Two Characterizations of Finite-State Dimension

Alexander Kozachinskiy, Alexander Shen

Two characterizations of finite-state dimension*

Alexander Kozachinskiy[1,4][0000−0002−9956−9023] and Alexander Shen[2,3][0000−0001−8605−7734]

[1] National Research University Higher School of Economics, Moscow, Russia
[2] LIRMM CNRS & University of Montpellier, alexander.shen@lirmm.fr
https://www.lirmm.fr/~ashen
[3] On leave from IITP RAS, Moscow
[4] Lomonosov Moscow State University

Abstract. In this paper we provide two equivalent characterizations of the notion of finite-state dimension introduced by Dai, Lathrop, Lutz and Mayordomo (2004). One of them uses Shannon's entropy of non-aligned blocks and generalizes old results of Pillai (1940) and Niven – Zuckerman (1951). The second characterizes finite-state dimension in terms of superadditive functions that satisfy some calibration condition (in particular, superadditive upper bounds for Kolmogorov complexity). The use of superadditive bounds allows us to prove a general sufficient condition for normality that easily implies old results of Champernowne (1933), Besicovitch (1935), Copeland and Erdös (1946), and also a recent result of Calude, Staiger and Stephan (2016).

Keywords: Finite-state dimension · Superadditive complexity functions · Normal sequences

1 Introduction

The notion of finite-state dimension of a bit sequence was introduced by Dai et al. [7] using finite-state gales. Later Bourke et al. [2] characterized the finite-state dimension in terms of Shannon entropies of aligned bit blocks (a prefix of the sequence is split into k-bit blocks for some k, and a random variable "uniformly chosen block" is considered).

In this paper we provide two new characterizations (equivalent definitions) of this notion. First (Section 2) we extend old results of Niven – Zuckerman [11] and Pillai [12] to the case of arbitrary finite-state dimension. These results were proven for normal sequences, i.e., sequences of finite-state dimension 1, and new tools (including Shearer-type inequality for entropies) are needed for the case of arbitrary finite-state dimension. Namely, we show (Theorem 1) that one can equivalently define the finite-state dimension using *non-aligned* blocks. For that, for a given n we consider a random variable "uniformly chosen k-bit factor"

* Supported by RaCAF ANR-15-CE40-0016-01 grant. The article was prepared within the framework of the HSE University Basic Research Program and funded by the Russian Academic Excellence Project '5-100'.

of the n-bit prefix of the sequence, take the $\liminf$ of its Shannon entropy as $n \to \infty$, divide this $\liminf$ by k and then take infimum (or limit) over k. We also provide examples showing that this equivalence works only in the limit ($k \to \infty$), not for blocks of fixed size.

The second characterization of finite-state dimension is given in Section 3. It does not use finite-state machines or entropies at all. We consider non-negative *superadditive* functions on bit strings, i.e., functions F such that $F(uv) \geqslant F(u) + F(v)$ for all u and v. Additionally we require some calibration property saying that F cannot be too small on too many inputs. Given a sequence $\alpha = \alpha_0\alpha_1\alpha_2\ldots$, we consider $\liminf_n F(\alpha_0\alpha_1\ldots\alpha_{n-1})/n$. We prove that the finite-state dimension of α is the infimum of these quantities taken over all F that satisfy our requirements.

The first example of a normal sequence was given by Champernowne [5]. It was the sequence 0 1 10 11 100 101 110 111 1000 1001 ... (concatenation of integers $0, 1, 2, 3, \ldots$ written in binary[5]). Later a more general class of examples was suggested by Copeland and Erdös [6]. In Section 4, using superadditive functions, we prove a general sufficient condition for normality (=finite-state dimension 1) for a sequence that is a concatenation of some finite strings x_1, x_2, x_3, etc. This sufficient condition is formulated in terms of Kolmogorov complexity of x_i: the average Kolmogorov complexity of strings $x_1, \ldots, x_k$ should have the same asymptotic growth as the average length of these strings (under some technical conditions; see the exact statement of Theorem 4). In [3] Calude, Salomaa and Roblot introduced the notion of automatic complexity and asked whether this notion can be used to characterize normality. This question was answered negatively in [4]. We give an alternative proof of this result using our sufficient condition for normality.

The notion of automatic complexity that can be used to characterize normality and finite-state dimension (and was the starting point for us) was introduced in [13]. A self-contained exposition, including the results of the current paper and other results about finite-state dimension, automatic complexity, finite-state a priori probability and martingales, as well as applications of these notions, will be included in the arxiv version of [13].

2 Non-aligned entropies

Consider a sequence $\alpha = \alpha_0\alpha_1\alpha_2\ldots$, and some positive integer k. We can split the sequence α into k-bit consecutive non-overlapping blocks (aligned version), or consider all k-bit substrings of α (non-aligned version, see below the exact definition). Then we consider limit frequencies of these blocks. In this way we get some distribution on the set $\{0,1\}^k$ of all k-bit blocks. We want to define the finite-state dimension of α as the limit of the normalized (i.e., divided by k) Shannon entropy of this distribution when k goes to infinity.

[5] In fact, Champernowne spoke about decimal notation and sequences of digits, but this does not make a big difference.

However, we should be more careful since these limit frequencies may not exist. Here is the exact definition. For every N take the first N blocks of length k and choose one of them uniformly at random. In this way we obtain a random variable taking values in $\{0,1\}^k$. Consider the Shannon entropy of this random variable (for the definition of Shannon entropy of a random variable see, e.g., [14, Chapter 7]). This can be done in an aligned (a) and non-aligned (na) settings, so we get two quantities: $H^{\mathrm{a}}_{k,N}(\alpha) = H(\alpha_{kI}\ldots\alpha_{kI+k-1})$, $H^{\mathrm{na}}_{k,N}(\alpha) = H(\alpha_I\ldots\alpha_{I+k-1})$, where $I \in \{0,\ldots,N-1\}$ (the block number) is chosen uniformly at random, and H denotes the Shannon entropy of the corresponding random variable.

