Finite-state independence and normal sequences

Nicolás Alvarez Verónica Becher Olivier Carton

February 1, 2017

Abstract

We consider the previously defined notion of finite-state independence and we focus specifically on normal words. We characterize finite-state independence of normal words in three different ways, using three different kinds of finite automata running on infinite words (Büchi automata): finite automata with two input tapes, selectors and shufflers. We give an algorithm to construct a pair of finite-state independent normal words. Unfortunately the algorithm has doubly exponential computational complexity.

1 Introduction

In [3] we introduced the notion of *finite-state independence* for any pair of infinite words. In this work we characterize this notion of independence specifically for *normal* words. As defined by Émile Borel [7] when we consider an alphabet with at least two symbols, an infinite word x is normal if all blocks of symbols of the same length occur in x with the same limiting frequency. The most famous normal word was given by Champernowne in [10],

$$0123456789101112131415161718192021 2223...$$

Bugeaud's book [8] gives a thorough presentation of normality and includes a list of references of many known constructions of normal words. Borel showed that, indeed, almost all words are normal. And in [3, Theorem 5.1] we showed that almost all pairs of normal words are finite-state independent.

A main result of this work, stated in Theorem 3.3, gives three characterizations of finite-state independence of normal words based on different kinds of deterministic finite automata. The first characterization establishes that two normal words are finite-state independent when the frequency of the states in the run of any deterministic finite automata with two input tapes is determined just by the automata, not by the input words.

The second characterization considers *selectors*, which are finite automata with two input tapes and one output tape. The symbols in the output tape are obtained as a selection of the symbols in the first input tape, while the symbols in the second input tape act as a consultive oracle. The characterization establishes that two normal words are finite-state independent exactly when any selector having them as input yields also a normal word. This result on selection by finite automata extends Agafonov's [1] and falls out of the deterministic rules that preserve normality given by Kamae and Weiss [13].

The third characterization given in Theorem 3.3 considers *shufflers*, which are finite automata with two input tapes and one output tape such that, after the run, the output tape contains all the symbols from the two normal words but shuffled. A general presentation of shufflers that are finite automata can be read in [17]. Shuffling here is not in the sense of Diaconis Persis, because symbols are not permuted. Instead, the notion of shuffling we use assumes two input words and the result of shuffling them is a new word that intercalates symbols from each of them, preserving the order in which they appear in the given words. The characterization given in in Theorem 3.3 proves that two normal words are finite-state independent exactly when every shuffling of them is also normal.

The other main result in this work, stated as Theorem 5.1, gives an algorithm to construct a pair of finite-state independent normal words, based on the characterization in terms of shufflers proved in Theorem 3.3. This algorithm outputs a pair of finite-state independent normal words (x, y) by outputting, at each step, one new symbol extending either the currently computed prefix of x or the currently computed prefix of y. Unfortunately, the computational complexity of this algorithm is doubly exponential, which means that to obtain the n-th symbol of the pair of finite-state independent normal words the algorithm performs a number of operations that is doubly exponential in n. Our construction of a pair of independent normal words has some similarity with the construction of sequences representing the fractional expansion of absolutely normal numbers (a number is absolutely normal if its expansion in each base greater than or equal to 2 is a normal word). The algorithm we give here has some similarity with Turing's algorithm for computing absolutely normal numbers [21, 4], which also has doubly exponential computational complexity.

The paper is organized as follows. In Section 2 we present the primary definitions of finite automata, normality and finite-state independende. In Section 3 we introduce the new needed definitions and we state Theorem 3.3 (the Characterization Theorem). Section 4 gives its proof. In Section 5, the last section, Theorem 5.1 gives the announced algorithm to compute a pair of finite-state independent normal words.

2 Primary definitions

Notation. Let A be finite set of symbols, that we refer as the alphabet. We write A^ω for the set of all infinite words in alphabet A, A^* for the set of all finite words, $A^{\leq k}$ for the set of all words of length up to k, and A^k for the set of words of length exactly k. The length of a finite word w is denoted by $|w|$. The positions of finite and infinite words are numbered starting at 1. To denote the symbol at position i of a word w we write $w[i]$ and to denote the substring of w from position i to j we write $w[i..j]$. The empty word is denoted by λ.

2.1 Automata

We consider finite automata running on tuple of infinite words with no accepting condition. As it will explained below, the only requirement on an infinite run to be accepting is that all labels to be infinite. In particular we consider k-tape automata, also known as k-tape transducers, for $k = 2$ and $k = 3$. We call them *k-automata.* A thorough presentation of these automata is in the books [16, 18]. We use 2-automata to compute functions from infinite words to infinite words. And we use 3-automata to compute functions either from pairs of infinite words to infinite words, or from infinite words to pairs of infinite words.

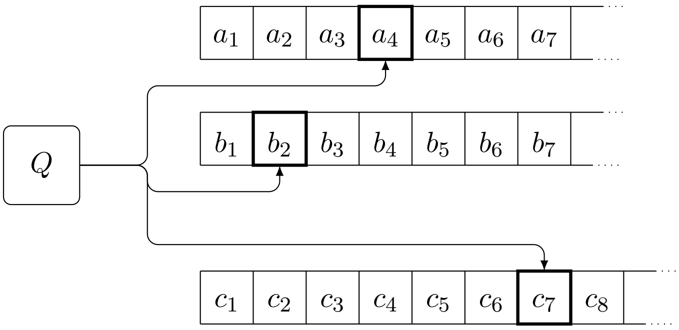

Figure 1: Working principle of a 3-automaton.

We give names such as *compressors*, *selectors shufflers* or *splitters* to some subclasses of these automata to emphasize their use. To simplify the presentation, we assume here that the input and output alphabets of all machines are the same alphabet A. A *k-automaton* is a tuple $\mathcal{A} = \langle Q, A, \delta, I \rangle$, where Q is the finite state set, A is the alphabet, δ is the transition relation, I the set of initial states. The set of transition relations is a subset of $Q \times (A \cup \{\lambda\})^k \times Q$. A transition is thus a tuple $\langle p, \alpha_1, \ldots, \alpha_k, q \rangle$ where p is its *starting state*, $\langle \alpha_1, \ldots, \alpha_k \rangle$ is its *label* and q is its *ending state*. Note that each α_i is here either a symbol a_i of the alphabet or the empty word λ. A transition is written $p \xrightarrow{\alpha_1,\ldots,\alpha_k} q$. As usual, two transitions are *consecutive* if the ending state of the first one is the starting state of the second one.

A finite *run* is a finite sequence of consecutive transitions

$$q_0 \xrightarrow{\alpha_{1,1},\ldots,\alpha_{k,1}} q_1 \xrightarrow{\alpha_{1,2},\ldots,\alpha_{k,2}} q_2 \cdots q_{n-1} \xrightarrow{\alpha_{1,n},\ldots,\alpha_{k,n}} q_n$$

The *label* of the run is the component-wise concatenation of the labels of the transitions. More precisely, it is the tuple $\langle u_1, \ldots, u_k \rangle$ where each u_j for $1 \leq j \leq k$ is equal to $\alpha_{j,1}\alpha_{j,2}\cdots\alpha_{j,n}$. Such a run is written shortly as $q_0 \xrightarrow{u_1,\ldots,u_k} q_n$.

An infinite *run* is an infinite sequence of consecutive transitions

$$q_0 \xrightarrow{\alpha_{1,1},\ldots,\alpha_{k,1}} q_1 \xrightarrow{\alpha_{1,2},\ldots,\alpha_{k,2}} q_2 \xrightarrow{\alpha_{1,3},\ldots,\alpha_{k,3}} q_3 \cdots$$

The *label* of the run is the component-wise concatenation of the labels of the transitions. More precisely, it is the tuple $\langle x_1, \ldots, x_k \rangle$ where each x_j for $1 \leq j \leq k$ is equal to $\alpha_{j,1}\alpha_{j,2}\alpha_{j,3}\cdots$. Note that some label x_j might be finite although the run is infinite since some transitions may have empty labels. The run is accepting if its first state q_0 is initial and each word x_j is infinite. Such an accepting run is written shortly $q_0 \xrightarrow{x_1,\ldots,x_k} \infty$. The tuple $\langle x_1, \ldots, x_k \rangle$ is accepted if there exists at least one accepting run with label $\langle x_1, \ldots, x_k \rangle$. Notice that there is no constraint on the states occurring infinitely often in an accepting run.

In this work we consider only deterministic k-automata whose transition function is determined by a subset of the k tapes. We start with some definitions. The *support* of a tuple $\langle \alpha_1, \ldots, \alpha_\ell \rangle$ in $(A \cup \{\lambda\})^\ell$ is the set of positions in the tuple symbols in A, For $1 \leq \ell \leq k$, the *ℓ-label* of a transition $p \xrightarrow{\alpha_1,\ldots,\alpha_k} q$ is the tuple $\langle \alpha_1, \ldots, \alpha_\ell \rangle$ and its *ℓ-support* is the support of its ℓ-label.

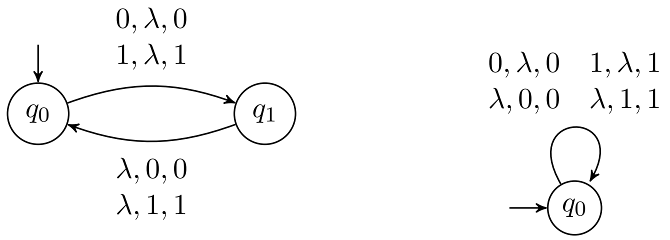

Figure 2: A 2-deterministic 3-automaton (left) and a non-deterministic 3-automaton (right)

We say that a k-automaton is *ℓ-deterministic*, with $1 \le \ell \le k$, if the following two conditions are fulfilled:

1. the set I of initial states is a singleton set;
2. for each state q, all transitions starting at q have the same ℓ-support but they have pairwise different ℓ-labels.

If the automaton is ℓ-deterministic, we call the common ℓ-support of all transitions starting from a state q the *ℓ-support* of q. The automaton is called *ℓ-complete* if for any tuple $\alpha = \langle \alpha_1, \ldots, \alpha_\ell \rangle$ and any state q such that the ℓ-support of α is equal to the ℓ-support of q, there is one transition starting at q with ℓ-label α.

The ℓ-determinism (respectively ℓ-completeness) guarantees that for each tuple $\langle x_1, \ldots, x_\ell \rangle$ of infinite words, there exists at most (respectively at least) one run such that first ℓ components of its label are $\langle x_1, \ldots, x_\ell \rangle$. However, this run might be not accepting since one of its labels is not infinite.

