Finite State Incompressible Infinite Sequences*

Cristian S. Calude[1], Ludwig Staiger[2], and Frank Stephan[3]

[1] Department of Computer Science, University of Auckland,
Private Bag 92019, Auckland, New Zealand
cristian@cs.auckland.ac.nz

[2] Martin-Luther-Universität Halle-Wittenberg, Institut für Informatik,
Von-Seckendorff-Platz 1, D - 06099 Halle, Germany
staiger@informatik.uni-halle.de

[3] Department of Mathematics and Department of Computer Science,
National University of Singapore, Singapore 119076, Republic of Singapore
fstephan@comp.nus.edu.sg

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 incompressibility, in particular the notion related to finite automaton based betting systems introduced by Schnorr and Stimm [28]. The paper concludes with a discussion of open questions.

1 Introduction

Algorithmic Information Theory (AIT) [7,18,25] 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.

* 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 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).

T V Gopal et al. (Eds.): TAMC 2014, LNCS 8402, pp. 50–66, 2014.

The incomputability of all descriptional complexities was 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 [6]. 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 [13,20,21,27].

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 [28] (see also [5,15]) and (b) a sequence is normal if and only if it is incompressible by any information lossless finite state compressor [33]. Doty and Moser [16,17] used computations with finite transducers for the definition of finite state dimension of infinite sequences. The NFA-complexity of a string [13] can be defined in terms of finite transducers that are called in [13] "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 [9,10] is utilised to define *finite state incompressible sequences*. We show basic connections of this new notion compared to standard complexity measures in Theorem 5: 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 13. 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.

Our notation follows standard textbooks [4,7]:

- 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 $x \in X^*$ is denoted by $|x|$.
- Sequences are denoted by $\mathbf{x}, \mathbf{y}$; the prefix of length n of the sequence $\mathbf{x}$ is denoted by $\mathbf{x} \restriction n$; the nth element of $\mathbf{x}$ is denoted by $\mathbf{x}(n)$.
- By $w \sqsubset u$ and $w \sqsubset \mathbf{y}$ we denote that w is a prefix of u and $\mathbf{y}$, respectively.
- If A, B are sets of strings then the concatenation is defined as $A \cdot B = \{xy : x \in A, y \in B\}$.
- A prefix-free set $A \subset X^*$ is a set with the property that for all strings $p, q \in X^*$, if $p, pq \in A$ then $p = pq$.

– By K, Km_D, and H we denote, respectively, the plain (Kolmogorov) complexity, the process complexity and the prefix-free complexity for appropriately fixed universal Turing machines.

2 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 $\mu(q_0,v)$.

A partially computable function S with a prefix-free domain mapping binary strings to admissible transducers is called an enumeration; for a σ in the domain of S, the admissible transducer assigned by S to σ is denoted as T_σ^S.

Definition 1 (Calude, Salomaa and Roblot [9,10]). A *perfect enumeration* S of all admissible transducers is a partially computable function with a prefix-free and computable domain mapping each binary string σ in the domain to an admissible transducer T_σ^S in a one-one and onto way.

Note that partially computable one-one functions with a computable range (as considered here) have also a computable inverse. 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 transducer represented by σ in S and σ' in S' and these representations satisfy $|\sigma'| < |\sigma| - c$, [9,10].

Definition 2. A *universal enumeration* S of transducers is a partially computable function with prefix-free domain whose range contains all admissible transducers such that for each further enumeration S' of admissible transducers there exists a constant c such that for all σ' in the domain of S', the transducer $T_{\sigma'}^{S'}$ equals to some transducer T_σ^S where σ is in the domain of S and $|\sigma| \leq |\sigma'| + c$.

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_σ^S is $T_{U(\sigma)}^{S'}$. The fact that U is a universal machine for prefix-free Kolmogorov complexity implies that also S is a universal enumeration of admissible transducers.

3 Complexity and Randomness

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

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

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 and the corresponding complexity, 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 [29] 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 $v \sqsubset w$, then $\psi(v) \sqsubset \psi(w)$. 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 $x \in \{0,1\}^*$.

