For submission to the Journal of Automata, Languages and Combinatorics
Created on March 31, 2020

# ON THE PROBABILITY OF BEING SYNCHRONIZABLE

Mikhail V. Berlinkov

*Institute of Mathematics and Natural Sciences,*
*Ural Federal University 620000 Ekaterinburg, Russia*
m.berlinkov@gmail.com

ABSTRACT

We prove that a random automaton with $n$ states and any fixed non-singleton alphabet is synchronizing with high probability. Moreover, we also prove that the convergence rate is exactly $1 - \Theta(\frac{1}{n})$ as conjectured by [Cameron, 2011] for the most interesting binary alphabet case.

*Keywords:* synchronizing automata, random mappings, random digraphs

## 1. Synchronizing automata

Suppose $\mathcal{A}$ is a complete deterministic finite automaton whose input alphabet is $A$ and whose state set is $Q$. The automaton $\mathcal{A}$ is called *synchronizing* if there exists a word $w \in A^*$ whose action *resets* $\mathcal{A}$, that is, $w$ leaves the automaton in one particular state no matter at which state in $Q$ it is applied: $q.w = q'.w$ for all $q, q' \in Q$. Any such word $w$ is called a *reset word* of $\mathcal{A}$. One can check that a word $(ab^3)^2a$ is reset for an automaton $\mathfrak{C}_4$ depicted on Figure 1 (left). For a brief introduction to the theory of synchronizing automata we refer the reader to the survey [15].

Synchronizing automata serve as transparent and natural models of error-resistant systems in many applications (coding theory, robotics, testing of reactive systems) and also reveal interesting connections with symbolic dynamics and other parts of mathematics. We take an example from [2]. Imagine that you are in a dungeon consisting of a number of interconnected caves, all of which appear identical. Each cave has a common number of one-way doors of different colours through which you may leave; these lead to passages to other caves. There is one more door in each cave; in one cave the extra door leads to freedom, in all the others to instant death. You have a map of the dungeon with the escape door identified, but you do not know in which cave you are. If you are lucky, there is a sequence of doors through which you may pass which takes you to the escape cave from any starting point.

This work is supported by Russian Foundation for Basic Research, grant no. 16-01-00795, and the Competitiveness Enhancement Program of Ural Federal University.

In terms of the provided example, the result of this paper is quite positive; we prove that for a uniformly at random chosen dungeon (automaton) there is a life-saving sequence (reset word) with probability $1 - O(\frac{1}{n^{0.5c}})$ where $n$ is the number of caves (states) and $c$ is the number of colours (letters). Moreover, we prove that the convergence rate is tight for the most interesting 2-colour case, thus confirming Cameron's conjecture [5]. Up to recently, the best results in this direction were much weaker: in [17] it was proved that random 4-letter automata are synchronizing with probability $p$ for a specific constant $p > 0$; in [16] it was proved that if a random automaton with $n$ states has at least $72\ln(n)$ letters then it is almost surely synchronizing. Recently, Nicaud [11] has shown (independently) by a completely different pure combinatoric techniques that a random $n$-state automaton with 2 letters is synchronizing with probability $1 - O(n^{-\frac{1}{8}+o(1)})$. Our results give a much better convergence rate.

## 2. The probability of being synchronizable

Let $Q$ stand for $\{1, 2, \dots, n\}$ and $\Sigma_n$ for the probability space of all unambiguous maps from $Q$ to $Q$ with the uniform probability distribution. Throughout this paper let $\mathcal{A} = \langle Q, \{a, b\}\rangle$ be a random automaton, that is, $\mathcal{A}$ is chosen uniformly at random from the set of all 2-letter automata with $n$ states. As we do not have any restrictions on $\mathcal{A}$, we can choose such an automaton by picking $a$ and $b$ independently and uniformly at random (u.a.r) from $\Sigma_n$, that is,

$$P(\mathcal{A} = \langle Q, \{a', b'\}\rangle) = P_{a \in \Sigma_n}(a = a')P_{b \in \Sigma_n}(b = b').$$

In its turn, the choice of a random mapping from $\Sigma_n$ is equivalent to choosing independently and u.a.r. an image for each state $q \in Q$. Here and below by independence of two objects $O_1(\mathcal{A})$ and $O_2(\mathcal{A})$ determined by an automaton, we mean the independence of the corresponding events $O_1(\mathcal{A}) = O_1$ and $O_1(\mathcal{A}) = O_2$, for each instances $O_1, O_2$ from the corresponding sets. Notice also that if two objects are defined by independent objects, they are also independent, e.g. the set of all self-loops by $a$ is independent of the letter $b$. We will extensively utilize this argument throughout the paper.

Our main result is the following theorem.

**Theorem 1.** *The probability of being synchronizable for* 2*-letter random automata with* $n$ *states equals* $1 - \Theta(\frac{1}{n})$.

## 3. Connectivity and the Upper Bound

Let us call *subautomaton* a terminal strongly-connected component of $\mathcal{A}$[1]. Call an automaton *weakly connected* if it has only one *subautomaton.* Observe that if an automaton is synchronizing, it must be weakly connected. Hence the following lemma gives the upper bound of Theorem 1.

[1] A strongly-connected component $S$ is terminal when $S \cdot u \subseteq S$ for every $u \in A^*$.

**Lemma 2.** *The probability that $\mathcal{A}$ is not weakly connected is at least $\Omega(\frac{1}{n})$.*

*Proof.* Let us count the number of automata having exactly one *disconnected loop*, that is automata having a state with only (two) incoming arrows from itself. We first choose a unique state $p$ of a disconnected loop in $n$ ways. The transitions for this state is defined in the unique way. For any other state $q$, we define transitions in $(n-2)^2$ ways by choosing the image for each letter to be any state except $p, q$, ensuring that $q$ is not disconnected and $p$ has no transitions from other states. Thus the probability of having exactly one disconnected loop is at least

$$\frac{n(n-2)^{2(n-1)}}{n^{2n}} = \frac{1}{n}\left(1 - \frac{2}{n}\right)^{2(n-1)} = \Theta\left(\frac{1}{n}\right).$$

This concludes the proof of the lemma as such automata are not weakly connected. □

The following lemma can be obtained as a consequence of [6][Theorem 3] but we present the proof here for the sake of completeness.

**Lemma 3.** *The number of states in each subautomaton of $\mathcal{A}$ is at least $n/4$ with probability $1 - O(\frac{1}{n})$.*

*Proof.* Given $1 \leq i < n/2$, there are $\binom{n}{i}$ ways to choose a set of states of a subautomaton of size $i$, then there are $i^{2i}$ ways to define transitions for both letters in the chosen subautomaton, and $n^{2(n-i)}$ ways to define transitions for the remaining states. Thus, the number of automata with a subautomaton of size $i$ is at most

$$N_i = \binom{n}{i} i^{2i} n^{2(n-i)}. \tag{1}$$

Let us consider the ratio $N_{i+1}/N_i$. We have

$$N_{i+1}/N_i = \frac{n-i}{i+1}\frac{(i+1)^{2(i+1)}}{i^{2i}n^2} = \frac{(n-i)(1+\frac{1}{i})^{2i}}{n^2} \leq \frac{e^2(n-i)(i+1)}{n^2}. \tag{2}$$

Trivial analysis shows that the maximum of (2) for $i \leq n/4$ is reached when $i$ is maximal, thus (2) is maximal for $i = n/4$ and equals $3e^2(1 + o(1))/16 < 1$. Using that, we have that the total number of automata with a sabautomaton of size smaller than $n/4$ is upper bounded by the sum of the geometric sequence with the common factor smaller than 1 and the first term equals $n^{2n-1}$. The lemma follows from the fact that the total number of binary automata with $n$ states is $n^{2n}$. □

**Lemma 4.** *With probability $1 - O(\frac{1}{n})$ there is only one strongly-connected subautomaton in $\mathcal{A}$.*

*Proof.* Suppose there are at least two strongly connected subautomata in $\mathcal{A}$ having sizes $s_1, s_2$ respectively. Notice that there are at most $3^n$ ways to choose subsets of states of these two subautomata, as this choice can be defined by coloring $Q$ into three colors. Then, transitions for states in these sabutomata can be defined in ${s_1}^{2s_1}$ and ${s_2}^{2s_2}$ ways resp., and for the rest $n - s_1 - s_2$ states they can be defined in at most $n^{2(n-s_1-s_2)}$ ways. As the total number of automata is $n^{2n}$, the probability of this happening can be upper bounded by

$$3^n \left(\frac{s_1}{n}\right)^{2s_1} \left(\frac{s_2}{n}\right)^{2s_2}, \tag{3}$$

where $s_1$ and $s_2$ both have size at least $n/4$ with probability $1-O(\frac{1}{n})$ due to Lemma 3. It can be easily shown that the maximum of (3) is reached when $s_1 = s_2$ when $s_1 + s_2$ is fixed. Similarly, using that the function $x^{cx} = e^{cx \ln x}$ is increasing for $x \geq 1/e$, one can deduce that (3) is maximal when $s_1 = s_2 = n/2$, and thus is upper bounded by

$$3^n \left(\frac{1}{2}\right)^n \left(\frac{1}{2}\right)^n = 3^n/4^n = o(1/n). \tag{4}$$

□

## 4. The Lower Bound

Now we turn to the proof of the lower bound of Theorem 1 by means of a top-down approach. In order to describe the plan, we need a few definitions which we elaborate on further in the proof. First, call a set of states $K \subseteq Q$ *synchronizable* if it can be mapped to one state by some word. Next, a pair of states $\{p, q\}$ is called *stable* if it cannot be mapped by any word into a non-synchronizable pair. Finally, the *underlying graph* of a letter (or mapping) $a$ is a digraph $\Gamma_a = \Gamma(Q, \{(q, q.a) \mid q \in Q\})$. Since it has common out-degree 1, the underlying graph consists of one or more maximal weakly connected components [2] called *clusters*, each one consisting of a unique cycle and trees rooted on this cycle (see Figure 1).

The plan is as follows. After recalling necessary well known inequalities in Subsection 4.1, we state an upper bound on the number of clusters of a random mapping in Subsection 4.2, which will be used throughout the paper. Then, in Subsection 4.3 we prove that having the set of *big* clusters for each letter synchronizable is enough to prove the lower bound. Next, in Subsection 4.4 we will show that this property can be provided by having a big enough set of *stable pairs* for each letter independent of the other letter. In Subsection 4.5 we prove that these sets of stable pairs can be built from just one stable pair independent of one of the letters. In Subsection 4.6 we will show that if the underlying graph of one of the letters has a unique highest 1-*branch*, then there is a pair of states completely defined by this letter which is additionally stable provided the *crown* of the 1-branch intersects with any *subautomaton.* Finally, in Subsection 4.7 we prove that with high probability one of the letters of a random automaton has a unique highest 1-branch whose crown is big enough to be reachable from any subautomaton with high probability.

