Finite-state independence

Verónica Becher Olivier Carton Pablo Ariel Heiber

October 16, 2017

Abstract

In this work we introduce a notion of independence based on finite-state automata: two infinite words are independent if no one helps to compress the other using one-to-one finite-state transducers with auxiliary input. We prove that, as expected, the set of independent pairs of infinite words has Lebesgue measure 1. We show that the join of two independent normal words is normal. However, the independence of two normal words is not guaranteed if we just require that their join is normal. To prove this we construct a normal word $x_1x_2x_3\ldots$ where $x_{2n} = x_n$ for every n. This construction has its own interest.

1 Introduction

In this work we introduce a notion of independence for pairs of infinite words, based on finite-state automata. We call it *finite-state independence*.

The concept of independence appears in many mathematical theories, formalizing that two elements are independent if they have no common parts. In classical probability theory the notion of independence is defined for random variables. In the case of random variables with finite range the notion of independence can be reformulated in terms of Shannon entropy function: two random variables are independent if the entropy of the pair is exactly the sum of the individual entropies. An equivalent formulation says that two random variables are independent if putting one as a condition does not decrease the entropy of the other one. In algorithmic information theory the notion of independence can be defined for finite objects using program-size (Kolmogorov–Chaitin) complexity. Namely, finite words x and y are independent if the program-size complexity of the pair (x, y) is close to the sum of program-size complexities of x and y. Equivalently, up to a small error term, two finite words are independent if the program-size complexity of one does not decrease when we allow the other as an oracle.

Algorithmic information theory also defines the notion of independence for random infinite words, as follows. Recall that, according to Martin-Löf's definition, an infinite word is random if it does not belong to any effectively null set; an equivalent characterization establishes that an infinite word is random if its prefixes have nearly maximal program-size complexity, which means that they are incompressible with Turing machines. Two random infinite words $x_1x_2\ldots$ and $y_1y_2\ldots$ are independent if their join $x_1y_1x_2y_2\ldots$ is random [23, 1], see also [17, Theorem 3.4.6] and [14] for independence on stronger notions of randomness. An equivalent definition establishes that two random infinite words are independent if the program-size complexity of the initial segment of one, conditioned on the other one, is nearly maximal. This means that one word remains incompressible even when using the other one as an oracle. See [16, 10, 22] for a thorough presentation of this material. While the notion of independence for random infinite words is well understood, algorithmic information theory has not provided a fully satisfactory definition of independence for arbitrary infinite words, see the discussion in [7].

Here we scale down the notion of independence given by algorithmic information theory by considering incompressibility by finite-state automata instead of incompressibility by Turing machines. Our definition builds on the theory of finite-state compression ratio introduced by Dai, Lathrop, Lutz and Mayordomo [9]. The finite-state compression ratio of an infinite word indicates how much it can be compressed by one-to-one finite-state transducers, which are finite-state

automata augmented with an output transition function such that the automata input–output behaviour is one-to-one (Huffman [12] called them lossless finite-state compressors). The infinite words that can not be compressed are exactly the Borel normal words (this result was first known from combining [20] and [9], see [4] for a direct proof). We say that two infinite words are finite-state independent if one does not help to compress the other using finite-state transducers. In Theorem 5.1 we show that the set of finite-state independent pairs of infinite words has Lebesgue measure 1, giving an elementary proof.

As expected, the join of two finite-state independent normal words is normal (Theorem 4.1). However, independence of two normal words is not guaranteed if we just require that their join is normal. To show this we construct a normal word x that is equal to the word $\text{even}(x)$ that consists of the symbols of x at even positions (Theorem 4.4). Thus, if $\text{odd}(x)$ consists of the symbols of x at odd positions, both $\text{odd}(x)$ and $\text{even}(x)$ are normal, and their join is normal. But $\text{odd}(x)$ and $\text{even}(x)$ are not independent: $\text{odd}(x)$ equals $\text{odd}(\text{even}(x))$. This phenomenon is not isolated: Alexander Shen (personal communication, August 2016) proved that for the set of words x such that $x = \text{even}(x)$, a word is normal with probability 1.

The notion of finite-state independence we present here is based just on deterministic finite-state transducers. It remains to investigate if non-deterministic finite-state transducers operating with an oracle can achieve different compression ratios. In the case of finite-state transducers with no oracle, it is already known that the deterministic and the non-deterministic models compress exactly the same words, namely, the non-normal words [3]. Some other models also compress exactly the same words, such as the finite-state transducers with a single counter [3] and the two-way transducers [8]. It is still unknown if there is a deterministic push-down transducer that can compress some normal words.

It also remains the question of how to characterize finite-state independence other than by the conditional compression ratio in finite-state automata. One would like a characterization in terms of a complexity function based on finite automata as those considered in [21] and [13].

2 Primary definitions

Let A be a finite set of symbols, the alphabet. We write A^ω for the set of all infinite words over A and A^k stands for the set of all words of length k. The length of a finite word w is denoted by $|w|$. The positions in finite and infinite words are numbered starting from 1. To denote the symbol at position i of a word w we write $w[i]$ and to denote the subword of w from position i to j we write $w[i..j]$. We use the customary notation for asymptotic growth of functions saying that $f(n)$ is in $O(g(n))$ if $\exists k > 0\ \exists n_0\ \forall n > n_0,\ |f(n)| \leq k|g(n)|$.

2.1 Normality

A presentation of the definitions and basic results on normal sequences can be read form [2]. Here we start by introducing the number of *occurrences* and the number of *aligned occurrences* of a word u in a word w.

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

$$|w|_u = |\{i : w[i..i + |u| - 1] = u\}|,$$
$$\|w\|_u = |\{i : w[i..i + |u| - 1] = u \text{ and } i \equiv 1 \bmod |u|\}|.$$

For example, $|aaaaa|_{aa} = 4$ and $\|aaaaa\|_{aa} = 2$. Aligned occurrences are obtained by cutting w in $|u|$-sized pieces starting from the left. Notice that the definition of aligned occurrences has the condition $i \equiv 1 \bmod |u|$ (and not $i \equiv 0 \bmod |u|$), because the positions are numbered starting from 1. Of course, when a word u is just a symbol, $|w|_u$ and $\|w\|_u$ coincide. Aligned occurrences can be seen as symbol occurrences using a power alphabet: if w is a word whose length is a multiple of r, then w can be considered as a word $\pi(w)$ over A^r by grouping its symbols into blocks of

length r. The aligned occurrences of a word u of length r in w then correspond to the occurrences of the symbol $\pi(u)$ in the word $\pi(w)$, and $\|w\|_u = |\pi(w)|_{\pi(u)}$.

