Finite State Incompressible Infinite Sequences

Cristian S. Calude[a], Ludwig Staiger[b], Frank Stephan[c]

[a] *Dept. of Computer Science, University of Auckland, Auckland, New Zealand*
[b] *Martin-Luther-Universität Halle-Wittenberg, Institut für Informatik, D - 06099 Halle, Germany*
[c] *Department of Mathematics and Department of Computer Science, National University of Singapore, Singapore 119076, Republic of Singapore*

Abstract

In this paper we define and study finite state complexity of finite strings and infinite sequences and study connections of these complexity notions to randomness and normality. We show that the finite state complexity does not only depend on the codes for finite transducers, but also on how the codes are mapped to transducers. As a consequence we relate the finite state complexity to the plain (Kolmogorov) complexity, to the process complexity and to prefix-free complexity. Working with prefix-free sets of codes we characterise Martin-Löf random sequences in terms of finite state complexity: the weak power of finite transducers is compensated by the high complexity of enumeration of finite transducers. We also prove that every finite state incompressible sequence is normal, but the converse implication is not true. These results also show that our definition of finite state incompressibility is stronger than all other known forms of finite automata based incompress-

---

Email addresses: `cristian@cs.auckland.ac.nz` (Cristian S. Calude), `staiger@informatik.uni-halle.de` (Ludwig Staiger), `fstephan@comp.nus.edu.sg` (Frank Stephan)

This work was done in part during C. S. Calude's visits to the Martin-Luther-Universität Halle-Wittenberg in October 2012 and the National University of Singapore in November 2013 and June 2014, and L. Staiger's visits to the CDMTCS, University of Auckland and the National University of Singapore in March 2013. The work was supported in part by NUS grant R146-000-181-112 (PI F. Stephan).

A preliminary version of this paper has appeared in T. V. Gopal, M. Agrawal, A. Li, B. S. Cooper (eds). *Proceedings of the 11th Annual Conference on Theory and Applications of Models of Computation*, Lecture Notes in Computer Science 8402, Springer, 2014, 50–66.

ibility, in particular the notion related to finite automaton based betting systems introduced by Schnorr and Stimm [35]. The paper concludes with a discussion of open questions.

Key words: AIT, finite state complexity and incompressibility, normality
2010 MSC: 68Q19, 68Q30, 68Q45

---

1. Introduction

Algorithmic Information Theory (AIT) [8, 20, 30, 27] uses various measures of descriptional complexity to define and study various classes of "algorithmically random" finite strings or infinite sequences. The theory, based on the existence of a universal Turing machine (of various types), is very elegant and has produced many important results.

The incomputability of all descriptional complexities is an obstacle towards more "down-to-earth" applications of AIT (e.g. for practical compression). One possibility to avoid incomputability is to restrict the resources available to the universal Turing machine and the result is resource-bounded descriptional complexity [7]. Another approach is to restrict the computational power of the machines used, for example, using context-free grammars or straight-line programs instead of Turing machines [15, 24, 25, 34].

The first connections between finite state machine computations and randomness have been obtained for infinite sequences. Agafonov [1] proved that every subsequence selected from a (Borel) normal sequence by a regular language is also normal. Characterisations of normal infinite sequences have been obtained in terms of finite state gamblers, information lossless finite state compressors and finite state dimension: (a) a sequence is normal if and only if there is no finite state gambler that succeeds on it [35] (see also [6, 17]) and (b) a sequence is normal iff it is incompressible by any information lossless finite state compressor [46]. Doty and Moser [18, 19] used computations with finite transducers for the definition of finite state dimension of infinite sequences. The NFA-complexity of a string [15] can be defined in terms of finite transducers that are called in [15] "NFAs with advice"; the main problem with this approach is that NFAs used for compression can always be assumed to have only one state.

The definition of *finite state complexity of a finite string* x in terms of a computable enumeration of finite transducers and the input strings used by transducers which output x proposed in [10, 11] is utilised to define *finite state incompressible sequences.* We show basic connections of this new notion compared to standard complexity measures in Theorem 9: It lies properly between the plain complexity as a lower bound and the prefix-free complexity as an upper bound in the case that the enumeration of transducers considered is a universal one. Furthermore, while finite state incompressibility depends on the enumeration of finite transducers, many results presented here are *independent* of the chosen enumeration. For example, we show that for every enumeration S every C_S–incompressible sequence is normal, Theorem 22. Furthermore, we can show that a sequence is Martin-Löf random iff it satisfies a strong incompressibility condition (parallel to the one for prefix-free Kolmogorov complexity) for every measure C_S based on some perfect enumeration S. One can furthermore transfer this characterisation to the measure C_S for universal enumerations S.

Finally, we illustrate the dependence of finite state complexity on the enumeration of finite transducers. We prove that in every sequence there are infinitely many finite state complexity dips when the complexity is based on some exotic enumerations.

2. Notation

In this section we introduce the notation used throughout the paper. By $\mathbb{N} = \{0, 1, 2, \ldots\}$ we denote the set of natural numbers. Its elements will be usually denoted by letters $i, \ldots, n$. By $\{0,1\}^*$ we denote the set of all binary strings (words) with ε denoting the empty string; $\{0,1\}^\omega$ is the set of all (infinite) binary sequences. The length of a finite string $x \in \{0,1\}^*$ is denoted by $|x|$. Sequences (infinite strings) are usually denoted by $\mathbf{x}, \mathbf{y}$; the prefix of length n of the sequence $\mathbf{x}$ is denoted by $\mathbf{x} \upharpoonright n$; the nth element of $\mathbf{x}$ is denoted by $\mathbf{x}(n)$.

For $w \in \{0,1\}^*$ and $\beta \in \{0,1\}^* \cup \{0,1\}^\omega$ let $w \cdot \eta$ be their *concatenation.* This concatenation product extends in an obvious way to subsets $L \subseteq \{0,1\}^*$

and $B \subseteq \{0,1\}^* \cup \{0,1\}^\omega$.

By $w \sqsubseteq u$ and $w \sqsubset \mathbf{y}$ we denote that w is a prefix of u and $\mathbf{y}$, respectively, and a prefix-free set $L \subset \{0,1\}^*$ is a set with the property that for all strings $p, q \in \{0,1\}^*$, if $p, pq \in L$ then $p = pq$.

3. Admissible Transducers and Their Enumerations

We consider transducers which try to generate prefixes of infinite binary sequences from shorter binary strings and consider hence the following transducers: An *admissible transducer* is a deterministic transducer given by a finite set of states Q with starting state q_0 and transition functions δ, μ with domain $Q \times \{0,1\}$, and say that the transducer on state q and current input bit a transitions to $q' = \delta(q,a)$ and appends $w = \mu(q,a)$ to the output produced so far.

One can generalise inductively the functions μ and δ by stating that $\mu(q,\varepsilon) = \varepsilon$ and $\mu(q,av) = \mu(q,a) \cdot \mu(\delta(q,a),v)$ for states q and input strings av with a being one bit; similarly, $\delta(q,\varepsilon) = q$ and $\delta(q,av) = \delta(\delta(q,a),v)$. The output $T(v)$ of a transducer T on input-string v is then $\mu(q_0,v)$.

Definition 1. *A partially computable function S mapping binary strings to admissible transducers is called an* enumeration *provided every admissible transducer T has a string $\sigma \in \mathrm{dom}(S)$; for a string $\sigma \in \mathrm{dom}(S)$, the admissible transducer assigned by S to σ is denoted as $S(\sigma) = T^S_\sigma$.*

If the domain $\mathrm{dom}(S)$ is a prefix-free subset of $\{0,1\}^*$ then we call S a prefix-free enumeration.

Next we introduce two subclasses of prefix-free enumerations, that is, enumerations S having a prefix-free domain $\mathrm{dom}(S)$.

Definition 2 (Calude, Salomaa and Roblot [10, 11]). *A* perfect enumeration *S of all admissible transducers is a partially computable function with a prefix-free and computable domain mapping each binary string $\sigma \in \mathrm{dom}(S)$ to an admissible transducer T^S_σ in an onto way.*

Note that partially computable functions with a computable range (as considered here) have a computable inverse, that is, for each input y from the range, an algorithm finds, by searching in parallel over all possible inputs, an

x which is mapped to y. It is known that there are perfect enumerations with a regular domain and that every perfect enumeration S can be improved to a better perfect enumeration S' such that for each c there is a transducer represented by σ in S and σ' in S' and these representations satisfy $|\sigma'| < |\sigma| - c$, [10, 11].

Definition 3. *A* universal enumeration *S of all transducers is a partially computable function with prefix-free domain such that for each other prefix-free enumeration S' of admissible transducers there exists a constant c such that for all σ' in the domain of S', the transducer $T^{S'}_{\sigma'}$ equals some transducer T^S_σ with $\sigma \in \mathrm{dom}(S)$ and $|\sigma| \leq |\sigma'| + c$.*

Note that perfect and universal enumerations are prefix-free enumerations. The construction of a universal enumeration S can be carried over from Kolmogorov complexity: If U is a universal machine for prefix-free Kolmogorov complexity and S' is a perfect enumeration of the admissible transducers, then the domain of S is the set of all σ such that $U(\sigma)$ is defined and in the domain of S' and T^S_σ is $T^{S'}_{U(\sigma)}$. Then, for every further enumeration S'' (also with prefix-free domain) there is a σ at most a constant longer than σ'' such that $U(\sigma)$ outputs an S'-program σ' with $T^{S''}_{\sigma''} = T^{S'}_{\sigma'}$ and so $T^S_\sigma = T^{S''}_{\sigma''}$, that is, there is a constant c such that each transducer with an index of length n in S'' has a further index of length up to $n + c$ in S. Thus U is universal.