[2] a maximal by size subgraph containing a state accessible from every state of the subgraph

### *4.1. Useful Asymptotics*

First, let us recall few asymptotic equations that will be used throughout the paper. The Stirling's approximation formula states that for $k > 0$

$$k! = \left(\frac{k}{e}\right)^k \sqrt{2\pi k}(1 + o(1)). \tag{st}$$

Using Taylor's expansion of a logarithm $\ln(1-x) = -\sum_{i=1}^{+\infty} \frac{x^i}{i}$, for $0 < k \leq m$ we get

$$\left(1 - \frac{k}{m}\right)^m = \exp\left(-\sum_{i=1}^{+\infty} \frac{k^i}{im^{i-1}}\right). \tag{exp1}$$

$$e^{-2k} \leq \left(1 - \frac{k}{m}\right)^m \leq e^{-k} \text{ if } k \leq \frac{m}{2}. \tag{exp2}$$

$$\left(1 - \frac{k}{m}\right)^m = e^{-k(1+o(1))} \text{ if } k = o(m). \tag{exp3}$$

$$\left(1 - \frac{k}{m}\right)^m = e^{-k}\Theta(1) \text{ if } k^2 = o(m). \tag{exp4}$$

Using (st), for the number of combinations for $0 < k < m$ we have

$$\begin{aligned}\binom{m}{k} &= \frac{m!}{(m-k)!k!} = \frac{\Theta(1)\left(\frac{m}{e}\right)^{m+1/2}}{\left(\frac{k}{e}\right)^{k+1/2}\left(\frac{m-k}{e}\right)^{m-k+1/2}} = \\ &= \frac{\Theta(1)m^m}{k^k(m-k)^{m-k}}\sqrt{\frac{m}{k(m-k)}} = \frac{\Theta(1)m^k}{k^k(1-\frac{k}{m})^{m-k}}\sqrt{\frac{m}{k(m-k)}}.\end{aligned} \tag{cmb0}$$

If $k = o(m)$, then due to (exp3), (cmb0) simplifies to

$$\binom{m}{k} = \frac{\Theta(1)\left(me^{1+o(1)}\right)^k(1-\frac{k}{m})^k}{k^k}\sqrt{\frac{1}{k}} = \frac{\Theta(1)\left(me(1+o(1))\right)^k}{k^k\sqrt{k}}, \tag{cmb1}$$

and if $k^2 = o(m)$, due to (exp4) it further simplifies to

$$\binom{m}{k} = \frac{\Theta(1)\left(me^{1+o(1)}\right)^k(1-\frac{k}{m})^k}{k^k}\sqrt{\frac{1}{k}} = \frac{\Theta(1)(me)^k}{k^k\sqrt{k}}. \tag{cmb2}$$

In the general case $0 < k < m$, as $m \leq O(1)k(m-k)$, $(1 - \frac{k}{m}) \leq 1$ and due to (exp2) we can upper bound (cmb0) as follows.

$$\binom{m}{k} \leq \frac{\Theta(1)m^k}{k^k(1-\frac{k}{m})^{m-k}}\sqrt{\frac{m}{k(m-k)}} \leq \frac{O(1)m^k e^{2k}}{k^k} = O(1)\left(\frac{e^2 m}{k}\right)^k. \tag{cmb3}$$

Notice that for $k \in \{0, m\}$, (cmb3) also holds if we assume $0^0 = 1$.

*4.2. The Cluster Structure of Underlying Graphs*

An example of an automaton with 4 states and the underlying graphs of its letters is given in Figure 1 – the underlying graph of $b$ (on the right) has only one cluster, while the underlying graph of $a$ (in the middle) consists of 3 clusters having the sets of vertices $\{0,1\}$, $\{2\}$, $\{3\}$, correspondingly. Clearly, each directed graph with $n$ vertices and constant out-degree 1 corresponds to a unique map from $\Sigma_n$. Thus we can consider $\Sigma_n$ as the probability space with the uniform distribution on all directed graphs with constant out-degree 1.

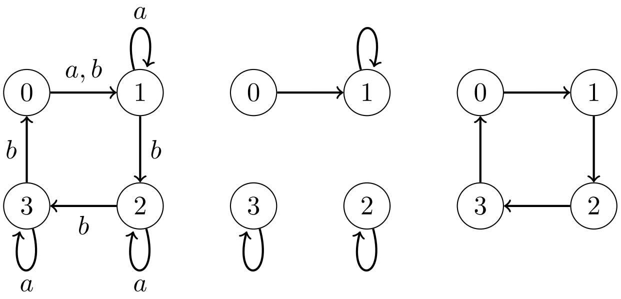

Figure 1: Left to right: an automaton $\mathfrak{C}_4$, the underlying graphs of its letters $a$ and $b$.

The number of clusters (or cycles) of a random mapping is known to be concentrated around $\frac{\ln n}{2}$ and accurate bounds were established for various rates of growth of the number of clusters (see e.g. [13]). We need a stronger upper bound for large deviations of the number of clusters.

**Lemma 5 (Nicaud, 2019).** *With probability* $1-o(\frac{1}{n^4})$*, a random digraph from* $\Sigma_n$ *has at most* $5\ln n$ *clusters.*

*Proof.* We will make use of powerful and beautiful theory of Analytic Combinatorics comprehensibly developed in [9]. Let $t_{n,k}$ be the number of mappings from $[n]$ to $[n]$ having $k$ clusters (component). The associated exponential bivariate generating series defined by

$$T(z,u) := \sum_{n\geq 0}\sum_{k\geq 0}\frac{t_{n,k}}{n!}z^n u^k$$

can be written, using the formal method (see e.g. [8]):

$$\begin{cases} T(z,u) = \exp\left(u\log\left(\frac{1}{1-C(z)}\right)\right), \\ C(z) = z\exp(C(z)) \end{cases} \tag{5}$$

As every coefficient $(t_{n,k})$ of the development of $T(z,u)$ near $(z,u) = (0,0)$ is nonnegative, we have, for every $0 < \rho < e^{-1}$ and every positive $\tau$:

$$T(\rho,\tau) = \sum_{n,k} \left([z^n u^k]T(z,u)\right) \rho^n \tau^k \geq \left([z^n u^k]T(z,u)\right) \rho^n \tau^k,$$

for every $n$ and every $k$. Hence (this is called a saddle-point bound):

$$\frac{t_{n,k}}{n!} = [z^n u^k]T(z,u) \leq \frac{T(\rho,\tau)}{\rho^n \tau^k} \tag{6}$$

This holds for any $0 < \rho < e^{-1}$, any positive $u$ and any $n$ and $k$. We choose

$$\rho = e^{-1}\left(1 - \frac{1}{2n}\right); \quad k = \lambda_n + i; \quad \lambda_n := \lceil \lambda \log n \rceil,$$

for any $i \geq 0$ and some $\lambda > 0$ to be fixed later. Due to (6), the probability that a random mapping with $n$ states has $k$ components is

$$\frac{t_{n,k}}{n^n} \leq \frac{n!}{n^n} \frac{T(\rho,\tau)}{\rho^n \tau^k}. \tag{7}$$

First observe that, due to (st) and the choice of $\rho$,

$$\frac{n!}{n^n \rho^n} = \frac{\sqrt{2\pi n}(1+o(1))}{\left(1-\frac{1}{2n}\right)^n} = \frac{\sqrt{2\pi n}}{e^{-1/2}}(1+o(1)). \tag{8}$$

Now we estimate $T(\rho,\tau)$. We use the classical development near $\rho = e^{-1}$ (see [8])

$$C(z) = 1 - \sqrt{2(1-ez)} + O(1-ez),$$

and thus $1 - C(\rho) = \frac{1}{\sqrt{n}} + O(\frac{1}{n})$, and thus from (5) we have

$$T(\rho,\tau) = \exp\left(\tau \log \frac{\sqrt{n}}{1 + O(1/\sqrt{n})}\right) = n^{\tau/2(1+o(1/n^{1/2}))}. \tag{9}$$

Putting together (7),(8) and (9), for some positive $\alpha$,

$$\frac{t_{n,k}}{n^n} \leq \alpha\sqrt{n}\ \frac{n^{\tau/2}}{n^{\lambda\log\tau}}\tau^{-i} = \alpha\ n^{\frac{1}{2}(\tau)-\lambda\log\tau}\tau^{-i}$$

If we choose $\tau = \lambda = 5$, we have $\frac{1}{2}(\tau+1) - \lambda\log\tau \leq -5$, so that

$$\frac{t_{n,k}}{n^n} \leq \alpha n^{-5} 5^{-i+1},$$

and therefore

$$\mathbb{P}_{g\in\Sigma_n}(g \text{ has more than } 5\log n \text{ clusters}) = \sum_{i\geq 0} \frac{t_{n,\lceil 5\log n\rceil + i}}{n^n} \leq \frac{\beta}{n^5},$$

for some positive $\beta$. □

### *4.3. Independent Synchronizable Sets*

A set of states $K \subseteq Q$ is *synchronizable* if it can be mapped to one state by some word. In contrast, a pair of states $\{p, q\}$ is called a *deadlock* if $p.s \neq q.s$ for each word $s$. Notice that a deadlock pair remains deadlock under the action of any word.

First we aim to show that for proving that $\mathcal{A}$ is synchronizing with probability $1 - O(\frac{1}{n})$, it is enough to show that with probability $1 - O(\frac{1}{n})$ for each letter there is a large synchronizable set of states which is completely defined by this letter. Given $x \in \{a, b\}$ and a constant $0 < \alpha_c < 1$ to be chosen later, we define $S_x$ to be the set of *big* clusters of $\Gamma_x$ – of size more than $n^{\alpha_c}$ states – and define $T_x$ to be the complement of $S_x$, or equivalently, $T_x$ to be the set of *small* clusters – the clusters containing at most $n^{\alpha_c}$ states. Since $S_x$ and $T_x$ are completely defined by $x$, both are independent of the other letter.

Due to Lemma 5, with probability $1 - O(\frac{1}{n})$ there are at most $5 \ln n$ clusters in $\Gamma_x$, whence $T_x$ contains at most $5 \ln{(n)} n^{\alpha_c}$ states with probability $1 - O(\frac{1}{n})$. Given a set of clusters $X$, denote by $\widehat{X}$ the set of states in the clusters of $X$.

**Theorem 6.** *If $\widehat{S_a}$ and $\widehat{S_b}$ are synchronizable and the underlying digraphs of $a$ and $b$ has at most $5 \ln n$ clusters each, then $\mathcal{A}$ is synchronizing with probability $1 - O(\frac{1}{n})$.*

*Proof.*

**Remark 7.** Let subsets $R, S \subseteq Q$ be defined independent of one another and $0 < k \leq |R|$. Then the probability that $|R \cap S| \geq k$ is at most

$$Pr(|R \cap S| \geq k) \leq \binom{|S|}{k}\binom{n}{|R| - k}/\binom{n}{|R|}. \tag{10}$$

In particular, if $|R| = O(1)$, the probability that $R$ overlaps with $S$ by at least $k$ states is at most

$$Pr(|R \cap S| \geq k) = O((|S|/n)^k). \tag{11}$$

*Proof.* As $R$ is independent of $S$, we may assume that $S$ is fixed and we choose $R$ at random. First we choose some $k$ states from $S$ in $\binom{|S|}{k}$ ways, then we choose the rest $|R| - k$ states of $R$ from all $n$ states in $\binom{n}{|R|-k}$ ways, and finally divide by $\binom{n}{|R|}$ – the total number of ways to choose $R$ from $Q$. This is an upper bound because if the overlap is larger than $k$, there are different choices of overlap that lead to the same choice of $R$. (11) easily follows from (cmb2). □ □

**Lemma 8.** *If a pair $\{p, q\}$ is independent*[3] *of one of the letters and $p$ and $q$ are independent of one another (under the condition that $p \neq q$), it is a deadlock with probability $O(\frac{\ln n}{n^{2-\alpha_c}})$.*

[3] It is assumed there is an event $E(\mathcal{A}) = \{p, q\}$, which doesn't matter in the context.