The 3-automaton at the left of Figure 2 accepts a triple $\langle x, y, z \rangle$ of infinite words over the alphabet $\{0, 1\}$ whenever z is the join of x and y; recall that the join of two infinite words $x = a_1 a_2 a_3 \cdots$ and $y = b_1 b_2 b_3 \cdots$ is the infinite word $z = a_1 b_1 a_2 b_2 a_3 \cdots$. This automaton is 2-deterministic. The 3-automaton pictured at the right of Figure 2 accepts a triple $\langle x, y, z \rangle$ of infinite words over the alphabet $\{0, 1\}$ whenever z is a shuffle of the symbols in x and y. This automaton is not 2-deterministic. Indeed the first condition on transitions is not fulfilled by the two transitions $q_0 \xrightarrow{0,\lambda,0} q_0$ and $q_0 \xrightarrow{\lambda,0,0} q_0$.

Let $\mathcal{A}$ be an ℓ-deterministic k-automaton. For each tuple $\langle x_1, \ldots, x_\ell \rangle$ of infinite words, there exists at most one tuple $\langle y_{\ell+1}, \ldots, y_k \rangle$ of infinite words such that the k-tuple $\langle x_1, \ldots, x_\ell, y_{\ell+1}, \ldots, y_k \rangle$ is accepted by $\mathcal{A}$. The automaton $\mathcal{A}$ realizes then a partial function from $(A^\omega)^\ell$ to $(A^\omega)^{k-\ell}$ and the tuple $\langle y_{\ell+1}, \ldots, y_k \rangle$ is denoted $\mathcal{A}(x_1, \ldots, x_\ell)$. The 1-deterministic 2-automata are also called sequential transducers in the literature. When a k-automaton is ℓ-deterministic, each transition is written

$$p \xrightarrow{\alpha_1,\ldots,\alpha_\ell|\beta_{\ell+1},\ldots,\beta_k} q$$

to emphasize that the first ℓ tapes are input tapes and that the $k - \ell$ remaining ones are output tapes.

Let $\mathcal{A}$ be a 1-deterministic 2-automaton. We say that $\mathcal{A}$ is a *compressor* if the (partial) function $x \mapsto \mathcal{A}(x)$ which maps x to the output $\mathcal{A}(x)$ is one-to-one.

The compression ratio of an infinite word x for $\mathcal{A}$ is given by the unique accepting run $q_0 \xrightarrow{u_1|v_1} q_1 \xrightarrow{u_2|v_2} q_2 \xrightarrow{u_3|v_3} q_3 \cdots$ where $x = u_1u_2u_3\cdots$ as

$$\rho_{\mathcal{A}}(x) = \liminf_{n\to\infty} \frac{|v_1v_2v_3\cdots|}{|u_1u_2u_3\cdots|}.$$

This compression ratio for a given automaton $\mathcal{A}$ can have any value. In particular, it can be greater than 1. An infinite word x is *compressible* by a 1-deterministic 2-automaton $\mathcal{A}$ if $\rho_{\mathcal{A}}(x) < 1$. The *compression ratio* of a given word x, $\rho(x)$, is the infimum of the compression ratios achievable by all one-to-one 1-deterministic 2-automata, namely,

$$\rho(x) = \inf\{\rho_{\mathcal{A}}(x) : \mathcal{A} \text{ is a one-to-one 1-deterministic 2-automaton}\}$$

For every infinite word x, $\rho(x)$ is less than or equal to 1, because there exists a compressor $\mathcal{A}_0$ which copies each symbol of the input to the output, so $\rho_{\mathcal{A}_0}(x)$ is equal to 1. The compression ratio of the word $x = 0^\omega$ is $\rho(x) = 0$ because for each positive real number ε there exists a compressor $\mathcal{A}$ such that $\rho_{\mathcal{A}}(x) < \varepsilon$. Notice that in this case the compression ratio equal to 0 is not achievable by any compressor $\mathcal{A}$. It follows from the results in [19, 11] that the words x with compression ratio $\rho(x)$ equal to 1 are the exactly the normal words. A direct proof of this result appears in [5, Characterization Theorem].

2.2 Normality

We start with the notation for the number of occurrences and the number of aligned occurrences of a given word.

Definition 2.1. For w and u two words, the number $|w|_u$ of *occurrences* of u in w and the number $\|w\|_u$ of *aligned occurrences* of u in w are respectively given by

$$\begin{aligned} |w|_u &= |\{i : w[i..i+|u|-1] = u\}|, \\ \|w\|_u &= |\{i : w[i..i+|u|-1] = u \text{ and } i = 1 \mod |u|\}|. \end{aligned}$$

For example, $|aaaaa|_{aa} = 4$ and $\|aaaaa\|_{aa} = 2$. Notice that the definition of aligned occurrences has the condition $i = 1 \mod |u|$ instead of $i = 0 \mod |u|$, because the positions are numbered starting at 1. Of course, when a word u is just a symbol, $|w|_u$ and $\|w\|_u$ coincide. Counting aligned occurrences of a word of length r over alphabet A is exactly the same as counting occurrences of the corresponding symbol over alphabet A^r. Precisely, consider alphabet A, a length r, and an alphabet B with $|A|^r$ symbols. The set of words of length r over alphabet A and the set B are isomorphic, as witnessed by the isomorphism $\pi : A^r \to B$ induced by the lexicographic order in the respective sets. Thus, for any $w \in A^*$ such that $|w|$ is a multiple of r, $\pi(w)$ has length $|w|/r$ and $\pi(u)$ has length 1, as it is just a symbol in B. Then, for any $u \in A^r$, $\|w\|_u = |\pi(w)|_{\pi(u)}$.

We now recall the definition of Borel normality [7] directly on infinite words. See the books [8, 14] for a thorough presentation of the material. An infinite word x is *simply normal* to word length ℓ if, for every $u \in A^\ell$,

$$\lim_{n\to\infty} \frac{\|x[1..n\ell]\|_u}{n} = |A|^{-\ell}.$$

An infinite word x is *normal* if it is simply normal to every word length. There are several other equivalent formulations of normality, they can be read from [8].

2.3 Independence

Roughly, two infinite words, possibly over different alphabets, are finite-state independent if no one helps to compress the other using 3-automata. In our setting, a *compressor* is a 2-deterministic 3-automata $\mathcal{A}$ such that for any fixed infinite word y, the function $x \mapsto \mathcal{A}(x, y)$ which maps x to the output $\mathcal{A}(x, y)$ is one-to-one. This guarantees that if y is known, x can be recovered from $\mathcal{A}(x, y)$. Note that we do not require that the function $(x, y) \mapsto \mathcal{A}(x, y)$ be one-to-one, which would be a much stronger assumption. For example, the 2-deterministic 3-automaton $\mathcal{C}$ which maps infinite words x and y to the infinite word z satisfying $z[i] = x[i] + y[i] \mod |A|$ for each $i \geq 1$ is, indeed a compressor but the function $(x, y) \mapsto \mathcal{C}(x, y)$ is not one-to-one.

Definition 2.2 ([3])**.** Let $\mathcal{A}$ be a compressor. For simplicity in the presentation we assume just one alphabet. However, it is possible to have three different alphabets, one for each input tape and one for the output tape. The *conditional compression ratio* of an infinite word x with respect to y in $\mathcal{A}$ is given by the unique accepting run

$$q_0 \xrightarrow{u_1,v_1|w_1} q_1 \xrightarrow{u_2,v_2|w_2} q_2 \xrightarrow{u_3,v_3|w_3} q_3 \cdots$$

such that $x = u_1u_2u_3 \cdots$ and $y = v_1v_2v_3 \ldots$ as

$$\rho_{\mathcal{A}}(x/y) = \liminf_{n\to\infty} \frac{|w_1w_2w_3 \cdots|}{|u_1u_2u_3 \cdots|}.$$

In case the input tape and the ouput tape have respective alphabets A and B of different sizes, the formula above should be multiplied by $\log|A|/\log|B|$. Notice that the number of symbols read from y, namely $|v_1v_2v_3 \cdots|$, is not taken into account in the value of $\rho_{\mathcal{A}}(x/y)$.

The *conditional compression ratio* of an infinite word x given an infinite word y, $\rho(x/y)$, is the infimum of the compression ratios $\rho_{\mathcal{A}}(x/y)$ of all compressors $\mathcal{A}$ with input x and oracle y.

Definition 2.3 ([3])**.** Two infinite words x and y, possibly over different alphabets, are *finite-state independent* if $\rho(x/y) = \rho(x)$, $\rho(y/x) = \rho(y)$ and the compression ratios of x and y are non-zero.

In the sequel instead of writing *finite-state independence* we simply write *independence.*

Notice that the compression ratios of x and y should not be zero. This means that a word x such that $\rho(x) = 0$ is independent of no word. Without this requirement, two words x and y such that $\rho(x) = \rho(y) = 0$ would be independent. In particular, each word x with $\rho(x) = 0$ would be independent of itself.

From the definition of independence follows that if the infinite words x and y are independent, each suffix of x is independent to each suffix of y.

3 Statement of the Characterization Theorem

3.1 Frequencies of states

We first introduce the definitions to characterize independence in terms of frequencies of states in runs of 2-deterministic 2-automata on normal words.

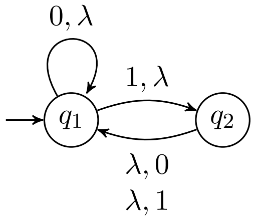

Figure 3: A 2-deterministic 2-automaton

Let $\mathcal{A}$ be a 2-deterministic 2-automaton and let x and y be two infinite words, possibly over different alphabets. Let γ be the run of $\mathcal{A}$ on x and y

$$q_1 \xrightarrow{\bar{a}_1,\bar{b}_1} q_2 \xrightarrow{\bar{a}_2,\bar{b}_2} q_3 \xrightarrow{\bar{a}_3,\bar{b}_3} q_4 \cdots$$

where each $\bar{a}_i$ and each $\bar{b}_i$ is either a symbol or the empty word and each $q_i \xrightarrow{\bar{a}_i,\bar{b}_i} q_{i+1}$ is a transition of $\mathcal{A}$.

With some abuse of notation let $|\gamma[1..n]|_q$ denote the number of occurrences of state q in the first n transitions of γ; that is, the cardinality of the set

$$\{i : 1 \le i \le n,\ q_i = q\}.$$

Similarly, for each transition $\tau = p \xrightarrow{\bar{a},\bar{b}} q$ let $|\gamma[1..n]|_\tau$ denote the number of occurrences of τ in the first n transitions of γ; that is, the cardinality of the set

$$\{i : 1 \le i \le n,\ q_i \xrightarrow{\bar{a}_i,\bar{b}_i} q_{i+1} = \tau\}.$$

We associate with 2-deterministic and 2-complete 2-automaton $\mathcal{A}$ a Markov chain described by a stochastic matrix M. Let A and B be the alphabets for the first and second tape of $\mathcal{A}$. The state set of the Markov chain is the state set Q of $\mathcal{A}$. The dimension of the matrix M is thus the number $|Q|$ of states and its rows and columns are indexed by element of Q. For two states p and q, the (p,q)-entry of M is the sum of the weights of all transitions from p to q where the weights are the following. The weight of a transition of the form $p \xrightarrow{a,\lambda} q$ (respectively $p \xrightarrow{\lambda,b} q$) is $1/|A|$ (respectively $1/|B|$) whereas the weight of a transition of the form $p \xrightarrow{a,b} q$ is $1/(|A||B|)$.