We recall the definition of Borel normality [5] for infinite words (see the books [6, 15] for a complete presentation). An infinite word x is *simply normal* to word length ℓ if all the blocks of length ℓ have asymptotically the same frequency of aligned occurrences, i.e., if for every $u \in A^\ell$,

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

An infinite word x is *normal* if it is simply normal to every length. Normality is defined here in terms of aligned occurrences but it could also be defined in terms of all occurrences. The equivalence between the two definitions requires a proof (see Theorem 4.5 in [6]).

2.2 Automata

We consider *k-tape automata*, also known as *k-tape transducers* when k is greater than 1 [18, 19]. We call them k-automata and we consider them for k equal to 1, 2, or 3. A k-automaton is a tuple $\mathcal{T} = \langle Q, A, \delta, I\rangle$, where Q is the finite state set, A is the alphabet, δ is the transition relation, I is the set of initial states. The transition relation is a subset of $Q \times (A \cup \{\varepsilon\})^k \times Q$. 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 of the alphabet or the empty word. A transition is written as $p \xrightarrow{\alpha_1,\ldots,\alpha_k} q$. As usual, two transitions are called *consecutive* if the ending state of the first is the starting state of the second.

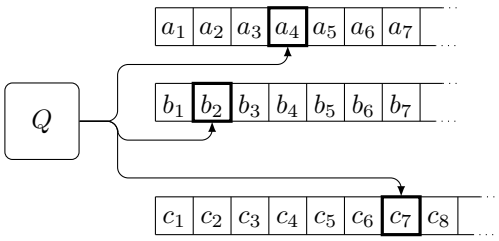

Figure 1: A 3-automaton and its tapes

.

An infinite run is an infinite sequence of consecutive transitions

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

The label of the run is the component-wise concatenation of the labels of the transitions given by the tuple $\langle x_1, \ldots, x_k\rangle$ where each x_j for $1 \le j \le k$ 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 as $q_0 \xrightarrow{x_1,\ldots,x_k} \infty$. The tuple $\langle x_1, \ldots, x_k\rangle$ is accepted if there exists at least one accepting run with label $\langle x_1, \ldots, x_k\rangle$. The 1-automata are the usual automata with ε-transitions and the 2-automata are the usual automata with input and output also known as transducers.

In this work we consider only deterministic k-automata. We actually consider k-automata where the transition is determined by a subset of the k tapes. Informally, a k-automaton is ℓ-deterministic, for $1 \le \ell \le k$, if the run is entirely determined by the contents of the first ℓ tapes. More precisely, a k-automaton is ℓ-deterministic if the following two conditions are fulfilled,

- the set I of initial states is a singleton set;
- for any two transitions $p \xrightarrow{\alpha_1,\ldots,\alpha_k} q$ and $p' \xrightarrow{\alpha'_1,\ldots,\alpha'_k} q'$ with $p = p'$,
  if $\alpha_j = \varepsilon$ for some $1 \le j \le \ell$, then $\alpha'_j = \varepsilon$
  if $\alpha_1 = \alpha'_1, \ldots, \alpha_\ell = \alpha'_\ell$, then $\alpha_{\ell+1} = \alpha'_{\ell+1}, \ldots, \alpha_n = \alpha'_n$ and $q = q'$.

The conditions on the transitions leaving a state p are the following. The first one requires that, among the first ℓ components, the ones with empty label are the same for all transitions leaving p. This means that each state determines (among the first ℓ tapes) the tapes from which a symbol is read (the ones with a symbol as label) and the tapes from which no symbol is read (the ones with empty label). The second condition is the usual one stating that two transitions leaving p and with the same labels in the first ℓ components must be the same.

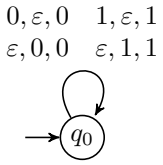

Figure 2: A non-deterministic 3-automaton for the shuffle.

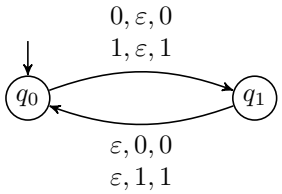

Figure 3: A 2-deterministic 3-automaton for the join.

Figures 2 and 3 show 3-automata that accept a triple $\langle x, y, z\rangle$ of infinite words over the alphabet $\{0, 1\}$. In Figure 2 the tuple $\langle x, y, z\rangle$ is accepted if z is a shuffle of x and y. In Figure 3 the tuple $\langle x, y, z\rangle$ is accepted if z is the join of x and y (the join of two infinite words $x = a_1a_2a_3\cdots$ and $y = b_1b_2b_3\cdots$ is the infinite word $z = a_1b_1a_2b_2a_3\cdots$.) The 3-automaton in Figure 3 is 2-deterministic but the one in Figure 2 is not because the two transitions $q_0 \xrightarrow{0,\varepsilon,0} q_0$ and $q_0 \xrightarrow{\varepsilon,0,0} q_0$ violate the required condition.

We assume each ℓ-deterministic k-automata $\mathcal{T}$ computes a partial function $(A^\omega)^\ell \to (A^\omega)^{k-\ell}$. Let $\mathcal{T}$ 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 $\mathcal{T}$. The automaton $\mathcal{T}$ 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 $\mathcal{T}(x_1, \ldots, x_\ell)$. The 1-deterministic 2-automata are also called *sequential transducers* in the literature. When a k-automaton is ℓ-deterministic, we write the transition as $p \xrightarrow{\alpha_1,\ldots,\alpha_\ell|\beta_{\ell+1},\ldots,\beta_k} q$ to emphasize the fact that the first ℓ tapes are input tapes and that the $k-\ell$ remaining ones are output tapes.

We comment here on transitions reading no symbol from the input tapes. Let $\mathcal{A}$ be a ℓ-deterministic k-automaton. Suppose that there exists a transition from state p to state q whose label has the form $\langle \varepsilon, \ldots, \varepsilon, \beta_{\ell+1}, \ldots, \beta_k\rangle$ with empty labels in the first ℓ components. Let us call these such a transition an ε^ℓ-transition. We claim that it is always possible to get rid of ε^ℓ-transitions. The automaton $\mathcal{A}$ being deterministic implies that this transition is the only transition leaving state p. If there exists a cycle made of ε^ℓ-transitions, this cycle is a dead end in the automaton and its states can be removed without changing the set of accepting runs of $\mathcal{A}$. Let us recall that it is required that all labels of an accepting state to be be infinite. Assume now that there is no cycle of such ε^ℓ-transitions. Removing the transition $p \xrightarrow{\varepsilon,\ldots,\varepsilon|\beta_{\ell+1},\ldots,\beta_k} q$ and adding a transition $p \xrightarrow{\alpha_1,\ldots,\alpha_\ell|\beta_{\ell+1}\gamma_{\ell+1},\ldots,\beta_k\gamma_k} r$ for each transition $q \xrightarrow{\alpha_1,\ldots,\alpha_\ell|\gamma_{\ell+1},\ldots,\gamma_k} r$ leaving q preserve the accepting paths and decrease the number of ε^ℓ-transitions. Completing the process until no ε^ℓ-transition remains remove all of them. In the rest of the paper, we always assume that ε^ℓ-transitions have been removed.