*Proof.* We will often refer to the fact that if a pair of states $\{p,q\}$ is a deadlock, then $|\widehat{T_a} \cap \{p,q\}| \geq 1$ and $|\widehat{T_b} \cap \{p,q\}| \geq 1$. Indeed, otherwise the pair would belong to either $\widehat{S_a}$ or $\widehat{S_b}$ which are synchronizable.

Suppose $\{p,q\}$ is independent of $a$. Let us consider a chain of states $\{p.a, q.a, p.a^2, q.a^2, \dots p.a^k, q.a^k\}$ for $k > 0$. Notice that if $p.a^i = q.a^i$ for some $i$, then $\{p,q\}$ is not a deadlock. Hence for each $i \leq k$ either $p.a^i$ or $q.a^i$ belongs to $\widehat{T_b}$.

If $\{p.a, q.a\} = \{p,q\}$, that is $a$ acts either as a transposition or as the identity on $\{p,q\}$, we upper bound the probability of this case by $O(\frac{1}{n^2})$ by Remark 7 applied to $\{p.a, q.a\}$ and $\{p,q\}$.

If $|\{p.a, q.a\} \cap \{p,q\}| = 1$, say $q.a = p$ and $p.a \notin \{p,q\}$, then the probability of $q.a = p$ is $O(1/n)$. Furthermore, $\{p.a^2, p.a\} = \{p,q\}.a^2$ is a deadlock and thus $|\{p.a^2, p.a\} \cap \widehat{T_b}| \geq 1$. Notice that $p.a$ may not be independent of $T_b$ in this case because $p.a \notin \{p,q\}$ and $\{p,q\}$ may depend on $b$, but without this condition it would. Thus, we have

$$Pr(|\widehat{T_b} \cap \{p.a^2, p.a\}| \geq 1 \mid p.a \notin \{p,q\}) = \frac{Pr(|\widehat{T_b} \cap \{p.a^2, p.a\}| \geq 1; p.a \notin \{p,q\})}{Pr(p.a \notin \{p,q\})}.$$

Using Remark 7, we have

$$Pr(|\widehat{T_b} \cap \{p.a^2, p.a\}| \geq 1; p.a \notin \{p,q\}) \leq Pr(|\widehat{T_b} \cap \{p.a^2, p.a\}| \geq 1) = O\left(\frac{\ln n}{n^{1-\alpha_c}}\right).$$

Also, we have that $Pr(p.a \notin \{p,q\}) = 1 - O(1/n)$. It remains to notice that $q.a = p$ is independent of $|\{p.a^2, p.a\} \cap \widehat{T_b}| \geq 1$. Other cases ($p.a = p$, $q.a = q$, $p.a = q$) can be proved in the same way.

If for all $k = \lceil \frac{2-\alpha_c}{1-\alpha_c} \rceil + 1$ all states in the chain are different, then for each $i = 1, 2, \dots, k$ there is a (distinct) state $r_i \in \widehat{T_b} \cap \{p.a^i, q.a^i\}$ (if both $p.a^i, q.a^i$ are in $\widehat{T_b}$, let $r_i = p.a^i$). As this chain is defined by images of $\{p,q\}$ by $a$, it is independent of $\widehat{T_b}$, by Remark 7, the probability that the chain overlaps with $\widehat{T_b}$ by at least $k$ states is at most, as $|\widehat{T_b}| = O(n^{\alpha_c} \ln n)$,

$$O((|\widehat{T_b}|/n)^k) = O((\ln n / n^{1-\alpha_c})^k) = O(1/n^{2-\alpha_c}).$$

Otherwise, there is the smallest $2 \leq i \leq k$ such that

$$|\{p.a^i, q.a^i\} \cap \{p, q, p.a, q.a, p.a^2, q.a^2, \dots p.a^{i-1}, q.a^{i-1}\}| > 0.$$

As $k$ is constant, by Remark 7 this happens with probability $O(1/n)$ and, additionally, we have that $|\{p.a, q.a\} \cap \widehat{T_b}| > 0$ which happens with probability $O(1/n^{1-\alpha_c})$ in the case $|\{p.a, q.a\} \cap \{p,q\}| = 1$ above. It remains to notice that those two events are independent as the former one is defined by $\{p.a, q.a\}$ and $\widehat{T_b}$ while the latter is defined by $\{p.a^i, q.a^i\}$ for $i > 1$ (as images of different states are chosen independently).

□ □

**Lemma 9.** *Given a constant $k$ and a letter $a$, the probability that there is a deadlock pair in $a$-clusters with cycles of size at most $k$ is at most $O(\frac{\ln^3 n}{n^{2-\alpha_c}})$.*

*Proof.* First notice that we can map a deadlock pair to a pair on cycles by applying $a^n$. Since the set of cycle states in these clusters is defined by the letter $a$, it is independent of the other letter. Thus the probability that at least one of the pairs from this set is deadlock is upper bounded by the product of the total number of such pairs (which is at most $25k^2 \ln^2 n$) with the probability of a pair independent of the letter $b$ to be a deadlock (which is $O(\frac{\ln n}{n^{2-\alpha_c}})$ by Lemma 8). The statement follows. □ □

Now let us bound the probability that $\mathcal{A}$ is not synchronizing. If this was the case, $\mathcal{A}$ would possess a deadlock pair $\{p, q\}$. Given a state $r$, denote by $c_r$ the cycle of the cluster containing $r$ in $\Gamma_a$ and by $s_r$ the length of this cycle. Denote also by $c_{r,i}$ the $i$-th state on the cycle $c_r$ for some order induced by the cycle $c_r$, i.e., $c_{r,i}.a = c_{r,i+1 \bmod s_r}$. Let $d$ be the g.c.d. of $s_p$ and $s_q$. Then for some $0 \leq x < d$ and all $0 < k_1, k_2, i \in Z_d = \{0, 1, \dots, d-1\}$, the pairs

$$\{c_{p,(i+k_1 d) \bmod s_p}, c_{q,(x+i+k_2 d) \bmod s_q}\} \text{ are deadlocks} \tag{12}$$

because we can get them as images of $\{p, q\}$. It follows that in each of these pairs at least one of the states is in $\widehat{T_b}$.

Since $k_1, k_2$ are arbitrary in (12), for each $i \in Z_d$ either $c_{p,(i+k_1 d) \bmod s_p} \in \widehat{T_b}$ for all $k_1$ or $c_{q,(x+i+k_2 d) \bmod s_q} \in \widehat{T_b}$ for all $k_2$. First, we choose $x$ in $d$ ways, and for some $k \in \{0, 1, \dots d\}$ we choose $k$-subset $I_p \subseteq \{0, 1, \dots d\}$ in $\binom{d}{k}$ ways such that $c_{p,(i+k_1 d) \bmod s_q} \in \widehat{T_b}$ for all integer $k_1$ and $i \in I_p$. Finally, we choose a set $I_q \subseteq \{0, 1, \dots d\}$ with $|I_q| = d - |I_p|$ [4]. Notice that with such a choice, there are $ks_p/d$ distinct states $c_{p,(i+k_1 d) \bmod s_p} \in \widehat{T_b}$ and $(d-k)s_q/d$ distinct states $c_{q,(x+i+k_2 d) \bmod s_q} \in \widehat{T_b}$.

Denote $z = \frac{ks_p + (d-k)s_q}{d}$. We can assume that $s_p \leq s_q$ implying $d \leq s_p \leq z$. Since $\widehat{T_b}$ is independent of $a$, the probability that the corresponding states from the cycles belong to $\widehat{T_b}$ equals $\binom{|\widehat{T_b}|}{z} / \binom{n}{z}$ (this is a particular case of Remark 7). Thus the probability of such configuration for two chosen clusters, with $s_p, s_q$ being their cycle lengths and $k$ being fixed, is at most

$$f(k) = d\binom{d}{k}\binom{d}{d-k}\binom{|\widehat{T_b}|}{z(k)} / \binom{n}{z(k)}. \tag{13}$$

Simple analysis would show that $f(k)$ increases by $k$ for $n$ big enough because $|\widehat{T_b}| = o(n)$. Hence, the probability (13) for all possible (at most $25 \ln^2 n$) choices of clusters and any choice of $k$ such that $z \geq \frac{1.1}{1-\alpha_c}$ (**Case 1**), can be upper bounded,

[4] We could define $I_q$ to be the complement of $I_p$ and thus wouldn't need to choose it but we do it to simplify description of an algorithm for synchronization testing later.

using (cmb2) and that $d \le s_p \le z, \sum_{k=0}^{d} \binom{d}{k}\binom{d}{d-k} \le 4^d$, as follows

$$(25\ln^2 n)s_p 4^{s_p}\binom{|\widehat{T_b}|}{s_p}\Big/\binom{n}{s_p} \le (25\ln^2 n)O\left(\left(\frac{|\widehat{T_b}|}{n}\right)^{s_p}\right) = o\left(\frac{1}{n}\right). \tag{14}$$

Now, consider the case $z \le \frac{1.1}{1-\alpha_c}$. If $k < d$ (**Case 2**), then $s_q \le zd \le z^2$ and thus both $s_p$ and $s_q$ are $O(1)$ (in terms of $n$) and we are done by Lemma 9.

Finally, in the case $k = d$ (**Case 3**), let us map a deadlock pair $\{p, q\}$ into a deadlock pair $\{p', q_{min}\}$ on the cycle where $q_{min}$ is the state with the smallest index on the cycle $s_q$ and $p' \in s_p$ is the corresponding state on $s_p$. This pair is a deadlock and independent of $b$, provided we have chosen the clusters of $p$ and $q$ with $s_p \le \frac{1.1}{1-\alpha_c}$ (in at most $25\ln^2 n$ ways) and a state $p' \in c_p$ (in at most $\frac{1.1}{1-\alpha_c}$ ways). Due to Lemma 8, the probability that one of these (at most $\frac{1.1}{1-\alpha_c}$) pairs is a deadlock is upper bounded by

$$25\ln^2 n\frac{1.1}{1-\alpha_c}O\left(\frac{1}{n^{2-\alpha_c}}\right) = O(1/n),$$

which completes the proof of the theorem. □

### *4.4. Stability Relation and Induced Colouring*

In view of Theorem 6, it remains to prove that $\widehat{S_a}$ and $\widehat{S_b}$ are synchronizable with probability $1 - O(\frac{1}{n})$. For this purpose, we use the notion of the *stability* relation introduced by Kari [10]. Recall that a pair of states $\{p, q\}$ is called *stable* if for every word $u$ there is a word $v$ such that $p.uv = q.uv$. The *stability* relation given by the set of stable pairs joined with a diagonal set $\{\{p, p\} \mid p \in Q\}$ is invariant under the actions of the letters and complete whenever $\mathcal{A}$ is synchronizing. It is also an equivalence relation on $Q$ because it is transitive and symmetric.