If the automaton $\mathcal{A}$ is strongly connected then the Markov chain is irreducible. By [20, Theorem 1.5], there exists a unique stationary distribution, that is, a line vector π such that $\pi M = \pi$ and $\sum_{q\in Q} \pi(q) = 1$. By definition, it is called the *stationary distribution* associated with the automaton $\mathcal{A}$. For example, the matrix of the associated Markov chain for the 2-automaton in Figure 3 is the 2×2-matrix M given by

$$M = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ 1 & 0 \end{pmatrix}$$

and the stationary distribution is thus given by $\pi(q_1) = 2/3$ and $\pi(q_2) = 1/3$.

Theorem 3.3 establishes that the frequencies of states in a run on normal independent words are given by the stationary distribution associated with the automaton. This means that the frequencies of states do not depend on the input words. This statement is analogous to [19, Lemma 4.5] but for 2-deterministic 2-automata.

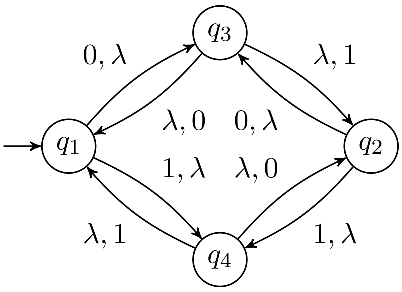

Figure 4: Another 2-deterministic 2-automaton

The following example shows that when two input words are normal but not independent then the frequency of states in the run depends on the input words. Consider the 2-deterministic and 2-complete 2-automaton pictured in Figure 4. The matrix of the associated Markov chain is the 4×4-matrix M given by

$$M = \frac{1}{2}\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

and the stationary distribution is thus given by $\pi(q_1) = \pi(q_2) = \pi(q_3) = \pi(q_4) = 1/4$. If the input words x and y are such that $x = y$, the run never visits the state q_2 and therefore the frequency of this state along the run is not equal to $1/4$.

3.2 Selecting

We present the definition of a selector that we use to characterize independence of normal words, to be given in Theorem 3.3. Given a normal infinite word, the problem of selection is how to select symbols from an infinite word so that the word defined by the selected symbols satisfies a designated property. An early result of Wall [22] shows that selecting the symbols of a normal word in the positions given by an arithmetical progression yields again a normal word. Agafonov [1] extended Wall's result and proved that any selection by finite automata preserves normality (a complete proof can be read for [5, Agafonov's Theorem] or see the slightly more general version [2, Theorem 7.1]). The selections admitted by Agafonov must be performed by an oblivious 1-deterministic 2-automaton. Oblivious means that the choice of selecting or not the next symbol only depends on the current state and not on the next symbol.

Other forms of selection by finite-automata do not preserve normality. For instance [2, Theorem 7.3] shows that the two-sided selection rule "select symbols in between two zeroes" from x, does not preserve normality.

In order to characterize independence we consider selection by a finite automata from an infinite word, conditioned to another infinite word that can be used in the selection process as an oracle.

Definition 3.1. A *selector* is a 2-deterministic 3-automaton such that each of its transitions has one the types $p \xrightarrow{a,\lambda|a} q$ (type I), $p \xrightarrow{a,\lambda|\lambda} q$ (type II), or $p \xrightarrow{\lambda,b|\lambda} q$ (type III) for two symbols $a, b \in A$. It is *oblivious* if all transitions starting at a given state have the same type.

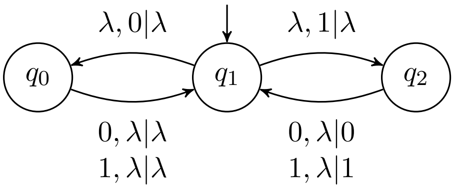

Figure 5: An oblivious selector

A transition of the type $p \xrightarrow{a,\lambda|a} q$ (type I) copies a symbol from the first input x to the output tape. A transition of the types $p \xrightarrow{a,\lambda|\lambda} q$ (type II) or $p \xrightarrow{\lambda,b|\lambda} q$ (type III) skips a symbol from either the first input x or the second input y. It follows then that the output word $z = \mathcal{S}(x, y)$ is obtained by selecting symbols from x. This justifies the terminology.

Since a selector is 2-deterministic, all transitions starting at a a given state either have type I and II or have type III. When it is oblivious it is not possible anymore that two transitions starting at the same state have types I and II. Whether or not a symbol is copied from the first input tape to the output tape only depends on the state and not on the symbol.

The automaton pictured in Figure 5 is an oblivious selector. It selects symbols from the first input x which are at a position where there is a symbol 1 in the second input y.

3.3 Shuffling

We present the definition of a shuffler we use to characterize independence of normal words in Theorem 3.3. An infinite word z is the shuffle of x and y if it can be factorized $z = u_1v_1u_2v_2u_3\cdots$ where the sequences of words $(u_i)_{i\geq 1}$ and $(v_i)_{i\geq 1}$ satisfy $x = u_1u_2u_3\cdots$ and $y = v_1v_2v_3\cdots$. We restrict to shuffles of words on the same alphabet, done by 2-deterministic 3-automata. We will prove that if x and y are normal words, x and y are independent exactly when any shuffle of them is also normal. The interleaving of the symbols from x and y must be driven by a deterministic and oblivious automaton reading x and y. Here oblivious means that the choice of inserting in the shuffled word z a symbol either from x or from y is only made upon the current state of the automaton and not upon the current symbols read from x and y.

Definition 3.2. A *shuffler* is a 2-deterministic 3-automaton such that each of its transitions has either the type $p \xrightarrow{a,\lambda|a} q$ (type I) or the type $p \xrightarrow{\lambda,a|a} q$ (type II).

Notice that the determinism of a shuffler $\mathcal{S}$ implies that for each its state p, all the transitions leaving p have the same type, either type I or type II. A transition of type I copies a symbol from the first input x to the output and a transition of type II copies a symbol from the second input y to the output. It follows then that the third word $z = \mathcal{S}(x, y)$ is obtained by shuffling x and y. This justifies the terminology.

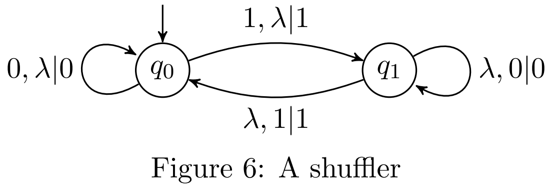

Figure 6: A shuffler

Consider infinite words $x = \overline{0011010001}\cdots$, $\qquad y = \underline{01000110001}\cdots$ and let $\mathcal{S}$ be the shuffler pictured in Figure 6. Then, the infinite word $z = \mathcal{S}(x, y)$ has the form

$$z = \overline{001}\underline{01}\overline{1}\underline{0001}\overline{01}\underline{1}\overline{0001}\underline{0001}\cdots$$

where the underlines and the overlines have been added to mark the origin of each symbol.

3.4 Characterization Theorem

Theorem 3.3 (Characterization Theorem)**.** *Let x and y be two normal words on the alphabets A and B. The following statements are equivalent.*

1. *The words x and y are independent.*
2. *For every strongly connected 2-deterministic automaton $\mathcal{A}$ and for every infinite run γ having as input a suffix of x and a suffix of y, $\mathcal{A}$ is 2-complete and the frequency of each state q in γ is*
$$\lim_{n\to\infty} \frac{|\gamma[1..n]|_q}{n} = \pi(q).$$
where π is the stationary distribution associated with $\mathcal{A}$.
3. *For any oblivious selector $\mathcal{S}$, the results $\mathcal{S}(x, y)$ and $\mathcal{S}(x, y)$ are also normal.*

Furthermore, if alphabets A and B are equal, the following statement is also equivalent.

4. *For any shuffler $\mathcal{S}$, the result $\mathcal{S}(x, y)$ is also normal.*

We present the proof of the theorem in the next section.

4 Proof of the Characterization Theorem

4.1 From independence to frequencies of states

Statement (2) of Theorem 3.3 considers runs when the automaton is strongly connected. If the automaton is not strongly connected, the run reaches a strongly connected component C without any transition leaving it, that is, made of recurrent states in the terminology of Markov chains. Therefore, this strongly connected component can be considered as a 2-deterministic 2-automaton. Furthermore, if the infinite words x and y are independent, each suffix of x is independent to each suffix of y. Statement (2) of Theorem 3.3 can be applied to a suffix of run which only visits states from C. The frequency of visits of each state in C is given by the stationary distribution of the Markov chain associated with C.

The first lemma says that it can be assumed that automata are in some normal form.

Lemma 4.1. *Each (2-deterministic) 2-automaton can be transformed to another that admits exactly the same infinite runs and it is such that all its transitions are of the form $p \xrightarrow{a,\lambda} q$ or $p \xrightarrow{\lambda,b} q$ for some symbols a and b.*

Proof. Transitions of the form $p \xrightarrow{a,b} q$ can be replaced by transitions of the form $p \xrightarrow{a,\lambda} q$ and $p \xrightarrow{\lambda,b} q$. Suppose that there exists a transition $p \xrightarrow{a,b} q$. Introduce a new state q_a and a transition $p \xrightarrow{a,\lambda} q_a$. Each transition $p \xrightarrow{a,b} q$ for $b \in A$ is then replaced by the transition $q_a \xrightarrow{\lambda,b} q$. In each run, the transition $p \xrightarrow{a,b} q$ can be replaced by the run $p \xrightarrow{a,\lambda} q_a \xrightarrow{\lambda,b} q$ of length 2. Furthermore, if the automaton is 1-deterministic, this transformation preserves the feature. □

The following lemma relates the frequency of a state with the frequency of the transitions starting at it. It is the first step towards the characterization through frequencies of states.