Let $\mathcal{T}$ be a 1-deterministic 2-automaton. To define the compression ratio of an infinite word $x = a_1a_2\ldots$ by $\mathcal{T}$, denoted by $\rho_{\mathcal{T}}(x)$, consider the unique accepting run

$$q_0 \xrightarrow{a_1|v_1} q_1 \xrightarrow{a_2|v_2} q_2 \xrightarrow{a_3|v_3} q_3 \cdots$$

of $\mathcal{T}$, where each a_i is a symbol in A, and each v_i is a finite word (possibly empty) of symbols in A. Then,

$$\rho_{\mathcal{T}}(x) = \liminf_{n\to\infty} \frac{|v_1v_2\cdots v_n|}{|a_1a_2\cdots a_n|}.$$

For a given infinite word x it may happen that for some automata $\mathcal{T}$, $\rho_{\mathcal{T}}(x)$ is greater than 1 and for some other automaton $\mathcal{T}'$, $\rho_{\mathcal{T}'}(x)$ is less than 1. We say that an infinite word x is *compressible* by a 1-deterministic 2-automaton $\mathcal{T}$ if $\rho_{\mathcal{T}}(x) < 1$. A 1-deterministic 2-automaton $\mathcal{T}$ is called *one-to-one* if the function which maps x to $\mathcal{T}(x)$ is one-to-one. The *compression ratio* of an infinite word x is the infimum of the compression ratios achievable by all one-to-one 1-deterministic 2-automata:

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

This compression ratio $\rho(x)$ is always less or equal to 1, as witnessed by the one-to-one compressor $\mathcal{C}_0$ which copies each symbol of the input to the output. For each infinite word x, the compression ratio $\rho_{\mathcal{C}_0}(x)$ is equal to 1.

The sequence $x = 0^\omega$, made entirely of zeros, has compression ratio $\rho(x) = 0$. This is because for each positive real number ε, there exists a compressor $\mathcal{C}$ such that $\rho_{\mathcal{C}}(x) < \varepsilon$. However, the compression ratio 0 is not achievable by any one-to-one 1-deterministic 2-automaton $\mathcal{C}$ for the following reason. Every such automaton $\mathcal{C}$ computes a function $A^\omega \to A^\omega$, and the compression ratio by $\mathcal{C}$ is the ratio between the output and the input in the cycle reached by the infinite run. If the input word x is ultimately periodic as for $x = 0^\omega$ and x is in the domain of $\mathcal{C}$ then the run on x is also ultimately periodic, and the compression ratio of x by $\mathcal{C}$ is non-zero. On the other extreme, the words with compression ratio equal to 1 are exactly the normal words.

3 Finite-state independence

To define the notion of finite-state independence we introduce the notion of conditional compression ratio. The conditional compression ratio of an infinite word x with respect to another infinite word y is the ratio of compression of x when y is used as an oracle. To define this notion, we consider 2-deterministic 3-automata such that two input tapes contain the words x and y and the output tape contains the result of the compression of x with the help of y.

A *compressor* is a 2-deterministic 3-automata $\mathcal{C}$ such that for any fixed infinite word y, the function $x \mapsto \mathcal{C}(x, y)$ which maps x to the output $\mathcal{C}(x, y)$ is one-to-one. This guarantees that, if y is known, x can be recovered from $\mathcal{C}(x, y)$. Note that we do *not* require that the function $(x, y) \mapsto \mathcal{C}(x, y)$ is one-to-one, which would be a much stronger requirement.

Definition 3.1. Let $\mathcal{C}$ be a compressor. The *conditional compression ratio* of an infinite word x with respect to y for $\mathcal{C}$ is determined by the unique accepting run

$$q_0 \xrightarrow{\alpha_1,\beta_1|w_1} q_1 \xrightarrow{\alpha_2,\beta_2|w_2} q_2 \xrightarrow{\alpha_3,\beta_3|w_3} q_3 \cdots$$

such that $x = \alpha_1\alpha_2\alpha_3\cdots$ and $y = \beta_1\beta_2\beta_3\ldots$, with each $\alpha_i, \beta_i \in (A \cup \epsilon)$, as

$$\rho_{\mathcal{C}}(x/y) = \liminf_{n\to\infty} \frac{|w_1w_2\cdots w_n|}{|\alpha_1\alpha_2\cdots\alpha_n|}.$$

Notice that the number of symbols read from y, the length of $\beta_1\beta_2\cdots\beta_n$, is *not* taken into account when defining $\rho_{\mathcal{C}}(x/y)$.

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

Notice that the plain compression ratio $\rho(x)$ and the conditional compression ratio $\rho(x/y)$ always exist and they are values between 0 and 1 (witnessed by the identity function).

The following proposition gives sufficient conditions for the maximum compression, when the conditional compression ratio is equal to zero.

Proposition 3.2. *If a function f is realizable by a 1-deterministic 2-automata then, for every x, $\rho(f(x)/x) = 0$.*

Proof. Assume that the function f is realized by a 1-deterministic 2-automaton $\mathcal{T}$, so $f(x) = \mathcal{T}(x)$ for every infinite word x. We fix a positive integer k and we construct a compressor $\mathcal{C}$ such that $\rho_{\mathcal{C}}(f(x)/x) = 1/k$. This compressor has two input tapes, the first one containing a word y and the second one the word x. It compresses y in a non-trivial way only when y is equal to $f(x)$. The automaton $\mathcal{C}$ proceeds as follows. It reads the infinite word x from the second input tape and computes $f(x)$ by simulating the automaton $\mathcal{T}$. While $f(x)$ coincides with y for the next k symbols, then $\mathcal{C}$ writes a 0. When there is a mismatch, $\mathcal{C}$ writes a 1 and then copies the remaining part of y to the output tape. Thus, if the mismatch occurs at position $m = kp+r$ with $1 \leq r \leq k$, the automaton $\mathcal{C}$ writes p symbols 0 before the mismatch, a symbol 1 and $y_{kp+1}y_{kp+2}y_{kp+3}\cdots$. □

The following proposition provides a sufficient condition for compressing x given y: some correlation between symbols in x and y at the same positions ensures $\rho(x/y) < 1$.