Then we apply the $\liminf_N$ as $N\to\infty$ and let $H^{\mathrm{a}}_k(\alpha) = \liminf_{N\to\infty} H^{\mathrm{a}}_{k,N}(\alpha)$ and $H^{\mathrm{na}}_k(\alpha) = \liminf_{N\to\infty} H^{\mathrm{na}}_{k,N}(\alpha)$. The following result says that both quantities $H^{\mathrm{a}}_k(\alpha)$ and $H^{\mathrm{na}}_k(\alpha)$, divided by the block length k, converge to the same value as $k\to\infty$, and this value can also be defined as $\inf_k H_k(\alpha)/k$ (both in aligned and non-aligned versions).

Theorem 1. *For every bit sequence α we have*

$$\lim_k \frac{H^{\mathrm{a}}_k(\alpha)}{k} = \inf_k \frac{H^{\mathrm{a}}_k(\alpha)}{k} = \lim_k \frac{H^{\mathrm{na}}_k(\alpha)}{k} = \inf_k \frac{H^{\mathrm{na}}_k(\alpha)}{k}.$$

This common value is called the *finite-state dimension of* α and denoted by $\mathrm{FSD}(\alpha)$. The original definition of finite-state dimension [7] was different, and the equivalence between it and the aligned version of the definition given above was shown in [2]. The equivalence between non-aligned and aligned versions seems to be new.

To prove this result, it is enough to prove two symmetric lemmas. The first one guarantees that if $H^{\mathrm{a}}_k(\alpha)/k$ is small (less than some threshold) for some k, then $H^{\mathrm{na}}_K(\alpha)/K$ is also small (less than the same threshold) for all sufficiently large K; the second says the same with aligned and non-aligned versions exchanged.

Lemma 1. *For every α, every k, every $K \geqslant k$:* $\frac{H^{\mathrm{na}}_K(\alpha)}{K} \leqslant \frac{H^{\mathrm{a}}_k(\alpha)}{k} + O\left(\frac{k}{K}\right)$.

Lemma 2. *For every α, every k, every $K \geqslant k$:* $\frac{H^{\mathrm{a}}_K(\alpha)}{K} \leqslant \frac{H^{\mathrm{na}}_k(\alpha)}{k} + O\left(\frac{k}{K}\right)$.

This two lemmas easily imply Theorem 1 by taking $\limsup_{K\to\infty}$ and then $\inf_k$ of both sides of both inequalities. So it remains to prove them.

Proof (of Lemma 1). Fix some sequence α, and consider some integer N. Take $I \in \{0,1\ldots,N-1\}$ uniformly at random and consider a random variable

$$\xi = \alpha_I\ldots\alpha_{I+K-1}$$

whose values are K-bit strings. By definition, the entropy of ξ is $H^{\mathrm{na}}_{K,N}(\alpha)$. Let us look at aligned k-bit blocks covered by the block ξ (i.e., the aligned k-bit blocks inside $I\ldots I+K-1$). The exact number of these blocks may vary depending on I, but there are at least $m = \lfloor K/k\rfloor - 1$ of them (if there were only $m-1$

complete blocks, plus maybe two incomplete blocks, then the total length would be at most $k(m-1)+2k-2 = km+k-2$, but we have $K/k \geqslant m+1$, i.e., $K \geqslant km+k$). We number the first m blocks from left to right and get m random variables $\xi_1, \ldots, \xi_m$ (defined on the same space $\{0, \ldots, N-1\}$). For example, ξ_1 is the leftmost aligned k-bit block of α in the interval $I \ldots I+K-1$. To reconstruct the value of ξ when all ξ_i are known, we need to specify the prefix and suffix of ξ that are not covered by ξ_i (including their lengths). This requires $O(k)$ bits of information, so

$$H^{\mathrm{na}}_{K,N}(\alpha) = H(\xi) \leqslant H(\xi_1) + \ldots + H(\xi_m) + O(k).$$

We will show that for each $s \in \{1, \ldots, m\}$ the distribution of the random variable ξ_s is close to the uniform distribution over the first $\lfloor N/k \rfloor$ aligned k-bit blocks of α. The standard way to measure how close are two distributions on the same set A is to measure the *statistical distance* between them, defined as

$$\delta(P,Q) = \frac{1}{2}\sum_{a \in A} \big|P(a) - Q(a)\big|.$$

We claim that (for each $s \in \{1, 2, \ldots, m\}$) the statistical distance between the distribution of ξ_s and the uniform distribution on the first $\lfloor N/k \rfloor$ aligned blocks converges to 0 as $N \to \infty$. First, let us note that for a fixed aligned block its probability to become s-th aligned block inside a random nonaligned block is exactly k/N (there are k possible positions for a random non-aligned block when this happens). The only exception to this rule are aligned blocks that are near the endpoints, and we have at most $O(K/k)$ of them. When we choose a random aligned block, the probability to choose some position is exactly $1/\lfloor N/k \rfloor$, so we get some difference due to rounding. It is easy to see that the impact of both factors on the statistical distance converges to 0 as $N \to \infty$. Indeed, the number of the boundary blocks is $O(K/k)$, and the bound does not depend on N, while the probability of each block (in both distributions) converges to zero.[6] Also, since $m = N/k$ and $m' = \lfloor N/k \rfloor$ differ at most by 1, the difference between $1/m$ and $1/m'$ is of order $1/m^2$, and converges to 0 even if multiplied by m (the number of blocks is about m).

Now we use the continuity (more precisely, the uniform continuity) of the entropy function and note that all $m = \lfloor N/k \rfloor - 1$ random variables in the right hand side are close to the uniform distribution on first $\lfloor N/k \rfloor$ aligned blocks (the statistical distance converges to 0), so

$$\liminf_{N\to\infty} H^{\mathrm{na}}_{K,N}(\alpha) \leqslant (\lfloor K/k \rfloor - 1) \liminf_{N\to\infty} H^{\mathrm{a}}_{k,\lfloor N/k \rfloor}(\alpha) + O(k),$$

and dividing by K we get the statement of Lemma 1. □

[6] More precisely, we should speak not about the probability of a given block, since the same k-bit block may appear in several positions, but about the probability of its appearance in a given position. Formally speaking, we use the following obvious fact: if we apply some function to two random variables, the statistical difference between them may only decrease. Here the function forgets the position of a block.