Lemma 4.2. *Let γ be the run of a 2-deterministic 2-automaton $\mathcal{A}$ on two independent normal words. Let p be a state of $\mathcal{A}$ and σ and σ' be two transitions starting at p. Let $(k_n)_{n\geq 0}$ be an increasing sequence of integers such that $\lim_{n\to\infty} |\gamma[1..k_n]|_p/k_n > 0$. Then*

$$\lim_{n\to\infty} \frac{|\gamma[1..k_n]|_\sigma}{|\gamma[1..k_n]|_p} = \lim_{n\to\infty} \frac{|\gamma[1..k_n]|_{\sigma'}}{|\gamma[1..k_n]|_p}.$$

Proof. For simplicity we assume that A is the binary alphabet $\{0,1\}$ but the proof can easily be extended to the general case. By Lemma 4.1, it can be assumed that each transition of $\mathcal{A}$ is of the form $p \xrightarrow{a,\lambda} q$ or $q \xrightarrow{\lambda,b} q$ for some symbols a and b. For the rest of the proof, transitions of the form $p \xrightarrow{a,\lambda} q$ are called of type I and transitions of the form $p \xrightarrow{\lambda,b} q$ are called of type II. Since the automaton is deterministic, all the transitions starting at each state q have the same type. A state q is said to be of type I (respectively II) if all transitions starting at q have type I (respectively II). By symmetry it can be assumed that all transitions starting at p, including σ and σ', are of type I.

We suppose by contradiction that the required equality does not hold and we claim that x and y are not independent. We show that x can be compressed given y. There is a lack of symmetry between x and y because transitions σ and σ' are of type I. By replacing $(k_n)_{n\geq 0}$ by one of its subsequences, it can be assumed that, for each transition τ, $\lim_{n\to\infty} |\gamma[1..k_n]|_\tau/k_n$ exists and that $\lim_{n\to\infty} |\gamma[1..k_n]|_\sigma/k_n \neq \lim_{n\to\infty} |\gamma[1..k_n]|_{\sigma'}/k_n$. Since the frequency of each state is equal to the sum of the frequencies of the transitions which start at it, the limit $\lim_{n\to\infty} |\gamma[1..k_n]|_q/k_n$ exists for each state q. Denote this limit by $\pi(q)$.

For each transition τ starting at a state q, let $\pi(\tau)$ be defined as follows.

$$\pi(\tau) = \begin{cases} \lim\limits_{n\to\infty} \dfrac{|\gamma[1..k_n]|_\tau}{|\gamma[1..k_n]|_q} & \text{if } \lim_{n\to\infty} |\gamma[1..k_n]|_q/k_n \neq 0 \\ \dfrac{1}{2} & \text{otherwise} \end{cases}$$

Since $\lim_{n\to\infty} |\gamma[1..k_n]|_\sigma/k_n \neq \lim_{n\to\infty} |\gamma[1..k_n]|_{\sigma'}/k_n$, $\pi(\sigma) \neq \pi(\sigma')$. Furthermore, the following equality holds for each state q.

$$\sum_{\tau \text{ starts at } q} \pi(\tau) = 1.$$

Since x is normal it suffices to show that $\rho(x/y) < 1$. Let ℓ be a block length to be fix later. Let γ be a finite run of length ℓ, so γ is a sequence $\tau_1\tau_2\cdots\tau_\ell$ of ℓ consecutive transitions. Let $\pi(\gamma)$ be defined as follows.

$$\pi(\gamma) = \begin{cases} \prod\limits_{\substack{\tau_i \text{ of type I} \\ 1\leq i\leq \ell}} \pi(\tau_i) & \text{if } \gamma \text{ has transition of type I} \\ 1 & \text{otherwise} \end{cases}$$

Let q be a state and $\bar{v}$ be a word of length ℓ. Let $\Gamma_{q,\bar{v}}$ be the set of runs of length ℓ, starting at q and reading a prefix of $\bar{v}$ on the second tape,

$$\Gamma_{q,\bar{v}} = \{\gamma : \gamma = q \xrightarrow{u,v} q', v \sqsubset \bar{v}, |u| + |v| = \ell\}.$$

Notice that the sets $\Gamma_{q,\bar{v}}$ are not always pairwise disjoint. The word v read by the run γ on the second tape can be the prefix of several words $\bar{v}$. If v is the prefix of both $\bar{v}$ and $\bar{v}'$, then the run γ belongs to both $\Gamma_{q,\bar{v}}$ and $\Gamma_{q,\bar{v}'}$.

We claim that for each state q and each word $\bar{v}$,

$$\sum_{\gamma \in \Gamma_{q,\bar{v}}} \pi(\gamma) = 1.$$

We prove it by induction on the length of the run, that we call ℓ. If $\ell = 0$, the only run $\gamma \in \Gamma_{q,\bar{v}}$ is the empty run so $\pi(\gamma) = 1$. Suppose now that $\ell \geq 1$. We distinguish two cases. First case: the transitions starting at q are of type I. Suppose first that the transitions starting at q are the two transitions $\tau_0 = q \xrightarrow{0,\lambda} q_0$ and $\tau_1 = q \xrightarrow{1,\lambda} q_1$. And suppose that $\bar{v} = \bar{v}'a$ where $\bar{v}' = v[1..\ell-1]$ and a is the last symbol of $\bar{v}$. The set $\Gamma_{q,\bar{v}}$ is then equal to the disjoint union $\Gamma_{q,\bar{v}} = \tau_0\Gamma_{q_0,\bar{v}'} \cup \tau_1\Gamma_{q_1,\bar{v}'}$. The result follows from the inductive hypothesis since $\pi(\Gamma_{q,\bar{v}}) = \pi(\tau_0)\pi(\Gamma_{q_0,\bar{v}'}) + \pi(\tau_1)\pi(\Gamma_{q_1,\bar{v}'}) = \pi(\tau_0) + \pi(\tau_1) = 1$. Second case: the transitions starting at q have type II. Suppose that $\bar{v} = a\bar{v}'$ where a is the first symbol of $\bar{v}$ and $\bar{v}' = v[2..\ell]$. The transition $\tau = q \xrightarrow{\lambda,a} q'$ is the first transition of each run in $\Gamma_{q,\bar{v}}$ and $\Gamma_{q,\bar{v}} = \tau\Gamma_{q',\bar{v}'}$. The result follows from the inductive hypothesis since $\pi(\Gamma_{q,\bar{v}}) = \pi(\Gamma_{q',\bar{v}'}) = 1$. Since $\sum_{\gamma\in\Gamma_{q,\bar{v}}} \pi(\gamma) = 1$, there exists, for each state q and each word $\bar{v}$, a prefix-free set $P_{q,\bar{v}} = \{w_{\gamma,\bar{v}} : \gamma \in \Gamma_{q,\bar{v}}\}$ such that $|w_{\gamma,\bar{v}}| \leq \lceil -\log \pi(\gamma)\rceil$ holds for each run $\gamma \in \Gamma_{q,\bar{v}}$. These words can be used to define a compressor $\mathcal{C}$ which runs as follows on two inputs. It simulates $\mathcal{A}$ and it has ℓ symbols of look ahead on the second tape. For each run γ of length ℓ, the compressor outputs $w_{\gamma,\bar{v}}$ on the third tape. The choice of $w_{\gamma,\bar{v}}$ depends on the look ahead $\bar{v}$.

We finally show that $\rho_{\mathcal{C}}(x/y) < 1$. The run γ of $\mathcal{A}$ on x and y can be factorized as $\gamma = \gamma_1\gamma_2\gamma_3\cdots$ where each run γ_i has length ℓ. The output of the compressor $\mathcal{C}$ is then $w_{\gamma_1,\bar{v}_1}w_{\gamma_2,\bar{v}_2}w_{\gamma_3,\bar{v}_3}\cdots$ where the words $\bar{v}_1, \bar{v}_2, \bar{v}_3, \ldots$ are the corresponding look ahead of ℓ symbols. Let $\varepsilon, \delta > 0$ be two positive real numbers. Let n be an integer large enough such that $|\gamma[1..k_n]|_\tau \leq (1+\delta)\pi(q)\pi(\tau)k_n$ for each transition τ starting at q. Then,

$$\begin{aligned}
|w_{\gamma_1,\bar{v}_1}\cdots w_{\gamma_n,\bar{v}_n}| &\leq \sum_{i=1}^{n} \lceil -\log \pi(\gamma_i)\rceil \\
&\leq n + \sum_{i=1}^{n} -\log \pi(\gamma_i) \\
&\leq n + \sum_{\tau \text{ of type I}} |\gamma[1..\ell n]|_\tau \log\frac{1}{\pi(\tau)} \\
&\leq \ell n \left[\frac{1}{\ell} + (1+\delta)\sum_{q \text{ of type I}} \pi(q) \sum_{\tau \text{ starts at } q} \pi(\tau)\log\frac{1}{\pi(\tau)}\right]
\end{aligned}$$

Then, for each state q,

$$\sum_{\tau \text{ starts at } q} \pi(\tau)\log\frac{1}{\pi(\tau)} \leq 1$$

and the relation is strict for $q = p$. Since $\pi(p) > 0$, for ε small enough, δ and ℓ can be chosen such that

$$\frac{1}{\ell} + (1+\delta)\sum_{q \text{ of type I}} \pi(q) \sum_{\tau \text{ starts at } q} \pi(\tau)\log\frac{1}{\pi(\tau)} \leq (1-\varepsilon)\sum_{q \text{ of type I}} \pi(q).$$

We obtain

$$|w_{\gamma_1,\bar{v}_1} \cdots w_{\gamma_{k_n},\bar{v}_{k_n}}| \leq (1-\varepsilon)\ell k_n \sum_{q \text{ of type I}} \pi(q).$$

Since $\sum_{q \text{ of type I}} \pi(q)$ is the limit of the ratio between the number of symbols read from x and the length of the run, we conclude $\rho_{\mathcal{C}}(x/y) < 1$. □

The following lemma states that if the conclusion of Lemma 4.2 holds then statement (2) of Theorem 3.3 also holds.

Lemma 4.3. *Let x and y be two normal words such that for each 2-deterministic 2-automaton $\mathcal{A}$, if p is a state of $\mathcal{A}$, σ and σ' are two transitions starting at p and $(k_n)_{n\geq 0}$ is an increasing sequence of integers such that $\lim_{n\to\infty} |\gamma[1..k_n]|_p/k_n > 0$. Then*

$$\lim_{n\to\infty} \frac{|\gamma[1..k_n]|_\sigma}{|\gamma[1..k_n]|_p} = \lim_{n\to\infty} \frac{|\gamma[1..k_n]|_{\sigma'}}{|\gamma[1..k_n]|_p}.$$

Then statement (2) of Theorem 3.3 holds.