Below in Lemma 4 we will show that universal enumerations S of all transducers exist.

4. Complexity and Randomness

Recall that the plain complexity (Kolmogorov) of a string $x \in \{0,1\}^*$ w.r.t. a partially computable function $\varphi : \{0,1\}^* \to \{0,1\}^*$ is $K_\varphi(x) = \inf\{|p| : \varphi(p) = x\}$. It is well-known that there is a universal partially computable function $U : \{0,1\}^* \to \{0,1\}^*$ such that

$$K_U(x) \leq K_\varphi(x) + c_\varphi$$

holds for all strings $x \in \{0,1\}^*$. Here the constant c_φ depends only on U and φ but not on the particular string $x \in \{0,1\}^*$. We will denote the complexity K_U simply by K. Furthermore, in the case that one considers only partially computable functions with prefix-free domain, there are also universal ones among them V, say, and the corresponding complexity K_V, called *prefix complexity* is denoted with H; like K, the prefix-free complexity H depends only up to a constant on the given choice of the underlying universal machine.

Schnorr [36] considered the subclass of partially computable prefix-monotone functions (or *processes*) $\psi : \{0,1\}^* \to \{0,1\}^*$, that is, functions which satisfy the additional property that for strings $v, w \in \mathrm{dom}(\psi)$, if $v \sqsubseteq w$, then $\psi(v) \sqsubseteq \psi(w)$. For this class of functions there is also a universal partially computable prefix-monotone function $W : \{0,1\}^* \to \{0,1\}^*$ such that for every further such ψ (with the same properties) there is a constant c_ψ, depending only on W and ψ, fulfilling

$$K_W(x) \leq K_\psi(x) + c_\psi, \tag{1}$$

for all binary strings $x \in \{0,1\}^*$. As in [20] we denote the complexity induced by the universal function by Km_D. Since processes are arbitrary partial computable functions and partial computable functions with prefix-free domain are processes, the following inequalities hold for all $x \in \{0,1\}^*$.

$$K(x) \leq Km_D(x) + O(1) \text{ and } Km_D(x) \leq H(x) + O(1) \tag{2}$$

Having introduced prefix-free universal partially computable functions V we can now show that universal enumerations S of all transducers in the sense of Definition 3 exist.

Lemma 4. *There is a universal enumeration S of all transducers.*

Proof. Let $(S_i)_{i\in\mathbb{N}}$ be an effective numbering of all enumerations with prefix-free domain, this time not requiring that these S_i have infinite domain. Now define a new prefix-free enumeration S as follows:

$$T^S_{0^i1\sigma} = \begin{cases} T^{S_i}_\sigma, & \text{if } \sigma \in \mathrm{dom}(S_i); \\ \text{undefined}, & \text{otherwise.} \end{cases}$$

For each i and each string x, if x has according to S_i the complexity $c = |\sigma\tau|$ which is witnessed by some σ in the domain of S_i and some input τ with $x = T^{S_i}_\sigma(\tau)$, then $x = T^S_{0^i1\sigma}(\tau)$ as well and x has, according to S, at most the complexity $|0^i 1\sigma\tau| = c + i + 1$. Thus $C_S(x) \leq C_{S_i}(x) + i + 1$, for all strings x where $C_{S_i}(x)$ is defined, hence S is universal. $\square$

Martin-Löf [28] introduced the notion of the random sequences in terms of tests and Schnorr — as cited by Chaitin [13] — characterised them in terms of prefix-free complexity; we take this characterisation as a definition. Furthermore, Schnorr [36] showed that the same definition holds for process complexity.

Definition 5 (Martin-Löf [28]; Schnorr [13, 36]). An infinite sequence $\mathbf{x} \in \{0,1\}^\omega$ is *Martin-Löf random* if there is a constant c such that $H(\mathbf{x} \upharpoonright n) \geq |n| - c$, for all $n \geq 1$. Equivalently, $\mathbf{x}$ is Martin-Löf random iff there is a constant c such that $Km_D(\mathbf{x} \upharpoonright n) \geq |n| - c$, for all $n \geq 1$.

5. Finite State Complexity

For a fixed admissible transducer T, one usually denotes the complexity $C_T(x)$ of a binary string x as the length of the shortest binary string y such that $T(y) = x$. The complexity C_T was proposed in [10, 11] to remedy the incomputability of Kolmogorov complexity (see more about other proposals in [10]). It can be viewed also as an example of the Minimal Description Length Principle [32, 23]. A description of a string x consists of a finite transducer T and another string y such as T "translates" y into x: $C_T(x)$ minimises the sum between the complexity of T and $|y|$.

This definition is now adjusted to enumerations S of admissible transducers.

Definition 6. *Let S be an enumeration of the admissible transducers. For each string x, the complexity $C_S(x)$ is the minimum $|\sigma| + |y|$ taken over all σ in the domain of S and y in the domain of T^S_σ such that $T^S_\sigma(y) = x$.*

This complexity is also called the *finite state complexity based on S* of a given string. Note that if S is universal and S' is any other prefix-free enumeration then there is a constant c such that

$$C_S(x) \leq C_{S'}(x) + c,$$

for all binary strings x. Thus the universal enumerations define an abstract finite state complexity in the same way as it is done for prefix-free, process or plain complexity. The next results relate the complexity C_S for universal enumerations S to the plain complexity K, the prefix-free complexity H and the process complexity Km_D.

We start with a simple property. Note that there is a fixed transducer T_τ^S such that $T_\tau^S(x) = x$, for all x. This implies the following.

Corollary 7. *For every enumeration S there is a constant c_S such that $C_S(x) \leq |x| + c_s$, for all $x \in \{0,1\}^*$.*

Next we give a useful construction combining an enumeration S with a computable partial function φ.

Let $\varphi : \{0,1\}^* \to \{0,1\}^*$ be a computable partial function with domain $\mathrm{dom}(\varphi)$ and S an enumeration of admissible transducers. Define a new enumeration $S[\varphi]$ in the following way:

$$\begin{array}{lcll} T_{0\sigma}^{S[\varphi]}(w) & = & T_\sigma^S(w), & \text{for } w \in \{0,1\}^*, \\ T_{1\sigma}^{S[\varphi]}(\varepsilon) & = & \varepsilon, & \text{if } \sigma \in \mathrm{dom}(\varphi), \\ T_{1\sigma}^{S[\varphi]}(\varepsilon) & = & \text{undefined}, & \text{otherwise, and} \\ T_{1\sigma}^{S[\varphi]}(aw) & = & \varphi(\sigma), & \text{for } a \in \{0,1\} \text{ and } w \in \{0,1\}^* . \end{array} \tag{3}$$

Since $T_\sigma^S = T_{0\sigma}^{S[\varphi]}$, every transducer appears as an image of the new mapping $S[\varphi]$, and, obviously, $S[\varphi]$ is an enumeration of transducers. Then, from Eq. (3) we obtain the following.

Lemma 8. *If S is a enumeration and φ is a computable partial function then the new enumeration $S[\varphi]$ has $\mathrm{dom}(S[\varphi]) = 0 \cdot \mathrm{dom}(S) \cup 1 \cdot \mathrm{dom}(\varphi)$ and $C_{S[\varphi]}(w) \leq K_\varphi(w) + 2$, for all $w \in \{0,1\}^*$.*

If, moreover, S is a prefix-free (perfect, universal) enumeration and $\mathrm{dom}(\varphi)$ is prefix-free then $S[\varphi]$ is also a prefix-free (perfect, universal) enumeration.

The next theorem shows that universal enumerations define intermediate complexities between the process and the prefix-free complexities.

Theorem 9. *Let S be a prefix-free universal enumeration of the admissible transducers. Then there are constants c, c' such that, for all binary strings x,*

$$Km_D(x) \leq C_S(x) + c,\ C_S(x) \leq H(x) + c'\,.$$

Furthermore, one cannot obtain equality up to constant for any of these inequalities.

Proof. For the first inequality, note that if $T_\sigma^S(y) = x$ then σ stems from a prefix-free set and hence there is a plain Turing machine ψ which on input p first searches for a prefix σ of p which is in $\mathrm{dom}(S)$ and, in the case that such a σ is found, outputs $T_\sigma^S(y)$ for the unique y with $\sigma y = p$. Thus the mapping from all σy to $T_\sigma^S(y)$ with $\sigma \in \mathrm{dom}(S)$ and $y \in \mathrm{dom}(T_\sigma^S)$ is partially computable and prefix-monotone. Thus $Km_D(x) \leq C_S(x) + c$ for some constant c.

Theorem 10 below implies that the first inequality is proper.