Proposition 3.3. *Let x and y be two infinite words. If there are three symbols c, c' and d and an increasing sequence $(m_n)_{n\geq 0}$ of integers such that*

$$\lim_{n\to\infty} \frac{|\{1 \leq i \leq m_n : x[i] = c, y[i] = d\}|}{m_n} \neq \lim_{n\to\infty} \frac{|\{1 \leq i \leq m_n : x[i] = c', y[i] = d\}|}{m_n},$$

then $\rho(x/y) < 1$.

We assume here that both limits exist; however, the same result holds if just one or none of the two limits exist.

Proof. By replacing the sequence $(m_n)_{n\geq 0}$ by some subsequence of it, we may assume without loss of generality that $\lim_{n\to\infty} |\{1 \leq i \leq m_n : x[i] = a, y[i] = b\}|/m_n$ exists for arbitrary symbols a and b. This limit is denoted by $\pi(a, b)$. The existence of all the limits $\pi(a, b)$ implies that for each symbol b,

$$\lim_{n\to\infty} |\{1 \leq i \leq m_n : y[i] = b\}|/m_n$$

also exists and

$$\lim_{n\to\infty} \frac{|\{1 \leq i \leq m_n : y[i] = b\}|}{m_n} = \sum_{a\in A} \pi(a, b).$$

This limit is denoted by $\pi_y(b)$. Define

$$\nu(a/b) = \begin{cases} \pi(a, b)/\pi_y(b) & \text{if } \pi_y(b) \neq 0 \\ 1/|A| & otherwise. \end{cases}$$

Let k be a block length to be fixed later. For two words $u = a_1 \cdots a_k$ and $v = b_1 \cdots b_k$ of length k, define

$$\nu(u/v) = \prod_{i=1}^{k} \nu(a_i/b_i).$$

Let us recall that a set P of words is prefix-free if no distinct words $u, v \in P$ satisfy that u is a prefix of v. Note that if P is prefix-free, every word $w \in A^*$ has at most one factorization $w = u_1 \cdots u_n$ where each u_i belongs to P. We will use the following well-known fact due to Huffman [12].

Let $p_1, \ldots, p_n$ be real numbers such that $0 \leq p_i \leq 1$ for each $1 \leq i \leq n$ and $\sum_{i=1}^{n} p_i = 1$, then there exist n distinct words $u_1, \ldots, u_n$ such that the set $\{u_1, \ldots, u_n\}$ is prefix-free and $|u_i| \leq \lceil -\log p_i \rceil$ for $1 \leq i \leq n$.

It is purely routine to check that $\sum_{u \in A^k} \nu(u/v) = 1$ for each word v of length k. Since $\sum_{u \in A^k} \nu(u/v) = 1$, there exists for each word v, a prefix-free set $\{w(u,v) : u, v \in A^k\}$ such that the relation $|w(u,v)| \leq \lceil -\log_{|A|} \nu(u/v) \rceil$ holds for each u and v. These words $w(u,v)$ are used by the transducer to encode the infinite word x with the help of y. The transducer reads x and y by blocks of length k. For each pair of blocks u and v, it outputs $w(u,v)$. This output can be decoded with the help of y because for each fixed block v, the possible blocks u are in one-to-one correspondence with the words $w(u,v)$.

We now evaluate the length of the output of the transducer. Let $p(n,u,v)$ be the number of occurrences of the pair (u,v) in x and y. Then,

$$p(n,u,v) = |\{1 \leq i \leq n-k : i = 1 \bmod k, x[i..i+k-1] = u, y[i..i+k-1] = v\}|$$

$$\begin{aligned}
\rho(x/y) &\leq \lim_{n\to\infty} \frac{1}{n} \sum_{u,v \in A^k} p(n,u,v)|w(u,v)| \\
&\leq \lim_{n\to\infty} \frac{1}{n} \sum_{u,v \in A^k} p(n,u,v)\lceil -\log_{|A|} \nu(u/v) \rceil \\
&\leq \frac{1}{k} + \lim_{n\to\infty} \frac{1}{n} \sum_{u,v \in A^k} p(n,u,v)(-\log_{|A|} \nu(u/v)) \\
&= \frac{1}{k} + \lim_{n\to\infty} \frac{1}{n} \sum_{\substack{u=a_1\cdots a_k \\ v=b_1\cdots b_k}} p(n,u,v)(-\log_{|A|} \prod_{i=1}^{k} \nu(a_i/b_i)) \\
&= \frac{1}{k} + \lim_{n\to\infty} \frac{1}{n} \sum_{a,b \in A} |\{1 \leq i \leq n : x[i] = a, y[i] = b\}| \log_{|A|} \frac{1}{\nu(a/b)} \\
&= \frac{1}{k} + \sum_{b \in A, \pi_y(b) \neq 0} \pi_y(b) \sum_{a \in A} \frac{\pi(a,b)}{\pi_y(b)} \log_{|A|} \frac{\pi_y(b)}{\pi(a,b)}
\end{aligned}$$

The previous to the last row results from counting correlated symbols instead of counting correlated blocks of length k. The last equality results from using $\nu(a/b) = \frac{\pi(a,b)}{\pi_y(b)}$ and applying the definition of $\pi(\cdot,\cdot)$ and $\pi_y(\cdot)$.

We have that for each symbol b, the sum

$$\sum_{a \in A} \frac{\pi(a,b)}{\pi_y(b)} \log_{|A|} \frac{\pi_y(b)}{\pi(a,b)} \leq 1.$$

However, for $b = d$, the above sum is strictly less than 1. Then, it follows that

$$\sum_{b \in A} \pi_y(b) \sum_{a \in A} \frac{\pi(a,b)}{\pi_y(b)} \log \frac{\pi_y(b)}{\pi(a,b)} < 1.$$

If k is chosen great enough, the conditional compression ratio $\rho(x/y)$ satisfies $\rho(x/y) < 1$. □

The definition of finite-state independence of two infinite words is based on the conditional compression ratio.

Definition 3.4. Two infinite words x and y 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.

Note that we require that the compression ratios of x and y are non-zero. This means that a word x such that $\rho(x) = 0$ cannot be part of an independent pair. Without this requirement, every two words x and y such that $\rho(x) = \rho(y) = 0$ would be independent. In particular, every word x with $\rho(x) = 0$ would be independent of itself.