Proof (of Lemma 2). Take $I \in \{0, 1 \ldots, N-1\}$ uniformly at random. We need an upper bound for $H^{\mathrm{a}}_{K,N}(\alpha)$, i.e., for $H(\alpha_{KI} \ldots \alpha_{KI+K-1})$. For that we use Shearer's inequality (see, e.g., [14, Section 7.2 and Chapter 10]). In general, this inequality can be formulated as follows. Consider a finite family of arbitrary random variables $\eta_0, \ldots, \eta_{m-1}$ indexed by integers in $\{0, \ldots, m-1\}$. For every $U \subset \{0, \ldots, m-1\}$ consider the tuple η_U of all η_u where $u \in U$. If a family of subsets $U_0, \ldots, U_{s-1} \subset \{0, \ldots, m-1\}$ covers each element of U at least r times, then

$$H(\eta_U) \leqslant \tfrac{1}{r}\left(H(\eta_{U_0}) + \ldots + H(\eta_{U_{s-1}})\right).$$

In our case we have K variables $\eta_0, \ldots \eta_{K-1}$ that are individual bits of a K-bit block $\alpha_{KI} \ldots \alpha_{KI+K-1}$ (for random I), i.e., $\eta_0 = \alpha_{KI}$, $\eta_1 = \alpha_{KI+1}$, etc. The set U contains all indices $0, \ldots, K-1$, and the sets U_i contains k indices $i, i+1, \ldots, i+k-1$ (where operations are performed modulo K, so there are U_i that combine the prefix and suffix of a random K-bit block). Each η_i is covered k times due to this cyclic arrangement. In other words, the variable η_{U_i} is a substring of the random string $\eta_U = \alpha_{KI} \ldots \alpha_{KI+K-1}$ that starts from ith position and wraps around if there is not enough bits. There are $k-1$ tuples of this "wrap-around" type (block of length k may cross the boundary in $k-1$ ways). These tuples are not convenient for our analysis, so we just bound their entropy by k. In this way we obtain the following upper bound:

$$H^{\mathrm{a}}_{K,N}(\alpha) = H(\alpha_{KI} \ldots \alpha_{KI+K-1}) \leqslant \\ \leqslant \frac{1}{k}\left(\sum_{s=0}^{K-k} H(\alpha_{KI+s} \ldots \alpha_{KI+s+k-1}) + (k-1)k\right).$$

Adding $k-1$ terms (replacing the wrap-around terms by some other entropies), we increase the right hand side:

$$H^{\mathrm{a}}_{K,N}(\alpha) \leqslant \frac{1}{k}\left(\sum_{s=0}^{K-1} H(\alpha_{KI+s} \ldots \alpha_{KI+s+k-1}) + (k-1)k\right).$$

Let us look at the variable $\alpha_{KI+s} \ldots \alpha_{KI+s+k-1}$ in the right hand side for some fixed s. It has the same distribution as the random non-aligned k-bit block $\alpha_J \ldots \alpha_{J+k-1}$ for uniformly chosen J in $\{0, \ldots, NK-1\}$ conditional on the event "$J \bmod K = s$":

$$H(\alpha_{KI+s} \ldots \alpha_{KI+s+k-1}) = H(\alpha_J \ldots \alpha_{J+k-1} \,|\, J \bmod K = s).$$

The average of these K entropies (for $s = 0, \ldots, K-1$) is the conditional entropy $H(\alpha_J \ldots \alpha_{J+k-1} \,|\, J \bmod K)$ that does not exceed the unconditional entropy. So we get

$$H^{\mathrm{a}}_{K,N}(\alpha) \leqslant \frac{1}{k}\left(K \cdot H^{\mathrm{na}}_{k,KN}(\alpha) + (k-1)k\right).$$

By taking the $\liminf$ as $N \to \infty$ we obtain

$$\frac{H^{\mathrm{a}}_{K}(\alpha)}{K} = \liminf_{N\to\infty} \frac{H^{\mathrm{a}}_{K,N}(\alpha)}{K} \leqslant \liminf_{N\to\infty} \frac{H^{\mathrm{na}}_{k,KN}(\alpha)}{k} + O\left(\frac{k}{K}\right).$$

However, the lim inf in the right hand side is taken over multiples of K and we want it to be over all indices. Formally, it remains to show that

$$\liminf_{N\to\infty} \frac{H^{\mathrm{na}}_{k,KN}(\alpha)}{k} = \liminf_{N\to\infty} \frac{H^{\mathrm{na}}_{k,N}(\alpha)}{k}$$

as the latter is by definition equal to $H^{\mathrm{na}}_k(\alpha)/k$. Indeed, the statistical distance between distributions on the first KN (non-aligned) blocks and the distribution on the first $KN + r$ blocks (where r the remainder modulo K) tends to zero since the first distribution is the second one conditioned on the event whose probability converges to 1 (i.e., the event "the randomly chosen block is not among the r last ones" whose probability is $KN/(KN + r)$). □

As we have mentioned, this result implies that non-aligned and aligned versions of normality (uniform distribution on non-aligned and aligned blocks) are equivalent. However, note the asymptotic nature of this argument: to prove that the distribution of (say) non-aligned k-bit blocks is uniform, it is not enough to know that aligned k-bit blocks have uniform distribution; we need to know that the distribution of K-bit blocks is uniform for arbitrarily large values of K. This is unavoidable, as the following result shows.

Theorem 2.

(a) *For all k there exists an infinite sequence α such that $H^{\mathrm{na}}_2(\alpha) < 2$ and $H^{\mathrm{a}}_i(\alpha) = i$ for all $i \leqslant k$.*
(b) *For all k there exists an infinite sequence α such that $H^{\mathrm{a}}_2(\alpha) < 2$ and $H^{\mathrm{na}}_i(\alpha) = i$ for all $i \leqslant k$.*

Proof. (a) Consider all k-bit strings. It is easy to arrange them in some order $B_0, B_1, \ldots$ such that the last bit of B_i is the same as the first bit of B_{i+1}, for all i, and the last bit of the last block is the same as the first bit of the first block. For example, consider (for every $x \in \{0,1\}^{k-2}$) four k-bit strings $0x0, 0x1, 1x1, 1x0$ and concatenate these 2^{k-2} quadruples in arbitrary order.