Let S be a universal enumeration of all admissible transducers and V be prefix-free universal mapping as mentioned in Section 4. Consider the enumeration $S[V]$. Then according to Lemma 8 $\mathrm{dom}(S[V]) = 0 \cdot \mathrm{dom}(S) \cup 1 \cdot \mathrm{dom}(V)$ is prefix-free and $C_{S[V]}(w) \leq H(w)+2$, for all $w \in \{0,1\}^*$. Since S is a universal enumeration, we have also $C_S(w) \leq C_{S[V]}(w)+c_1 \leq H(w)+c_1+2$.

Since $C_S(w) \leq |w|+c$, for some constant c, and $H(w)-|w|$ is unbounded (cf. [8, 20]) one cannot reverse the second inequality to an equality up to constant. □

The properness of one inequality was missing in the previous result. It follows from the following theorem.

Theorem 10. *There is a prefix-monotone partially computable function ψ such that for every prefix-free enumeration S and each constant c there is a binary string x with $K_\psi(x) < C_S(x) - c$.*

Proof. Let Ω be the infinite (Martin-Löf random) binary expansion of a Chaitin Omega number [13] and let Ω_s be an approximation to Ω from the left for s steps. Now define

$$\psi(x) = 0^{\min\{s : x \leq_{lex} \Omega_s\}}.$$

Here $x \leq_{lex} A$ if either A extends x or if for the first $k \in dom(x)$ with $x(k) \neq A(k)$ it holds that $x(k) = 0$ and $A(k) = 1$. Note that this function is partially computable and furthermore it is prefix-monotone. It is defined on all x with $x \leq_{lex} \Omega$. Note that for $x = \Omega \upharpoonright n$, $\psi(x)$ coincides with the convergence module $c_\Omega(n) = \min\{s : \forall m < n\, [\Omega_s(m) = \Omega(m)]\}$.

The goal of the construction is now to show that for all constants c and all prefix-free enumerations S of admissible transducers, almost all prefixes $x \sqsubset \Omega$ satisfy that $\psi(x)$ is larger than the length of any value $T^S_\sigma(y)$ with $|\sigma y| \leq |x| + c$. So fix one prefix-free enumeration S.

The first ingredient for this is to use that for almost all σ, if $T^S_\sigma(y)$ is longer than $\psi(\Omega \upharpoonright |\sigma| + |y| - c)$ then y is shorter than $|\sigma|$. Assume by way of contradiction that this is not be true and that there are infinitely many n with corresponding σ, y such that $n = |\sigma| + |y| - c$ and $|T^S_\sigma(y)| \geq \psi(\Omega \upharpoonright n) = c_\Omega(n)$ and $|\sigma| \leq n/2$. Now one can compute from σ and $|y|$ the maximum length s of an output of $T^S_\sigma(z)$ with $|z| \leq |y|$ and then take $\Omega \upharpoonright n$ as $\Omega_s \upharpoonright n$. Hence $H(\Omega \upharpoonright n)$ is, up to a constant, bounded by $|\sigma| + 2\log(|y|)$ which is bounded by $n/2 + 2\log n$ plus a constant, in contradiction to the fact that $H(\Omega \upharpoonright n) \geq n$ for almost all n. Thus the above assumption cannot be true.

Hence, for the further proof, one has only to consider transducers whose input is at most as long as the code. The corresponding definition would be to let, for each $\sigma \in \mathrm{dom}(S)$, $\varphi(\sigma)$ be the length of the longest output of the form $T^S_\sigma(y)$ with $y \leq |\sigma|$.

Now assume by way of contradiction that there are a constant c and infinitely many $x \sqsubset \Omega$ such that there exists a σ with $|\psi(x)| \leq \varphi(\sigma)$ and $|\sigma| \leq |x| + c$. Then one can construct a prefix-free machine V with the same domain as S such that $V(\sigma)$ for all $\sigma \in \mathrm{dom}(S)$ outputs $z = \Omega_{\varphi(\sigma)} \upharpoonright |\sigma| - c$. As $|\sigma| \leq |x| + c$ it follows that z is a prefix of x and a prefix of Ω.

The domains of V and S are the same, hence V is a partially computable function with prefix-free domain which has for infinitely many prefixes $z \sqsubset \Omega$ an input σ of length up to $|z| + 2c$ with $V(\sigma) = z$, that is, which satisfies $H_V(z) \leq |z| + 2c$ for infinitely many prefixes z of Ω. This again contradicts the fact that Ω is Martin-Löf random, hence this does not happen.

Note that $K_\psi(x) \leq Km_D(x) + c'$ for some constant c'. Now one has, for almost all n that the string $u_n = 0^{c_\Omega(n)}$ satisfies $u_n = \psi(\Omega \upharpoonright n)$ and

$K_\psi(u_n) = n$ and $Km_D(u_n) \le n + c'$ while, for all S and c and almost all n, $C_S(u_n) > n + c$, hence $C_S(u_n) - Km_D(u_n)$ goes to ∞ for $n \to \infty$. So C_S and Km_D cannot be equal up to constant for any prefix-free enumeration S of admissible transducers. $\square$

Furthermore, for enumerations S having a computable domain $\mathrm{dom}(S)$, one can show that there is an algorithm to compute C_S.

Proposition 11. *Let S be an enumeration of the admissible transducers and let $\mathrm{dom}(S)$ be computable. Then the mapping $x \mapsto C_S(x)$ is computable.*

Proof. We have $C_S(x) \le |x| + c$ for some constant c. Now $C_S(x)$ is the length of the shortest σy with $\sigma \in \mathrm{dom}(S)$, $y \in \{0,1\}^*$, $|\sigma y| \le |x| + c$ and $T^S_\sigma(y) = x$. Due to the length-restriction $|\sigma y| \le |x| + c$, the search space is finite and due to the computability of $\mathrm{dom}(S)$ the search can be carried out effectively. $\square$

6. Complexity of Infinite Sequences

Martin-Löf randomness can be formalised using both prefix-free Kolmogorov complexity and process complexity, see Definition 5. Therefore it is natural to ask whether such a characterisation does also hold for the C_S complexity. As an easy consequence of Definition 5 and the sandwich property of Theorem 9 one obtains the following.

Lemma 12. *Let $\mathbf{x} \in \{0,1\}^\omega$. Then $\mathbf{x}$ is Martin-Löf random iff for every universal enumeration of transducers S there is a constant c depending only on $\mathbf{x}$ and S such that for all $n \in \mathbb{N}$ the condition $C_S(\mathbf{x} \restriction n) \ge n - c$ holds.*

One can, however, define randomness also in terms of weaker enumerations.

Theorem 13. *The following statements are equivalent:*

(a) *The sequence $\mathbf{x}$ is not Martin-Löf random;*

(b) *There is a perfect enumeration S such that for every $c > 0$ and almost all $n > 0$ we have $C_S(\mathbf{x} \restriction n) < n - c$;*

(c) *There is a perfect enumeration S such that for every $c > 0$ there exists an $n > 0$ with $C_S(\mathbf{x} \upharpoonright n) < n - c$.*

Proof. If $\mathbf{x}$ is Martin-Löf random then according to Lemma 12 for every prefix-free enumeration S it holds $C_S(\mathbf{x} \upharpoonright n) \geq n - c$ for some constant c and all n. Hence none of the conditions (b) or (c) is satisfied.

Now assume that $\mathbf{x}$ is not Martin-Löf random. Let V be a universal prefix-free machine and $H_V = H$. Using V we define the following enumeration S of finite transducers:

> For $\sigma\eta$ such that $\sigma \in \mathrm{dom}(V)$ and $\mathrm{time}(V(\sigma)) = |\eta|$, let $T^S_{\sigma\eta}$ be defined as the transducer which maps every non-empty string w to $V(\sigma)\eta w$.

Here $\mathrm{time}(V(\sigma))$ denotes the time till the computation stops; S is computable and prefix-free because $\mathrm{dom}(V)$ is prefix-free and $\mathrm{dom}(S) = \{\sigma\eta : \sigma \in \mathrm{dom}(V) \wedge |\eta| = \mathrm{time}(V(\sigma))\}$.

If the sequence $\mathbf{x}$ is not Martin-Löf random, then for every $c > 0$ there is an $n > 0$ such that $H(\mathbf{x} \upharpoonright n) < n - c$. Hence, for $c > 0$ we have $n > 0$, $\sigma \in \{0,1\}^*$, $s > 0$ such that $V(\sigma) = \mathbf{x} \upharpoonright n$, $|\sigma| < n - c$ and $\mathrm{time}(V(\sigma)) = s$.

Define $\eta \in \{0,1\}^s$ via $\big(\mathbf{x} \upharpoonright (n-c) + s\big) = V(\sigma)\eta$. Then $T^S_{\sigma\eta}(w) \sqsubset \mathbf{x}$ whenever $V(\sigma)\eta w \sqsubset \mathbf{x}$. Thus $C_S(\mathbf{x} \upharpoonright n + s') < n + s' - c$ for all $s' \geq s$, and the conditions (b) and (c) hold. □

Corollary 14. *A sequence $\mathbf{x}$ is Martin-Löf random iff for every prefix-free enumeration S there is a constant c such that for every $n \geq 1$ the inequality $C_S(\mathbf{x} \upharpoonright n) \geq n - c$ holds true.*

Furthermore, there is a perfect enumeration S which satisfies that a sequence $\mathbf{x}$ is Martin-Löf random iff for every $n \geq 1$ the inequality $C_S(\mathbf{x} \upharpoonright n) \geq n - c$ holds true.