Proof. Let $\mathcal{A}$ be a strongly connected 2-deterministic 2-automaton. We first show that if there is an infinite run γ in $\mathcal{A}$ on suffixes x' and y' of x and y, then $\mathcal{A}$ must be 2-complete. Suppose by contradiction that $\mathcal{A}$ is not 2-complete. Suppose that $x = ux'$ and $y = vy'$. By adding a few states and transitions to $\mathcal{A}$, we construct a 2-deterministic automaton $\mathcal{A}'$ which first read u and v and then does as $\mathcal{A}$. The run of $\mathcal{A}'$ on x and y is equal to $\rho\gamma$ for some finite run ρ. We first claim that each state q in $\mathcal{A}$ has non-zero frequency. Let P be the subset of states $\{q : \liminf_{n\to\infty} \frac{|\gamma[1..k_n]|_p}{n} > 0\}$. The hypothesis implies that each reacable from a state in P is also in P. Since $\mathcal{A}$ is strongly connected, all states of $\mathcal{A}$ are in P. Now suppose by constradiction that $\mathcal{A}$ is not complete. It can be made complete by adding transitions which are not used in the run γ. This contracdict the hypothesis.

To prove the statement about frequencies of states, it is sufficient to show that for each increasing sequence of integers $(k_n)_{n\geq 0}$ such that $\lim_{n\to\infty} |\gamma[1..k_n]|_q/k_n$ exists, this limit is equal to $\pi(q)$. Let $(k_n)_{n\geq 0}$ be such sequence. Replace $(k_n)_{n\geq 0}$ by one of its sub-sequences so that $\lim_{n\to\infty} |\gamma[1..k_n]|_q/k_n$ exists for each state q. It has already been shown in the previous paragraph that this limit cannot be 0.

We introduce two sequences $(v_n)_{n\geq 0}$ and $(v'_n)_{n\geq 0}$ of line vectors and a sequence $(M_n)_{n\geq 0}$ of matrices. For each state q, the q-entries of the vectors v_n and v'_n are given by $v_n(q) = |\gamma[1..k_n]|_q/k_n$ and $v'_n(q) = |\gamma[2..k_n+1]|_q/k_n$. For each pair of states p and q, the (p,q)-entry of M_n is the sum over all transitions τ from p to q of the ratio $|\gamma[1..k_n]|_\tau/|\gamma[1..k_n]|_p$. A routine check yields that $v_nM_n = v'_n$ holds for each integer $n \geq 1$. Both sequences $(v_n)_{n\geq 0}$ and $(v'_n)_{n\geq 0}$ converge to the same line vector v given by $v(q) = \lim_{n\to\infty} |\gamma[1..k_n]|_q/k_n$. From the hypothesis, the sequence $(M_n)_{n\geq 0}$ converges to the matrix M of the Markov chain associated with $\mathcal{A}$. Taking limits gives that $vM = v$. By the unicity of the stationary distribution of M, $v(q) = \pi(q)$ holds for each state q. □

Proof of Theorem 3.3, (1) implies (2). The proof that the statement (1) of Theorem 3.3 implies the statement (2) follows directly from Lemmas 4.2 and 4.3. □

4.2 From frequencies of states to independence

To prove that statement (2) implies statement (1) in Theorem 3.3 we shall use the following lemmas that consider strongly connected components of automata and stationary distributions. A strongly connected component of an automaton is called *final* if no transition leaves it.

Lemma 4.4. *Assume statement (2) of Theorem 3.3 holds. Then, any infinite run in* 2*-deterministic automaton on x and y reaches a final strongly connected component.*

Proof. Suppose that the infinite run γ in the 2-deterministic automaton $\mathcal{A}$ never reaches a final strongly connected component. There is a suffix of the run that remains in a non final strongly connected component C. Consider the restriction $\mathcal{A}'$ of $\mathcal{A}$ to the state set C. As C is not final, $\mathcal{A}'$ is not complete. This is a contradiction. □

Lemma 4.5. *If $\mathcal{A}$ is strongly connected, then the restriction of $\mathcal{A}_{k,\ell}$ to the set $Q \times A^k \times B^\ell$ is also strongly connected.*

Proof. Let (q,u,v) and (q',u',v') be two states in $Q \times A^k \times B^\ell$. There exist a word w in $A^* \cup B^*$ and states r of $\mathcal{A}$ such that either $q \xrightarrow{uw,v} r$ or $q \xrightarrow{u,vw} r$ is a finite run in $\mathcal{A}$. By symmetry, it can be assumed that $q \xrightarrow{uw,v} r$ is a finite run in $\mathcal{A}$. Since $\mathcal{A}$ is strongly connected, there exists a run $r \xrightarrow{u'',v''} q'$. Then

$$(q,u,v) \xrightarrow{uwu''u',vv''v'} (q',u',v')$$

is a run in $\mathcal{A}_{k,\ell}$. □

Lemma 4.6. *If $\mathcal{A}$ is strongly connected and π is its stationary distribution, then the stationary distribution of $\mathcal{A}_{k,\ell}$ is given by $\pi(q,u,v) = \pi(q)/|A|^k|B|^\ell$ for each state $(q,u,v) \in Q \times A^k \times B^\ell$.*

Proof. Le $M = (m_{p,q})$ be the $Q \times Q$-matrix of $\mathcal{A}$. Each entry $m_{p,q}$ is equal to either

$$|\{a : p \xrightarrow{a,\lambda} q\}|/|A| \text{ or } |\{b : p \xrightarrow{\lambda,b} q\}|/|B|$$

The vector π is the unique vector satisfying $\pi M = \pi$ and $\sum_{q \in Q} \pi(q) = 1$. Let (q,u,v) be a fixed state. for each transition $p \xrightarrow{a,\lambda} q$, there is a transition $(p,au',v) \xrightarrow{a',\lambda} (q,u,v)$ where $u = u'a'$ (u' is the prefix of length $k-1$ of u and a is its last symbol). □

Proof of Theorem 3.3, (2) implies (1). Let x and y be two normal words such that statement (2) of Theorem 3.3 holds. We will show that x and y are independent. It is sufficient to show that x cannot be compressed with the help of y, since exchanging the roles of x and y we obtain the other incompressibility result.

Let $\mathcal{C}$ be a 2-deterministic 3-automaton. Let q_0 be the initial state of $\mathcal{C}$. Let γ be the run of $\mathcal{C}$ on x and y and let z be the output of $\mathcal{C}$ along γ, that is, $z = \mathcal{C}(x,y)$. Let $\varepsilon > 0$ be a positive real number. We claim that the compression ratio $\rho_{\mathcal{C}}(x/y)$ satisfies $\rho_{\mathcal{C}}(x/y) > 1 - \varepsilon$. Since this holds for each $\varepsilon > 0$, this shows that $\rho_{\mathcal{C}}(x/y) \geq 1$. It can be assumed that $\mathcal{C}$ is strongly connected. Otherwise, the run on x and y in $\mathcal{C}$ reaches, by Lemma 4.4, a final strongly connected component of $\mathcal{C}$. Making the same reasoning with the suffix of run in this strongly connected would prove that suffixes of x and y are not independent, proving that x and y are also not independent.

Let k be a positive integer to be fixed later. Since y is normal, there exists a constant $K > 0$ such that if $u \sqsubset x$, $v \sqsubset y$ and $w \sqsubset z$ (u, v and w are prefixes of x, y and z respectively) such that

$$q_0 \xrightarrow{u,v|w} q$$

then $|v| \leq K|u|$. The run γ is decomposed

$$q_0 \xrightarrow{u_1,v_1|w_1} q_1 \xrightarrow{u_2,v_2|w_2} q_2 \xrightarrow{u_3,v_3|w_3} \cdots$$

where $|u_i| = k$ for each integer $i \geq 1$. Note that the lengths of each word v_i and each word w_i are arbitrary. Our aim is to prove that for N large enough $|w_1 \cdots w_N| \geq (1-\varepsilon)|u_1 \cdots u_N|$.

Let ℓ be the integer $\lceil kK/\varepsilon \rceil$. By definition of ℓ, the cardinality of the set $\{i \leq N : |v_i| > \ell\}$ is less than εN. Otherwise we would have $|v_1 \cdots v_N| > K|u_1 \cdots u_N|$ which contradicts the definition of the constant K. The indices i such that $|v_i| > \ell$ are ignored in the sequel. Let v'_i be the prefix of length ℓ of the infinite word $v_i v_{i+1} v_{i+2} \cdots$. Unless $|v_i| > \ell$, v_i is a prefix of v'_i. Let $v' \in B^\ell$ be a fixed word of length ℓ. The cardinality of the set

$$X_{v'} = \{u \in A^k : \exists p, q \ \ p \xrightarrow{u,v|w} q, v \sqsubset v' \text{ and } |w| < (1-\varepsilon)k\}$$

is bounded by $|Q|^2|A|^{k(1-\varepsilon)}$. The integer k is chosen such that $|A|^k - |Q|^2|A|^{k(1-\varepsilon)}$ is greater than $(1-\varepsilon)|A|^k$. To distinguish states which can occur in the sequence $(q_i)_{i \geq 0}$, we introduce a new automaton $\mathcal{A}'$. Its state set is $Q \times \{0, \ldots, k-1\}$ and its transitions are defined as follows.

$$\begin{array}{lll} (q,i) \xrightarrow{a,\lambda|w} (q', i+1 \bmod k) & \text{if } q \xrightarrow{a,\lambda|w} q' \text{ in } \mathcal{A} \\ (q,i) \xrightarrow{\lambda,b|w} (q', i) & \text{if } q \xrightarrow{\lambda,b|w} q' \text{ in } \mathcal{A} \end{array}$$

Note that the stationary distribution of the new automaton $\mathcal{A}'$ does not satisfy $\pi(q,i) = \pi(q)/k$ because some states may be unreachable. However $\sum_q \pi(q,i) = 1/k$ for each $0 \leq i < k$.