Then consider a periodic sequence with period $B_0B_1 \ldots B_{2^k-1}$. Obviously all aligned k-bit blocks have the same frequency, so $H^{\mathrm{a}}_k(\alpha) = k$. However, for non-aligned bit blocks of length 2 we have two cases: this pair can be completely inside some B_i, or be on the boundary between blocks. The pairs of the first type are balanced (since we have all possible k-bit blocks), but the boundary pairs could be only 00 or 11 due to our construction. So the non-aligned frequency of these two blocks is $1/4+\Omega(1/k)$, and for two other blocks we have $1/4-\Omega(1/k)$, so $H^{\mathrm{na}}_2(\alpha) < 2$.

However, in this construction we do not necessarily have that $H^{\mathrm{a}}_i(\alpha) = i$ for $i < k$. But this is easy to fix. Note that $H^{\mathrm{a}}_k(\alpha) = k$ implies $H^{\mathrm{a}}_i(\alpha) = i$ whenever i is a divisor of k. So we can just use the same construction with blocks of length $k!$ instead of k.

(b) Now let us consider a sequence constructed in the same way, but blocks $B_0, B_1, \ldots, B_{2^k-1}$ go in the lexicographical ordering. First let us note that all k-bit blocks have the same *non-aligned* frequencies in the periodic sequence with

period $B_0B_1 \dots B_{2^k-1}$. (For aligned k-blocks it was obvious, but the non-aligned case needs some proof.) Indeed, consider some k-bit string U; we need to show that it appears exactly k times in the (looped) sequence $B_0B_1 \dots B_{2^k-1}$. In fact, it appears exactly once for each position modulo k. For example, it appears once among the blocks B_i. Why the same it true for some other position $s \bmod k$ where the $k-s$ first bits of U appear as a suffix of B_{i-1} and the last s bits of U appear as a prefix of B_i? Note that $(k-s)$-bit suffixes of $B_0, B_1, B_2, \dots$ form a cycle modulo 2^{k-s}, so the first $k-s$ bits of U uniquely determine the *last* $k-s$ bits of B_i, whereas the first s bits of B_i are just written in the s-bit suffix of U.

This implies that non-aligned frequencies for all k-bit blocks are the same. Therefore, they are the same also for all smaller values of k. In particular, we can assume for the rest that k is odd.

Now let us consider *aligned* blocks of size 2. We will show that aligned frequency of the block 10 in the sequence $B_0B_1 \dots B_{2^k-1}$ is $1/4 - \Omega(1/k)$. Since k is odd (see above), when we cut our sequence into blocks of size 2, there are "border" blocks that cross the boundaries between B_i and B_{i+1}, and other non-border blocks. Each second boundary is crossed (between B_0 and B_1, then B_2 and B_3, and so on), so the border blocks *all have the first bit* 0. In particular, 10 never appears on such positions. This creates discrepancy of order $1/k$ for 10, and we should check that it is not compensated by non-boundary blocks. In the blocks B_i with even i we delete that last bit and cut the rest into bit pairs. After deleting the last bit we have all possible $(k-1)$-bit strings, so no discrepancy arises here. In the blocks B_i with odd i we delete the first bit, and then cut the rest into bit pairs. In the last pair the last bit is 1 (since i is odd), so once again we never have 10 here, as required (the other positions are balanced). □

3 Superadditive complexity measures

The finite-state dimension is a scaled-down version of effective Hausdorff dimension [8]. The effective Hausdorff dimension of a sequence $\alpha = \alpha_0\alpha_1 \dots$ can be equivalently defined as the $\liminf \mathrm{C}(\alpha_0 \dots \alpha_{N-1})/N$, where C stands for the Kolmogorov complexity function [9,10]. We use here plain complexity, but prefix, a priori or monotone complexity (see, e.g., [14, Chapter 6]) will work as well, since they all differ only by $O(\log n)$ for n-bit strings (see, e.g., [14] for more details about Kolmogorov complexity and effective dimension). It is natural to look for a similar characterization of finite-state dimension in terms of compressibility. Such a characterization was given in [7, Section 7]. However, it did not use a complexity notion that can replace C in the definition of effective Hausdorff dimension, using finite-state compressors instead. A suitable complexity notion was introduced in [13], and it indeed gives the desired characterization. We may also use superadditive upper bounds for Kolmogorov complexity. In this extended abstract we present only a version that does not mention Kolmogorov complexity or finite-state machines at all.

Consider a non-negative function F defined on strings. Recall that F is *superadditive* if $F(xy) \geqslant F(x) + F(y)$ for all x and y. We call F *calibrated* if for

every n the sum $\sum 2^{-F(x)}$ taken over all strings x of length n does not exceed some constant (not depending on n).

Theorem 3. *Let $\alpha = \alpha_0\alpha_1\alpha_2\ldots$ be an infinite bit sequence. Then*

$$\mathrm{FSD}(\alpha) = \inf_F \left(\liminf_{N\to\infty} \frac{F(\alpha_0 \ldots \alpha_{N-1})}{N} \right),$$

where the infimum is taken over all superadditive calibrated $F\colon \{0,1\}^* \to [0,+\infty)$.

Proof. We start with an upper bound for the finite-state dimension. Let F be a superadditive calibrated function. We need to show that

$$\mathrm{FSD}(\alpha) \leqslant \liminf_{N\to\infty} \frac{F(\alpha_0 \ldots \alpha_{N-1})}{N}.$$

Since $\mathrm{FSD}(\alpha)$ can be defined as $\lim_k H^{\mathrm{a}}_k(\alpha)/k$, it is enough to prove that

$$H^{\mathrm{a}}_k(\alpha)/k \leqslant \liminf_{N\to\infty} \frac{F(\alpha_0 \ldots \alpha_{N-1})}{N} + O(1/k) \qquad (*)$$

for all k. Fix some $k \in \mathbb{N}$. We can split $\alpha_0 \ldots \alpha_{N-1}$ into $M = \lfloor N/k \rfloor$ aligned k-bit blocks $b_1, \ldots, b_M$ and a tail of length less than k. Since F is superadditive, its value of $\alpha_0 \ldots \alpha_{N-1}$ is at least the sum of its values on blocks $b_1, \ldots, b_M$ (plus the value on the tail; it is non-negative and we ignore it). So we need a lower bound for the sum $F(b_1) + \ldots + F(b_M)$.