The second clause of this result shows that the measure C_S, for perfect S, combines features of prefix-free Kolmogorov complexity and a minimum description length: On one hand it permits to define the Martin-Löf random sequences in a very natural way and, on the other hand, the complexity $C_S(x)$ can be effectively computed for every $x \in \{0,1\}^*$. However, C_S cannot replace the prefix-free Kolmogorov complexity to single out the random finite strings: the set of random strings is immune, hence it cannot be defined by a computable measure like C_S, as that would result in a decidable set. In this way, the measure obtained here is the best possible.

7. Finite State Complexity Based on Exotic Enumerations

Most of the previous results have used the complexity C_S based on prefix-free enumerations S. If we drop the prefix-freeness condition the complexity can behave in a different way. First we investigate the relation between the complexity C_S and plain Kolmogorov complexity K.

Lemma 15. *Let S be a not necessary prefix-free enumeration of all admissible transducers. Then we have the following upper bound on the plain complexity K: there is a constant c such that for all $w \in \{0,1\}^*$ we have:*

$$K(w) \leq C_S(w) + 2\log|w| + c.$$

Proof. Let $\gamma : \mathbb{N} \to \{0,1\}^*$ be a computable prefix-free encoding of the natural numbers. We may assume that for all n, $|\gamma(n)| \leq 2\log n + 2$.

Given S we define a computable partial function φ as follows:

$$\varphi(\pi) = \begin{cases} T^S_\sigma(p), & \text{if } \pi = \gamma(|\sigma|) \cdot \sigma \cdot p, \text{ and} \\ \varepsilon, & \text{otherwise.} \end{cases}$$

Then

$$K_\varphi(T^S_\sigma(p)) \leq |\sigma| + |p| + 2\log|\sigma| + 2 \leq C_S(T^S_\sigma(p)) + 2\log C_S(T^S_\sigma(p)) + 2.$$

Now the assertion follows from $C_S(w) \leq |w| + c_S$ (see Corollary 7). □

We illustrate this phenomenon by showing that in every sequence there exist infinitely many complexity dips, a phenomenon discovered by Martin-Löf [29] for the plain (Kolmogorov) complexity. As this is readily seen, for enumerations like $S[U]$, see Lemma 8, we restrict our considerations to enumerations where $\mathrm{dom}(S)$ is computable.

By $\mathrm{string}(i)$ we denote the binary string obtained by removing the leading

1 from the binary representation of the integer $i \geq 1$. For an enumeration of admissible transducers S define the following modified enumeration S':

$$T_{0\sigma}^{S'}(p) = T_{\sigma}^{S}(p), \text{ for } \sigma \in S, p \in \{0,1\}^*, \tag{4}$$

$$T_{1\rho}^{S'}(p) = \begin{cases} \text{string}(|\rho|) \cdot T_{1\rho}^{S'}(p'), & \text{if } p = 0p', \\ \rho \cdot T_{1\rho}^{S'}(p'), & \text{if } p = 1p', \\ \varepsilon, & \text{otherwise.} \end{cases} \tag{5}$$

The transducer $T_{1\rho}^{S'}$ realises, as one can easily see, a homomorphism from $\{0,1\}^*$ to $\{0,1\}^*$ mapping 0 onto string($|\rho|$) and 1 onto ρ.

Next we use the following illustration of [22, Theorem 1] ([8, Theorem 6.10]).

Lemma 16. *Every infinite sequence $\mathbf{x} \in \{0,1\}^\omega$ has infinitely many prefixes of the form* string($|w|$)w*, with $w \in \{0,1\}^*$.*

Proof. The proof follows [22, Theorem 1]. Fix $\mathbf{x} \in \{0,1\}^\omega$. Choose an integer $\ell \geq 1$ and put $v = \mathbf{x} \restriction \ell$. Then $v = \text{string}(n)$ for a unique integer $n \geq 1$. Next define w' to be the prefix of length $\ell + n$ of of $\mathbf{x}$, that is, $\mathbf{x} \restriction (\ell + n) = w' = v \cdot w = \text{string}(|w|)w$. □

Theorem 17. *There exist enumerations S' having a computable domain* dom(S') *such that for every infinite sequence $\mathbf{x} \in \{0,1\}^\omega$ there are infinitely many prefixes $v_i \sqsubset \mathbf{x}$ such that $|v_i| - C_{S'}(v_i) > i$.*

Proof. First observe that for $w \in \{0,1\}^*, |w| \geq 2$, according to Eq. (5) we have $T_{1w}^{S'}(01) = \text{string}(|w|)w$ and thus $C_{S'}\big(\text{string}(|w|)w\big) \leq |w| + 3$, but $\big|\text{string}(|w|)w\big| \geq |w| + \lfloor \log_2 |w| \rfloor$. The assertion follows from Lemma 16. □

Complexity dips cannot be avoided even when we consider only transducers which satisfy the condition $|\mu(q,a)| \leq m$, for all $(q,a) \in Q \times \{0,1\}$, that is, the output can always be at most m times as long as the input. We call these transducers *m-bounded.* We denote by $C_S^{(m)}$ the variant of C_S which looks at complexity using only *m-bounded transducers.*

Theorem 18. *There exist enumerations S of admissible 2–bounded transducers having a computable domain* dom(S) *such that for every infinite sequence $\mathbf{x} \in \{0,1\}^\omega$ there are infinitely many prefixes $v_i \sqsubset \mathbf{x}$ such that $|v_i| - C_S^{(2)}(v_i) > i$.*

Proof. Similar to the proof of Lemma 16 define $v = \text{string}(n)$ to be the prefix of length 2ℓ of $\mathbf{x}$ and then append the next $2n$ symbols of $\mathbf{x}$. This construction shows that for every infinite sequence $\mathbf{x} \in \{0,1\}^\omega$ there are infinitely many prefixes of the form string$\left(\frac{|w|}{2}\right)$ w where the lengths of string$\left(\frac{|w|}{2}\right)$ and w are even.

Let $w = a_1 \dots a_{2n}$ and $v = b_1 \dots b_{2\ell}$ and define the transducer $T_{1w'}$ where $w' = a_2 a_4 \dots a_{2n}$ as $T_{1w'} = (\{0,1\}, \{s_1, \dots, s_{\bar{n}}\}, s_1, \delta, \mu)$ with $\bar{n} = \ell + n$ and

$$\begin{array}{rclrcll} \delta(s_i, a) & = & s_{i+1}, & \mu(s_i, a) & = & ab_{2i}, & \text{for } i = 1, \dots, \ell, \\ \delta(s_i, a) & = & s_{i+1}, & \mu(s_i, a) & = & aa_{2i}, & \text{for } i = \ell+1, \dots, \bar{n}, \\ \delta(s_{\bar{n}+1}, a) & = & s_{\bar{n}+1}, & \mu(s_{\bar{n}+1}, a) & = & a, & a \in \{0,1\}. \end{array}$$

This construction is depicted in Figure 1 below.

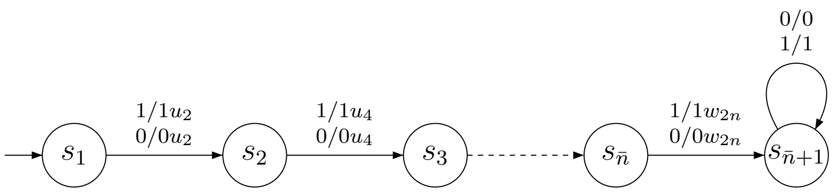

Figure 1: Transducer $T_{1w'}$ where $|w'| = n$ and $\bar{n} = \ell + n$

The transducer $T_{1w'}$ is 2-bounded and

$$T_{1w'}(b_1 b_3 \dots b_{2\ell-1} a_1 a_3 \dots a_{2n-1}) = b_1 \dots b_{2\ell} \cdot a_1 \dots a_{2n} = \text{string}\left(\tfrac{|w|}{2}\right) w.$$

A construction similar to Eqs. (4) and (5) shows that

$$\left|\text{string}\left(\tfrac{|w|}{2}\right) w\right| - C_S^{(2)}\left(\text{string}\left(\tfrac{|w|}{2}\right) w\right) \geq \ell - 3$$

is unbounded for suitably chosen prefixes of $\mathbf{x} \in \{0,1\}^\omega$. $\square$

8. Finite State Incompressibility and Normality

In this section we define finite state incompressible sequences and prove that each such sequence is normal. Given an enumeration S of all admissible transducers, a sequence $\mathbf{x} = x_1 x_2 \dots x_n \dots$ is C_S*–incompressible* if $\liminf_n C_S(\mathbf{x} \restriction n)/n = 1$. This definition resembles in some sense the definition of (asymptotic) Kolmogorov complexity $\underline{\kappa}(\mathbf{x}) = \liminf_n K(\mathbf{x} \restriction n)/n$

investigated in [33, 38]. From Levin's Theorem 3.4 of [47] one deduces that this quantity coincides with Lutz's [26] constructive dimension. For more details see [41, 42].

