(c)+2} + (n - W(c) + 2)p^{\bar{0}}(c) - (n + 2)p^{\bar{0}}(c) \\
&\leqslant 2\alpha^{n-W(c)+2} - W(c)p^{\bar{0}}(c) \\
&\leqslant 2\alpha^{n-W(c)+2} - 2p^{\bar{0}}(c) \\
&\leqslant 0.
\end{aligned}$$

Note that if cells 0 and $n - W(c) + 1$ do not fire and cell 1 fires then $|M| \geqslant W(c) + 1$. This event occurs with probability $\alpha(1 - \alpha)^2$ and thus $\Pr\{|\Delta(F(c))| \geqslant 1\} \geqslant \alpha(1 - \alpha)^2$.

Theorem 2. *If $\alpha \leqslant 0.5$ then the expected convergence time of* FLIP-IF-NOT-ALL-0 *on any initial configuration of size n is $O(n^2\alpha^{-1})$.*

Proof. Using lemma 1, lemma 3 and 4, we obtain that if $\alpha \leqslant 0.5$, the stochastic process reaches a configuration of potential 0 after $O(n^2\alpha^{-1})$ iterations on expectation. Note that the factor $(1-\alpha)^2$ is negligible since $\alpha \leqslant 0.5$. The only configuration of potential 0 is $\bar{0}$.

4 Proving an upper bound

In this section, we will demonstrate that FLIP-IF-NOT-ALL-0 convergences on exponential time on expectation when α is close enough to 1. To achieve this goal we will use a coupling with oriented percolation. This coupling will be done such that if the open cluster of the percolation is large enough then the dynamics of the automaton convergences in exponential time on expectation. Since we will use percolation on a non-standard topology, we start by adapting here the classical results from percolation theory for this topology. We pursue by the construction of the coupling.

4.1 Percolation

Consider a probability p, an integer n and the infinite randomly labeled oriented graph $\mathbb{L}(p,n) = (V,E)$ (see figure 2) where $V = \{(i,j) \in \mathbb{N} \times \{0,\dots,n\} : i+j \text{ is odd}\}$ is called the set of *sites* and E the set of *bonds*. For all sites $(i,j) \in V$, i is the *height* of the site and j is the *width* of the site. The height and width of a bond is the height and width of its origin. For all $(i,j) \in V$, there are oriented bonds between site (i,j) and sites $(i+1,j-1)$ (if $j \neq 0$), $(i+1,j+1)$ (if $j \neq n$), $(i-1,j-1)$ (if $j \geqslant 0$ and $i > 0$) and $(i-1,j+1)$ (if $j \neq n$ and $i > 0$). Bonds from a site of height i to a site of height $i+1$ have a probability p to be labeled *open* and a probability $1-p$ to be labeled *closed*. These probabilities are independent and identically distributed. Bonds from a site of height i to a site of height $i-1$ are labeled *closed*.

An open path of a randomly labeled graph is a path where all edges are open. We denote by C the open cluster of $\mathbb{L}(p,n)$: C contains all sites such that there exists an open path from a site of height 0 to this site.

In the next section we will use this percolation grid to prove the existence of the exponential convergence time phase in FLIP-IF-NOT-ALL-0. To achieve this goal, we will only need theorem 3. The following definitions are only introduced to prove this theorem and will not be used subsequently. Our proof relies on a classic method to obtain upper bounds for oriented percolation. Note that this proof is very similar to the one used in [15] but for the clarity of the paper we prefer to give here the whole adaptation of this proof.