Martin-Löf [23] introduced the notion of the random sequences in terms of tests and Schnorr — as cited by Chaitin [11] — characterised them in terms of prefix-free complexity; we take this characterisation as a definition. Furthermore, Schnorr [29] showed that the same definition holds for process complexity.

Definition 3 (Martin-Löf [23]; Schnorr [11,29]). 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 one can say that $\mathbf{x}$ is Martin-Löf random iff there is a constant c such that $Km_D(\mathbf{x} \upharpoonright n) \geq |n| - c$, for all $n \geq 1$.

4 Complexity Based on Transducers

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 $T(y) = x$. This definition is now adjusted to enumerations S of admissible transducers.

Definition 4. Let S be an enumeration of the admissible transducers. For each string x, the S-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_σ^S such that $T_\sigma^S(y) = x$.

This S-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 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 and plain complexity. The next result relates the complexity C_S for universal enumerations S to the plain complexity K, the prefix-free complexity H and the process complexity Km_D.

Theorem 5. *Let S be a universal enumeration of the admissible transducers. Then there are constants c, c', c'' such that, for all binary strings x,*

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

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

Proof. For the first inequality, note that if $T^S_\sigma(y) = x$ 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 $T^S_\sigma(y)$ for the unique y with $\sigma y = p$. Thus the mapping from all σy to $T^S_\sigma(y)$ with $\sigma \in \mathrm{dom}(S)$ and $y \in \mathrm{dom}(T^S_\sigma)$ is partially computable and well-defined. The inequality follows then from the universality of the plain Kolmogorov complexity K. One can furthermore see that ψ is also prefix-monotone and therefore also witnessing that $Km_D(x) \leq C_S(x) + c'$ for some constant c'.

To see that the first inequality is proper, note that $K(x) \leq Km_D(x) + c'''$ but there is no constant c'''' such that $Km_D(x) \leq K(x) + c''''$ for all x [29].

Theorem 6 below implies that the second inequality is proper.

Let S' be a fixed perfect enumeration of all admissible transducers; it is known that S' exists [9,10]. The inequality $C_S(x) \leq H(x) + c''$ might be obtained by choosing an enumeration S which for every p in the domain of a prefix-free universal machine U assigns to $p0$ a transducer mapping ε to $U(p)$ and, in the case that $U(p) \in \mathrm{dom}(S')$, to $p1$ the transducer $T^{S'}_{U(p)}$. Clearly, if $U(p) = x$ then $T^S_{p0}(\varepsilon) = x$ and therefore $C_S(x) \leq |p| + 1$. This enumeration of transducers is universal.

Furthermore, there is a fixed code σ for the transducer realising the identity $(T^S_\sigma(x) = x)$, hence $C_S(x) \leq |x| + |\sigma|$ for all x. It is known that this bound is not matched by longer and longer prefixes of Chaitin's Ω with respect to H, hence one cannot reverse the third 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 6. *There is a prefix-monotone partially computable function ψ such that for every enumeration S and each constant c there is a binary string x with $K_\psi(x) < C_S(x) - c$.*

Proof. Let Ω be Chaitin's random set and let Ω_s be an approximation to Ω from the left for s steps. Now define

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

Note that this function is partially computable and furthermore it is 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 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 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 $n/2$ 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 correspdonding definition would be to let, for each $\sigma \in \mathrm{dom}(S)$, $\phi(\sigma)$ 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 $|\psi(x)| \leq \phi(\sigma)$ 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 $z = \Omega_{\phi(\sigma)} \upharpoonright |\sigma| - c$. 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 $Km_D(u_n) \leq n + c'$ 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 enumeration S of admissible transducers. □

Furthermore, for perfect enumerations S, one can show that there is an algorithm to compute C_S.