Proposition 19. *Every Martin-Löf random sequence is C_S–incompressible for every enumeration S, but the converse implication is not true.*

Proof. If $\mathbf{x}$ is a Martin-Löf random sequence, then $\liminf_n K(\mathbf{x} \restriction n)/n = 1$, so by Lemma 15, $\mathbf{x}$ is C_S–incompressible. Next we take a Martin-Löf random sequence $\mathbf{x}$ and modify it to be not random: define $\mathbf{x}'(n) = 0$ whenever n is a power of 2 and $\mathbf{x}'(n) = \mathbf{x}(n)$, otherwise. Clearly, $\mathbf{x}'$ is not Martin-Löf random, but $\liminf_n K(\mathbf{x} \restriction n)/n = 1$, so $\mathbf{x}$ is C_S–incompressible for every enumeration S of all admissible transducers. $\square$

A sequence is *normal* if all digits are equally likely, all pairs of digits are equally likely, all triplets of digits equally likely, etc. This means that the sequence $\mathbf{x} = x_1 x_2 \dots x_n \dots$ is normal if the frequency of every string y in $\mathbf{x}$ is $2^{-|y|}$, where $|y|$ is the length of y.

Lemma 20. *If the sequence $\mathbf{x}$ is not normal, then there exist a transducer T_σ^S and a constant α with $0 < \alpha < 1$ (depending on $\mathbf{x}, \sigma, S$) such that for infinitely many integers $n > 0$ we have $C_{T_\sigma^S}(\mathbf{x} \restriction n) < \alpha \cdot n$.*

Proof. It is known (see [18, 19, 35]) that if the sequence $\mathbf{x}$ is not normal, then there exist a transducer T_σ^S, a sequence $\mathbf{y}$, and a real $\alpha \in (0, 1)$ such that $\lim_{m\to\infty} T_\sigma^S(\mathbf{y} \restriction m) = \mathbf{x}$ and for infinitely many $m > 0$

$$T_\sigma^S(\mathbf{y} \restriction m) \sqsubset \mathbf{x} \text{ and } m < \alpha \cdot |T_\sigma^S(\mathbf{y} \restriction m)|.$$

Consequently, for infinitely many $m > 0$

$$C_{T_\sigma^S}(T_\sigma^S(\mathbf{y} \restriction m)) \le m < \alpha \cdot |T_\sigma^S(\mathbf{y} \restriction m)|,$$

hence $C_{T_\sigma^S}(\mathbf{x} \restriction n) < \alpha \cdot n$ for infinitely many $n > 0$ because $T_\sigma^S(\mathbf{y} \restriction m) \sqsubset \mathbf{x}$ for infinitely many $m > 0$. $\square$

Example 21. Ambos-Spies and Busse [2, 3] as well as Tadaki [44] investigated infinite sequences $\mathbf{x}$ which can be predicted by finite automata in a certain way. The formalisations result in the following equivalent characterisations for a sequence $\mathbf{x}$ to be finite state predictable:

- The sequence $\mathbf{x}$ can be predicted by a finite automaton in the sense that every state is either passing or has a prediction on the next bit and when reading $\mathbf{x}$ the finite automaton makes infinitely often a correct prediction and passes in those cases where it does not make a correct prediction, that is, it never predicts wrongly.

- There is a finite automaton which has in every state a label from $\{0,1\}^*$ such that, whenever the automaton is in a state with a non-empty label w then some of the next bits of $\mathbf{x}$ are different from the corresponding ones in w.

- The sequence $\mathbf{x}$ is the image $T(\mathbf{y})$ for some binary sequence $\mathbf{y}$ and a finite transducer T which has only labels of the form (a, aw) with $a \in \{0,1\}$ and $w \in \{0,1\}^*$ and where in the translation from $\mathbf{y}$ into $\mathbf{x}$ infinitely often a label (a, aw) with $w \neq \varepsilon$ is used.

- There is a finite connected automaton with binary input alphabet such that not all states of it are visited when reading $\mathbf{x}$.

- $\mathbf{x}$ fails to contain some string w as a substring.

The last item makes clear that the class of finite state predictable sequences is the complement of the class of disjunctive [21] or rich sequences [39]. All Borel normal sequences are disjunctive whereas not all disjunctive sequences are Borel normal. An example is the sequence $\mathbf{x} = \prod_{w \in \{0,1\}^*} 0^{|w|!} \cdot w$ from [37] which contains considerably more occurrences of zeros than ones.

In [37, 39] the set of non-disjunctive sequences is characterised as the union of nullsets or, equivalently, nowhere dense sets definable (or accepted) by finite automata.

Theorem 22. *Every C_S–incompressible sequence is normal.*

Proof. Assume that the sequence $\mathbf{x}$ is not normal. According to Lemma 20 there exist $\alpha \in (0,1)$ and $\sigma \in \mathrm{dom}(S)$ such that for infinitely many integers $n > 0$ we have $C_{T_\sigma^S}(\mathbf{x} \upharpoonright n)) < \alpha \cdot n$. For these n it also holds that $C_S(\mathbf{x} \upharpoonright n) < \alpha \cdot n + |\sigma|$. Since $\alpha < 1$, $\mathbf{x}$ is not C_S–incompressible. $\square$

9. How Large is the Set of Incompressible Sequences?

It is natural to ask whether the converse of Theorem 22 is true. The results in [1, 6, 35, 46] discussed in Introduction might suggest a positive answer. In fact, the answer is *negative.*

To prove this result we will use binary *de Bruijn strings* of order $r \geq 1$ which are strings w of length $2^r + r - 1$ over alphabet $\{0, 1\}$ such that any binary string of length r occurs as a contiguous substring of w (exactly once). It is well-known that de Bruijn strings of any order exist, and have an explicit construction [16, 45]. For example, 00110 and 0001011100 are de Bruijn strings of orders 2 and 3 respectively.

Note that de Bruijn strings are derived in a circular way, hence their prefix of length $r - 1$ coincides with the suffix of length $r - 1$. Denote by $B(r)$ the prefix of length 2^r of a de Bruijn string of order r. The examples of de Bruijn strings of orders 2 and 3 previously presented are derived from the strings $B(2) = 0011$ and $B(3) = 00010111$, respectively. Thus the string $B(r) \cdot B'(r)$, where $B'(r)$ is the length $r - 1$ prefix of $B(r)$, contains every binary string of length string r exactly once as a substring.

In [31] it is shown that every sequence of the form

$$\mathbf{b}_f = B(1)^{f(1)} B(2)^{f(2)} \ldots B(n)^{f(n)} \ldots$$

is normal provided that the function $f : \mathbb{N} \to \mathbb{N}$ is increasing and satisfies the condition $f(i) \geq i^i$, for all $i \geq 1$. Moreover, in this case the real $0.\mathbf{b}_f$ is a Liouville number, i.e. it is a transcendental real number with the property that, for every positive integer n, there exist integers p and q with $q > 1$ and such that $0 < |0.\mathbf{b}_f - \frac{p}{q}| < q^{-n}$.

Lemma 23. *Every string w, $B(1) \sqsubseteq w \sqsubset \mathbf{b}_f$ can be represented in the form*

$$w = B(1)^{f(1)} B(2)^{f(2)} \ldots B(n-1)^{f(n-1)} B(n)^j w' \qquad (6)$$

where $n \geq 1$, $1 \leq j \leq f(n)$ and $|w'| < 2^{n+1} = |B(n+1)|$.

Proof. Indeed, in the case

$$B(1)^{f(1)} B(2)^{f(2)} \ldots B(n-1)^{f(n-1)} \sqsubseteq w \sqsubset B(1)^{f(1)} B(2)^{f(2)} \ldots B(n)^{f(n)}$$

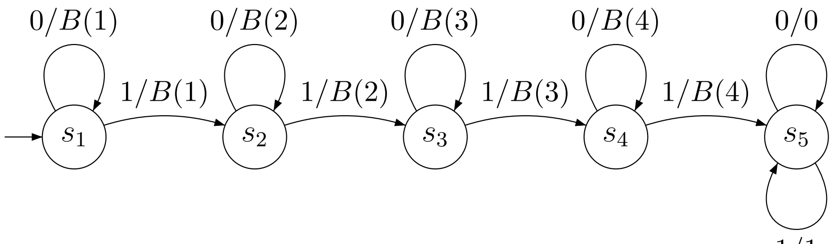

Figure 2: Block representation of the transducer T_4.

we can choose $w' \sqsubset B(n)$, and if

$$B(1)^{f(1)}B(2)^{f(2)}\dots B(n)^{f(n)} \sqsubseteq w \sqsubset B(1)^{f(1)}B(2)^{f(2)}\dots B(n)^{f(n)}B(n+1)$$

we can choose $w' \sqsubset B(n+1)$. □

Next we show that there are normal sequences which are simultaneously Liouville numbers and compressible by transducers, that is, the converse of Theorem 22 is false. This also proves that C_S–incompressibility is stronger than all other known forms of finite automata based incompressibility, cf. [1, 6, 17, 35, 46]. In view of the second inequality of Theorem 9, for universal enumerations S this follows from the existence of computable normal sequences, cf. [4, 5]. Here we show that this holds for all enumerations.