Given a pair $\{p, q\}$, either $\{p, q\}$ is in one $a$-cluster or the states $p$ and $q$ belong to different $a$-clusters. In the latter case, we say that $\{p, q\}$ *connects* these $a$-clusters. Suppose there exists a *large* set $Z_a$ of distinct pairs (namely, $|Z_a| \ge n^{\beta_s}$ where $1-\alpha_c < \beta_s < 0.5$ will be chosen later) that are independent of $a$ and stable with probability $1 - O(\frac{1}{n})$. Consider the graph $\Gamma(S_a, Z_a)$ with the set of vertices $S_a$, and draw an edge between two clusters if and only if some pair from $Z_a$ connects them.

**Lemma 10.** *Let $Z_a$ be a set of at least $n^{\beta_s}$ distinct pairs independent of $a$; then $\Gamma(S_a, Z_a)$ is connected with probability $1 - O(\frac{1}{n})$. If additionally all cycle pairs of one of the clusters from $S_a$ are stable, then $\widehat{S_a}$ is synchronizable* [5].

*Proof.* The latter statement follows from the definition of $S_a$ and the transitivity of the stability relation. Indeed, if $\Gamma(S_a, Z_a)$ is connected, all cycle pairs of the cycles of $S_a$ are stable. Since each pair of $S_a$ can be mapped to a cycle pair of $S_a$, $\widehat{S_a}$ is synchronizable.

[5] In particular, the existence of a loop among cycles of $S_a$ is enough.

Let us turn to the first statement. Since $Z_a$ is independent of $a$, we can choose $Z_a$ uniformly at random for a given random mapping $a$, and estimate the probability that $\Gamma(S_a, Z_a)$ is not connected for that choice. The choice of $Z_a$ can be done as follows. We first choose $2|Z_a|$ states and then randomly join different pairs of chosen states.

Arguing by contradiction, suppose that there is a set of clusters $S' \subsetneq S_a$ such that for $G = union(S')$, we have $|G| \leq 0.5n$ and each pair $\{p, q\} \in Z_a$ either belongs to $G$ or does not intersect with $G$. Notice also that $|G| > n^{\alpha_c}$, because $G$ must contain at least one cluster from $S_a$ and all clusters in $S_a$ contain more than $n^{\alpha_c}$ states.

Denote $m = |Z_a|$. Let $k_1$ pairs from $Z_a$ belong to $G$ and $k_2 = m - k_1$ pairs do not belong to $G$. The probability of such event is at most, $g$ being the size of $G$,

$$2^{5 \ln n} \frac{\binom{g}{2k_1}\binom{n-g}{2k_2}(2k_1)!!(2k_2)!!}{\binom{n}{2m}(2m)!!}(1 + o(1)). \tag{15}$$

Indeed, due to Lemma 5, with probability $1 - O(\frac{1}{n})$ we can choose $G$ (as a subset of clusters) in at most $2^{5 \ln n}$ ways. Then we choose $2k_1$ states from $G \cap Z_a$ in $\binom{g}{2k_1}$ ways and a perfect matching on them in $(2k_1)!!$ ways. Similarly, we choose $k_2$ pairs from $(Q \setminus G) \cap Z_a$ in $\binom{n-g}{2k_2}(2k_2)!!$ ways; next we divide it by the total number of ways to choose $|Z_a|$ pairs $\binom{n}{2m}(2m)!!$.

Using that $(2x)!! = \prod_{i=1}^{x}(2i) = 2^x x!$ and $\binom{x}{y} = \frac{x!}{(x-y)!y!}$, (15) can be upper bounded as follows

$$\begin{aligned} O(1)n^5 \frac{g!(n-g)!(n-2m)!(2m)!}{n!(2k_1)!(g-2k_1)!(2k_2)!(n-g-2k_2)!} \frac{2^{k_1}k_1!2^{k_2}k_2!}{2^m m!} = \\ = O(1)n^5 \frac{g!(n-g)!(n-2m)!(2m)!}{n!(2k_1)!(g-2k_1)!(2k_2)!(n-g-2k_2)!} \frac{k_1!k_2!}{m!}. \end{aligned} \tag{16}$$

As $n - g > g$, it can be easily shown that (16) decreases by $k_1$ provided $m = k_1 + k_2$ is fixed. For $k_1 = 0$, (16) is reduced to

$$O(1)n^5 \frac{g!(n-g)!(n-2m)!(2m)!}{n!g!(2m)!(n-g-2m)!} = O(n^5)\frac{(n-g)!(n-2m)!}{n!(n-g-2m)!}, \tag{17}$$

Using Stirling's formula (st) for (17), we get

$$O(n^5)\frac{(n-g)!(n-2m)!}{n!(n-g-2m)!} = O(n^5)\frac{(n-g)^{n-g}(n-2m)^{n-2m}}{n^n(n-g-2m)^{n-g-2m}}\sqrt{\frac{(n-g)(n-2m)}{n(n-g-2m)}}. \tag{18}$$

Notice that for $x^2 = o(z)$, we have that

$$(1 - \frac{x}{z})^{z-x} = e^{-x}\frac{1}{(1 - \frac{x^2}{z})} = e^{-x}(1 + o(1)) \tag{19}$$

Hence using that $m^2 = o(n)$ and $n - g \geq 0.5n$, we can simplify (18) as

$$O(n^5)\frac{(n-g)^{2m}(1-\frac{2m}{n})^{n-2m}}{n^{2m}(1-\frac{2m}{n-g})^{n-g-2m}} = O(n^5)(1-\frac{g}{n})^{2m} \leq O(n^5)(1-\frac{n^{\alpha_c}}{n})^{2m}. \tag{20}$$

Finally, using Taylor's expansion for natural logarithm, we upper bound (20) as

$$O(n^5)e^{2m\ln(1-n^{\alpha_c-1})} = O(n^5)e^{-2n^{\beta_s}n^{\alpha_c-1}(1+o(1))}. \tag{21}$$

It remains to recall that $\beta_s + \alpha_c - 1 > 0$, and thus (21) is $o(\frac{1}{n^r})$ for any $r > 0$. □

When a letter is clear from the context, we call a pair of states *cycle pair* if both states belong to (probably different) cycles in the underlying digraph of the letter.

**Lemma 11.** *Suppose there exists a set of at least $n^{\beta_s}$ distinct pairs $Z_a$ independent of $a$, which are stable with probability $1 - O(\frac{1}{n})$ and $\Gamma(S_a, Z_a)$ is connected; then with probability $1 - O(\frac{1}{n})$ cycles from $S_a$ are all stable, and thus $\widehat{S_a}$ is synchronizable.*

*Proof.* Let $n_s$ denote the number of clusters in $S_a$ and $z$ denote the number of pairs in $Z_a$. Suppose the cycles of $S_a$ have states from exactly $d$ stable classes. We have to upper bound the probability that $d > 1$. Using that the stability relation is an equivalence and that the graph of cycles is connected by stable pairs, we have the following properties.

(I) All cycles have states from each of $d$ stability classes. Indeed, suppose a stable pair $\{p, q\}$ has $p$ in one cycle and $q$ in another. When $p.a^i$ for $i > 0$ runs through all the classes of one cycle, $q.a^i$ must run through the same classes in the other cycle. The statement follows from connectivity now.

(II) The length of all cycles form $S_a$ is divided by $d$ (follows from (I)).

(III) We can enumerate these stable classes from 0 to $d-1$ such that the class indexed $i$ is mapped to $(i + 1 \mod d)$ by $a$.

Let us colour each cycle state from $S_a$ in the colour corresponding to its stable class. We extend this colouring to the whole $\widehat{S_a}$ as follows. We colour a state $p$ in the same colour as the (cycle) state $p.a^{nd}$. With this definition, we have that (III) holds for all states from $\widehat{S_a}$.

**Remark 12.** The action of $a$ maps a pair $\{p, q\}$ in a monochrome pair if and only if $\{p, q\}$ is monochrome.

*Proof.* Indeed, let $\{p, q\}$ be a cycle pair. If $\{p.a, q.a\}$ is monochrome, then it is stable, whence $\{p.a^t, q.a^t\}$ is stable and monochrome for all $t > 0$. If we take $t$ being the product of all cycles lengths, we get a pair $\{p, q\}$, which thus must be monochrome. For non-cycle pairs, the statement follows from the fact that $a^{nd}$ is homomorphic map into the set of cycle states. □ □

It follows from the definition of the colouring and Remark 12 that each stable pair in $\widehat{S_a}$ must be monochrome. Denote by $s_i$ the number of pairs $Z_a$ coloured by $i$ for $i = 0, 1, \ldots, d-1$. Without loss of generality, suppose $s_0$ is maximal among $s_i$.

Since $Z_a$ is independent of $a$, we can choose $Z_a$ uniformly at random for a given random mapping $a$ and estimate the probability that $Z_a$ yields a colouring into $d > 1$ colours. The choice of $Z_a$ can be done in two stages. We first randomly choose $2z$ states and then randomly choose a perfect matching on the chosen states.

**Case 1.** $s_0 \leq \gamma z$ for $0.9 < \gamma < 1$. Consider the probability that for a chosen set of states in $Z_a$, there is a colouring (satisfying above properties) and a perfect matching on its states such that all pairs of the matching are monochrome. The probability is upper bounded by

$$nd^{n_s} \frac{\prod_{i=0}^{d-1} (2s_i)!!}{(2z)!!} = nd^{n_s} \frac{\prod_{i=0}^{d-1} (s_i)!}{z!} \tag{22}$$

Indeed, first we choose $d$ in at most $n$ ways, then we determine a colouring for each of $n_s$ clusters by choosing a colour of one of its cycle states in $d$ ways (other colours are uniquely determined according to (iii)), then for each colour $i$ we choose a perfect matching in $(2s_i)!!$ ways. Finally, we divide it by the total number $(2z)!!$ of all perfect matchings on all $2z$ states from $Z_a$. Notice that we do not have to choose the values of $s_i$ in this case, because they are defined by the colouring and the choice of the set of states in $Z_a$.

Also the number of perfect matchings is constant for every choice of a set of states, so we don't need to multiply (and divide) by the number of possible choices of states from $Z_a$.