The dual graph $\mathbb{L}^d(p,n) = (V^d, E^d)$ (see figure 2) is a randomly labeled graph where $V^d = \{(i,j) \in \mathbb{N} \times \{0,\dots,n\} : i+j \text{ is even}\}$ and for all $(i,j) \in V^d$, there are bonds between site (i,j) and sites $(i+1,j-1)$ (if $j \neq 0$) and $(i+1,j+1)$ (if

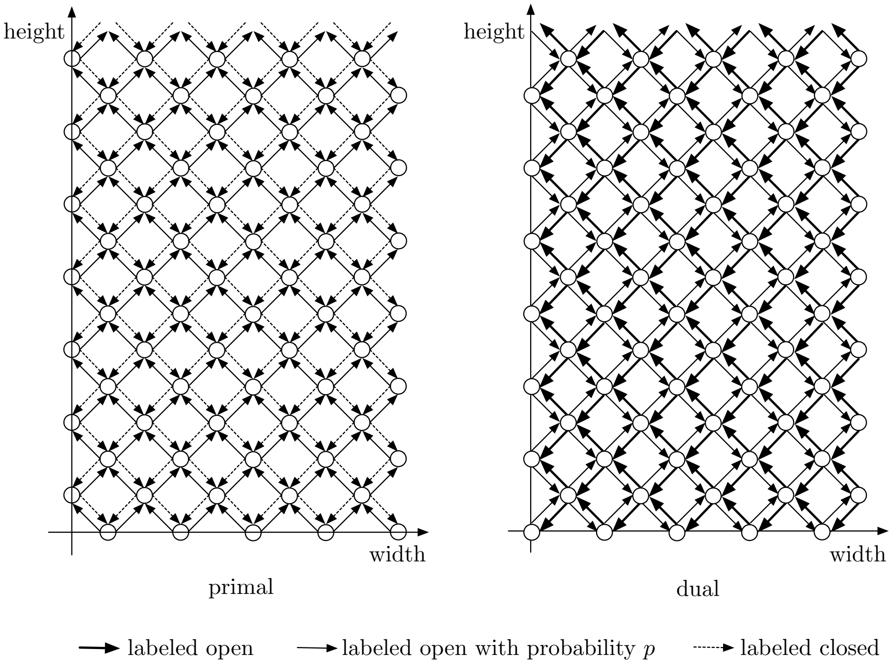


Fig. 2. A percolation graph with its dual graph.

$j \neq n$). For all $(i, j) \in V^d$, there are oriented bonds between site (i, j) and sites $(i+1, j-1)$ (if $j \neq 0$), $(i+1, j+1)$ (if $j \neq n$), $(i-1, j-1)$ (if $j \geqslant 0$ and $i > 0$) and $(i-1, j+1)$ (if $j \neq n$ and $i > 0$). Bonds from a site of width i to a site of width $i+1$ have a probability p to be labeled *open* and a probability $1-p$ to be labeled *closed*. These probabilities are independent and identically distributed. Bonds from a site of width i to a site of width $i-1$ are labeled *open*. We denote by C^d the open cluster of $\mathbb{L}^d(p, n)$: C^d contains all sites such that there exists an open path from a site of width 0 to this site.

By superposing $\mathbb{L}(p, n)$ and $\mathbb{L}^d(p, n)$, we can note that $\mathbb{L}^d$ is obtained by rotating by $\frac{\pi}{2}$ clockwise the bonds of $\mathbb{L}(p, n)$ in their center, *i.e.* we consider the bijection $E \rightarrow E^d$ where $\forall i, j \in V$:

- bond from site (i,j) to (i+1,j+1) is associated to bond from (i+1,j) to (i,j+1);
- bond from site (i+1,j) to (i,j+1) is associated to bond from (i+1,j+1) to (i,j);
- bond from site (i+1,j+1) to (i,j) is associated to bond from (i,j+1) to (i+1,j);
- bond from site (i,j+1) to (i+1,j) is associated to bond from (i,j) to (i+1,j+1);

It is possible to design a coupling between $\mathbb{L}(p, n)$ and $\mathbb{L}^d(p, n)$ such that a bond of $\mathbb{L}(p, n)$ is open if and only if the corresponding bond of $\mathbb{L}^d(p, n)$ is closed. Since all the edges from a site of width j to a site of width $j-1$ are open in the dual graph, we have the following fact:

Fact 3 *If site (i,j) belongs to C^d then for all $(k,l) \in V^d$ such that $i-j+l \leqslant k \leqslant i+j+l$, site (k,l) belongs to C^d.*