Theorem 24. *For every enumeration S there are normal sequences $\mathbf{x}$ such that $\lim_{n\to\infty} C_S(\mathbf{x} \upharpoonright n)/|n| = 0$, so $\mathbf{x}$ is C_S–compressible.*

Proof. Define the transducer $T_n = (\{0,1\}, \{s_1, \dots, s_{n+1}\}, s_1, \delta_n, \mu_n)$ as follows:

$$\begin{array}{rclrcll} \delta_n(s_i,0) & = & s_i, & \mu_n(s_i,0) & = & B(i), & \text{for } i \le n, \\ \delta_n(s_i,1) & = & s_{i+1}, & \mu_n(s_i,1) & = & B(i), & \text{for } i \le n, \\ \delta_n(s_{n+1},a) & = & s_{n+1}, & \mu_n(s_{n+1},a) & = & a, & \text{for } a \in \{0,1\}\,. \end{array}$$

For example, the transducer T_4 is presented in Figure 2. Let σ_n be an encoding of T_n according to S. Choose a function $f : \mathbb{N} \to \mathbb{N}$ which satisfied the following two conditions for all $n \ge 1, i > 1$:

$$f(n) \ge \max\{|\sigma_{n+1}|, n^n, 2^{n+2}\} \text{ and } f(i) \ge 2 \cdot f(i-1). \tag{7}$$

Finally, let $p_i = 0^{f(i)-1}1$ and $p'_j = 0^{j-1}1$. Eq. (6) shows that

$$T_n(p_1 \dots p_{n-1}p'_j w') = B(1)^{f(1)} \dots B(n-1)^{f(n-1)} B(n)^j w'$$

is a prefix of the normal sequence $\mathbf{x} = \mathbf{b}_f$. We then have:

$$\begin{aligned} |T_n(p_1 \dots p_{n-1}p'_j w')| &= \sum_{i=1}^{n-1} 2^i f(i) + 2^n j + |w'| \\ &\geq 2^{n-1} f(n-1) + 2^n j, \end{aligned}$$

and

$$\begin{aligned} &|\sigma_n| + |p_1 \dots p_{n-1} \cdot p'_j \cdot w'| \\ &\quad = |\sigma_n| + \sum_{i=1}^{n-1} |p_i| + |p'_{n-1}| + |w'| \\ &\quad \leq f(n-1) + 2f(n-1) + j + f(n-1) \\ &\quad = 4f(n-1) + j. \end{aligned}$$

This shows that for every prefix w of $\mathbf{b}_f$ presented in the form (6) as

$$w = B(1)^{f(1)} \dots B(n-1)^{f(n-1)} \cdot B(n)^j \cdot w',$$

we have $B(1) \sqsubset w \sqsubset \mathbf{b}_f$ and (by using the inequality $\frac{a+b}{c+d} \leq \max\left\{\frac{a}{c}, \frac{b}{d}\right\}$, when $0 < a, b, c, d$):

$$\frac{C_S(w)}{|w|} \leq \frac{4f(n-1) + j}{2^{n-1} f(n-1) + 2^n j} \leq \frac{4}{2^{n-1}}.$$

This shows that $\lim_{n\to\infty} C_S(\mathbf{x} \upharpoonright n)/|n| = 0$. $\square$

In the proof of Theorem 24 we have used an arbitrary function f satisfying (7). Of course, there exist computable and incomputable such functions.

Corollary 25. *For every enumeration S there are normal and C_S–compressible computable and incomputable sequences.*

Similar to Section 7 we consider also the variant $C_S^{(m)}$ of C_S which looks at complexity using only *m-bounded transducers.* The following result is a sample result for this area.

Theorem 26. *For every enumeration S of all 2-bounded admissible transducers, there are normal sequences $\mathbf{x}$ such that $\lim_{n\to\infty} C_S^{(2)}(\mathbf{x}\restriction n)/n = 1/2$.*

Proof. We start from the transducers T_n defined in the proof of Theorem 24 and we split every long output $B(i)$ of T_n into 2^{i-1} pieces of length 2. Formally, we replace the states $s_i, i \leq n$, by sub-transducers $A_i = (\{0,1\}, R_i, r_{i,1}, \delta_n^{(i)}, \mu_n^{(i)})$ where $R_i = \{r_{i,1}, \ldots, r_{i,2^{i-1}}\}$,

$$\begin{array}{ll} \delta_n^{(i)}(r_{i,j}, a){=}r_{i,j+1}, & \mu_n^{(i)}(r_{i,j}, a){=}u_{i,j}, j < 2^i,\ a < 2, \\ \delta_n^{(i)}(r_{i,2^{i-1}}, 0){=}r_{i,1}, & \mu_n^{(i)}(r_{i,2^{i-1}}, 0){=}u_{i,2^{i-1}}, \\ \delta_n^{(i)}(r_{i,2^{i-1}}, 1){=}r_{i+1,1}, & \mu_n^{(i)}(r_{i,2^{i-1}}, 1){=}u_{i,2^{i-1}}, \end{array}$$

and $B(i) = u_{i,1} \ldots u_{i,2^{i-1}}$ with $|u_{ij}| = 2$. Observe that the transition with input 1 on state $r_{i,2^{i-1}}$ leads to the initial state of the next sub-transducer (for $i = n$ this leads to state $r_{n+2,1} = s_{n+1}$ of T_n).

Then, the new transducer is defined as follows:

$$Q_n = \bigcup_{i=1}^{n} R_i \cup \{s_{n+1}\}, q_{0n} = r_{1,1},$$

$$\delta_n' = \bigcup_{i=1}^{n} \delta_n^{(i)} \cup \{(s_{n+1}, 0, s_{n+1}), (s_{n+1}, 1, s_{n+1})\}$$

and

$$\mu_n' = \bigcup_{i=1}^{n} \mu_n^{(i)} \cup \{(s_{n+1}, 0, 0), (s_{n+1}, 1, 1)\}.$$

Again let σ_n' be an encoding of T_n' in S, and let $\bar{p}_i = (0^{2^{i-1}})^{f(i)-1}0^{2^{i-1}-1}1$ where $f : \mathbf{N} \to \mathbf{N},\ f(n) \geq \max\{|\sigma_{n+1}'|, n^n, 2^{n+2}\},\ f(i) \geq 2 \cdot f(i-1)$, is as in the proof of Theorem 24. Let $\bar{p}_{i,j}' = (0^{2^{i-1}})^{j-1}0^{2^{i-1}-1}1$.

Furthermore, let $B(1) \sqsubseteq w \sqsubset \mathbf{b}_f$. According to Eq. (6) we have:

$$w = B(1)^{f(1)} \ldots B(n-1)^{f(n-1)} B(n)^j w' = T_n'(\bar{p}_1 \ldots \bar{p}_{n-1} \bar{p}_j' w').$$

We then have:

$$\begin{array}{rcl} |T_n'(\bar{p}_1 \ldots \bar{p}_{n-1}(0^{j-1})1 \cdot w')| & = & \sum_{i=1}^{n-1} 2^i \cdot f(i) + 2^n j + |w'| \\ & \geq & \sum_{i=1}^{n-1} 2^i \cdot f(i) + 2^n j, \end{array}$$

and

$$\begin{aligned} C_S^{(m)}(w) &\leq |\sigma_n'| + \textstyle\sum_{i=1}^{n-1} 2^{i-1} f(i) + 2^{n-1} j + |w'| \\ &\leq f(n-1) + \textstyle\sum_{i=1}^{n-1} 2^{i-1} f(i) + 2^{n-1} j + f(n-1), \end{aligned}$$

finally obtaining

$$\begin{aligned} \frac{C_S^{(m)}(w)}{|w|} &\leq \frac{\sum_{i=1}^{n-2} 2^{i-1} f(i) + 2^{n-1} j + (2^{n-2}+2) f(n-1)}{\sum_{i=1}^{n-2} 2^{i} f(i) + 2^{n} j + 2^{n-1} f(n-1)} \\ &\leq \frac{2^{n-2}+2}{2^{n-1}}. \end{aligned}$$

This proves that $\lim_{t\to\infty} C_S^{(2)}(\mathbf{x} \restriction t)/t = 1/2$. $\square$

Theorem 26 can be easily generalised to m-bounded complexity thereby yielding the bound $\lim_{n\to\infty} C_S^{(m)}(\mathbf{x} \restriction n)/n = 1/m$. Moreover, the results of Theorems 24 and 26 can be also generalised to arbitrary (output) alphabets Y. Here the circular de Bruijn strings of order n, $CB_{|Y|}(n)$, have length $|Y|^n$.

In connection with Theorem 24, we can ask whether the finite state complexity of each sequence $\mathbf{x}$ representing a Liouville number satisfies the inequality $\limsup_{n\to\infty} C_S(\mathbf{x} \restriction n)/n < 1$. The answer is negative: Example 12 of [40] shows that there are sequences $\mathbf{x}$ representing Liouville numbers having $\limsup_{n\to\infty} K(\mathbf{x} \restriction n)/n = 1$, hence by Theorem 9, $\limsup_{n\to\infty} C_S(\mathbf{x} \restriction n)/n = 1$.

The following result complements Theorem 24: the construction is valid for every enumeration, but the degree of incompressibility is slightly smaller.

Theorem 27. *There exists an infinite, normal and computable sequence* $\mathbf{x}$ *which satisfies the condition* $\liminf_{n\to\infty} C_S(\mathbf{x} \restriction n)/n = 0$, *for all prefix-free enumerations* S.