4 Join independence is not enough

Recall that the *join* of two infinite words $x = a_1a_2a_3\cdots$ and $y = b_1b_2b_3\cdots$ is the infinite word $a_1b_1a_2b_2\cdots$ obtained by interleaving their symbols. It is denoted $x \vee y$. A possible definition of independence for normal words x and y would be to require their join $x \vee y$ to be normal. We call this notion *join independence*. This definition of independence would be natural since it mimics the definition of independence in the theory of algorithmic randomness, where two random infinite words are independent if their join is random [23], see also [17, Theorem 3.4.6]. One can ask whether a similar result holds for our definition. Is it true that two normal words are finite-state independent if and only if their join is normal? It turns out that finite-state independence implies join independence. But the converse fails.

We use the following notation. For a given infinite word $x = a_1a_2a_3\ldots$, we define infinite words $\mathrm{even}(x)$ and $\mathrm{odd}(x)$ as $a_2a_4a_6\cdots$ and $a_1a_3a_5\cdots$ respectively. Similarly, for a finite word w, $\mathrm{even}(w)$ and $\mathrm{odd}(w)$ are the words appearing on the even and the odd positions in w. For example, $x = \mathrm{even}(x)$ means that $a_n = a_{2n}$ for all n.

Theorem 4.1. *Let x and y be normal. If x and y are finite-state independent then $x \vee y$ is normal.*

Proof. Suppose $x \vee y$ is not normal. Then, there is a length k such that a word of length k does not have aligned frequency 2^{-k} in $x \vee y$. We can assume without loss of generality that the length is even (increasing it if necessary), so we assume that some word of length $2k$ does not have aligned frequency 2^{-2k}. Split $x \vee y$ into blocks of size $2k$:

$$x \vee y = (u_1 \vee v_1)(u_2 \vee v_2)\cdots$$

where $x = u_1u_2\ldots$ and $y = v_1v_2\ldots$ are the respective decomposition in k-blocks. Consider a sequence of positions $(m_n)_{n\geq 1}$ in $x \vee y$ such that each block of length $2k$ has an aligned frequency, and let $u \vee v$ be a block of length $2k$ whose frequency f along $(m_n)_{n\geq 1}$ is not 2^{-2k}. Let $u' \vee v$ be another block of length $2k$ with frequency along $(m_n)_{n\geq 1}$ different from f. (Notice that if $u \vee v$ and all blocks $u' \vee v$ have the all the same frequency f along $(m_n)_{n\geq 1}$ then, necessarily, there is a block v' such that $u \vee v'$ has frequency along $(m_n)_{n\geq 1}$ different from f, so we exchange the roles of x and y). Then, at the positions $(m_n/2)_{n\geq 1}$

$$\lim_{n\to\infty} \frac{|\{1 \leq i \leq m_n/2 : x[i..i+k-1] = u, y[i..+k-1] = v\}|}{m_n} \neq$$
$$\lim_{n\to\infty} \frac{|\{1 \leq i \leq m_n/2 : x[i..+k-1] = u', y[i..i+k-1] = v\}|}{m_n},$$

By the argument in Proposition 3.3 we conclude that $\rho(u_1u_2\ldots/v_1v_2\ldots) < 1$. By considering a compressor that reads blocks of length k we obtain

$$\rho(x/y) = \rho(u_1u_2\ldots/v_1v_2\ldots) < 1.$$

□

Actually, the proof of Theorem 4.1 gives a stronger result.

Proposition 4.2. *If y is normal and $\rho(x/y) = 1$, then $x \vee y$ is normal.*

We now show that there are two words, that are join independent but not finite-state independent. The proof is based on the existence of normal word x such that $x = \mathrm{even}(x)$, that we prove in Theorem 4.4 below.

Theorem 4.3. *There exists two normal words x and y such that $x \vee y$ is normal but x and y are not independent.*

Proof. By Theorem 4.4, proved below, there is a normal word x such that $x = \mathrm{even}(x)$. Let $y = \mathrm{odd}(x)$ and $z = \mathrm{even}(x)$. Since x is normal and $x = y \vee z$, the words y and z are join independent. Since $y = \mathrm{odd}(x)$ and $x = \mathrm{even}(x) = z$, we have the equality $y = \mathrm{odd}(z)$. This implies, by Proposition 3.2, that $\rho(y/z) = 0$; hence, y and z are not finite-state independent. □

4.1 Construction of a normal word x such that $x = \text{even}(x)$

Theorem 4.4. *There is a normal word x such that $x = \text{even}(x)$.*

Here we prove this existence by giving an explicit construction of a normal word $x = a_1a_2a_3\cdots$ over the alphabet $\{0,1\}$ such that $a_{2n} = a_n$ for every n. The construction can be easily extended to an alphabet of size k to obtain a word $a_1a_2a_3\cdots$ such that $a_{kn} = a_n$ for each integer $n \geq 1$. Beware that the construction, as it is presented below, cannot provide a word $a_1a_2a_3\cdots$ over a binary alphabet such that $a_{3n} = a_n$ (some more sophisticate one is needed, but it can be done; the probabilistic argument also works).

A finite word w is called *ℓ-perfect* for an integer $\ell \geq 1$, if $|w|$ is a multiple of ℓ and all words of length ℓ have the same number of aligned occurrences $|w|/(\ell 2^\ell)$ in w.

Lemma 4.5. *Let w be an ℓ-perfect word such that $|w|$ is a multiple of $\ell 2^{2\ell}$. Then, there exists a 2ℓ-perfect word z of length $2|w|$ such that $\text{even}(z) = w$.*

Proof. Since $|w|$ is a multiple of $\ell 2^{2\ell}$ and w is ℓ-perfect, for each word u of length ℓ, $\|w\|_u$ is a multiple of 2^ℓ. Consider a factorization of $w = w_1w_2\cdots w_r$ such that for each i, $|w_i| = \ell$. Thus, $r = |w|/\ell$. Since w is ℓ-perfect, for any word u of length ℓ, the set $\{i : w_i = u\}$ has cardinality $r/2^\ell$. Define z of length $2|w|$ as $z = z_1z_2\cdots z_r$ such that for each i, $|z_i| = 2\ell$, $even(z_i) = w_i$ and for all words u and u' of length ℓ, the set $\{i : z_i = u' \vee u\}$ has cardinality $r/2^{2\ell}$. This latter condition is achievable because, for each word u of length ℓ, the set $\{i : \text{even}(z_i) = u\}$ has cardinality $r/2^\ell$ which is a multiple of 2^ℓ, the number of possible words u'. □

Corollary 4.6. *Let w be an ℓ-perfect word for some even integer ℓ. Then there exists an ℓ-perfect word z of length $2|w|$ such that $\text{even}(z) = w$.*