Let $\mathcal{A}$ be 2-deterministic 2-automaton and let k and ℓ be two positive integers. We introduce a new automaton $\mathcal{A}_{k,\ell}$. Its state set is $Q \times A^{\leq k} \times \{\lambda\} \cup Q \times A^k \times B^{\leq \ell}$ and its transitions are defined as follows.

$$\begin{array}{rl} (q,u,\lambda) \xrightarrow{a,\lambda} (q,ua,\lambda) & \text{if } |u| < k \\ (q,u,v) \xrightarrow{\lambda,b} (q,u,vb) & \text{if } |u| = k \text{ and } |v| < \ell \\ (q,au',v) \xrightarrow{a',\lambda} (q,u'a',v) & \text{if } |u'| = k-1,\ |v| = \ell \text{ and } q \xrightarrow{a,\lambda} q' \text{ in } \mathcal{A} \\ (q,u,bv') \xrightarrow{\lambda,b'} (q,u,v'b') & \text{if } |u| = k,\ |v'| = \ell - 1 \text{ and } q \xrightarrow{\lambda,b} q' \text{ in } \mathcal{A} \end{array}$$

Note that the states in $Q \times A^{\leq k} \times \{\lambda\} \cup Q \times A^k \times B^{<\ell}$ are obviously transient. The purpose of these states is to gather the first k symbols of x and the first ℓ symbols of y to reach the state (q_0, u, v) where q_0 is the initial state of $\mathcal{A}$ and u and v are the prefixes of x and y of length k and ℓ respectively. By Lemmas 4.5 and 4.6, the length of the output along the run γ is at least $(1-\varepsilon)^4 kN$,

$$\begin{aligned}\sum_{i=1}^{N}|w_i| &\geq (1-\varepsilon)\sum_{i=1,|v_i|\leq\ell}^{N}|w_i| \\ &\geq \frac{(1-\varepsilon)^2 N}{|A|^k|B|^\ell}\sum_{v_i\in B^\ell}\sum_{u_i\in A^k}|w_i| \\ &\geq \frac{(1-\varepsilon)^2 N}{|A|^k|B|^\ell}\sum_{v\in B^\ell}\sum_{u\in A^k}(1-\varepsilon)k \\ &\geq \frac{(1-\varepsilon)^2 N}{|A|^k|B|^\ell}\sum_{v\in B^\ell}(1-\varepsilon)|A|^k(1-\varepsilon)k \\ &\geq (1-\varepsilon)^4 kN.\end{aligned}$$ □

4.3 From independence to selecting and back

Proof of Theorem 3.3, (1) implies (3). We need to prove that selection from a normal word x with an independent normal oracle y preserves normality. Mutatis mutandis this proof is same as that given in [2, Theorem 7.1], but now one should consider 2-deterministic 3-automata, and the normal word y as a consultive oracle. □

Proof of Theorem 3.3, (3) implies (1). Suppose that x and y are not independent. Since it has already been proved that statements (1) and (2) are equivalent, it can be assumed that statement (2) does not hold. By Lemma 4.3, theres is a 2-deterministic automaton $\mathcal{A}$ with the following property. Let γ be the run of $\mathcal{A}$ on x and y. There exist a state p of $\mathcal{A}$ and σ and σ' two transitions starting at p, an increasing sequence $(k_n)_{n\geq 0}$ of integers such that $\lim_{n\to\infty}|\gamma[1..k_n]|_p/k_n > 0$ and

$$\lim_{n\to\infty}\frac{|\gamma[1..k_n]|_\sigma}{|\gamma[1..k_n]|_p} \neq \lim_{n\to\infty}\frac{|\gamma[1..k_n]|_{\sigma'}}{|\gamma[1..k_n]|_p}.$$

Since $\mathcal{A}$ is 2-deterministic, it can be assumed that all transitions starting at q read symbols from the same tape. The automaton $\mathcal{A}$ can be turned into a selector as follows. Transitions starting at q select the digit they read but all other transitions do not select the digit they read. The previous inquality shows that the output of the obtained selector is not even simply normal. This is a contradiction with the hypothesis. □

We end this section with the following result that shows that independence of two normal words implies independence of one and a word that results from selection of the other.

Proposition 4.7. *Let x and y be normal and independent words. If y' is obtained by oblivious selection from y, then x and y' are still independent.*

Proof. We show that if x and y' are not independent, then x and y are also not independent. We suppose that x and y' are not independent. This means either that x can be compressed with the help of y' or that y' can be compressed with the help of x. Suppose first that x can be compressed by a compressor $\mathcal{C}$ with the help of y'. Combining this compressor with the selector $\mathcal{S}$ which selects y' from y yields a compressor $\mathcal{C}'$ which compresses x with the help y.

Indeed, this compressor $\mathcal{C}'$ skips symbols from y which are not selected by $\mathcal{S}$ and simulates $\mathcal{C}$ on those symbols which are selected by $\mathcal{S}$.

Suppose second that y' can be compressed by a compresor $\mathcal{C}$ with the help of x. We claim that y can also be compressed with the help of x. The selector $\mathcal{S}$ which selects y' from y is used as a splitter to splits y into y' made of the selected symbols and y'' made of the non selected symbols. Then, the compressor $\mathcal{C}$ is used to compress y' with the help of x into a word z. Finally, words z and y'' are merged into a word z' by blocks of length m from same length m. Each block of length of length m contains either m symbols from z or m symbols from y'' plus an extra symbol indicating whether the block contains symbols from z or symbols from y''. The combination of all these automata yields a compressor which compresses y with the help of x. □

4.4 From frequencies of states to shuffling and back

If two normal words x and y are on different alphabets then, in general, their shuffling $\mathcal{S}(x, y)$ is not normal. For instance, if x and y are words on different alphabets their join is not normal. Thus, we assume now a unique alphabet.

Exchanging the input and output tapes of a shuffler $\mathcal{S}$ gives a 1-deterministic 3-automaton that we call the *splitter* corresponding to $\mathcal{S}$. This is due to the very special form of the transitions of shufflers. If the output $z = \mathcal{S}(x, y)$ of the shuffler $\mathcal{S}$ on inputs x and y is fed to the corresponding splitter, the two outputs are x and y. The fact that the corresponding splitter is 1-deterministic yields the following lemma which really requires that the alphabets on the two tapes are equal.

Lemma 4.8. *Let $\mathcal{S}$ be a shuffler and q one of its states. For each finite word w, there is exactly one run of length $|w|$ starting at q and outputting w.*

Proof of Theorem 3.3, (2) implies (4). Suppose x and y are normal. Let γ be the run of the shuffler $\mathcal{S}$ with inputs x and y and let ℓ be a given length. For each state q of $\mathcal{S}$ and each word w of length ℓ, there exists by Lemma 4.8 a unique run $\sigma_{q,w}$ starting at state q and outputting w.

For each word w of length ℓ, the number of occurrences of w in the prefix $z[1..n]$ of z is given by

$$|z[1..n]|_w = \sum_{q \in Q} |\gamma[1..n]|_{\sigma_{q,w}}.$$

We first claim that the run γ reaches a strongly connected component with no leaving transition. Suppose by contradiction that the run remains in a strongly connected component with at least one leaving transition. Let $\mathcal{A}$ be the automaton made of this strongly connected component without the leaving transitions. Therefore, there is no stationary distribution associated with $\mathcal{A}$ and this a contradiction with statement (2). Thus, it can be assumed without loss of generality that $\mathcal{S}$ is strongly connected. The ratio $|z[1..n]|_w/n$ is given by

$$\frac{|z[1..n]|_w}{n} = \sum_{q \in Q} \frac{|\gamma[1..n]|_q}{n} \frac{|\gamma[1..n]|_{\sigma_{q,w}}}{|\gamma[1..n]|_q}.$$

We claim that for any two paths σ and σ' of the same length ℓ and starting at the same state q,

$$\lim_{n\to\infty} \frac{|\gamma[1..k_n]|_\sigma}{|\gamma[1..k_n]|_p} = \lim_{n\to\infty} \frac{|\gamma[1..k_n]|_{\sigma'}}{|\gamma[1..k_n]|_p}.$$

Consider the automaton $\mathcal{S}_\ell$ defined as follows. The state set of $\mathcal{S}_\ell$ is the set of runs of length ℓ in $\mathcal{S}$. There is a transition from a run σ to a run σ' in $\mathcal{S}_\ell$ if there is a transition τ in $\mathcal{S}$ such that σ' is the suffix of length ℓ of $\sigma\tau$. An easy computation shows that the stationary distribution of $\mathcal{S}_\ell$ is given by $\pi_\ell(\sigma) = \pi(q)/2^\ell$ where π is the stationary distribution of $\mathcal{S}$. Applying statement (2) to $\mathcal{S}_\ell$ yields the equality. It follows that the ratio $|z[1..n]|_w/n$ has also a limit which does depend on w. Since this holds for each length ℓ, the infinite word z is normal. □

Proof of Theorem 3.3, (4) implies (1). Suppose that x and y are not independent and x is compressible with the help of y. Let $\mathcal{A}$ be the compressor such that $\rho_{\mathcal{A}}(x/y) < \rho(x)$. Consider the shuffler $\mathcal{S}$ that mimics $\mathcal{A}$ and copies each digit of x (respectively of y) as soon as it is read by $\mathcal{A}$. We claim that $\mathcal{S}(x, y)$ is compressible, hence not normal. For compressing $\mathcal{S}(x, y)$, first define a splitter $\mathcal{S}'$ exchanging the inputs and outputs in the transition of $\mathcal{S}$. Thus, $\mathcal{S}'(\mathcal{S}(x, y)) = (x, y)$. By composing $\mathcal{S}'$ with $\mathcal{A}$ we can compress x using y and obtain a compressed word x'. Let m be the block size used in this compression. Finally, words y and x' are merged into a word z interleaving a block of m symbols from x with a block of m symbols from y. Since the hypothesis ensures x is compressible, so is word z. From this word z we can recover (x', y), from which we can recover (x, y) and then obtain $S(x, y)$, as required. □

5 Construction of a pair of independent normal words

We now present the second main result in this work.

Theorem 5.1. *For every alphabet A, there is an algorithm that computes a pair of independent normal words.*

To prove Theorem 5.1 we give an explicit algorithm based on the characterization of independent normal words in terms of shufflers (Theorem 3.3 statement (4)). The algorithm we present here is an adaptation of Turing's algorithm for computing an absolutely normal number [21, 4]. But instead of computing the expansion of a number that is normal in every integer base here we compute a pair of normal infinite words such that every shuffling of them is normal. We start with auxiliary definitions and some properties. We write log for the logarithm in base e and $\log_b$ for any other base b.

Definition 5.2. 1. For any shuffler $\mathcal{S}$ define the set

$$E_{\mathcal{S}_i}(\varepsilon, \gamma, n) = \left\{(x, y) \in A^\omega \times A^\omega : \left| |\mathcal{S}_i(x, y)[1..n]|_\gamma - n/|A|^{|\gamma|} \right| < \varepsilon n \right\}.$$

2. Assume an enumeration of shufflers $\mathcal{S}_1, \mathcal{S}_2, \ldots$ and define the set

$$F(\varepsilon, t, \ell, n) = \bigcap_{i=1}^{t} \bigcap_{r=1}^{\ell} \bigcap_{\gamma \in |A|^r} E_{\mathcal{S}_i}(\varepsilon, \gamma, n).$$

3. For each positive integer n, let $\ell_n = (\log_{|A|} n)/3$, $t_n = n$ and $\varepsilon_n = 2\sqrt{(\log n \log_{|A|} n)/n}$.

$$F_n = F(\varepsilon_n, t_n, \ell_n, n).$$

Lemma 5.3 (Lemma 8 in [21], adapted from Theorem 148 in [12]). *Let r and n be positive integers. For every real ε such that $6/\lfloor n/r\rfloor \le \varepsilon \le 1/|A|^r$ and for every $\gamma \in A^r$, if $N(\gamma,i,n) = |\{w \in A^n : |w|_\gamma = i\}|$ then*

$$\sum_{0\le i\le n/|A|^r-\varepsilon n} N(\gamma,i,n) + \sum_{n/|A|^r+\varepsilon n\le i\le n} N(\gamma,i,n) < \ 2|A|^{n+2r-2} r e^{-|A|^r\varepsilon^2 n/6r}.$$

For a word $u \in A^*$ we denote by $[u]$ the set of infinite words that start with u, and we call it the cylinder determined by u,

$$[u] = \{x \in A^\omega : x[1..|u|] = u\}.$$

For the cartesian product of two cylinders $[u] \times [v]$ we write $([u],[v])$, and we call the pair of cylinders determined by (u,v).