Proof. Fix a computable enumeration $(T_m)_{m\geq 1}$ of all admissible transducers such that each T_m has at most m states and each transition in T_m from one state to another has only labels which produce outgoing strings of at most length m (that is, complicated transducers appear sufficiently late in the list).

Now define a sequence of strings α_n such that each α_n is the length-lexicographic first string longer than n such that for all transducers T_m with

$1 \leq m \leq n$, for all states q of T_m and for each string γ of less than n bits, there is no string β of length below $\frac{n-1}{n} \cdot |\alpha_n|$ such that $\gamma T_m(q, \beta)$ is α_n or an extension of it. Note that these α_n must exist, as every sufficiently long prefix of the Champernowne sequence meets the above given specifications due to Champernowne sequence normality [14]. Furthermore, $\alpha_0 = 0$ as the only constraint is that α_0 is longer than 0. An easy observation shows that also $|\alpha_n| \leq |\alpha_{n+1}|$, for all n.

In what follows we will use an acceptable numbering of all partially computable functions from natural numbers to natural numbers of one variable $(\varphi_e)_{e \geq 1}$. Now let f be a computable function from natural numbers to natural numbers satisfying the following conditions:

Short: For all $t \geq 1$, $|\alpha_{f(t)}| \leq \sqrt{t}$.

Finite-to-one: For all $n \geq 1$ and almost all $t \geq 1$, $f(t) > n$.

Match: $\forall n \, \forall e < n \, \exists t \, [\varphi_e(n) < \infty \implies t > \varphi_e(n) \wedge f(t) = n \wedge f(t+1) = n \wedge \ldots \wedge f(t^2) = n]$.

In order to construct f, consider first a computable one-one enumeration $(e_0, n_0, m_0), (e_1, n_1, m_1), \ldots$ of the set $\{(e, n, m) : e < n \wedge \varphi_e(n) = m\}$. The function f is now constructed in stages where the requirement "Short" is satisfied all the time, the requirement "Finite-to-one" will be a corollary of the way the function is constructed and the requirement "Match" will be satisfied for the k-th constraint (e_k, n_k, m_k) in the k-th stage.

> In the k-th stage, let s_k be the first value where $f(s_k)$ was not defined in an earlier stage and let t_k be the first number such that $t_k > s_k + m_k$ and $|\alpha_{n_k}| \leq \sqrt{t_k}$. Having these properties, for u with $s_k \leq u < t_k$, let $f(u)$ be the maximal ℓ with $|\alpha_\ell| \leq \sqrt{\max\{1, u\}}$, and for u with $t_k \leq u \leq t_k^2$, let $f(u) = n_k$.

It is clear that the function f is computable. Next we verify that it satisfies the required three conditions.

Short: This condition, which is more or less hard-coded into the algorithm, directly follows from the way t_k is selected and $f(u)$ is defined in the two cases.

Finite-to-one: The inequality $f(u) \leq n$ is true only in stages k where for some u either $|\alpha_{n+1}| > \sqrt{s_k}$ or $n_k \leq n$; both conditions happen only for finitely many stages k.

Match: For each n and e with $\varphi_e(n)$ being defined, there is a stage k such that $(e_k, n_k, m_k) = (e, n, \varphi_e(n))$. The choice of t_k makes then f to be equal to n_k on $t_k, t_k + 1, \ldots, t_k^2$ and furthermore $t_k > \varphi_{e_k}(n_k)$.

Let $\mathbf{x}$ be the sequence $\alpha_{f(0)}\alpha_{f(1)}\alpha_{f(2)} \ldots$ which is obtained by concatenating all the strings $\alpha_{f(n)}$ for the n in default order. It is clear that $\mathbf{x}$ is computable.

Consider any enumeration S of transducers. Choose e such that $\varphi_e(n)$ takes the value the length of the code of that transducer T_n which has the starting state q and a further state q' and follows the following transition table:

state	input	output	new state
q	0	ε	q'
q	1	α_n	q
q'	0	0	q
q'	1	1	q

As φ_e is total, there is for each $n > e$ a t larger than the code of the transducer T_n such that $f(t), f(t+1), \ldots, f(t^2)$ are all n. Now $\sigma = \alpha_{f(0)} \ldots \alpha_{f(t^2)}$ can be generated by T_n by a code of the form $\beta = 0\sigma(0)0\sigma(1) \ldots 0\sigma(u-1)1^{t^2-t}$ where u is the length of $\alpha_{f(0)}\alpha_{f(1)} \ldots \alpha_{f(t-1)}$. The length of β is $2u + t^2 - t$. Note that $u \leq t \cdot \sqrt{t}$ by the condition "Short" and therefore $|\beta| \leq t^2 + t^{3/2} - t$ while the string σ generated from β by the transducer T_n has at least the length $(t^2 - t) \cdot |\alpha_n|$ which is at least $(t^2 - t) \cdot (n + 1)$. Furthermore, the representation of T_n in S has at most length t, thus

$$C_S(\sigma)/|\sigma| \leq (t^2 + t^{3/2})/(n \cdot (t^2 - t)) \leq \frac{2}{n}.$$

It follows that $\liminf_{n\to\infty} C_S(\mathbf{x} \upharpoonright n)/n = 0$.

Next we prove that $\mathbf{x}$ is normal. Fix a transducer T_m. Then, for every $n > m$, there is a sufficiently large t such that $(n-1) \cdot t$ of the first $n \cdot t$ values $s < n \cdot t$ satisfy $f(s) > n$. Fix such a t and let $\beta = \beta_0\beta_1 \ldots \beta_{n \cdot t}$ be such that $\beta_0 \ldots \beta_s$ is the shortest prefix of β with T_m producing from the starting state and input $\beta_0 \ldots \beta_s$ an extension of $\alpha_{f(0)} \ldots \alpha_{f(s)}$. Note that the image of $\beta_0 \ldots \beta_s$ is at most $m-1$ symbols longer than $\alpha_{f(0)} \ldots \alpha_{f(s)}$. Let $\sigma = \alpha_{f(0)} \ldots \alpha_{f(t \cdot n)}$. One can prove by induction that for all s with $f(s) \geq n$ we have

$$|\beta_s| \geq \frac{n-1}{n} \cdot |\alpha_{f(s)}|,$$

and for all s where $f(s) < n$ we have

$$|\alpha_{f(s)}| \leq |\sigma|/(t \cdot n).$$

It follows that $|\beta| \geq \frac{(n-1)^2}{n^2} \cdot |\sigma|$ and therefore we have sufficiently long prefixes of $\mathbf{x}$ which are concatenations of the strings $\alpha_{f(0)} \ldots \alpha_{f(t \cdot n)}$, all having complexity relative to T_m near 1. Furthermore, the length difference between any given prefix and a prefix of such a form is smaller than the square root of the length and therefore one can conclude that the sequence is incompressible with respect to each fixed transducer T_m. Hence, by Theorem 22, it is normal. □

The proof method in Theorem 27 can be adapted to obtain the following result.

Theorem 28. *There exists a perfect enumeration S and a sequence which is computable, normal and C_S–incompressible.*

Proof. The sequence of the T_n and α_n is defined as in the proof of Theorem 27; furthermore, it is assumed that the listing of the T_n is one-one. However, f is chosen such that it satisfies the following three conditions:

Short: For all $t \geq 1$, $|\alpha_{f(t)}| \leq \sqrt{t}$.

Finite-to-one: For all $n \geq 1$ and almost all $t \geq 1$, $f(t) > n$.

Monotone: For all $t \geq 1, f(t) \leq f(t+1)$.

This is achieved by selecting

$$f(t) = \max\{m : |\alpha_m| \leq \sqrt{t}\}.$$

It is clear that f is computable and satisfies the conditions "Short" and "Monotone". The condition "Finite-to-one" follows from the observation that $f(t) > n$ for all t with $|\alpha_{n+1}| \leq \sqrt{t}$ and the fact that almost all t satisfy this condition.

As above one can see that whenever $f(t) > n$ and $m \leq n$ then $T_m(\beta)$ extends $\alpha_{f(0)}\alpha_{f(1)} \ldots \alpha_{f(n \cdot t)}$ only if $|\beta| \geq (n-1)^2/n^2$. Now one makes S such that the transducer T_m has the code word $0^m 1^{m^2 \cdot t_m}$ for the first t_m such that $f(t_m) > m$. It can be concluded that $C_{T_m}(\sigma)/|\sigma| \geq (m-1)^2/m^2 \cdot |\sigma|$, for all prefixes σ of $\mathbf{x}$ and that $C_{T_m}(\sigma)/|\sigma|$ goes to 1 for longer and longer prefixes of $\mathbf{x}$. Thus the sequence $\mathbf{x}$ is normal and furthermore $\mathbf{x}$ is incompressible with respect to the here chosen S. □

10. Conclusion and Open Questions

Enumerations are — in the context of this paper — computable listings of all admissible transducers. We have investigated two main notions of enumerations: the arbitrary ones and the prefix-free ones. The prefix-free ones turned out to be the far more natural notion and among these, we were specifically interested in two special cases: the perfect enumerations (which have a decidable domain, are surjective and have a computable inverse) and the universal enumerations (which optimise the codes for the transducers up to a constant for the best possible value). We have showed that Martin-Löf randomness of infinite sequences can be characterised with both of these types of enumerations. Furthermore, we have related the finite-state complexity based on universal enumerations with the prominent notions of algorithmic description complexity of binary strings. Finite-state complexities based on some exotic enumerations behave like the plain (Kolmogorov) complexity.