As $(k_1 + k_2)! \geq k_1!k_2!$ One can easily observe that the maximum of the right hand side of (22) (for $d > 0$) is reached with the smallest number of non-zero values among $s_i$. As $s_0$ is the largest among $s_i$, (22) is upper bounded by

$$\begin{aligned} nd^{n_s} \frac{s_0!(z-s_0)!}{z!} &= nd^{n_s} / \binom{z}{s_0} = [\text{due to (cmb0)}] \\ = nd^{n_s} \frac{(\gamma z)^{\gamma z}((1-\gamma)z)^{(1-\gamma)z}}{z^{\gamma z}} O(\sqrt{z}) &= [n^{\beta_s} \leq z \leq n, d \leq n, 0 < \gamma < 1] \\ &= O(n^{5 \ln n + 2}) \gamma^{\gamma n^{\beta_s}} = O\left(\frac{1}{n}\right). \end{aligned} \tag{23}$$

**Case 2.** $s_0 \geq \gamma z$. Let $\omega_0$ be the total number of 0-coloured states in $\widehat{S_a}$. First, consider the case $\omega_0 \leq 0.9\omega$, where $\omega$ is the number of states in $\widehat{S_a}$. The probability of such colouring is at most

$$znd^{n_s} \binom{\omega_0}{2s_0} \binom{n - \omega_0}{2(z - s_0)} / \binom{n}{2z}. \tag{24}$$

Indeed, first we choose $s_0$ in at most $z$ ways, then we choose $d$ in at most $n$ ways and determine a colouring in $d^{n_s}$ ways (as in **Case 1**). With this choice being made, as $Z_a$ is independent of $a$, it is also independent of the set of 0-coloured states within this colouring. Thus the probability that these two sets has $2s_0$ states in overlap is at most, as in Remark 7,

$$\binom{\omega_0}{2s_0} \binom{n - \omega_0}{2(z - s_0)} / \binom{n}{2z}.$$

If $z = s_0$, (24) can be upper bounded as

$$n^{1+\beta_s} e^{5\ln^2 n} \binom{\omega_0}{2s_0} / \binom{n}{2s_0} \leq n^{(5\ln n+2)} \frac{\omega_0!}{(\omega_0 - 2s_0)!} \frac{(n-2s_0)!}{n!} \leq$$
$$\leq n^{(5\ln n+2)} \left(\frac{\omega_0}{n-2s_0}\right)^{2s_0} \leq n^{(5\ln n+2)} (0.1 + o(1))^{2\gamma n^{\beta_s}} = o(1/n). \quad (25)$$

For $z > s_0$, since $z^2 = o(n)$ and $n - \omega_0 \geq 0.1n$, by (cmb2), (24) can be upper bounded as

$$n^{1+\beta_s} d^{5\ln n} \binom{\omega_0}{2s_0} \left(\frac{e(n-\omega_0)}{2(z-s_0)}\right)^{2(z-s_0)} \sqrt{z} / \left(\frac{en}{2z}\right)^{2z} \leq$$
$$\leq n^{(5\ln n+3)} \binom{\omega_0}{2s_0} / \left(\frac{en}{2z}\right)^{2s_0} = n^{(5\ln n+3)} \left(\frac{z\omega_0}{es_0 n}\right)^{2s_0} \frac{\sqrt{\omega_0}}{(1 - \frac{2s_0}{\omega_0})^{\omega_0 - 2s_0}}. \quad (26)$$

Now, if $\omega_0 \geq \sqrt{n}$ then $s_0 = o(\omega_0)$ and thus $(1 - \frac{2s_0}{\omega_0})^{\omega_0 - 2s_0} = e^{2s_0(1+o(1))}$. Hence (26) can by upper bounded by

$$n^{(5\ln n+4)} \left(\frac{z\omega_0(1+o(1))}{ns_0}\right)^{2s_0} \leq n^{(5\ln n+4)} \left(\frac{0.9(1+o(1))}{\gamma}\right)^{2\gamma n^{\beta_s}} = o\left(\frac{1}{n}\right). \quad (27)$$

If $\omega_0 < \sqrt{n}$, as in (cmb3), we have that $(1 - \frac{2s_0}{\omega_0})^{\omega_0 - 2s_0} \leq e^{4s_0}$, and (26) can by upper bounded as

$$n^{(5\ln n+4)} \left(\frac{ez\omega_0}{ns_0}\right)^{2s_0} \leq n^{(5\ln n+4)} \left(\frac{e}{\gamma\sqrt{n}}\right)^{2\gamma n^{\beta_s}} = o\left(\frac{1}{n}\right). \quad (28)$$

**Case 2.1.** $\omega_0 > 0.9\omega$. The probability of a corresponding colouring for this case is at most

$$\frac{\omega\binom{\omega}{\omega_0}(\omega - \omega_0)^{\omega_0}\omega^{\omega - \omega_0}}{\omega^\omega}. \quad (29)$$

Indeed, first we choose $\omega_0$ in less than $\omega$ ways, and then we choose a subset of 0-coloured states in $\binom{\omega}{\omega_0}$ ways. Then for each of 0-coloured state we choose a non 0-coloured image in $\omega - \omega_0$ ways (the colour of the image must be equal to $d - 1 \neq 0$), and for the remained $\omega - \omega_0$ states we choose an arbitrary image in $\omega$ ways, finally we divide it by $\omega^\omega$, the total number of ways to choose images for $\omega$ states in $\widehat{S_a}$. Using (cmb0) and monotonic descending of (29) by $\omega_0$ (for $\omega_0 > 0.5\omega$), (29) is upper bounded by

$$\frac{\omega\binom{\omega}{0.9\omega}(0.1\omega)^{0.9\omega}\omega^{0.1\omega}}{\omega^\omega} \leq \frac{O(\omega)\omega^\omega(0.1\omega)^{0.9\omega}\omega^{0.1\omega}}{(0.9\omega)^{0.9\omega}(0.1\omega)^{0.1\omega}\omega^\omega} \leq$$
$$\leq \frac{O(\omega)(0.1)^{0.9\omega}}{(0.9)^{0.9\omega}(0.1)^{0.1\omega}} = O(\omega)\left(\frac{10^{0.1}}{9^{0.9}}\right)^\omega \leq O(n)10^{(0.1-0.9\log_9 10)\Theta(n)} = o(1/n). \quad (30)$$

This completes the proof of the lemma. □

### *4.5. Searching for more Stable Pairs*

Due to results of Subsection 4.4 and Theorem 6, it remains to prove that there exists such $Z_a$ and $Z_b$ with probability $1 - O(\frac{1}{n})$. We now prove that having just one stable pair which is independent of one of the letters is enough to get enough such pairs for each of the letters.

**Lemma 13.** *If $\mathcal{A}$ has a stable pair $\{p, q\}$ independent of $b$; then for any constant $k > 0$ with probability $1 - O(\frac{1}{n})$ there are $k$ distinct stable pairs independent of $a$.*

*Proof.* Consider the chain of states $p.b, q.b, \dots p.b^{k+1}, q.b^{k+1}$. Since $\{p, q\}$ is independent of $b$, the probability that all states in this chain are different is

$$\left(1 - \frac{2}{n}\right)\left(1 - \frac{3}{n}\right)\dots\left(1 - \frac{2k+3}{n}\right) \geq \left(1 - \frac{2k+3}{n}\right)^{2(k+1)} = 1 - O\left(\frac{1}{n}\right).$$

Since $\{p, q\}$ is independent of $b$, each pair $\{p.b^i, q.b^i\}$ for $1 \leq i \leq k+1$ in this chain is independent of $a$. □

**Lemma 14.** *If for some $0 < \epsilon < 1$ the automaton $\mathcal{A}$ has $k = \lceil \frac{1}{2\varepsilon} \rceil$ distinct stable pairs independent of $b$; then with probability $1 - O(\frac{1}{n})$ there are $f = \lceil n^{0.5-\epsilon} \rceil$ distinct stable pairs independent of $a$.*

*Proof.* We start with the first pair $\{p_1, q_1\}$ and build a chain

$$p_1.b, q_1.b, p_1.b^2, q_1.b^2 \dots$$

until a next state coincide with anyone already in the chain or in $k$ given pairs. If it happens, we take the second pair $\{p_2, q_2\}$ and continue in the same fashion. We stop as soon as we have $2f$ states in this chain or we ran out of $k$ given pairs. The probability that the latter happens, $f_i$ being the number of steps successfully made before taking $(i+1)$-th pair, can be upper bounded by, as $\frac{(2f_i+2k)}{n}$ is the upper bound of 'collision' with previous states for $i$-th pair,

$$\frac{(2f_1 + 2k)}{n}\frac{(2f_2 + 2k)}{n}\dots\frac{(2f_k + 2k)}{n} \leq \left(\frac{2(f+k)}{n}\right)^k = \frac{O(1)f^k}{n^k}. \tag{31}$$

As there are $\binom{f}{k} = O(f^k)$ ways to choose values of $f_i$, the final bound is

$$\frac{O(1)f^{2k}}{n^k} = O\left(\frac{1}{n^{2k\varepsilon}}\right) = O(1/n), \tag{32}$$

which proves the lemma as pairs in the built chain are independent of $a$ by construction. □

**Theorem 15.** *For any $\beta < 0.5$, if $\mathcal{A}$ has a stable pair $\{p, q\}$ independent of one of the letters (say $b$), then with probability $1 - O(\frac{1}{n})$ for each letter $x \in \{a, b\}$, there is a set of at least $n^{\beta}$ distinct stable pairs independent of $x$.*

*Proof.* If we apply Lemma 13 for a stable pair independent of $b$, we get a set $I$ of $k$ distinct stable pairs independent of $a$. Then, we apply Lemma 13 to the first pair of $I$ to get a set of $k$ distinct stable pairs independent of $b$. Thus, we can apply Lemma 14 to either of $a$ or $b$ and get $\lceil n^{\beta} \rceil$ stable pairs independent of the other letter. □

As Theorem 15 works for any $\beta < 0.5$, we can let $\beta_s = 0.45, \alpha_c = 0.95$ to satisfy all constraints.

### *4.6. Highest Trees and Stable Pairs*

To use Theorem 15 we need to find a stable pair completely defined by one of the letters whence independent of the other one. For this purpose, we reuse ideas from Trahtman's solution [14] of the famous Road Colouring Problem. A subset $A \subseteq Q$ is called an $F$-clique of $\mathcal{A}$ if it is a maximal by inclusion set of states such that each pair of states from $A$ is a deadlock. It is proved in Trahtman [14] that if $\mathcal{A}$ is strongly connected then all $F$-cliques have the same size. We also need to reformulate [14, Lemma 2] for our purposes.

**Lemma 16.** *If $A$ and $B$ are two distinct $F$-cliques in a strongly connected automaton such that $A \setminus B = \{p\}, B \setminus A = \{q\}$ for some states $p, q$; Then $\{p, q\}$ is a stable pair.*

*Proof.* Arguing by contradiction, suppose there is a word $u$ such that $\{p.u, q.u\}$ is a deadlock. Then $(A \cup B).u$ is an $F$-clique because all pairs are deadlocks. Since $p.u \neq q.u$, we have $|(A \cup B).u| = |A| + 1 > |A|$ contradicting the maximality of $A$. □

Given a digraph $g \in \Sigma_n$ and an integer $c > 0$, call a *$c$-branch* of $g$ any (maximal) subtree of a tree of $g$ with the root of height $c$ [6]. For instance, the trees are exactly 0-branches. Let $T$ be a highest $c$-branch of $g$ and $h$ be the height of the second highest $c$-branch. Let us call the *$c$-crown* of $g$ the (probably empty) forest consisting of all the states of height at least $h + 1$ in $T$. For example, the digraph $g$ presented on Figure 2 has two highest 1-branches rooted in states $6, 12$. Without the state 14, the digraph $g$ would have the unique highest 1-branch rooted at state 6, having the state 8 as its 1-crown.

The following theorem is an analogue of Theorem 2 from [14] for 1-branches instead of trees and a relaxed condition on the connectivity of $\mathcal{A}$.

**Theorem 17.** *Suppose the underlying digraph of the letter $a$ has a unique highest 1-branch $T$ and its 1-crown intersects with some (strongly-connected) subautomaton. Denote by $r$ the root of $T$ and by $q$ the predecessor of the root of the tree containing $T$ on the $a$-cycle [7]. Then $\{r, q\}$ is stable and independent [8] of $b$.*

[6] The height of a vertex in a tree is the number of edges from the vertex to the root.

[7] It follows that both $q.a, r.a$ coincides with the root of the tree containing $T$.