How do we get such a bound? We know that the sum of $2^{-F(b)}$ (taken over all blocks b of length k) is bounded by some constant c that does not depend on k. Assume first for simplicity that this constant is 1 and all values of F are integers. Then there exists a prefix-free code for all k-bit blocks where every block b has code of length at most $F(b)$. Then the sum $F(b_1) + \ldots + F(b_M)$, divided by M, is an average code length for the distribution with entropy $H^{\mathrm{a}}_{k,M}(\alpha)$, therefore

$$F(b_1) + \ldots + F(b_M) \geqslant M H^{\mathrm{a}}_{k,M}(\alpha),$$

and

$$F(\alpha_0 \ldots \alpha_{N-1}) \geqslant \lfloor N/k \rfloor H^{\mathrm{a}}_{k,\lfloor N/k \rfloor}(\alpha).$$

Now, dividing both sides by N and taking the $\liminf$, we get the desired inequality $(*)$ even without $O(1/k)$ term. This term appears when we recall that the sum of $2^{-F(b)}$ over all blocks of length k is bounded by a constant (instead of 1) and that the values of F are not necessary integers. To rescue the argument, we need to add some constant to F and perform rounding that adds a constant term to the average code length bound. We get

$$F(b_1) + \ldots + F(b_M) \geqslant M(H^{\mathrm{a}}_{k,M}(\alpha) - O(1))$$

and

$$F(\alpha_0 \ldots \alpha_{N-1}) \geqslant \lfloor N/k \rfloor (H^{\mathrm{a}}_{k,\lfloor N/k \rfloor}(\alpha) - O(1)).$$

Dividing by N, we get a correction of order $O(1/k)$, as claimed.

For the other direction, we need to assume that $H^{\mathrm{a}}_k(\alpha)/k$ is small (less than some threshold) for some k and construct a calibrated superadditive function F such that $\liminf F(\alpha_0 \ldots \alpha_{N-1})/N$ is small (does not exceed the same threshold). For that, we need some general method to construct superadditive calibrated functions. This method is a finite-state version of the a priori complexity notion from algorithmic information theory [14, Section 5.3]. Here it is.

Consider a finite set S of vertices (states). Assume that each vertex has two outgoing edges labeled by $(0, p_0)$ and $(1, p_1)$, where p_0 and p_1 are some non-negative reals such that $p_0 + p_1 = 1$. Then we may consider a probabilistic process: being in state s, the machine emits 0 (with probability p_0) or 1 (with probability p_1), and changes state following the corresponding edge. In addition to such a labeled graph G, fix some state $s \in S$ as an initial state. Then we get a probabilistic algorithm that emits bits, and the corresponding measure $P_{G,s}$ on the space of bit sequences. Let $P_{G,s}(u)$ be the probability of the event "starting from s, the process emits a bit sequence with prefix u". For each k the sum of $P_{G,s}(u)$ over all strings u of length k is exactly 1, so the function $u \mapsto -\log_2 P_{G,s}(u)$ is calibrated. However, it may not be superadditive. To get superadditivity, we take the maximum probability over all initial states s.

Lemma 3. *Let G be a labeled graph of the described type, and all probabilities on labels are positive.*[7] *Then the function $F_G(u) = -\log \max_{s \in S} P_{G,s}(u)$ is calibrated and superadditive.*

Proof (of Lemma 3). (Calibration) Since $\max_{s \in S}$ does not exceed $\sum_{s \in S}$, we conclude that the sum of $2^{-F_G(u)}$ over all strings of given length does not exceed the number of states.

(Superadditivity) We need to prove that

$$\max_{s \in S} P_{G,s}(uv) \leqslant \max_{s \in S} P_{G,s}(u) \cdot \max_{s' \in S} P_{G,s'}(v).$$

We need an upper bound for $P_{G,s}(uv)$ for each s. Indeed, the probability of emitting uv starting from s is equal to the product of the probability of emitting u, starting from s, and the conditional probability of emitting v if u was emitted before. The first probability is $P_{G,s}(u)$ (and does not exceed the maximal value taken over all s). The second probability is $P_{G,s'}(v)$, where s' is the state s' after emitting u. Lemma 3 is proven. □

Now assume that $H^{\mathrm{a}}_k(\alpha)/k$ (for some k) is less than some threshold β. This means that there exists a sequence of prefixes of α such that the entropies of corresponding aligned distributions on $\{0,1\}^k$ converge to some number less than βk. Compactness arguments show that we may assume that the corresponding distributions on $\{0,1\}^k$ converge to some distribution Q whose entropy $H(Q)$ is less that βk. Assume for now that all blocks have positive Q-probabilities. Consider a probabilistic process that generates a concatenation of independent k-bit

[7] This is a technical condition needed to avoid infinities in the logarithms.

strings each having distribution Q. To generate one string according to Q, we generate its bits sequentially, with corresponding conditional probabilities. So the state is the sequence of bits that are already generated; the states form a tree. Finally, generating the last (kth) bit of this string, we return to the initial state (the root of this tree) and are ready to generate new independent strings with the same distribution.

If G is the labeled graph constructed in this way, all labels are positive (recall that we assume that all Q-values are positive). If s is the root, then $P_{G,s}(b_0 \dots b_{m-1}) = Q(b_0) \cdot \ldots \cdot Q(b_{m-1})$ for arbitrary k-bit blocks $b_0, \dots, b_{m-1}$. Now let $b_0 b_2 \dots b_{m-1}$ be the prefix of α from the subsequence of prefixes where the corresponding distributions converge to Q. If $F(u)$ is defined as $-\log P_{G,s}(u)$, then $F(b_0 \dots b_{m-1}) = \sum_{i=0}^{m-1}(-\log Q(b_i))$. Recall that the frequencies of all k-bit blocks among $b_0, \dots, b_{m-1}$ converge to Q. Therefore,

$$F(b_0 \dots b_{m-1}) = (H(Q) + o(1))m < \beta k m$$

for sufficiently large m such that the prefix $b_0 \dots b_{m-1}$ belongs to the subsequence. Dividing both sides by the length km, we get $\liminf_N F(\alpha_0 \dots \alpha_{N-1})/N \leqslant \beta$. The only problem is that $F(u)$ may not be superadditive, but we can replace it by a smaller superadditive calibrated function $-\log P_G(u)$ (taking the maximum of probabilities over all states).