The results of Sections 8 and 9 show that our definition of finite state incompressibility is stronger than all other known forms of finite automata based incompressibility, in particular the notion related to finite automaton based betting systems introduced by Schnorr [35].

The following three questions are left open: Are there an enumeration S, a computable sequence $\mathbf{x}$ and a constant c such that $C_S(\sigma) > |\sigma| - c$, for all prefixes σ of $\mathbf{x}$? This would mean that, with respect to S, some computable sequence $\mathbf{x}$ behaves like a Martin-Löf random one (in other enumerations). One can also ask the converse question, for which enumerations S is it true that every sequence satisfying $C_S(\mathbf{x} \upharpoonright n) \geq n - c$ is Martin-Löf random? Note that every universal and also some perfect enumeration satisfy this condition. What is the relation between C_S–incompressible sequences and ε–random sequences, [9, 12, 43]? Note that some ε–random sequences can be finite-state predictable by not having a certain substring, cf. [38, 44], hence they can be compressed by a single transducer; this is, however, not true for all ε–random sequences. In particular it would be interesting to ask whether one can say that $\mathbf{x}$ is ε–random iff for every perfect enumeration S there is a constant c such that for all n, $C_S(\mathbf{x} \upharpoonright n) \geq \varepsilon \cdot n - c$.

Acknowledgments

The authors would like to thank Sanjay Jain and the anonymous referees for helpful comments.

References

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

[2] Ambos-Spies, K., Busse, E.: Automatic forcing and genericity: On the diagonalization strength of finite automata, *Proceedings of DMTCS 2003*, LNCS 2731, Springer, 97–108 (2003).

[3] Ambos-Spies, K., Busse, E.: Computational aspects of disjunctive sequences, *Proceedings of MFCS 2004*, LNCS 3153, Springer, 711–722 (2004).

[4] Becher, V., Figueira, S.: An example of a computable absolutely normal number, *Theoretical Computer Science* 270: 947–958 (2002).

[5] Becher, V., Heiber, P., Slaman, T. A.: A computable absolutely normal Liouville number, *Mathematics of Computation* (2014), in press.

[6] Bourke, C., Hitchcock, J. M., Vinodchandran, N. V.: Entropy rates and finite-state dimension, *Theoretical Computer Science* 349, 3: 392–406 (2005).

[7] Buhrman, H., Fortnow, L.: Resource-bounded Kolmogorov complexity revisited, *Proceedings of STACS 1997*, LNCS 1200, Springer, 105–116 (1997).

[8] Calude, C. S.: *Information and Randomness. An Algorithmic Perspective*, 2nd ed., Springer, Berlin (2002).

[9] Calude, C. S., Hay, N. J., Stephan, F.: Representation of left-computable ε–random reals, *Journal of Computer and System Sciences* 77: 812–839 (2011).

[10] Calude, C. S., Salomaa, K., Roblot, T. K.: Finite state complexity, *Theoretical Computer Science* 412: 5668–5677 (2011).

[11] Calude, C. S., Salomaa, K., Roblot, T. K.: State-size hierarchy for FS-complexity, *International Journal of Foundations of Computer Science*, 23, 1: 37–50 (2012).

[12] Calude, C. S., Staiger, L., Terwijn, S. A.: On partial randomness, *Ann. Pure Appl. Logic*, 138: 20–30 (2006).

[13] Chaitin, G. J.: A theory of program size formally identical to information theory. *Journal of the Association for Computing Machinery*, 22: 329–340 (1975).

[14] Champernowne, D. G.:. The construction of decimals normal in the scale of ten, *Journal of the London Mathematical Society*, 8: 254–260 (1933).

[15] Charikar, M., Lehman, E., Liu, D., Panigrahy, R., Prabhakaran, M., Rasala, A., Sahai, A., Shelat, A.: Approximating the smallest grammar: Kolmogorov complexity in natural models, *Proceedings of STOC 2002*, ACM Press, 792–801 (2002).

[16] de Bruijn, N.: A combinatorial problem, *Proceedings of the Koninklijke Nederlandse Akademie van Wetenschappen* 49: 758–764 (1946).

[17] Dai, J. J., Lathrop, J. I., Lutz, J. H., Mayordomo, E.: Finite-state dimension, *Theoretical Computer Science* 310: 1–33 (2004).

[18] Doty, D., Moser, P.: Finite-state dimension and lossy compressors, `arxiv:cs/0609096v2` (2006).

[19] Doty, D., Moser, P.: Feasible Depth. *Proceedings of CiE 2007*, LNCS 4497, Springer, 35–44 (2007).

[20] Downey, R., Hirschfeldt D.: *Algorithmic Randomness and Complexity*, Springer, Heidelberg (2010).

[21] Jürgensen, H., Shyr, H.-J., Thierrin, G.: Disjunctive ω-languages, *Elektron. Informationsverarb. Kybernet.* 19: 267–278 (1983).

[22] Katseff, H. P.: Complexity dips in random infinite binary sequences, *Information and Control* 38, 3: 258–263 (1978).

[23] Koppel, M., Atlan, H.: An almost machine-independent theory of program-length complexity, sophistication, and induction, *Inf. Sci.* 56: 23–33 (1991)

[24] Lehman, E.: *Approximation Algorithms for Grammar-based Compression*, PhD Thesis, MIT (2002).

[25] Lehman, E., Shelat, A.: Approximation algorithms for grammar-based compression, *Proceedings of SODA 2002,* SIAM Press, 205–212 (2002).

[26] Lutz, J. H.: The dimensions of individual strings and sequences. *Inform. and Comput.* 187(1): 49–79 (2003).

[27] Li, M, Vitányi, P: *An Introduction to Kolmogorov Complexity and Its Applications*, 3rd ed., Springer (2007).

[28] Martin-Löf, P.: The definition of random sequences, *Information and Control* 9: 602–619 (1966).

[29] Martin-Löf, P.: Complexity oscillations in infinite binary sequences, *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete* 19: 225-230 (1971).

[30] Nies, A.: *Computability and Randomness*, Clarendon Press, Oxford (2009).

[31] Nandakumar, S., Vangapelli, S. K.: Normality and finite-state dimension of Liouville numbers, *Theory of Computing Systems* (2014), 1–11, doi=10.1007/s00224-014-9554-8.

[32] Rissanen, J.: *Information and Complexity in Statistical Modeling*, Springer, New York, 2007.

[33] Ryabko, B. Ya.: Noise-free coding of combinatorial sources, Hausdorff dimension and Kolmogorov complexity, *Problemy Peredachi Informatsii* 22(3): 16–26 (1986).

[34] Rytter, W.: Application of Lempel-Ziv factorization to the approximation of grammar-based compression, *Theoretical Computer Science* 302: 211–222 (2002).

[35] Schnorr, C. P., Stimm, H.: Endliche Automaten und Zufallsfolgen, *Acta Informatica* 1: 345–359 (1972).

[36] Schnorr, C. P.: Process complexity and effective randomness tests, *Journal of Comput. System Sciences* 7: 376–388, (1973).

[37] Staiger, L.: Reguläre Nullmengen, *Elektron. Informationsverarb. Kybernet.* 12: 307–311 (1976).

[38] Staiger, L.: Kolmogorov complexity and Hausdorff dimension. *Inform. and Comput.*, 103(2): 159–194, (1993).

[39] Staiger, L.: Rich ω-words and monadic second-order arithmetic. In Mogens Nielsen and Wolfgang Thomas, editors, *Computer Science Logic (Aarhus, 1997)*, *Lecture Notes in Computer Science* 1414, Springer, 1998, 478–490.

[40] Staiger, L.: The Kolmogorov complexity of real numbers, *Theoretical Computer Science* 284: 455–466 (2002).

[41] Staiger, L.: Constructive dimension equals Kolmogorov complexity, *Inform. Process. Lett.* 93: 149–153 (2005).

[42] Staiger, L.: The Kolmogorov complexity of infinite words, *Theoretical Computer Science* 383: 187–199 (2007).

[43] Staiger, L.: On Oscillation-free ε-random sequences, *Electr. Notes Theor. Comput. Sci.*, 221: 287–297 (2008).

[44] Tadaki, K.: Phase Transition and Strong Predictability, in Oscar H. Ibarra, Lila Kari and Steffen Kopecki, editors, *Unconventional Computation and Natural Computation - 13th International Conference, UCNC 2014, Proceedings*, LNCS 8553, Springer, 340–352.

[45] van Lint, J. H., Wilson, R. M.: *A Course in Combinatorics,* Cambridge University Press (1993).

[46] Ziv, J., Lempel, A.: Compression of individual sequences via variable-rate coding, *IEEE Transactions on Information Theory* 24: 530–536 (1978).

[47] Zvonkin, A. K., Levin, L. A.: Complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms, *Russian Math. Surveys* 25: 83–124 (1970).