Proof. Since w is ℓ-perfect, it is also $\ell/2$-perfect. Furthermore, if u and v are words of length $\ell/2$ and ℓ respectively then $\|w\|_u = 2^{\ell/2+1}\|w\|_v$. Thus, the hypothesis of Lemma 4.5 is fulfilled with $\ell/2$. □

Corollary 4.7. *There exist a sequence $(w_n)_{n\geq 1}$ of words and a sequence of positive integers $(\ell_n)_{n\geq 1}$ such that $|w_n| = 2^n$, $\text{even}(w_{n+1}) = w_n$, w_n is ℓ_n-perfect and $(\ell_n)_{n\geq 1}$ is non-decreasing and unbounded. Furthermore, it can be assumed that $w_1 = 01$.*

Proof. We start with $w_1 = 01$, $\ell_1 = 1$, $w_2 = 1001$ and $\ell_2 = 1$. For each $n \geq 2$, if $\ell_n 2^{2\ell_n}$ divides $|w_n|$, then $\ell_{n+1} = 2\ell_n$ and w_{n+1} is obtained by Lemma 4.5. Otherwise, $\ell_{n+1} = \ell_n$ and w_{n+1} is obtained by Corollary 4.6 . Note that the former case happens infinitely often, so $(\ell_n)_{n\geq 1}$ is unbounded. Also note that each ℓ_n is a power of 2. □

Lemma 4.8 (Theorem 148 [11])**.** *Let A be an alphabet of b symbols. Let $p(k,r,j)$ be the number of words of length k with exactly j occurrences of a given word of length r, at any position:*

$$p(k,r,j) = \left| \bigcup_{u\in A^r} \{w \in A^k : |w|_u = j\} \right|$$

For every integer r greater than or equal to 1, for every integer k large enough and for every real number ε such that $6/\lfloor k/r \rfloor \leq \varepsilon \leq 1/b^r$,

$$\sum_{i:\ |i-k/b^r| \geq \varepsilon k} p(k,r,i) < 2\ b^{k+2r-2}r\ e^{-b^r\varepsilon^2 k/6r}.$$

Lemma 4.9 (Theorem 4.6 [6])**.** *Let A be an alphabet. An infinite word x is normal if and only if there is a positive number C such that, for every every word u,*

$$\limsup_{N\to\infty} \frac{|x[1..N]|_u}{N} < \frac{C}{|A|^{|u|}},$$

Finally, the next lemma is similar to Lemma 4.9 but with aligned occurrences.

Lemma 4.10. *Let A be an alphabet. An infinite word x is normal if and only if there is a positive number C such that, such that for infinitely many lengths ℓ, for every word w of length ℓ,*

$$\limsup_{N\to\infty} \frac{\|x[1..\ell N]\|_w}{N} < \frac{C}{|A|^\ell}.$$

Proof. The implication from left to right is immediate from the definition of normality. We prove the other. Fix alphabet A with b symbols, fix x and C. Assume that for infinitely many lengths ℓ, for every word w of length ℓ, the stated condition holds. Equivalently,

$$\limsup_{N\to\infty} \frac{\|x[1..N]\|_w}{N} < \frac{C}{\ell|A|^\ell}. \tag{*}$$

We will prove that for every word u, of any length,

$$\limsup_{N\to\infty} \frac{|x[1..N]|_u}{N} < \frac{C}{|A|^u}.$$

and conclude that x is normal by Lemma 4.9. The task is to switch from aligned occurrences to non-aligned occurrences. For this we consider long consecutive blocks and the fact that most of them have the expected number of non-aligned occurrences of small blocks inside.

Fix a length r and a word u of length r. Let ℓ be any length greater than r for which (*) holds. Fix ε. We group the words of length ℓ in good and bad for r and ε. The bad ones deviate from the expected number of occurrences of some word of length r by $\varepsilon\ell$ or more. The good ones do not. Lemma 4.8 bounds the number of these bad words.

We use that each bad word has at most $\ell - r + 1$ occurrences of u; each good word has at most $\ell/b^r + \varepsilon\ell/b^r$ occurrences of u; and in between any of two consecutive blocks of length ℓ there are at most $r-1$ occurrences of u.

$$\begin{aligned}
\frac{|x[1..N]|_u}{N} &< \frac{1}{N}\sum_{w\in A^\ell} \|x[1..N]\|_w(|w|_u + (r-1)) \\
&= \frac{1}{N}\sum_{\text{bad } w} \|x[1..N]\|_w\big(|w|_u + (r-1)\big) + \frac{1}{N}\sum_{\text{good } w} \|x[1..N]\|_w\big(|w|_u + (r-1)\big) \\
&< \frac{1}{N}(\ell - r + 1 + r - 1)\sum_{\text{bad } w} \|x[1..N]\|_w + \frac{1}{N}\left(\frac{\ell}{b^r} + \varepsilon\ell + r - 1\right)\sum_{\text{good } w} \|x[1..N]\|_w \\
&< \frac{1}{N}\ \ell\ \left(2b^\ell\ b^{2r-2} r\ e^{-b^r\varepsilon^2\ell/(6r)}\right)\frac{CN}{\ell b^\ell} + \frac{1}{N}\left(\frac{\ell}{b^r} + \varepsilon\ell + r - 1\right)\frac{N}{\ell} \\
&= 2b^{2r-2}re^{-b^r\varepsilon^2\ell/(6r)}C + \left(\frac{1}{b^r} + \varepsilon + \frac{r-1}{\ell}\right).
\end{aligned}$$

For ℓ large enough and $\varepsilon = \ell^{-1/3}$ the values $2b^{2r-2}re^{-b^r\varepsilon^2\ell/(6r)}C$ and $\left(\varepsilon + \frac{r-1}{\ell}\right)$ are arbitrarily small. So,

$$\limsup_{N\to\infty} \frac{|x[1..N]|_u}{N} < \frac{C}{b^r}.$$

□

Proof of Theorem 4.4. Let $(w_n)_{n\geq 1}$ be a sequence given by Corollary 4.7. Let $x = 11w_1w_2w_3\cdots$ We first prove that x satisfies $x = \text{even}(x)$. Note that $x[2^k + 1..2^{k+1}] = w_k$ for each $k \geq 1$ and $x[1..2^{k+1}] = 11w_1\cdots w_k$. The fact that $w_n = \text{even}(w_{n+1})$ implies $x[2n] = x[n]$, for every $n \geq 3$. The cases for $n = 1$ and $n = 2$ hold because $x[1..4] = 1101$.