[8] Stability of the pair depends on $b$ because whether 1-crown is reachable depends on $b$, but the

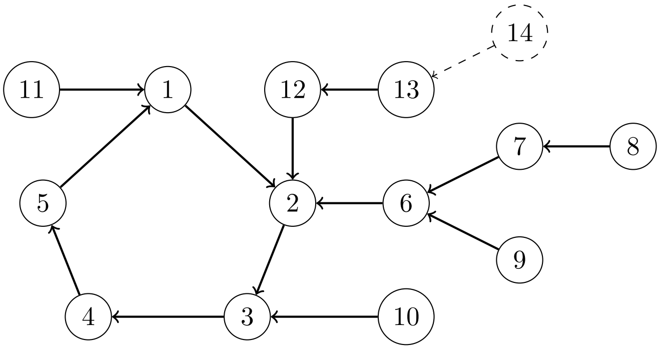

Figure 2: A digraph with one cycle and a unique highest tree.

*Proof.* Let $p$ be some state in the intersection of 1-crown of $T$ and a subautomaton $\mathcal{B}$. Then $q$ and all states from the corresponding cycle by $a$ belong to $\mathcal{B}$ because they are reachable from $p \in \mathcal{B}$. If $\mathcal{B}$ is synchronizing, we are done. Otherwise, there is an $F$-clique $F_0$ of size at least 2, and since $\mathcal{B}$ is strongly connected, there is another $F$-clique $F_1$ containing $p$. Since $p \in F_1$, $F_1 \cap T$ is not empty. Let $g$ be a state with maximal height $h$ from $F_1 \cap T$. As $F_1$ is an $F$-clique, all other states from $F_1 \cap T$ must have smaller height, as otherwise $a^h$ would merge some pair in $F_1$.

Let us consider the $F$-cliques $F_2 = F_1.a^{h-1}$ and $F_3 = F_2.a^L$ where $L$ is the least common multiplier of all cycle lengths in $\Gamma_a$. By the choice of $L$ and $F_2$, we have

$$F_2 \setminus F_3 = \{g.a^{h-1}\} = \{r\} \text{ and } F_3 \setminus F_2 = \{q\}.$$

Hence, by Lemma 16 the pair $\{r, q\}$ is stable. Since this pair is completely defined by the unique highest 1-branch of $a$ and the letters are chosen independently, this pair is independent of $b$. □

### *4.7. Finding Unique Highest Reachable 1-branch*

Due to Theorems 6,15 and Lemmas 10,11, it remains to show that we can use Theorem 17, that is, with probability $1 - O(\frac{1}{n})$ the underlying digraph of one of the letters has a unique highest 1-branch and the 1-crown of this 1-branch is reachable from $F$-cliques (if $F$-cliques exist). The crucial idea in the solution of the Road Coloring Problem [14] was to show that each *admissible* digraph can be *coloured* into an automaton satisfying the above property (for trees) and then use Theorem 17 to reduce the problem. In order to apply Theorem 17, we need the probabilistic version of the combinatorial result from [14].

**Theorem 18.** *Let $g \in \Sigma_n$ be a random digraph, $c > 0$, and $H$ be the $c$-crown of $g$ having $r$ roots. Then $|H| > 2r > 0$ with probability $1 - \Theta(1/\sqrt{n})$, in particular,*

pair is defined solely by $a$ and thus independent of $b$. In what follows, for simplicity we write ‘stable pair independent of’.

*a highest c-branch is unique and higher than all other c-branches of $g$ by 2 with probability $1 - \Theta(1/\sqrt{n})$.*

The proof of the above theorem has been attempted in a draft [3] (Theorem 12).

Since the letters of $\mathcal{A}$ are chosen independently, the following corollary of Theorem 18 is straightforward.

**Corollary 19.** *With probability $1 - O(\frac{1}{n})$ the underlying digraph of one of the letters (say $a$) satisfies Theorem 18.*

Due to Lemma 4, with probability $1 - O(\frac{1}{n})$ there is just one (strongly-connected) subautomaton $\mathcal{B}$ in $\mathcal{A}$. In order to use Theorem 17 and thus complete the proof of Theorem 1, it remains to show that the 1-crown of the underlying digraph of $a$ intersects with $\mathcal{B}$. Thus the following statement completes the proof of Theorem 1.

**Theorem 20.** *The 1-crown of the underlying digraph of $a$ intersects with each subautomaton with probability $1 - O(\frac{1}{n})$.*

*Proof.*

Let $g \in \Sigma_n$ and $H$ be the 1-crown of $g$. Let $d = |H|$ and $j$ be the number of roots in $H$. Due to Corollary 19, one of the letters (say $a$) satisfies Theorem 18 with probability $1 - O(\frac{1}{n})$, that is, $d > 2j$ for $g = \Gamma_a$ with probability $1 - O(\frac{1}{n})$. By Lemma 4 and Lemma 3, there is only one subautomaton $\mathcal{C}$ and its size is at least $n/4$. Therefore, there are at least $\Theta(n^{2n})$ of automata satisfying both constraints.

Arguing by contradiction, suppose that among such automata there are more than $\Theta(n^{2n-1})$ automata $\mathcal{A}$ such that their 1-crown does not intersect with $\mathcal{C}$. Denote this set of automata by $L_n$. For $1 \le j < d$, denote by $L_{n,d,j}$ the subset of automata from $L_n$ with the 1-crown having exactly $d$ vertices and $j$ roots. By definition $j < 0.5d$, and the conditions on the size of the subautomaton implies that $d < 3n/4$. Thus we have

$$\sum_{d=2}^{\lfloor 3n/4 \rfloor} \sum_{j=1}^{\lfloor 0.5d \rfloor} |L_{n,d,j}| = |L_n|. \tag{33}$$

Given an integer $n/4 \le m < 3n/4$, let us consider the set of all $m$-states automata whose letter $a$ has the unique highest 1-branch which is higher by 1 than a second highest 1-branch (or equivalently the 1-crown has only root vertices). Due to Theorem 18, there are at most $O(m^{2m-0.5})$ of such automata. Denote this set of automata by $K_m$. By $K_{m,j}$ denote the subset of automata from $K_m$ with exactly $j$ vertices in the 1-crown. Again, we have

$$\sum_{j=1}^{m-1} |K_{m,j}| = |K_m|. \tag{34}$$

Each automaton $\mathcal{A}$ from $L_{n,d,j}$ can be obtained from $K_{m,j}$ for $m = n - (d - j)$ by *growing its crown* as follows. Let us take an automaton $\mathcal{B} = (Q_b, \Sigma)$ from $K_{m,j}$

with no subautomaton of size less than $n/4$. First we extend $Q_b$ with missing $d-j$ states by selecting their insertion positions in $Q_b$. There are $\binom{n}{d-j}$ ways to do this. Let us denote this set $H_b$. The indices of the states from $Q_b$ are shifted according to the positions of the inserted states, that is, the index $q$ is shifted to the number of chosen indices $z \leq q$ for $H_b$. Next, we choose an arbitrary forest with $H_b$ as the set of internal vertices and $j$ roots which belong to the 1-crown of $\mathcal{B}$ in at most $jd^{d-j-1}$ ways (follows from Cayley's formula). Notice that we have completely chosen the action of the letter $a$.

Next we choose some subautomaton $M$ of $\mathcal{B}$ and redefine arbitrarily the image by the letter $b$ for all states from $Q_b \setminus M$ to the set $Q_b \cup H_b$ in $n^{m-|M|}$ ways. Within this definition, all automata from $K_{m,j}$ which differ only by $b$-transitions from $Q_b \setminus M$ may lead to the same automaton from $L_{n,d,j}$. Given a subautomaton $M$, denote such class of automata by $K_{m,j,M}$. There are exactly $m^{m-|M|}$ automata from $K_{m,j}$ in each such class. Since $|M| \geq n/4$ and subautomata cannot overlap, $\mathcal{B}$ can appear in at most 4 such classes.

Thus we have completely chosen both letters and obtained each automaton $\mathcal{A}$ in $L_{n,d,j}$. Therefore, for the automaton $\mathcal{B}$ and one of its subautomaton $M$ of size $z \geq n/4$, we get at most

$$\binom{n}{d-j} jd^{d-j-1} n^{m-z}$$

automata from $L'_{n,d,j}$ each at least $m^{m-z}$ times, where $L'_{n,d,j}$ is the set of automata containing $L_{n,d,j}$ without the constraint on the number of states of a subautomaton. Notice that we get each automaton from $L_{n,d,j}$ while $\mathcal{B}$ runs over all automata from $K_{n-(d-j),j}$ with no subautomaton of size less than $n/4$. Thus we get that

$$|L_{n,d,j}| \leq \sum_{z=\lceil n/4 \rceil}^{n} \sum_{M,|M|=z} \sum_{\mathcal{B} \in K_{m,j,M}} \frac{\binom{n}{d-j} jd^{d-j-1} n^{m-z}}{m^{m-z}}. \tag{35}$$

Since each automaton $\mathcal{B} \in K_{m,j}$ with no subautomaton of size less than $n/4$ appears in at most 4 of $K_{m,j,M}$, we get

$$|L_{n,d,j}| \leq 4|K_{m,j}| \max_{n/4 \leq z \leq m} \frac{\binom{n}{d-j} jd^{d-j-1} n^{m-z}}{m^{m-z}} = 4|K_{m,j}| \frac{\binom{n}{d-j} jd^{d-j-1} n^{m-n/4}}{m^{m-n/4}}. \tag{36}$$

Using (33) and (34), we get

$$|L_n| \leq 4 \sum_{d=2}^{\lfloor 3n/4 \rfloor} \sum_{j=1}^{\lfloor 0.5d \rfloor} |K_{m,j}| \frac{\binom{n}{d-j} jd^{d-j-1} n^{m-n/4}}{m^{m-n/4}} \leq$$

$$\leq 4 \sum_{d=2}^{\lfloor 3n/4 \rfloor} \max_{j \leq 0.5d} |K_m| \frac{\binom{n}{d-j} jd^{d-j} n^{m-n/4}}{m^{m-n/4}}. \tag{37}$$

Let $x = d - j$. Then $1 \le x \le 0.5d \le 3/8n$. Using (cmb0), we get that,

$$\binom{n}{d-j} = \Theta(1)\frac{n^n}{x^x(n-x)^{n-x}}\sqrt{\frac{n}{x(n-x)}} \le O(1)\frac{n^n}{x^{x+0.5}(n-x)^{n-x}} \tag{38}$$

Using that $|K_m| = O(m^{2m-0.5})$, for each term of (37) we have

$$\begin{aligned} & m^{2m-0.5} j d^{d-j}\binom{n}{d-j}\left(\frac{n}{m}\right)^{m-n/4} \le \\ & \le (n-x)^{2(n-x)-0.5} j d^x \frac{n^n}{x^{x+0.5}(n-x)^{n-x}}\left(\frac{n}{n-x}\right)^{3/4n-x} \le \\ & \le (n-x)^{n/4-0.5} j d^x \frac{n^{7/4n-x}}{x^{x+0.5}} \le (1-\frac{x}{n})^{n/4} j d^x \frac{n^{2n-x-0.5}}{x^{x+0.5}} \le \exp f(x,j), \end{aligned} \tag{39}$$

where

$$f(x,j) = \frac{n}{4}\ln(1-\frac{x}{n}) + \ln j + x\ln d + (2n-x-0.5)\ln n - (x+0.5)\ln x. \tag{40}$$

The derivative of $f'_x(x, d-x)$ is

$$-\frac{x}{4(1-\frac{x}{n})} + \frac{1}{d-x} + \ln d - \ln n - (1 + \ln x). \tag{41}$$

As $0.5d \le x \le d \le 3n/4$, for $n$ big enough $f'_x(x, d-x) < 0$. It follows that

$$\max_{j \le 0.5d} \exp f(x,j) = \max_{0.5d \le x \le d} \exp f(x, d-x) \le \exp f(0.5d, 0.5d). \tag{42}$$

Thus we have that (37) is upper bounded by

$$\begin{aligned} O(1)\sum_{d=2}^{\lfloor 3n/4 \rfloor} \exp f(0.5d, 0.5d) \le O(1)\sum_{d=2}^{\lfloor 3n/4 \rfloor}(1-\frac{d}{2n})^{n/4} d^{d/2+1}\frac{n^{2n-(d+1)/2}}{d^{(d+1)/2}} \le \\ \le O(1)\sum_{d=2}^{\lfloor 3n/4 \rfloor}(1-\frac{d}{2n})^{n/4}\sqrt{d}n^{2n-(d+1)/2}. \end{aligned} \tag{43}$$

It can be easily shown that the sum (43) is dominated be the first term (for $d = 2$) which is $O(1)n^{2n-1.5}$. This contradicts $|L_n| \ge \Theta(n^{2n-1})$, and the theorem follows. □

## 5. Testing for Synchronization in Linear Expected Time

In this section we show that following the proof of Theorem 1 we can decide, whether or not a given $n$-state automaton $\mathcal{A}$ is synchronizing in linear expected time in $n$. Notice that the best known deterministic algorithm (basically due to Černý [7]) for this problem is quadratic on the average and in the worst case.