Lemma 5. *Let h be the minimal height of a site of width n of C^d. Then C contains at least one site of height $h-1$.*

Proof. For all $t \in \{0,\dots,h-1\}$, we denote by r_t the site of V^d with the smallest width among sites of height t which does not belong to C^d. By fact 3, the difference of width between site r_t an r_{t+1} is 1 or -1. We consider sites l_t of $\mathbb{L}(p,n)$ such that l_t is of height t and the width of l_t is equal to the width of r_t minus 1. This path is an open path of $\mathbb{L}(p,n)$ from a site of height 0 to a site of height $h-1$.

Lemma 6. *If $p \geqslant \frac{16^2-1}{16^2}$, there exists $\epsilon > 0$ such that the probability that C^d contains a site (i,n) with $i \leqslant 2^n+1$ is less than $1-\epsilon$.*

Proof. Consider a site i of width 0 and a site j of width n, we denote by $p(i,j)$ the probability that there is an open path from site i to site j in $\mathbb{L}^d(p,n)$. The length of a path from i to j is at least $n-1$. Consider $L \geqslant n-1$ then since the degree of a vertex of $\mathbb{L}^d$ is less than 4 there is less than 3^L path of length L from site i to site j. Moreover, for each of these path, at least half of its links are from a site of width i to a site of width $i+1$. Thus,

$$p(i,j) \leqslant \sum_{L=n-1}^{\infty} 3^L (1-p)^{\frac{L}{2}} \leqslant \frac{(3\sqrt{(1-p)})^n}{1-3\sqrt{(1-p)}}.$$

Consider $L^d = \{(i,0) \in V^d : i \leqslant 2^n - n + 2\}$ and $R^d = \{(i,n) \in V^d : i \leqslant 2^n+1)\}$. Since the cardinality of $L^d \times R^d$ is less than 4^n, the probability that there is an open path from a site of L^d to a site of R^d is less than $\frac{(12\sqrt{(1-p)})^n}{1-3\sqrt{(1-p)}}$. When $p \geqslant \frac{16^2-1}{16^2}$, there exists $\epsilon > 0$ such that this probability is less than $1-\epsilon$. From fact 3, C^d does not contain a site (i,n) such that $i \leqslant 2^n$.

Theorem 3. *If $p \geqslant \frac{16^2-1}{16^2}$ then there exists $\epsilon > 0$ such that C contains a site of height 2^n with probability ϵ.*

Proof. By lemma 6, there exists $\epsilon > 0$ such that the probability that C^d contains a site of width n and height less than 2^n+1 is at least ϵ when $p \geqslant \frac{16^2-1}{16^2}$. By lemma 5, this event means that C contains a site of height 2^n.

Note that the bound of $\frac{16^2-1}{16^2}$ could be improved by more refined tools from percolation theory. Nevertheless, these tools will only improve the final result of theorem 5 by a negligible amount.

4.2 Coupling

Consider a random sequence of configurations $(c^t)_{t\geqslant 0}$ evolving under FLIP-IF-NOT-ALL-0 and α-asynchronous dynamics. The size of the configuration is n. Consider a percolation grid $\mathbb{L}(p,n)$. Consider the following mapping (see figure 3):