We prove that x is normal. Consider an arbitrary index n_0. By construction, w_{n_0} is ℓ_{n_0}-perfect and for each $n \geq n_0$, w_n is also ℓ_{n_0}-perfect. For every word u of length ℓ_{n_0} and for every $n \geq n_0$,

$$\|x[1..2^{n+1}]\|_u \leq \|x[1..2^{n_0}]\|_u + \|w_{n_0} \dots w_n\|_u$$

Then, for every N such that $2^n \leq N < 2^{n+1}$ and $n \geq n_0$,

$$\begin{aligned}
\frac{\|x[1..N]\|_u}{N/\ell_{n_0}} &\leq \frac{\|x[1..2^{n+1}]\|_u}{N/\ell_{n_0}} \\
&\leq \frac{\|x[1..2^{n_0}]\|_u + \|w_{n_0} \dots w_n\|_u}{N/\ell_{n_0}} \\
&\leq \frac{\|x[1..2^{n_0}]\|_u}{2^n/\ell_{n_0}} + \frac{\|w_{n_0} \dots w_n\|_u}{2^n/\ell_{n_0}} \\
&= \frac{\|x[1..2^{n_0}]\|_u}{2^n/\ell_{n_0}} + \frac{(2^{n_0} + \dots + 2^n)/(\ell_{n_0} 2^{\ell_{n_0}})}{2^n/\ell_{n_0}} \\
&= \frac{\|x[1..2^{n_0}]\|_u}{2^n/\ell_{n_0}} + \frac{2^{n+1} - 2^{n_0}}{2^n 2^{\ell_{n_0}}} \\
&< \frac{\|x[1..2^{n_0}]\|_u}{2^n/\ell_{n_0}} + \frac{2}{2^{\ell_{n_0}}}.
\end{aligned}$$

For large values of N and n such that $2^n \leq N < 2^{n+1}$, the expression

$$\frac{\|x[1..2^{n_0}]\|_u}{2^n/\ell_{n_0}}$$

becomes arbitrarily small. We obtain for every word u of length ℓ_{n_0},

$$\limsup_{N \to \infty} \frac{\|x[1..N]\|_u}{N/\ell_{n_0}} \leq 3\ 2^{-\ell_{n_0}}.$$

Since the choice of ℓ_{n_0} was arbitrary, the above inequality holds for each ℓ_n. Since $(\ell_n)_{n \geq 1}$ is unbounded, the hypothesis of Lemma 4.10 is fulfilled, with $C = 3$, so we conclude that x is normal. □

Alexander Shen (personal communication, August 2016) proved that almost all binary words satisfying $x = \text{even}(x)$ are normal. The argument in his proof works if the distances between different occurrences of the same repeated symbol grow sufficiently fast. For example, his argument can be also used to prove that almost all binary words satisfying $x_{3n} = x_n$ are normal.

5 Almost all pairs are independent

The next theorem establishes that almost all pairs of infinite words are independent.

Theorem 5.1. *The set $I = \{(x, y) : x$ and y are independent$\}$ has measure 1.*

To prove it we use that if the oracle y is normal then the number of symbols read from y, is linearly bounded by the number of symbols read from the input x. This property, stated in Lemma 5.3 below, requires the notion of a finite run and the notion of a forward pair.

A finite run of a k-automaton $\mathcal{T}$ is a finite sequence of consecutive transitions

$$q_0 \xrightarrow{a_{1,1},\dots,a_{k,1}} q_1 \xrightarrow{a_{1,2},\dots,a_{k,2}} q_2 \to \cdots \to q_{n-1} \xrightarrow{a_{1,n},\dots,a_{k,n}} q_n.$$

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

Let $\mathcal{T}$ be a 2-deterministic 3-automaton whose state set is Q. Let v be a finite word. A pair $(p, a) \in Q \times A$ is called a *forward pair* of v if there is a finite run $p \xrightarrow{au,v|w} q$ for some finite words u and w and some state q. A finite word v is called a *forward word* if it has a maximum number of forward pairs. Since the number of pairs $(p, a) \in Q \times A$ is finite, there exist forward words. Note that, in case the automaton is total, every extension of a forward word is also a forward word. However, not every prefix of a forward word is a forward word.

Lemma 5.2. *Let $\mathcal{T}$ be a 2-deterministic 3-automaton and let x and y be two infinite words such that the run $\gamma = q_0 \xrightarrow{x,y|z} \infty$ is accepting. Let v be a forward word for $\mathcal{T}$. For each factorization $\gamma = q_0 \xrightarrow{u_1,v_1|w_1} q_1 \xrightarrow{u_2,v_2|w_2} q_2 \xrightarrow{x_1,y_1|z_1} \infty$ such that v occurs in v_2, u_2 is non-empty.*

Proof. It suffices to prove the result when the word v_2 is equal to v. Suppose by contradiction that u_2 is empty. Let a be the first symbol of x_1. Since u_2 is empty, the pair (q_1, a) is not forward pair of v. Since x_1 is infinite, there exists a right extension vv' of v such that (q_1, a) is a forward pair of vv'. This contradicts the fact that v has a maximum number of forward pairs. □

Lemma 5.3. *Let $\mathcal{T}$ be a 2-deterministic 3-automaton and let x and y be two infinite words such that the run $\gamma = q_0 \xrightarrow{x,y|z} \infty$ is accepting. If y is normal, there is a constant K depending only on $\mathcal{T}$ such that for any factorization $\gamma = q_0 \xrightarrow{u_1,v_1|w_1} q_1 \xrightarrow{x_1,y_1|z_1} \infty$ such that $|u_1|$ is long enough, $|v_1| \leq K|u_1|$.*

Proof. Let v be a forward word for $\mathcal{T}$. Since y is normal there is a positive constant k less than 1 such that the number of disjoint occurrences of v in $y[1..n]$ is greater than kn for n large enough. By the previous lemma, $|u_1| \geq k|v_1|$ holds. The result holds then with $K = 1/k$. □

We write μ for the Lebesgue measure.