**Theorem 21.** *There is a deterministic algorithm for deciding whether or not a given $k$-letter automaton is synchronizing having linear in $n$ expected time. Moreover, for $k > 1$ the proposed algorithm is optimal in expected time up to a constant factor.*

*Proof.* First, let's establish the lower bound for the complexity. To do this, we need to make precise the computational model. We consider that algorithms can query their input using questions of the form 'What is the image of the state $q$ by the letter $\alpha$¿. Let $query(q, \alpha)$ denote such a query. Our lower bounds results are stated as a lower bound on the number of queries required to complete the computations. That is, we do not take into account computation steps other than querying the input, which is not a problem as we aim at proving lower bounds and since in any classical way to represent a deterministic automaton, our queries are indeed the way to access the input data.

**Lemma 22.** *Any deterministic algorithm that decides whether a given $n$-state $k$-letter automaton $\mathcal{A}$ is synchronizing (for $k > 1$) performs on average (at least) linear in $n$ number of queries.*

*Proof.* First notice that it is necessary to verify that $\mathcal{A}$ is weakly connected to claim that $\mathcal{A}$ is synchronizing. To ensure that, it is required to verify (directly or indirectly) that there is no disconnected state (having connections only from itself, see Lemma 2). To do that, it is necessary to check that a state has either an incoming transition from another state or outgoing transition to another state. If an automaton is synchronizing, an algorithm would have to make at least $\lceil n/2 \rceil$ such queries, as it cannot claim it is synchronizing before ensuring that it is weakly connected.

Due to Theorem 1, it happens with probability $1 - O(\frac{1}{n})$. Thus the average time complexity must be at least $n/2(1 - O(1/n)) = \Theta(n)$ which completes the proof. □ □

It follows from Lemma 22 that linear expected time on average cannot be improved for any algorithm that tests for a synchronization.

We now turn to describing the algorithm. The idea of is to subsequently check that all conditions used in Theorem 1 holds for $\mathcal{A}$; if so, we return 'Yes'; otherwise, we run aforementioned quadratic-time algorithm for $\mathcal{A}$. Since the probability that any of these conditions is not met is $O(\frac{1}{n})$, the overall expected time is linear in $n$ if all conditions can be verified in linear time.

Thus it remains to prove that all conditions required in Theorem 1 for an automaton to be synchronizable can be checked in linear time. We first describe the algorithm for $k = 2$ and then explain how to generalize it for $k > 1$.

First we call Tarjan's linear algorithm [12] on the underlying digraph of $\mathcal{A}$ to find *minimal strongly connected components* [9] (MSCC) and, if there are several MSCC, we return 'No' because $\mathcal{A}$ is not weakly connected (whence not synchronizing) in this case. Otherwise, there is a unique MSCC $\mathcal{B}$ and $\mathcal{A}$ is synchronizing whenever $\mathcal{B}$ is. Thus all further calculations can be performed with the automaton $\mathcal{B}$. Due to Lemma 3, we may also assume that $\mathcal{B}$ has at least $n/4$ states.

[9] strongly-connected subgraphs with no outgoing edges

Let us now fix one of the letters, say $a$, and consider its underlying graph. In this graph each state $q \in Q$ is located in some cluster $cluster(q)$. Let $cl(q)$ denotes the cycle length in this cluster. Assuming cycle vertices are indexed in the clockwise direction from 0 to $cl(q) - 1$, let $tree(q)$ denote the tree in which $q$ is located and $root(q)$ denote the index of the root vertex defined above. Finally, let $height(q)$ denote the height of a vertex $q$ in its tree (which is 0 for cycle vertices). Both $cluster(q)$ and $tree(q)$ are needed only to compare whether them for two states, so we can label them with index of a particular state that belong to them.

We want to calculate the *cluster structure*, that is, for each $q \in Q$ we want to compute $root(q), tree(q)$, $cluster(q), cl(q)$ and $height(q)$.

As a secondary information, we compute the number of clusters and their sizes as well as the unique highest tree if it exists.

**Lemma 23.** *The cluster structure of a letter $x \in \Sigma$ can be calculated in linear in $n$ time.*

*Proof.* In each step we choose an unobserved state $p \in Q$ [10], set $cluster(p) = p$ and walk by the path

$$p = p_0, p_1 = p_0.x, \dots, p_m = p_{m-1}.x$$

in the underlying digraph of $x$ until we encounter a state $p_m$ such that $p_m = p_k = p_k.x^{m-k}$ for some $k < m$. It follows that the length of the cycle is $cl(p) = m - k$. Then we set $root(p_i) = i - k$ for $k \leq i \leq m$. After that, for each cycle state $q$ we run *Breadth First Search* (BFS) in the tree $tree(q)$ by reversed arrows, and at $j$-th step we set for a current state $s$:

$$height(s) = j, tree(s) = q, cluster(s) = p, root(s) = root(q), cl(s) = cl(p).$$

We process a full cluster by this subroutine. Since we observe each state only in one subroutine and at most twice, the algorithm is linear. Clearly we can simultaneously evaluate the number of clusters and check whether there is a unique highest tree. □ □

We may assume that the number of clusters does not exceed $5 \ln n$ due to Lemma 5. If the unique highest tree has been found for one of the letters, we can compute in linear time the highest 1-branch in this tree, for instance, applying the same algorithm on this tree instead of the whole graph. Hence using the cluster structure, one can check in linear time that one of the letters (say $a$) in $\mathcal{A}$ satisfies Theorem 18. This is not the case with probability $O(\frac{1}{n})$ due to Corollary 19. Due to Theorem 20, some states of the crown of $a$ belong to $\mathcal{B}$ with probability $1 - O(\frac{1}{n})$.

For the letter $a$ and its highest 1-branch $T$, we find a pair $\{r, q\}$ where $r$ is the root of $T$ and $q$ is the predecessor of the root of the tree containing $T$ on the $a$-cycle. The pair $\{r, q\}$ is stable by Theorem 17 and independent of $b$.

[10] This can be done in amortized linear time by maintaining a bit mask of unobserved states and a queue of states. We pop states from the queue until we find one which is unobserved.

Next, following the proof of Theorem 15, we try to *extend* $\{r, q\}$ to sets $Z_a, Z_b$ of $n^{0.45}$ distinct stable pairs each, independent for $a$ and $b$ respectively. The maximum number of pairs that we need to observe during this procedure is bounded by $O(n^{0.45})$, whence this step can be done in linear time. Again, due to Theorem 15, we fail with probability $O(\frac{1}{n})$ at this stage.

Recall that, given $x \in \{a, b\}$, $S_x$ is the set of clusters of $\Gamma_x$ containing more than $n^{0.45}$ states and $T_x$ is the complement of $S_x$. Given a pair $\{p, q\}$, either $\{p, q\}$ in one $a$-cluster or the states $p$ and $q$ belong to different $a$-clusters. In the latter case, we say that $\{p, q\}$ *connects* these $a$-clusters. Consider the graph $\Gamma(S_a, Z_a)$ with the set of vertices $S_a$, and draw an edge between two clusters if and only if some pair from $Z_a$ connects them. Since $|S_a| \leq 5 \ln n$ and $|Z_a| \leq n^{0.45} + 1$, one can construct the graph $\Gamma(S_a, Z_a)$ and verify that it is connected in linear time by *Depth First Search* (DFS) yielding the spanning tree of $\Gamma(S_a, Z_a)$ simultaneously. Due to Lemma 10, we fail here with probability $O(\frac{1}{n})$.

Next, we calculate the greatest common divisor $d$ of the cycle lengths of the clusters in $S_a$. Using the Euclidean algorithm it can be done in $O(\ln^2 n)$ time (at most $|S_a| \leq 5 \ln n$ runs each having logarithmic time complexity in terms of the maximal cycle size).

Due to Lemma 11 if $d > 1$, we additionally have to verify that no colouring of $S_a$ is possible in $d$ colours which would preserve monochrome pairs under the action of $a$. Suppose there is such a colouring. Let $\sigma$ be a partition on $Q$ defined by the letter $a$ as follows. States $p, q$ are in the same $\sigma$-class if and only if $p.a^n = q.a^n$. Thus for each cluster $C_i$ with the cycle length $s_i$, all states of $C_i$ are partitioned into $s_i$ classes of equivalence $C_{i,j}$ for $j \in \{0, 1, \dots, s_i - 1\}$ such that $C_{i,j}.a \subseteq C_{i,(j+1 \mod s_i)}$. We can assume that $C_{i,0}$ contains a state $p$ with $root(p) = 0$ which implies that $q \in C_{i,root(q)}$. Notice that each such class must be monochrome and $d \mid s_i$ for all $i$ (see (ii) of Lemma 11), and each cycle of $S_a$ has states of each colour (see (i) of Lemma 11). Hence there must be $0 \leq x_i \leq d - 1$ such that $C_{i,x_i}$ is 0-coloured. Then, for a given state $q$ its colour can be computed as

$$d - x_i + root(q) - (height(q) \mod d) \mod d.$$

Let $\{p, q\}$ be a stable pair such that $p \in C_{i,root(p)}, q \in C_{j,root(q)}$. Then $p.a^{nd-root(p)+x_i} \in C_{i,(x_i+nd \mod s_i)}$ which is 0-coloured whence

$$q.a^{nd-root(p)+x_i} \in C_{j,(root(q)+nd+x_i-root(p) \mod s_j)}$$

must be 0-coloured too. Hence $root(q)+nd+x_i-root(p) = x_j$ modulo $d$ or equivalently

$$d \mid (root(p) - root(q)) - (x_{cluster(p)} - x_{cluster(q)}). \tag{44}$$

Thus, it is enough to check whether (44) holds for some $x_i \mid i \in \{1, 2, \dots, |S_a|\}$ (such that $0 \leq x_i \leq d - 1$) and for all pairs $\{p, q\} \in S$.