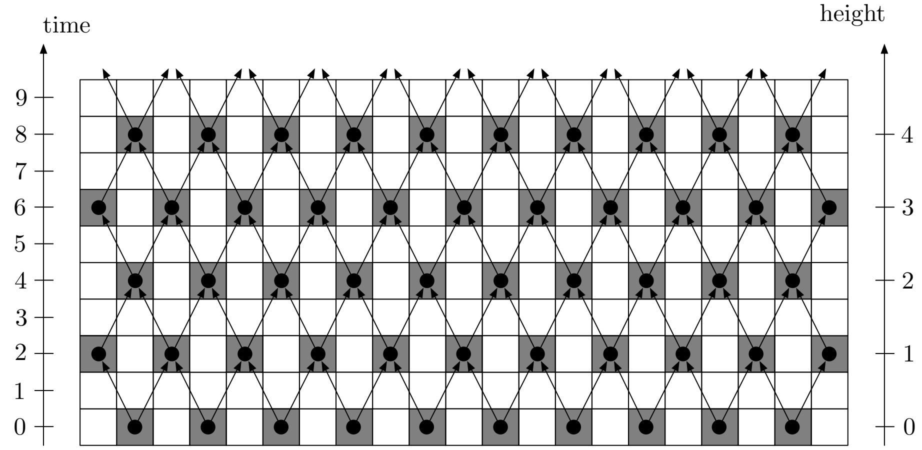


Fig. 3. the mapping from sites to cells.

Definition 7 (Mapping). *We define* $g : V \to \{0, n\}^{\mathbb{N}}$ *as the injection which associates the percolation site* (i, j) *to the cell* j *of configuration* c^{2i}.

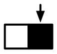
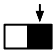
Our aim is to design a coupling such that cells of $g(C)$ are active. The coupling will be defined recursively according to time and height. We denote by C^t the sites of height t which are in the open cluster C.

Definition 8 (Correspondence criterion). *We say that a space-time diagram* $(c^t)_{t>0}$ *and a labeled directed graph* $\mathbb{L}(p)$ *satisfy the* correspondence criterion at step t *if and only if the cells of* $g(C^t)$ *have at least one of their neighbors in a different state. We say that they satisfy the correspondence criterion if and only if they satisfy the correspondence criterion for all* $t \geqslant 0$.

Note that, satisfying the correspondence criterion implies that the cells of $g(C)$ are all active. The coupling will be define such if the correspondence criterion is true at time t, it remains true at time $t + 1$. To achieve this goal efficiently, we will consider only local criteria.

Definition 9 (Candidate). *A site is a* candidate *of height* $t + 1$ *if and only if at least one of its predecessors is in* C^t. *We denote by* $\hat{C}^{t+1}$ *the set of candidates of height of* $t + 1$.

Definition 10 (Constrained cells). *A cell* c_i^t *is* constrained *at time* t *if and only if* $c_i^{t+2} \in g(V)$ *and* $g^{-1}(c_i^{t+2})$ *is in* $\hat{C}^{t+1}$.

We have to find a way such that constrained cells possess a neighbor in a different state than themselves after two iterations of FLIP-IF-NOT-ALL-0. We will have to consider different patterns. For the rest of the paper, we will use the following kind of notation  to represent the patterns. Here,  designs a set of two consecutive cells i and $i+1$ such that $c_i = 0$, $c_{i+1} = 1$ and the arrowed cell $i+1$ is constrained.

Definition 11 (block). *A block is a set of cells which are in the neighborhood* $\square\overset{\downarrow}{\blacksquare}$, $\overset{\downarrow}{\blacksquare}\square$, $\blacksquare\overset{\downarrow}{\square}$, $\overset{\downarrow}{\square}\blacksquare$, $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$, $\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}$, $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$, $\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}$, $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}$, $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}$, $\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$ *or* $\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$.

Lemma 7. *For any configuration, it is possible to compute a set of blocks such that any constrained cell of the configuration is in exactly one block.*

Proof. To prove that any constrained cell can belong to a block, we enumerate all the possible neighborhoods of the cell (see figure 4).

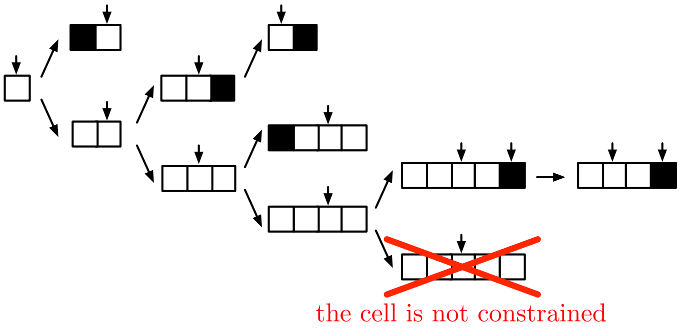


Fig. 4. Proof that any constrained cell can be assigned in a block.