This ends the proof for the case when Q is everywhere positive. If not, we may consider another distribution Q' that is close to Q but has all positive probabilities. Then $F(b_0 \dots b_{m-1})$ will be bigger, and the increase is Kullback – Leibler divergence between Q and Q'. So we just need to make this divergences less than $\beta k - H(Q)$.

Theorem 3 is proven. □

4 Sufficient condition for normality

Assume that some non-empty strings $x_1, x_2, \dots$ are given, and consider the infinite sequence $\varkappa = x_1 x_2 \dots$ obtained by their concatenation. The following theorem provides some conditions that guarantee that $\varkappa$ is a normal sequence.

Theorem 4. *Let L_n be the average length of the first n strings, i.e., $L_n = (|x_1| + \ldots + |x_n|)/n$. Let C_n be the average Kolmogorov complexity of the same strings, i.e., $C_n = (\mathrm{C}(x_1) + \ldots + \mathrm{C}(x_n))/n$. Assume that $|x_n|/(|x_1| + \ldots + |x_{n-1}|) \to 0$ and $L_n \to \infty$ as $n \to \infty$. If $C_n/L_n \to 1$ as $n \to \infty$, then $\varkappa = x_1 x_2 \dots$ is normal. In general,* $\mathrm{FSD}(\varkappa) \geqslant \liminf_{n\to\infty} C_n/L_n$.

Recall that normal sequences can be defined as sequences of finite-state dimension 1.

For example, in the Champernowne sequence the string x_n is the binary representation of n. It is easy to check all three conditions (the latter one uses that the average Kolmogorov complexity of k-bit strings is $k - O(1)$).

This theorem and its proof require some notions and results from algorithmic information theory (all needed information can be found, e.g., in [14]): the notion of plain Kolmogorov complexity $\mathrm{C}(x)$ in used in its statement, the notion of a priori complexity (the logarithm of the continuous a priori probability) is used in the proof. However, this theorem has a corollary that can be formulated without Kolmogorov complexity. For that we consider a random variable i uniformly distributed in $\{1, \ldots, n\}$, random variable x_i whose value are binary strings, and replace C_n by the entropy H_n of this variable. (If all x_i are different, this entropy is $\log n$.) Again, if $H_n/L_n \to 1$, then $\varkappa$ is normal, and $\mathrm{FSD}(\varkappa) \geqslant \liminf H_n/L_n$ in the general case. To derive this corollary, we note that the difference between a priori and prefix complexity is negligible (logarithmic compared to length, see below the comparison between a priori and plain complexities), and prefix complexity provides a prefix-free code for the random variable x_i (with random i), so the average length of the code is at least the Shannon entropy of this variable.

Proof (of Theorem 4). To prove this result, we need to recall the proof of Theorem 3 and note that we can restrict the $\inf_F$ in the right hand side to functions F that are computable upper bounds for the a priori complexity up to $O(1)$ precision (see [14, Section 5.1] for the definition). Indeed, in the proof we have constructed a distribution on the Cantor space (product of distribution Q on k-bit blocks). If Q were computable, then all the transition probabilities in the graph G we constructed would be computable, and $P_{G,s}$ would be a computable measure on the Cantor space for each s, therefore its negative logarithm would be an upper bound for a priori complexity (up to $O(1)$ precision), and the same is true for the minimum over (finitely many) states s.

However, we may not assume that Q is computable: it is the limit distribution in a sequence of prefixes and may be arbitrary. Still (see the discussion above) we may always choose Q' that is close to Q, is computable (even rational) and has non-zero probabilities.

Therefore it remains to show that for every F that is a superadditive upper bound for a priori complexity, the liminf of $F(u)/|u|$, where u is a prefix of $\varkappa$, is at least $\liminf_n C_n/L_n$. If u ends on the block boundary, i.e., if $u = x_1 \ldots x_n$ for some n, then

$$F(u) = F(x_1 \ldots x_n) \geqslant F(x_1) + \ldots + F(x_n) \geqslant \mathrm{KA}(x_1) + \ldots + \mathrm{KA}(x_n) - O(n),$$

where KA is a priori complexity (we use superadditivity of F and recall that F is an upper bound for KA up to $O(1)$ additive term). Assume for a while that we have plain complexity C in this inequality. Then we may continue and write $F(u) \geqslant \mathrm{C}(x_1) + \ldots + \mathrm{C}(x_n) - O(n) = nC_n - O(n)$ and $|u| = nL_n$, so $F(u)/|u| \geqslant C_n/L_n - O(1/L_n)$, and the last term is $o(1)$, since $L_n \to \infty$ as $n \to \infty$.

Now we should consider u that do not end on the block boundary. We can delete the last incomplete block and get slightly shorter u'. For this u' we use the same bound as before, and due to the superadditivity it works as a bound for u. However, we have $|u|$ in the denominator, not $|u'|$. This does not change

the lim inf, since we assume that $|x_n| = o(|x_1| + \ldots + |x_{n-1}|)$, so the length of the incomplete block is negligible compared to the total length of previous complete blocks, and the correction factor converges to 1.