Let us show how this property can be checked in linear time. Consider the spanning tree $T$ of $\Gamma(S_a, Z_a)$ (which we have computed previously) and recall that each edge of $\Gamma(S_a, Z_a)$ corresponds to a pair from $Z_a$. We start from the root $r$ of $T$ and set

$x_{cluster(r)} = 0$. Next, we traverse the edges of the tree $T$ using DFS. For each next edge and a corresponding pair $\{p, q\} \in Z_a$, we have that either $x_{cluster(p)}$ or $x_{cluster(q)}$ is already defined. This allows to determine the other index between 0 and $d-1$ in the unique way to satisfy (44). While traversing the tree, we define all $x_i$. After all $x_i$ are defined, we can check (44) for all the pairs from $Z_a$. Clearly, the success of the procedure does not depend on the choice of $x_{cluster(r)}$. Since there are at most $n^{0.45}+1$ of pairs in $Z_a$, this routine can be done in linear time. Due to Lemma 11, we fail with probability $O(\frac{1}{n})$.

Thus, due to Lemma 11, we may assume that all clusters of $\Gamma_a$ of size at least $n^{0.45}$ are contained in one *synchronizing class* $\widehat{S_a}$, i.e. each pair from $\widehat{S_a}$ can be synchronized. Moreover, since $S_a$ is defined by the letter $a$, this class is independent of $b$. We can do the same for the letter $b$ and obtain the corresponding set $S_b$ with the same properties.

It remains to prove that we can check the sufficient conditions for automaton being synchronizable following the proof of Theorem 6 in linear time. Notice that we can decide whether a state belongs to $\widehat{T_a}$ or $\widehat{T_b}$ in constant time.

Let $c_p, c_q$ be two (possibly equal) $a$-cycles, $s_p, s_q$ being their respective lengths, and $d$ be the g.c.d. of $s_p$ and $s_q$ as in Theorem 6. Using the Euclidean algorithm $d$ can be computed in $O(\ln^2 n)$ time. Notice that if both $s_p$ and $s_q$ are greater than $n^{\alpha_c}$, then both clusters belong to $S_a$ and thus cannot contain a deadlock pair. Hence without loss of generality, we can assume that $s_p \leq n^{\alpha_c}$ and $s_p \leq s_q$.

Now, for a cycle $c_p$ we compute the set of indices $I_p \subseteq Z_d = \{0, 1, \ldots, d-1\}$ such that for each $i \in I_p$ we have $c_{p,(i+k_1 d) \bmod s_p} \in \widehat{T_b}$ for all $k_1$. Then we do the same for $I_q$. This clearly can be done in $O(s_p + s_q)$ time. Then, if $|I_p| + |I_q| < d$, we know that there cannot be a deadlock pair as we would be able to map such a pair to one that belongs to $\widehat{S_b}$ (see Theorem 6 for the details). Notice also that in total we consider at most $25 \ln^2 n$ of pairs of clusters and the complexity for each pair is $O(d)$ which is $O(n^{\alpha_c}) = O(n^{0.95})$ whence the overall complexity is linear.

If $|I_p| + |I_q| \geq d$, let us denote $k = |I_p|$ and $z = \frac{ks_p + (d-k)s_q}{d}$ as in Theorem 6. If $z \geq \frac{1.1}{1-\alpha_c}$ (**Case 1** of Theorem 6), we can just fallback to the general quadratic algorithm as this happens with probability $O(1/n)$. If $z < \frac{1.1}{1-\alpha_c}$ and $k < d$ (**Case 2** of Theorem 6), then we check all pairs $p \in c_p, q \in c_q$ following Lemma 8 which can be done in constant time. As a result we either get that no pair is a deadlock or that we can fallback to the general algorithm due to the proof of **Case 2** of Theorem 6.

Finally, in the case $k = d$ we take a state $q_{min}$ with the minimal index on the cycle $c_q$ and check whether any of the pairs $\{p', q_{min}\}$ for $p' \in c_p$ can be a deadlock following Lemma 8. If we find such a pair we again can fallback to the general algorithm due to the proof of **Case 3** of Theorem 6. Otherwise, we know there cannot be a deadlock cycle pair $\{p, q\}$ for $p \in c_p, q \in c_q$ and thus in the corresponding clusters. As in this case $s_p = z \leq \frac{1.1}{1-\alpha_c}$ we need to check only constant number of pairs, each in constant time.

If we did not fail up to this moment, we return 'Yes'. The correctness of the algorithm now follows from Theorem 6. Thus we have shown that we can confirm all the required properties in linear time and fail with $O(\frac{1}{n})$ probability. This concludes the proof for the 2-letter alphabet case.

Suppose we have an automaton $\mathcal{A} = \langle Q, \{a_1, a_2, \dots, a_k\}\rangle$ for $k > 2$. In this case, we run the aforementioned algorithm for the 2-letter alphabet case for the automaton $\mathcal{A}_1 = \langle Q, \{a_1, a_2\}\rangle$ but in the case of failure at some stage, we neither execute the quadratic algorithm nor return 'No'. Instead, we consider the automaton for the next two letters $\mathcal{A}_2 = \langle Q, \{a_3, a_4\}\rangle$ and continue this way while there are two other letters. If at some iteration, the considered automaton is synchronizing, we return 'Yes'. In the opposite case, in the end we just run the quadratic algorithm having complexity $O(n^2k)$ for the entire automaton $\mathcal{A}$. Since the letters are chosen independently, this happens with probability $O(\frac{1}{n^{\lceil k/2\rceil}})$. Since $k > 2$, the overall expected complexity $O(\frac{n^2k}{n^{\lceil k/2\rceil}})$ is linear again. □

Pavel Ageev, a former master student of Mikhail Volkov, has implemented a modified version of the above algorithm [1]. He relaxed some conditions in the properties we have to check, namely, the property that stable pairs (found according to Section 4.5) consist of pairwise distinct states. Clearly, this relaxation does not affect correctness of the algorithm. He then launched the modified algorithm on 1000 of random binary automata with $n$ states for $n \in \{1000, 2000, \dots, 10000\}$. The results are shown in the following table.

| meaning / $n/1000$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| average *#bad automata* per $n$ automata | 5.09 | 4.84 | 4.76 | 4.82 | 5.07 | 5.26 | 4.76 | 5.13 | 5.00 | 5.32 |
| elapsed ms. per good automaton | 1 | 2 | 3 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

These experiments indirectly confirms that random binary automaton is synchronizing with probability $1 - \frac{\alpha(n)}{n}(1 + o(1))$ where $\alpha(n) \leq 0.0055$.

## 6. Conclusions

Theorem 1 gives an exact order of the convergence rate for the probability of being synchronizable for 2-letter automata up to a constant factor. It is fairly easy to verify that the convergence rate for $t$-size alphabet case ($t > 1$) is $1 - O(\frac{1}{n^{0.5t}})$ because the main restriction comes from the probability of having a unique 1-branch for some letter. Thus perhaps the most natural open question here is about the tightness of the convergence rate $1 - O(\frac{1}{n^{0.5t}})$ for the $t$-letter alphabet case.

Since only weakly connected automata can be synchronizing, the second natural open question is about the convergence rate for random weakly connected automata of being synchronizable for the uniform distribution. Especially, binary alphabet is of certain interest because the upper bound for this case comes from a non-weakly connected case. We predict exponentially small probability of not being synchronizable for this case and $\Theta(\frac{1}{n^{k-1}})$ for random $k$-letter automata (for $k > 1$).

Another challenging problem concerns a generalization of synchronization property. Namely, given $d \geq 1$ what is the probability that for uniformly at random chosen strongly (or weakly) connected automaton – the minimum rank of words is equal to $d$. Notice that the case $d = 1$ corresponds to the original problem of synchronization, and for $d = n$ it is the probability that all letters are permutations, which is $(n!/n^n)^k \sim n^{k/2}e^{-nk}$. We believe that for $d > 1$ the probability is exponentially close to 1 and, if one could prove it, this would lead to the positive answer to the aforementioned hypothesis for the weakly connected case.

## 7. Acknowledgements

The author is grateful to Mikhail Volkov for permanent support in the research and also to Marek Szykuła, Cyril Nicaud, Dominique Perrin, Marie-Pierre Béal, Pavel Ageev and Julia Mikheeva for their interest and useful suggestions. The author is also thankful to anonymous referees of the conference version [4] of the present work for remarks and comments which helped to improve the presentation of the results.

## References

[1] P. Ageev, Implementation of the algorithm for testing an automaton for synchronization in linear expected time. *Journal of Automata, Languages and Combinatorics* **24** (2019) 2-4, 139–152.

[2] J. Araújo, W. Bentz, P. J. Cameron, Groups synchronizing a transformation of non-uniform kernel. *Theoretical Computer Science* **498** (2013), 1 – 9.

[3] M. Berlinkov, Highest Trees of Random Mappings. *ArXiv e-prints* (2015).

[4] M. Berlinkov, On the probability of being synchronizable. In: S. Govindarajan, A. Maheshwari (eds.), *Algorithms and Discrete Applied Mathematics*. Springer International Publishing, Cham, 2016, 73–84.

[5] P. J. Cameron, Dixon's Theorem and random synchronization. *ArXiv e-prints* (2011).

[6] A. Carayol, C. Nicaud, Distribution of the number of accessible states in a random deterministic automaton. In: T. W. Christoph Dürr (ed.), *STACS'12 (29th Symposium on Theoretical Aspects of Computer Science).* 14, LIPIcs, Paris, France, 2012, 194–205.

[7] J. Černý, Poznámka k homogénnym eksperimentom s konečnými automatami. *Matematicko-fyzikálny Časopis Slovenskej Akadémie Vied* **14** (1964) 3, 208–216. In Slovak.

[8] P. Flajolet, A. M. Odlyzko, Random mapping statistics. In: J.-J. Quisquater, J. Vandewalle (eds.), *Advances in Cryptology — EUROCRYPT '89*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1990, 329–354.

[9] P. Flajolet, R. Sedgewick, *Analytic Combinatorics.* 1 edition, Cambridge University Press, USA, 2009.

[10] J. Kari, Synchronization and stability of finite automata. *J. UCS* **8** (2002) 2, 270–277.

[11] C. Nicaud, Fast Synchronization of Random Automata. In: K. Jansen, C. Mathieu, J. D. P. Rolim, C. Umans (eds.), *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*. Leibniz International Proceedings in Informatics (LIPIcs) 60, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2016, 43:1–43:12.

[12] R. Tarjan, Depth first search and linear graph algorithms. *Siam Journal On Computing* **1** (1972) 2.

[13] A. Timashov, Asymptotic expansions for the distribution of the number of components in random mappings and partitions. *Discrete Mathematics and Applications* **21** (2011) 3, 291–301.

[14] A. Trahtman, The road coloring problem. *Israel Journal of Mathematics* **172** (2009) 1, 51–60.

[15] M. Volkov, Synchronizing automata and the černý conjecture. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, 11–27.

[16] Y. Zaks, E. Skvortsov, Synchronizing random automata. *Discrete Mathematics and Theoretical Computer Science* **Vol. 12 no. 4** (2010). http://dmtcs.episciences.org/514

[17] Y. Zaks, E. Skvortsov, Synchronizing random automata on a 4-letter alphabet. *Journal of Mathematical Sciences* **192** (2013) 3, 303–306.