Now, consider a set B of blocks such that each constrained cell appears in at least one block of B. We show how to find a set B' of blocks such that redundancies can be eliminated, *i. e.* each cell appears in exactly one block of B'. If a constrained cell belongs to two different blocks of B, we modify the two blocks like this:

- if a constrained cell belongs to block $\square\overset{\downarrow}{\blacksquare}$ and another block then remove block $\square\overset{\downarrow}{\blacksquare}$. Apply the same method to blocks $\overset{\downarrow}{\blacksquare}\square$, $\blacksquare\overset{\downarrow}{\square}$ or $\overset{\downarrow}{\square}\blacksquare$.
- if the white constrained cell of a block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$ is the same white constrained cell of a block $\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}$, then these two blocks can be replaced by block $\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$.

- if the black constrained cell of a block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$ is the same black constrained cell of a block $\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}$, then this two blocks can be replaced by block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}$.
- if the white constrained cell of a block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$ is also the left white constrained cell of a block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}$, then remove block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$.
- if the white constrained cell of a block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$ is also the right white constrained cell of a block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}$, then the pattern $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$ can be decomposed as $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\,\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$.
- if the black constrained cell of a block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$ is also the left black constrained cell of a block $\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$, then the pattern $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$ can be decomposed as $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\,\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$.
- if the white constrained cell of a block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$ is also the white constrained cell of a block $\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$, then remove block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$.
- the analysis of block $\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare\overset{\downarrow}{\square}$ is symmetric to the analysis of block $\overset{\downarrow}{\square}\blacksquare\overset{\downarrow}{\blacksquare}\blacksquare$.
- if the white constrained cell of a block $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$ is the same white constrained cell of a block $\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}$, then these two blocks can be replaced by block $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}$.
- if the black constrained cell of a block $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$ is the same black constrained cell of a block $\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}$, then this two blocks can be replaced by block $\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$.
- if the black constrained cell of a block $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$ is also the left black constrained cell of a block $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}$, then remove block $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$.
- if the black constrained cell of a block $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$ is also the right black constrained cell of a block $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}$, then the pattern $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$ can be decomposed as $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square\,\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$.
- if the white constrained cell of a block $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$ is also the left white constrained cell of a block $\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$, then the pattern $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$ can be decomposed as $\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square\,\overset{\downarrow}{\blacksquare}\square\overset{\downarrow}{\square}\square$.

- if the white constrained cell of a block ■□□□ is also the right white constrained cell of a block □□□■□□□, then remove block ■□□□.
- the analysis of block □□□■ is symmetric to the analysis of block ■□□□.
- if the white constrained cell of a block ■□□□■ is also the left white constrained cell of a block □□□■□□□, then the pattern ■□□□■□□□ can be decomposed as ■□□□ ■□□□.
- if the white constrained cell of a block ■□□□■ is also the right white constrained cell of a block □□□■□□□, then the pattern □□□■□□□■ can be decomposed as □□□■ □□□■.
- if the black constrained cell of a block □■■■□ is also the left black constrained cell of a block ■■■□■■■, then the pattern □■■■□■■■ can be decomposed as □■■■ □■■■.
- if the black constrained cell of a block □■■■□ is also the right black constrained cell of a block ■■■□■■■, then the pattern ■■■□■■■□ can be decomposed as ■■■□ ■■■□.
- there are no other possible redundancies between blocks.

Note that these operations either remove a block or replace a block by smaller ones. Thus, the process ends and all redundancies are removed.

Lemma 8. *For any block, there is a probability at least α^{12} that all the constrained cell of the block possess a neighbor in a different state than themselves after two iterations of* FLIP-IF-NOT-ALL-0*.*

Proof. Figure 5 shows that for any block, all the constrained cell of the block can satisfy the correspondence criterion by firing at most 12 cells of the block during the next two steps.