Theorem 5.4. *For each normal word y, the set $\{x : \rho(x/y) < \rho(x)\}$ has Lebesgue measure 0.*

Proof. Fix y normal. Since for every x, $\rho(x) \leq 1$ (see comment after Definition 3.1), it suffices to prove $\{x : \rho(x/y) < 1\}$ has measure 0. The inequality $\rho(x/y) < 1$ holds if there exists a 2-deterministic 3-automaton that compresses x using y as oracle. For a 2-deterministic 3-automaton $\mathcal{T}$, let Q be its state set and let K be the constant obtained by Lemma 5.3. For any integer n and any positive real number $\varepsilon < 1$, let $X_{n,\varepsilon}$ be defined by

$$X_{n,\varepsilon} = \{x : q_0 \xrightarrow{u,v|w} q, u = x[1..n], v = y[1..n'] \text{ and } |w| < (1-\varepsilon)n\}.$$

We claim that $\mu(X_{n,\varepsilon}) \leq |Q|Kn2^{-n\varepsilon}$. The number of configurations $\langle q, n, n'\rangle$ is at most $|Q|Kn2^{(1-\varepsilon)n}$ because there are $|Q|$ possibles states, $n' \leq Kn$, and there are at most $2^{(1-\varepsilon)n}$ words of length smaller than $(1-\varepsilon)n$. Two words x and x' with different prefixes of length n can not reach the same configuration. If they did, the rest of the run would be the same and this would contradict the injectivity of $\mathcal{T}$. Therefore, $X_{n,\varepsilon}$ is contained in at most $|Q|Kn2^{(1-\varepsilon)n}$ cylinders of measure 2^{-n}.

Observe that for n fixed, the inclusion $X_{n,\varepsilon} \subseteq X_{n,\varepsilon'}$ holds for $\varepsilon' \leq \varepsilon$. Let $\varepsilon(n)$ be a decreasing function which maps each integer n to a real number such that

$$\sum_{n\geq 0} n2^{-\varepsilon(n)n} < \infty,$$

for instance $\varepsilon(n) = 1/\sqrt{n}$. For each k, define

$$\tilde{X}_k = \bigcup_{n\geq k} X_{n,\varepsilon(n)}.$$

By the choice of the function $\varepsilon(n)$, $\lim_{k\to\infty} \mu(\tilde{X}_k) = 0$. We claim that each word x compressible with $\mathcal{T}$ with normal oracle y belongs to $\tilde{X}_k$ for each k. It suffices to prove that it belongs to infinitely many sets $X_{n,\varepsilon(n)}$. Suppose x is compressed with ratio $1-\delta$, for some $\delta > 0$. Then there is an infinite sequence of integers $(n_j)_{j\geq 0}$ such that the configurations $\langle q, n_j, n'_j\rangle$ reached

after reading the prefix of length n_j and outputting w_j, with $|w_j| < (1-\delta)n_j$. Let j_0 be such that $\varepsilon(n_{j_0}) < \delta$. Then, for each for $j \geq j_0$, we have $x \in X_{n_j,\varepsilon(n_j)}$. Since this holds for each of the countably many 3-automata $\mathcal{T}$, we conclude that the measure of the set of words compressible with normal oracle y is null. □

Proof of Theorem 5.1. By definition of independence, the complement $\bar{I} = A^\omega \times A^\omega \setminus I$ of I can be decomposed as $\bar{I} = J_1 \cup J_2 \cup J_3 \cup J_4$ where the sets J_1, J_2, J_3 and J_4 are defined by the following equations.

$$J_1 = \{(x,y) : \rho(x) = 0\} \qquad J_2 = \{(x,y) : \rho(y) = 0\}$$
$$J_3 = \{(x,y) : \rho(x/y) < \rho(x)\} \qquad J_4 = \{(x,y) : \rho(y/x) < \rho(y)\}$$

The sets J_1 and J_2 satisfy $\mu(J_1) = \mu(J_2) = 0$. By symmetry, the sets J_3 and J_4 satisfy $\mu(J_3) = \mu(J_4)$. To show that $\mu(I) = 1$, it suffices then to show that $\mu(J_3) = 0$.

The measure of J_3 is then given by

$$\mu(J_3) = \iint_{x,y} 1_{J_3}\, dxdy = \int_y \left(\int_x 1_{J_3}\, dx \right) dy = \int_y f(y)\, dy$$

where the function f is defined by $f(y) = \mu(\{x : \rho(x/y) < \rho(x)\})$. By Theorem 5.4, $f(y)$ is equal to 0 for each normal word y. Since the set of normal words has measure 1, $f(y)$ is equal to 0 for almost all y. It follows that $\mu(J_3) = 0$. This concludes the proof of the theorem. □

Acknowledgements. The authors acknowledge Alexander Shen for many fruitful discussions. The authors are members of the Laboratoire International Associé INFINIS, CONICET/Universidad de Buenos Aires–CNRS/Université Paris Diderot. Becher is supported by the University of Buenos Aires and CONICET.

References

[1] B. Bauwens, A. Shen, and H. Takahashi. Conditional probabilities and van Lambalgen theorem revisited. Submitted, 2016.

[2] V. Becher and O. Carton. Normal numbers and computer science. In Valérie Berthé and Michel Rigó, editors, *Sequences, Groups, and Number Theory*, Trends in Mathematics Series. Birkhauser/Springer, 2017.

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

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

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

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

[7] C. S. Calude and M. Zimand. Algorithmically independent sequences. *Information and Computation*, 208(3):292 – 308, 2010.

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

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

[10] R. G. Downey and D. Hirschfeldt. *Algorithmic randomness and complexity.* Theory and Applications of Computability. New York, NY: Springer. xxvi, 855 p., 2010.

[11] G. H. Hardy and E. M. Wright. *An introduction to the theory of numbers.* Oxford University Press, Sixth Edition 2008.

[12] D. Huffman. A method for the construction of minimum-redundancy codes. In *Institute of Radio Engineers*, volume 40:9, pages 1098–1101, 1952.

[13] K. Hyde and B. Kjos-Hanssen. Nondeterministic automatic complexity of almost square-free and strongly cube-free words. In Z. Cai, A. Zelikovsky, and A. Bourgeois, editors, *Computing and Combinatorics: 20th International Conference, COCOON 2014, Atlanta, GA, USA, August 4-6, 2014. Proceedings*, pages 61–70. Springer International Publishing, Cham, 2014.

[14] S. Kautz. *Degrees of random sets.* PhD thesis, Cornell University, 1991.

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

[16] M. Li and P. Vitanyi. *An Introduction to Kolmogorov Complexity and Its Applications.* Springer Publishing Company, Incorporated, 3 edition, 2008.

[17] A. Nies. *Computability and randomness.* Clarendon Press, 2008.

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

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

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

[21] J. Shallit and M. Wang. Automatic complexity of strings. *J. Autom. Lang. Comb.*, 6(4):537–554, April 2001.

[22] A. Shen, V. Uspensky, and N. Vereshchagin. *Kolmogorov complexity and algorithmic randomness.* Submitted, 2016.

[23] M. van Lambalgen. *Random sequences.* PhD thesis, University of Amsterdam, 1987.

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

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

Pablo Ariel Heiber
Departamento de Computación, Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires & CONICET, Argentina.
pheiber@dc.uba.ar