Proposition 7. *Let S be a perfect enumeration of the admissible transducers. Then the mapping $x \mapsto C_S(x)$ is computable.*

Proof. Note that there is a fixed transducer T^S_τ such that $T^S_\tau(x) = x$ for all x. Now $C_S(x)$ is the length of the shortest σy with $\sigma \in \mathrm{dom}(S)$, $y \in \{0,1\}^*$,

$|\sigma y| \leq |\tau x|$ and $T^S_\sigma(y) = x$. Due to the length-restriction $|\sigma y| \leq |\tau x|$, the search space is finite and due to the perfectness of the enumeration S, the search can be carried out effectively. □

5 Complexity of Infinite Sequences

Martin-Löf randomness can be formalised using both prefix-free Kolmogorov complexity and process complexity, see Definition 3. Therefore it is natural to ask whether such a characterisation does also hold for the C_S complexity. The answer is affirmative as given in the following theorem.

Theorem 8. *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* $C_S(\mathbf{x} \restriction n) < n - c$*;*
(d) *For every universal enumeration* S *and for every* $c > 0$ *and almost all* $n > 0$ *we have* $C_S(\mathbf{x} \restriction n) < n - c$*;*
(e) *For every universal enumeration* S *and for every* $c > 0$ *there exists an* $n > 0$ *with* $C_S(\mathbf{x} \restriction n) < n - c$.

Proof. If $\mathbf{x}$ is Martin-Löf random then, as noted after Definition 3, $Km_D(\mathbf{x} \restriction n) \geq n - c$ for some constant c and all n. It follows that, for every enumeration S, from Theorem 5 that $C_S(\mathbf{x} \restriction n) \geq n - c'$ for some constant c' and all n. Hence non of the conditions (a-e) is satisfied.

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

> For $\sigma\eta$ such that $\sigma \in \mathrm{dom}(U)$ and $\mathrm{time}(U(\sigma)) = |\eta|$, let T^S_σ be defined as the trnasucer which maps every string τ to $U(\sigma)\eta\tau$.

Here $\mathrm{time}(U(\sigma))$ denotes the time till the computation stops; S is computable and prefix-free because $\mathrm{dom}(U)$ is prefix-free.

If the sequence $\mathbf{x}$ is not Martin-Löf random, then for every $c > 0$ there exists an $n > 0$ such that $H(\mathbf{x} \restriction n) < n - c$. Hence, for every $c > 0$ there exist $n > 0$, $\sigma \in \{0,1\}^*$, $s > 0$ such that $U(\sigma) = \mathbf{x} \restriction n$, $|\sigma| < n - c$ and $\mathrm{time}(U(\sigma)) = s$. Consequently, for every $c > 0$ there exist $n > 0$, $\sigma \in \{0,1\}^*$, $s > 0$ and $\eta \in \{0,1\}^s$ such that $\sigma\eta \in \mathrm{dom}(S)$, $|\sigma| < n - c$, $T^S_{\sigma\eta}(\varepsilon) = \mathbf{x} \restriction (n + s)$, hence for every $c > 0$ there exist $n, s > 0$ such that $C_S(\mathbf{x} \restriction (n + s)) < n + s - c$. We have showed that for every $c > 0$ and almost all $m > 0$, $C_S(\mathbf{x} \restriction m) < m - c$. Thus (b) holds. If S' is a universal enumeration, then $C_S(x) \leq C_{S'}(x) + c''$ for some constant c'' and all binary strings x. Hence (d) holds. Furthermore, (b) implies (c) and (d) implies (e). So (a-e) hold. Hence the conditions (a-e) are equivalent. □

Corollary 9. *A sequence* $\mathbf{x}$ *is Martin-Löf random iff for every enumeration* S *there is a constant* c *such that for every* $n \geq 1$ *the inequality* $C_S(\mathbf{x} \restriction n) \geq n - c$ *holds true.*

6 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 $\mathbf{x} = x_1 x_2 \cdots x_n \cdots$ is C_S*–incompressible* if $\liminf_n C_S(\mathbf{x} \upharpoonright n)/n = 1$.