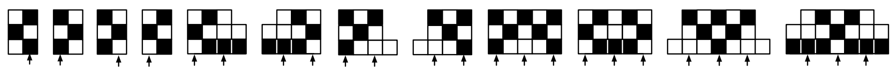


Fig. 5. How to validate the correspondence criterion for any block. The cells drawn at time t and $t+1$ are either inactive or fired.

With the previous lemma, we can achieve our coupling.

Theorem 4. *If* $\alpha \geqslant \sqrt[12]{1-(1-p)^6}$ *then we can define a coupling such that the correspondence criterion is true for all* $t \in \mathbb{N}$.

Proof. To prove this result, we show that if $\alpha \geqslant \sqrt[12]{1-(1-p)^6}$ then there is a coupling such that the correspondence criterion is true for all $t \geqslant 0$. Consider $n \in \mathbb{N}$ and the initial configuration c^0 where $c^0_i = 1$ if and only if $i = 0 \bmod 2$. The correspondence criterion is true at time 0. We suppose that the correspondence criterion is true at time t. We explain how to build a coupling such that this criterion stays true at time $t+1$. Using lemma 7, it is possible to find a set B of blocks such that each constrained cell appears exactly in one block of B. Each cell which is not in a set of B fires with probability α independently from the other cells. Each bond which does not end to a candidate at time $t+2$ is open with probability p independently from the other bonds. Now consider a block of B, we consider a random variable X uniformly distributed between 0 and 1 and independent from any other random variable of the coupling. According to lemma 8, there is a probability at least α^{12} that all the constrained cell of a block possess a neighbor in a different state than themselves after two iterations of FLIP-IF-NOT-ALL-0. Since $\alpha \geqslant \sqrt[12]{1-(1-p)^6}$, and a block contains at most three constrained cells, it is possible to define a coupling using the random variable X (see [19]) such that:

- each cell fires with probability α independently from other cell;
- each bond is open with probability p independently from other bond;
- if $0 \leqslant X \leqslant 1-(1-p)^6$ then the cells of the block fire such that all the constrained cell of a block possesses a neighbor in a different state than themselves after two iterations of FLIP-IF-NOT-ALL-0;
- if $1-(1-p)^6 < X \leqslant 1$ then all the bonds are closed.

Theorem 5. *Consider a sequence of configurations* $(c^t)_{t\geqslant 0}$ *evolving under rule* FLIP-IF-NOT-ALL-0 *and* α*-asynchronous dynamics where* c^0 *contains at least one black cell and one white cell, if* $\alpha \geqslant \sqrt[12]{1-(1-(\frac{16^2-1}{16^2})^2)^6}$ *then* $\mathbb{E}[T] = \Omega(2^n)$.

Proof. Consider that $\alpha \geqslant \sqrt[12]{1-(1-\frac{16^2-1}{16^2})^6}$ then its possible to define the coupling between FLIP-IF-NO-ALL-0 and $\mathbb{L}(p,n)$ with $p > \frac{16^2-1}{16^2}$. According to theorem 4, there exists an initial configuration such that the correspondence criterion is true for all $t \geqslant 0$. According to lemma 3, there exists $\epsilon > 0$ such that there is a site of height 2^n in the open cluster with probability at least ϵ. According to the coupling definition, the probability that at least one cell is active in c^{2^n} is greater than ϵ.

5 Future Works

To conclude, we presented here the first formal proof of a phase transition in simple stochastic cellular automata. We were not able to determine the exact value of α_c but this question is still open for oriented percolation as well. The proof presented here cannot be directly extended for proving polynomial time convergence in expectation for $0.5 \leqslant \alpha \leqslant \alpha_c$ because lemma 2 became false when $\alpha > 0.5$.