Finally, the difference between plain and a priori complexity is $O(\log m)$ for strings of length m. Therefore, we get a bound (for prefixes $u = x_1 \ldots x_n$)

$$\frac{F(u)}{|u|} \geqslant \frac{\mathrm{KA}(x_1) + \ldots + \mathrm{KA}(x_n) - O(n)}{|x_1| + \ldots + |x_n|} \geqslant$$
$$\geqslant \frac{\mathrm{C}(x_1) + \ldots + \mathrm{C}(x_n) - O(\log |x_1| + \ldots + \log |x_n|) - O(n)}{|x_1| + \ldots + |x_n|}.$$

Both O-terms do not change the limit; we have already discussed this for $O(n)$ (recall that n is small compared to the total length, since $L_n \to \infty$), and the convexity of logarithm (Cauchy inequality) allows us to write

$$\frac{\log |x_1| + \ldots + \log |x_n|}{|x_1| + \ldots + |x_n|} \leqslant \frac{n \cdot \log \left(|x_1|/n + \ldots + |x_n|/n\right)}{|x_1| + \ldots + |x_n|} = \frac{\log L_n}{L_n} \to 0.$$

Theorem 4 is proven. □

As we have noted, this sufficient condition implies the normality of the Champernowne number [5]. It is also easy to see that Copeland – Erdös criterion [6] can be derived in the same way. In this result some integers are skipped, but in such a way that the bit length of the ith remaining integer is still $(1 + o(1)) \log i$, and the sufficient condition can be still applied. More work is needed to derive the result of Besicovitch [1] saying that that concatenated binary representations of perfect squares form a normal number. For this example x_m is a binary representation of m^2, has length about $2 \log m$ and complexity about m, so we get only the lower bound $1/2$ for its finite-state dimension from Theorem 4. To prove normality, we should split the string x_m into two halves of the same length $x_m = y_m z_m$. It is easy to see that the most significant half of m^2 determines m almost uniquely, so the complexity of y_m is close to the complexity of m. For z_m it is not the case: if m has j trailing zeros in the binary representation, then m^2 has $2j$ trailing zeros and its complexity decreases at least by $j - O(1)$ compared to the complexity of m. A simple analysis shows that this estimate is exact, and since the average number of trailing zeros in a random s-bit string is $O(1)$, we get the required bound.

Now let us give more details. Let z_m be the suffix of x_m of length $\lfloor \log_2 m \rfloor + 1$, i.e., the length of z_m is exactly the length of the binary representation of m, and let $y_m \in \{0,1\}^*$ be the corresponding prefix, i.e., $x_m = y_m z_m$. Note that the length of y_m is $\log_2 m + O(1)$. Therefore, the average length of $y_1, z_1, \ldots, y_m, z_m$ is $\log_2 m + O(1)$, and it remains to show that the average Kolmogorov complexity of these strings is $\log m \cdot (1 - o(1))$. We will do this by showing that the average of conditional complexities $C(i|y_i), C(i|z_i)$ over $i \in \{1, \ldots, m\}$ is $O(\log \log m)$. Since we already know that the average of $C(i)$ over $i \in \{1, \ldots, m\}$ is $\log_2 m + O(1)$, this would give the desired bound. Indeed, this follows from the chain rule:

$$C(y_i) \geqslant C(i) - C(i|y_i) - O(\log \log m), \qquad C(z_i) \geqslant C(i) - C(i|z_i) - O(\log \log m).$$

For the first part we will show not only that the average of $C(i|y_i)$ is at most $O(\log\log m)$, but that the same is true for *every* i. Indeed, assume that you know y_i and the length of the binary representation of i (let us denote this quantity by k). Then there is at most $O(1)$ different j of length k such that $y_j = y_i$. Indeed, the difference between i^2 and j^2 is $|i^2 - j^2| = \Omega(|i-j| \cdot 2^k)$. On the other hand, by definition we have that $i^2 = 2^k y_i + z_i, j^2 = 2^k y_i + z_j$, which means that that difference between i^2 and j^2 is $|z_i - z_j| = O(2^k)$. Therefore, if for the k-bit number j we have $y_j = y_i$, then j differs from i only by some constant. We need only to specify the length of the binary representation of i, using $O(\log\log m)$ bits.

As we mentioned earlier, we need a more complicated argument to show that the average of $C(i|z_i)$ is $O(\log\log m)$. The reason is that it is true only for averages: there are some i such that $C(i|z_i)$ is of order $\log m$. We have to show somehow that the number of "bad" i is negligible. To do so we need the following technical lemma.

Lemma 4. *Let $t(n)$ denote the largest natural number d such that n is divisible by 2^d (i.e., $t(n)$ is the number of trailing zeros in the binary representation of n). Then for every $a \in \mathbb{N}$ the number of $x \in \{0, 1, \ldots, 2^k - 1\}$ such that $x^2 \equiv a^2 \pmod{2^k}$ is at most $O(2^{t(a)})$.*

Proof. Indeed, assume that a has z trailing zeros and $x^2 = a^2 \pmod{2^k}$ for some $x \in \{0, 1 \ldots, 2^k - 1\}$. Then $x^2 - a^2 = (x-a)(x+a)$ is a multiple of 2^k, therefore $x - a$ is a multiple of 2^u and $x + a$ is a multiple of 2^v for some u, v such that $u + v = k$. Then $2a = (x + a) - (x - a)$ is a multiple of $2^{\min(u,v)}$, so $\min(u, v) \leqslant z - 1$. Then $\max(u, v) \geqslant k - z - 1$, so one of $x - a$ and $x + a$ is a multiple of 2^{k-z-1}, and each case contributes at most $2^{z+1} = O(2^z)$ solutions for the equation $x^2 = a^2 \pmod{2^k}$. □

This lemma implies that $C(i|z_i) = O(t(i) + \log\log m)$. Indeed, assume that z_i and the length of the binary representation of i (denoted by k in the sequel) are given. Suppose that j is a k-bit number satisfying $z_j = z_i$. Then, as $i^2 = 2^k \cdot y_i + z_i, j^2 = 2^k \cdot y_j + z_i$, the difference between i^2 and j^2 is the multiple of 2^k. By Lemma 4 the number of such j is $O(2^{t(a)})$, i.e., specifying one of them requires $t(a) + O(1)$ bits.

As the average of $t(i)$ is $O(1)$, this gives the required bound for the average value of $C(i|z_i)$.

Calude, Salomaa and Roblot [3, Section 6] define a version of automatic complexity in the following way. A deterministic transducer (finite automaton that reads an input string and at each step produces some number of output bits) maps a description string to a string to be described, and the complexity of y is measured as the minimal sum of the sizes of the transducer and the input string needed to produce y; the minimum is taken over all pairs (transducer, input string) producing y. The size of the transducer is measured via some encoding, so the complexity function depends on the choice of this encoding. "It will be interesting to check whether finite-state random strings are Borel normal" [3, p. 5677]. Since normality is defined for infinite sequences, one probably

should interpret this question in the following way: is it true that normal infinite sequences can be characterized as sequences whose prefixes have finite-state complexity close to length?