Proposition 5.4. *For every shuffler $\mathcal{S}$, every n, r, ε such that ε such that $6/\lfloor n/r\rfloor \le \varepsilon \le 1/|A|^r$ and every $\gamma \in A^r$,*

$$\mu(E_{\mathcal{S}}(\varepsilon,\gamma,n)) > 1 - 2|A|^{2r-2} r e^{-|A|^r\varepsilon^2 n/6r}.$$

Proof. Consider the set

$$P(\varepsilon,\gamma,n) = \left\{ w \in A^n : \Big| |w|_\gamma - n/|A|^{|\gamma|} \Big| < \varepsilon n \right\}.$$

Then,

$$\begin{aligned} E_{\mathcal{S}}(\varepsilon,\gamma,n) &= \bigcup_{w\in P(\varepsilon,\gamma,n)} \{([u],[v]) : |u|+|v| = n \text{ and } \forall x \in [u] \forall y \in [v],\ \mathcal{S}(x,y) \in [w]\} \\ &= \bigcup_{w\in P(\varepsilon,\gamma,n)} \mathcal{S}^{-1}([w]). \end{aligned}$$

Thus,

$$\mu(E_{\mathcal{S}}(\varepsilon,\gamma,n)) = \sum_{w\in P(\varepsilon,\gamma,n)} \mu(\mathcal{S}^{-1}([w])) = |P(\varepsilon,\gamma,n)|\ |A|^{-n}.$$

Finally, Lemma 5.3 gives the needed upper bound for $|\overline{P}(\varepsilon,\gamma,n)|$. □

For any set B we write $\overline{B}$ to denote its complement.

Proposition 5.5. *For any ε, t, ℓ and n, such that $6/\lfloor n/\ell\rfloor \le \varepsilon \le 1/|A|^\ell$,*

$$\mu(F(\varepsilon,t,\ell,n)) > 1 - 2t|A|^{3\ell-1} e^{-\varepsilon^2 n/(3\ell)}.$$

Proof. By Defininition 5.2,

$$\mu(\overline{F}(\varepsilon,t,\ell,n)) \le \sum_{i=1}^{t}\sum_{r=1}^{\ell}\sum_{\gamma\in A^r} \mu(\overline{E_{\mathcal{S}_i}}(\varepsilon,\gamma,n)).$$

The number of terms of this triple sum is bounded by

$$\sum_{i=1}^{t}\sum_{r=1}^{\ell}\sum_{\gamma\in A^r} 1 = \sum_{i=1}^{t}\sum_{r=1}^{\ell} |A|^r < \sum_{i=1}^{t} \frac{|A|^{\ell+1}-1}{|A|-1} < \sum_{i=1}^{t} |A|^{\ell+1} = t|A|^{\ell+1}.$$

From the lower bound given in Proposition 5.4 we obtain that for every shuffler $\mathcal{S}$ and for every word $\gamma \in A^{\leq \ell}$,

$$\mu(\overline{E_{\mathcal{S}}}(\varepsilon, \gamma, n)) < 2|A|^{2\ell-2}\ell e^{-\varepsilon^2 n/(3\ell)}.$$

Therefore,

$$\mu(\overline{F}(\varepsilon, t, \ell, n)) < 2t|A|^{3\ell-1}e^{-\varepsilon^2 n/(3\ell)}. \qquad \square$$

Recall that Definition 5.2 defines $\ell_n = (\log_{|A|} n)/3$, $t_n = n$, $\varepsilon_n = 2\sqrt{(\log n \log_{|A|} n)/n}$ and $F_n = F(\varepsilon_n, t_n, \ell_n, n)$.

Proposition 5.6. *Let $n_{\text{start}} = \min\{n : \varepsilon_n \geq 6/\lfloor n/\ell_n \rfloor\}$. Then for every $n \geq n_{\text{start}}$, $\ell_n, t_n \geq 1$,*

$$\mu(F_n) \geq 1 - 1/n^2.$$

Proof. To apply Proposition 5.5 it is required that $6/\lfloor n/\ell_n \rfloor \leq \varepsilon_n \leq 1/|A|^{\ell_n}$. Let n_{start}= $\min\{n : \varepsilon_n \geq 6/\lfloor n/\ell_n \rfloor\}$. Then, for every $n \geq n_{start}$ the required inequality holds. So, Application of Proposition 5.5 yields

$$\begin{aligned}
\mu(\overline{F_n}) &\leq 2t_n|A|^{3\ell_n-1}e^{-\varepsilon^2 n/(3\ell_n)} \\
&\leq t_n\ |A|^{3\ell_n}e^{-\varepsilon^2 n/(3\ell_n)} \\
&= n|A|^{(\log_{|A|} n)}e^{-4n(\log n)(\log_{|A|} n)/(n \log_{|A|} n)} \\
&= n^2\ e^{-4\log n} \\
&= \frac{1}{n^2}. \qquad \square
\end{aligned}$$

If n_{start} is as determined by Proposition 5.6, then $\bigcap_{n \geq n_{start}} F_n$ is not empty and consists just of pairs of independent normal words. We can actually show that the intersection of a subsequence of F_n's with n growing at most exponentially, also consists just of pairs of independent normal words. The next definition fixes n_0 as $\log n_{start}$ and defines the sets G_n that will be used in the proof of Theorem 5.1.

Definition 5.7. Let $n_0 = \log_{|A|} \min\{n : \varepsilon_n \geq 6/\lfloor n/\ell_n \rfloor\}$. We define a sequence $(G_n)_{n\geq 0}$ of finite sets of pairs of cylinders in $A^\omega \times A^\omega$, such that for every n, $G_{n+1} \subseteq G_n$ and

$$G_n = \bigcap_{j=0}^{n} F_{|A|^{n_0+j}}$$

Lemma 5.8. *The set $\bigcap_{n\geq 0} G_n$ consists exclusively of independent normal words.*

Proof. Fix n_0 as defined in Definition 5.7. Suppose $(u, v) \in \bigcap_{n\geq 0} G_n$. To show that u and v are independent we show that for any shuffler $\mathcal{S}$, $\mathcal{S}(u, v)$ is a normal sequence. Fix a finite word $w \in A^*$. Pick m_0 such that if i is the index of $\mathcal{S}$ in the enumeration of shufflers, $t_{m_0} \geq i$, $\ell_{m_0} \geq |w|$, $m_0 \geq n_0$ and $\varepsilon_{m_0} < 1/|A|^{|w|}$.

Let's see that for any m greater than m_0 the following holds. Let k be such that $|A|^k \leq m < |A|^{k+1}$. Then, using that $(u, v) \in F_{|A|^{k+1}}$,

$$\begin{aligned}\frac{|\mathcal{S}(u,v)[1..m]|_w}{m} &< \frac{|\mathcal{S}(u,v)[1..|A|^{k+1}]|_w}{m}\\ &< \frac{1}{m}|A|^{k+1}\left(\frac{1}{|A|^{|w|}} + \varepsilon_{m_0}\right)\\ &\leq \frac{|A|^{k+1}}{|A|^k}\frac{2}{|A|^{|w|}}\\ &= \frac{2|A|}{|A|^{|w|}}.\end{aligned}$$

This implies that

$$\limsup_{m\to\infty} \frac{|\mathcal{S}_i(u,v)[1..m]|_w}{m} < \frac{2|A|}{|A|^{|w|}}.$$

We conclude that $\mathcal{S}(u, v)$ is normal applying Theorem 4.6 in [8] which establishes that a word x is normal if, and only if, there exists a positive number C such that for every finite word w,

$$\limsup_{m\to\infty} \frac{|x[1..m]|_w}{m} \leq \frac{C}{|A|^{|w|}}.$$

Hence, taking C equal to $2|A|$ we obtain that $S(u, v)$ is normal. Now we prove that both, u and v, are normal too. Consider the selector $\mathcal{S}'$ defined as the splitter that reverses $\mathcal{S}$ and then ignores the second output tape. That is, if $\mathcal{S}(u, v) = z$ then $\mathcal{S}'(z) = u$. Since $\mathcal{S}(u, v)$ is normal, by Agafonov's theorem u is normal. A similar argument proves that v is also normal. We proved that every $(u, v) \in \bigcap_{n\geq 0} G_n$ is a pair of normal words satisfying that for every shuffler $\mathcal{S}$, $\mathcal{S}(u, v)$ is normal. By Theorem 3.3 item (4), u, v are independent. □

Proof of Theorem 5.1. For clarity we present the proof for the alphabet $A = \{0, 1\}$, hence $|A| = 2$. It is straightforward transfer the proof to any alphabet of an arbitrary size. We prove that Algorithm 5.9 constructs of a pair of independent normal words. From the algorithm is immediate that the sequence $(I_n)_{n\geq 0}$ is such that for every n, $I_{n+1} \subset I_n$, and $\mu(I_{n+1}) = \mu(I_n)/2$. We show that for every n, $\mu(I_n \cap G_n) > 0$. We prove by induction that for every n,

$$\mu(G_n \cap I_n) > 2^{-2n-1}.$$

For the base case, $n = 0$, $\mu(G_0 \cap I_0) = 1 > 2^{-1}$. For the inductive step, $n + 1$, since

$$\mu(\overline{F_{2^{n_0+n+1}}}) < \frac{1}{(2^{n_0+n+1})^2} = 2^{-2(n_0+n+1)} < 2^{-2(n+1)},$$

we have

$$\begin{aligned}\mu(G_{n+1} \cap I_n) &= \mu(G_n \cap I_n \cap F_{2^{n_0+n+1}})\\ &> 2^{-2n-1} - 2^{-2(n+1)}\\ &= 2^{-2(n+1)}.\end{aligned}$$

Proof of Theorem 5.1

Algorithm: Construction of a pair of normal independent words using shufflers
Input: No input
Output: A sequence $I_n = ([u_n],[v_n])_{n\geq 0}$, such that $u_n, v_n \in \{0,1\}^*$, $|u_n| + |v_n| = n$ and $\bigcap_{i\geq 0} I_n$ contains a unique pair (u,v) of independent normal words.