First results on the analysis of phase transition in α-asynchronous cellular automata were presented in [19]. This study was made on FLIP-IF-NOT-ALL-EQUAL. The behavior of this automata seems more interesting than the one of FLIP-IF-NOT-ALL-0. Unfortunately, its analysis is harder because there are two stable configurations for FLIP-IF-NOT-ALL-EQUAL: $\bar{0}$ and $\bar{1}$. Lemma 2 does not apply: black cells can be inactive. Thus, the proof presented cannot be extended for this automata and its analysis remains open.

Acknowledgements: Thanks to Nicolas Schabanel for the simulations of FLIP-IF-NOT-ALL-0 and his useful comments.

References

1. P. Chassaing and L. Gerin. Asynchronous cellular automata and brownian motion. In *Proceedings of AofA 2007*, volume AH of *DMTCS Proceedings*, pages 385–402, 2007.
2. M. Cook. Universality in elementary cellular automata. *Complex system*, 15:1–40, 2004.
3. N. Schabanel D. Regnault and É. Thierry. Progresses in the analysis of stochastic 2D cellular automata: A study of asynchronous 2D minority. *Theoretical Computer Science*, 410:4844–4855, 2009.
4. N. Fatès. Experimental study of elementary cellular automata dynamics using the density parameter. *Discrete Mathematics and Theoretical Computer Science*, AB:155–166, 2003.
5. N. Fatès. Asynchronism induces second order phase transitions in elementary cellular automata. *Journal of Cellular Automata*, 4(1):21–38, 2009.
6. N. Fatès. Stochastic cellular automata solve the density classification problem with an arbitrary precision. In *Proceedings of STACS 2011*, pages 284–295, 2011.
7. N. Fatès, M. Morvan, N. Schabanel, and É. Thierry. Asynchronous behaviour of double-quiescent elementary cellular automata. In *Proc. of MFCS'2005*, volume 3618 of *LNCS*, pages 316–327. Springer, 2005.
8. N. Fatès, M. Morvan, N. Schabanel, and É. Thierry. Fully asynchronous behavior of double-quiescent elementary cellular automata. *Theoretical Computer Science*, 362:1–16, 2006.
9. N. Fatès, D. Regnault, N. Schabanel, and É. Thierry. Asynchronous behaviour of double-quiescent elementary cellular automata. In *Proceedings of LATIN'2006*, volume 3887 of *LNCS*. Springer, 2006.
10. P. Gács. Reliable cellular automata with self-organization. *Journal of Statistical Physics*, 103(1/2):45–267, 2001.
11. P. Gács and J. Reif. A simple three-dimensional real-time reliable cellular array. *Journal of Computer and System Sciences*, 36(2):125–147, 1988.
12. D. R. Karger. Global min-cuts in $\mathcal{RNC}$, and other ramifications of a simple mincut algorithm. In *Proc. of the 4th Annual ACM-SIAM Symposium on Discrete algorithms*, pages 21–30, 1993.

13. J. Kari. Theory of cellular automata: a survey. *Theoretical Computer Science*, 334:3–33, 2005.
14. M. Land and R. K. Belew. No perfect two-state cellular automata for density classification exists. *Physical review letters*, 74:5148–5150, 1995.
15. E. Mossel and S. Roch. Slow emergence of cooperation for win-stay lose-shift on trees. *Machine Learning*, 67(1-2):7–22, 2006.
16. K. Nagel and M. Schreckenberg. A cellular automaton model for freeway traffic. *Journal de Physique I*, 2(12):2221–2229, 1992.
17. N. Ollinger and G. Richard. 4 states are enough! *Theoretical Computer Science*, 412:22–32, 2011.
18. D. Regnault. Abrupt behavior changes in cellular automata under asynchronous dynamics. In *Proceedings of 2nd European Conference on Complex Systems*, 2006.
19. D. Regnault. Directed percolation arising in stochastic cellular automata. In *Proceedings of MFCS 2008*, volume 5162 of *LNCS*, pages 563–574, 2008.
20. P. Sarkar. A brief history of cellular automata. *ACM Computing Surveys*, 32(1):80–107, March 2000.
21. A. Toom. Stable and attractive trajectories in multicomponent systems. *Advances in Probability*, 6:549–575, 1980.