It turns out [4] that this is only a sufficient condition, not a criterion. More precisely, there is a normal sequence such that finite-state complexity of its first n bits is $o(n)$. This example is also an easy consequence of Theorem 4. Indeed, let us denote the complexity defined in [3] by $\mathrm{CSR}(x)$. It depends on the choice of the encoding for transducers, but the following theorem is true for every encoding, so we assume that some encoding is fixed and omit it in the notation.

Theorem 5 ([4]). (a) *If a sequence $\alpha = a_0a_1\ldots$ is not normal, then there exists some $c < 1$ such that the $\mathrm{CSR}(a_0\ldots a_{n-1}) < cn$ for infinitely many n.*

(b) $\liminf \mathrm{CSR}(b_0\ldots b_{n-1})/n = 0$ *for some normal sequence $\beta = b_0b_1\ldots$*

Proof. The first part of the statement can be proven using Shannon coding in the same way as in [13]. For the second part we construct an example of a normal sequence using Champernowne's idea and Theorem 4. The sequence will have the form $\beta = (B_1)^{n_1}(B_2)^{n_2}\ldots$; here B_i is the concatenation of all strings of length i (say, in lexicographical ordering, but this does not matter), and n_i is a fast growing sequence of integers.

To choose n_i, let us note first that for a periodic sequence (of the form XY^∞) the CSR-complexity of its prefixes of the form XY^k is o(length). Indeed, we may consider a transducer that first outputs X, then outputs Y for each input bit 1. So $\mathrm{CSR}(XY^m) = m + O(1)$, and the compression ratio is about $1/|Y|$. To get o(length), we use Y^c for some constant c as a period to improve the compression.

Now consider the complexity/length ratio for the prefixes of β if the sequence n_i grows fast enough. Indeed, assume that $n_1, n_2, \ldots, n_k$ are already chosen and we now choose the value of n_{k+1}. We may use the bound explained in the previous paragraph and let $X = (B_1)^{n_1}\ldots(B_k)^{n_k}$ and $Y = B_{k+1}$. For sufficiently large n_{k+1} we get arbitrarily small complexity/length ratio. (Note that good compression is guaranteed only for some prefixes; when increasing k, we need to switch to another transducer, and we know nothing about the length of its encoding.)

It remains to apply Theorem 4 to show that for some fast growing sequence $n_1, n_2, \ldots$ the sequence β is normal. We apply the criterion by splitting B_k into pieces of length k (so all strings of length k appear once in this decomposition of B_k). We already know that the average Kolmogorov complexity of the pieces in B_k is $k - O(1)$ (and the length of all pieces is k). This is enough to satisfy the conditions of Theorem 4 *if $x_1\ldots x_n$ ends on the boundary of the block B_k*. But this is not guaranteed; in general we need also to consider the last incomplete group of blocks that form a prefix of some B_k. The total length of these blocks is bounded by $|B_k|$, i.e., by $k2^k$. We need this group to be short compared to the rest, and this will be guaranteed if n_{k-1} (the lower bound for the length of the previous part) is much bigger than $k2^k$. And we assume that n_k grow very fast, so this condition is easy to satisfy. Theorem 5 is proven. □

References

1. Besicovitch, A.: The asymptotic distribution of the numerals in the decimal representation of the squares of the natural numbers. Mathematische Zeitschrift **39**(1), 146–156 (1935). https://doi.org/10.1007/BF01201350
2. Bourke, C., Hitchcock, J.M., Vinodchandran, N.: Entropy rates and finite-state dimension. Theoretical Computer Science **349**(3), 392–406 (2005). https://doi.org/10.1016/j.tcs.2005.09.040
3. Calude, C.S., Salomaa, K., Roblot, T.K.: Finite state complexity. Theoretical Computer Science **412**(41), 5668–5677 (2011). https://doi.org/10.1016/j.tcs.2011.06.021
4. Calude, C.S., Staiger, L., Stephan, F.: Finite state incompressible infinite sequences. Information and Computation **247**, 23–36 (2016). https://doi.org/10.1016/j.ic.2015.11.003
5. Champernowne, D.G.: The construction of decimals normal in the scale of ten. Journal of the London Mathematical Society **1**(4), 254–260 (1933). https://doi.org/10.1112/jlms/s1-8.4.254
6. Copeland, A.H., Erdös, P.: Note on normal numbers. Bulletin of the American Mathematical Society **52**(10), 857–860 (1946). https://doi.org/10.1090/S0002-9904-1946-08657-7
7. Dai, J.J., Lathrop, J.I., Lutz, J.H., Mayordomo, E.: Finite-state dimension. Theoretical Computer Science **310**(1-3), 1–33 (2004). https://doi.org/10.1016/S0304-3975(03)00244-5
8. Lutz, J.H.: Dimension in complexity classes. SIAM Journal on Computing **32**(5), 1236–1259 (2003). https://doi.org/0.1137/S0097539701417723
9. Lutz, J.H.: The dimensions of individual strings and sequences. Information and Computation **187**(1), 49–79 (2003). https://doi.org/10.1016/S0890-5401(03)00187-1
10. Mayordomo, E.: A Kolmogorov complexity characterization of constructive Hausdorff dimension. Information Processing Letters **84**(1), 1–3 (2002). https://doi.org/10.1016/S0020-0190(02)00343-5
11. Niven, I., Zuckerman, H., et al.: On the definition of normal numbers. Pacific Journal of Mathematics **1**(1), 103–109 (1951). https://doi.org/10.2140/pjm.1951.1.103
12. Pillai, S.: On normal numbers. Proceedings of the Indian Academy of Sciences, Section A **12**(2), 179–184 (1940). https://doi.org/10.1007/BF03173913
13. Shen, A.: Automatic Kolmogorov complexity and normality revisited. In: International Symposium on Fundamentals of Computation Theory. pp. 418–430. Springer (2017). https://doi.org/10.1007/978-3-662-55751-8_33
14. Shen, A., Uspensky, V.A., Vereshchagin, N.: Kolmogorov complexity and algorithmic randomness, vol. 220. American Mathematical Soc. (2017). https://doi.org/10.1090/surv/220