Let $\mathcal{S}_1, \mathcal{S}_2, \ldots$ be a enumeration of shufflers.
For each $n \geq 1$, let $\ell_n = (\log n)/3$, $\varepsilon_n = 2\sqrt{(\log n \log_2 n)/n}$ and

$$F_n = \bigcap_{i=1}^{n} \bigcap_{\gamma\in 2^{\leq \ell_n}} E_{\mathcal{S}_i}(\varepsilon_n,\gamma,n), \text{ where}$$

$$E_{\mathcal{S}_i}(\varepsilon_n,\gamma,n) = \{(x,y)\in\{0,1\}^\omega \times \{0,1\}^\omega : \left| \, |\mathcal{S}_i(x,y)[1..n]|_\gamma - n/2^{|\gamma|} \right| < n\varepsilon_n\}.$$

Let $n_0 = \log_2 \min\{n : \varepsilon_n \geq 6/\lfloor n/\ell_n \rfloor\}$. We write λ for the empty word.

```
begin
    $n \leftarrow 0$
    $I_0 \leftarrow ([\lambda],[\lambda])$
    $G_0 \leftarrow ([\lambda],[\lambda])$
    repeat
        $([u_n],[v_n]) \leftarrow I_n$
        if $n$ is even then
            $I_n^0 \leftarrow ([u_n0],[v_n])$
            $I_n^1 \leftarrow ([u_n1],[v_n])$
        else
            $I_n^0 \leftarrow ([u_n],[v_n0])$
            $I_n^1 \leftarrow ([u_n],[v_n1])$
        $G_{n+1} \leftarrow G_n \cap F_{2^{n_0+n+1}}$;
        if $\mu(I_n^0 \cap G_{n+1}) > 2^{-2n+1}$ then
            $I_{n+1} \leftarrow I_n^0$
        else
            $I_{n+1} \leftarrow I_n^1$
        print $I_{n+1}$
        $n \leftarrow n+1$
    forever
end
```

Algorithm 5.9: Construction of a pair of normal independent words using shufflers

Then, at least one of $G_{n+1} \cap I_n^0$ and $G_{n+1} \cap I_n^1$ must have measure greater than $2^{-2(n+1)-1}$, as required. Since $(I_n)_{n\geq 0}$ is a nested sequence of intervals of strictly decreasing but positive measure, and for every n, $\mu(G_n \cap I_n) > 0$, we conclude that

$$\bigcap_{n\geq 0} I_n = \bigcap_{n\geq 0} G_n \cap I_n$$

contains a unique pair (u, v). And by Lemma 5.8 all the elements in $\bigcap_{n\geq 0} G_n$ are pairs of independent normal words. This concludes the proof. □

5.1 Computational complexity

Algorithm 5.9 computes a sequence $(I_n)_{n\geq 0}$ of pairs of cylinders in $\{0,1\}^\omega \times \{0,1\}^\omega$ such that $\bigcap_{i\geq 0} I_n$ contains a unique pair (u, v) of independent words. We now establish its computational complexity.

Proposition 5.10. *Algorithm 5.9 has doubly exponential complexity: to output n symbols of the independent normal words u and v the algorithm performs a number of mathematical operations that is doubly exponential in n.*

Proof. As in Turing's original construction, the complexity of each step of our algorithm is dominated by the computation of the set $F_{n_0+2^{2^n+1}}$, which is doubly exponential. Notice that the measures of the inspected sets can be calculated in simply exponential time, and the rest of the computation takes constant time.

The construction works by taking a sequence of "good sets" $(G_n)_{n\geq 0}$ and a sequence $(I_n)_{n\geq 0}$ of pairs of cylinders in $\{0,1\}^\omega \times \{0,1\}^\omega$. For the initial step, $n = 0$, $\mu(G_0) = 1, \mu(I_0) = 1$, and $\mu(G_0 \cap I_0) = 1$. For subsequent steps, we refine G_n into G_{n+1} and choose one suitable half of I_n to be I_{n+1}. We now find out the length s_n of the shuffling that need to be inspected at step n of the algorithm. At step n, $G_{n+1} = G_n \cap F_{s_n}$ and $\mu(G_{n+1}) \geq \mu(G_n) - \mu(\overline{F_{s_n}})$. The algorithm chooses the half of I_n whose intersection with G_{n+1} is at least $(\mu(G_n) - \mu(\overline{F_{s_n}}))/2$. We need that for each n, this measure is positive:

$$\Big((((\mu(G_0) - \mu(\overline{F_{s_0}}))/2 - \mu(\overline{F_{s_1}}))/2 - \mu(\overline{F_{s_2}}))/2 \ldots - \mu(\overline{F_{s_{n-1}}})\Big)/2 > 0$$

$$2^{-n} - 2^{-(n-1)}\mu(\overline{F_{s_0}}) - \ldots - 2^{-1}\mu(\overline{F_{s_{n-1}}}) > 0$$

Multiplying by 2^n

$$1 - 2\mu(\overline{F_{s_0}}) - \ldots - 2^{n-1}\mu(\overline{F_{s_{n-1}}}) > 0$$

$$\sum_{n=1}^{\infty} 2^n \mu(\overline{F_{s_{n-1}}}) < 1.$$

Therefore, we require $\sum_{n=1}^{\infty} 2^n \mu(\overline{F_{s_{n-1}}}) < 1$ while Proposition 5.6 establishes that $\mu(\overline{F_{s_{n-1}}}) < 1/s_{n-1}^2$. Thus, we require $s_{n-1} \geq 2^n$, which shows the needed exponential growth in the index of the sets F_{s_n}. Notice that the algorithm fixes $s_n = 2^{n+1}$. and the computation of the set F_{s_n} requires the inspection of 2^{s_n} words of length s_n. Then at step n the algorithm performs a number of operations that is doubly exponential in n. Finally notice that at step n the algorithm outputs n symbols in the form of two words u_n, v_n, such that $|u| + |v| = n$. □

6 Conclusion

As a conclusion, we would like to mention a few open problems. The following questions remain to be investigated.

1. The characterization of finite-state independence of normal words given in Theorem 3.3 uses deterministic finite automata with no extra memory (counters, stack). Determine if the same characterization holds for the non-deterministic version of the same finite automata. We have pursued this line of investigation in [2, 9] for the characterization of normality in terms of incompressibility by finite-automata and essentially we found that, without extra memory, non-determinism does not add compressibility power.
2. Give a purely combinatorial characterization of finite-state independence of normal words. We aim at a condition on the two sequences that is defined in combinatorial terms, without mentioning automata (in the same way that the definition of normality can be stated in terms of frequency of blocks).
3. There are efficient algorithms that compute absolutely normal numbers with nearly quadratic complexity as [6] or, as recently announced, in poly-logarithmic linear complexity [15]. It may be possible to adapt those algorithms to efficiently compute a pair of independent normal sequences.
4. Construct a normal word that is finite-state independent to some given normal word. That is, given a word that has been proved to be normal, as Champernowne's word, we aim to construct another normal word that is finite-state independent to it.

Acknowledgements. The authors are members of the Laboratoire International Associé INFINIS, CONICET/Universidad de Buenos Aires–CNRS/Université Paris Diderot. The present paper was completed while Becher visited the Erwin Schrödinger International Institute for Mathematics and Physics, Austria. Becher is supported by the University of Buenos Aires and CONICET.

References

[1] V. N. Agafonov. Normal sequences and finite automata. *Soviet Mathematics Doklady*, 9:324–325, 1968.

[2] V. Becher, O. Carton, and P. A. Heiber. Normality and automata. *Journal of Computer and System Sciences*, 81(8):1592–1613, 2015.

[3] V. Becher, O. Carton, and P. A. Heiber. Finite-state independence. arXiv:1611.03921. Submitted, 2016.

[4] V. Becher, S. Figueira, and R. Picchi. Turing's unpublished algorithm for normal numbers. *Theoretical Computer Science*, 377(1-3):126–138, 2007.

[5] V. Becher and P. A. Heiber. Normal numbers and finite automata. *Theoretical Computer Science*, 477:109–116, 2013.

[6] V. Becher, P.A. Heiber, and T. Slaman. A polynomial-time algorithm for computing absolutely normal numbers. *Information and Computation*, 232:1–9, 2013.

[7] É. Borel. Les probabilités dénombrables et leurs applications arithmétiques. *Rendiconti del Circolo Matematico di Palermo*, 27:247–271, 1909.

[8] Y. Bugeaud. *Distribution Modulo One and Diophantine Approximation.* Series: Cambridge Tracts in Mathematics 193. Cambridge University Press, 2012.

[9] O. Carton and P. A. Heiber. Normality and two-way automata. *Information and Computation*, 241:264–276, 2015.

[10] D. Champernowne. The construction of decimals normal in the scale of ten. *J. London Math. Soc.*, s1-8(4):254–260, 1933.

[11] J. Dai, J. Lathrop, J. Lutz, and E. Mayordomo. Finite-state dimension. *Theoretical Computer Science*, 310:1–33, 2004.

[12] G. H. Hardy and E. M. Wright. *An introduction to the theory of numbers.* Oxford University Press, Oxford, sixth edition, 2008.

[13] T. Kamae and B. Weiss. Normal numbers and selection rules. *Israel Journal of Mathematics*, 21(2):101–110, 1975.

[14] L. Kuipers and H. Niederreiter. *Uniform distribution of sequences.* Wiley-Interscience, New York, 1974.

[15] J. Lutz and E. Mayordomo. Computing absolutely normal numbers in nearly linear time. arXiv:1611.05911, 2016.

[16] D. Perrin and J.-É. Pin. *Infinite Words.* Elsevier, 2004.

[17] J.-E. Pin. *Relational morphisms, transductions and operations on languages*, pages 34–55. Springer Berlin Heidelberg, Berlin, Heidelberg, 1989.

[18] J. Sakarovitch. *Elements of automata theory.* Cambridge University Press, 2009.

[19] C. P. Schnorr and H. Stimm. Endliche automaten und zufallsfolgen. *Acta Informatica*, 1:345–359, 1972.

[20] E. Senata. *Non-negative Matrices ans Markov Chains.* Springer, 2006.

[21] A. Turing. A note on normal numbers. In J.L.Britton, editor, *Collected Works of A.M. Turing: Pure Mathematics*, pages 117–119. North Holland, Amsterdam, 1992. with notes of the editor in 263–265.

[22] D. D. Wall. *Normal Numbers.* PhD thesis, University of California, Berkeley, California, 1949.

Nicolás Alvarez
ICIC - Universidad Nacional del Sur, CONICET
Departamento de Ciencias en Ingeniería de la Computación
naa@cs.uns.edu.ar

Verónica Becher
Departmento de Computación, Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires & ICC, CONICET, Argentina.
vbecher@dc.uba.ar

Olivier Carton
Institut de Recherche en Informatique Fondamentale
Université Paris Diderot
Olivier.Carton@irif.fr