Proposition 10. *Every Martin-Löf random sequence is C_S–incompressible for all enumerations S, but the converse implication is not true.*

Proof. If $\mathbf{x}$ is a Martin-Löf random sequence, then $\liminf_n K(\mathbf{x} \upharpoonright n)/n = 1$, so by Theorem 5, $\mathbf{x}$ 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 $\liminf_n K(\mathbf{x} \upharpoonright n)/n = 1$, so $\mathbf{x}$ is C_S–incompressible for every enumeration S of all admissible transducers. □

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 $\mathbf{x} = x_1 x_2 \cdots x_n \cdots$ is normal if the frequency of every string y in $\mathbf{x}$ is $2^{-|y|}$, where $|y|$ is the length of y.

Lemma 11. *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 $C_{T_\sigma^S}(\mathbf{x} \upharpoonright n) < \alpha \cdot n$.*

Proof. According to [16,17,28], 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 $\lim_{m\to\infty} T_\sigma^S(\mathbf{y} \upharpoonright m) = \mathbf{x}$ and for infinitely many $m > 0$

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

Consequently, for infinitely many $m > 0$

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

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

Example 12. Ambos-Spies and Busse [2,3] as well as Tadaki [31] 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.
- $\mathbf{x}$ fails to contain some string w as a substring.
- There is a finite connected automaton with binary input alphabet such that not all states of it are visited when reading $\mathbf{x}$.
- 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.

Finite state predictable sequences are not normal and, by the work of Schnorr and Stimm [28], there is a finite-automaton martingale which succeeds on such a sequence. Furthermore, there are sequences which are not normal but also not finite-state predictable. An example can be obtained by translating the decimal Champernowne sequence $\mathbf{y}$ [12] into a binary sequence $\mathbf{x}$ such that $\mathbf{x}(k) = 1$ iff $\mathbf{y}(k) \in \{1, 2, \ldots, 9\}$ and $\mathbf{x}(k) = 0$ iff $\mathbf{y}(k) = 0$; now the resulting $\mathbf{x}$ is not normal; however, $\mathbf{x}$ contains every substring as a substring and is thus also not finite-state predictable.

Theorem 13. *Every C_S–incompressible sequence is normal.*

Proof. Assume that the sequence $\mathbf{x}$ is not normal. According to Lemma 11 there exist $\alpha \in (0,1)$ and $\sigma \in \mathrm{dom}(S)$ such that for infinitely many integers $n > 0$ we have $C_{T^S_\sigma}(\mathbf{x} \upharpoonright n)) < \alpha \cdot n$. 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. □

7 How Large Is the Set of Incompressible Sequences?

It is natural to ask whether the converse of Theorem 13 is true. The results in [1,5,28,33] 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 substring of w (exactly once). It is well-known that de Bruijn strings of any order exist, and have an explicit construction [14,32]. 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 [26] it is shown that every sequence of the form

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

is normal provided that the function $f : \mathbf{N} \to \mathbf{N}$ 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 14. *Every string* $w,\ B(1) \sqsubseteq w \sqsubset \mathbf{b}_f$ *can be represented in the form*

$$w = B(1)^{f(1)} B(2)^{f(2)} \cdots B(n-1)^{f(n-1)} B(n)^j w' \tag{2}$$

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

Proof. Indeed, in the case

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

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

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

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 13 is false. This also proves that C_S–incompressibility is stronger than all other known forms of finite automata based incompressibility, cf. [1,5,15,28,33].

Theorem 15. *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 $T_n = (\{0,1\}, \{s_1, \ldots, s_{n+1}\}, s_1, \delta_n, \mu_n)$ as follows:

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

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

$$f(n) \geq \max\{|\sigma_{n+1}|, n^n, 2^{n+2}\} \text{ and } f(i) \geq 2 \cdot f(i-1). \tag{3}$$

Finally, let $p_i = 0^{f(i)-1}1$ and $p'_j = 0^{j-1}1$. Eq. (2) shows that

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

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

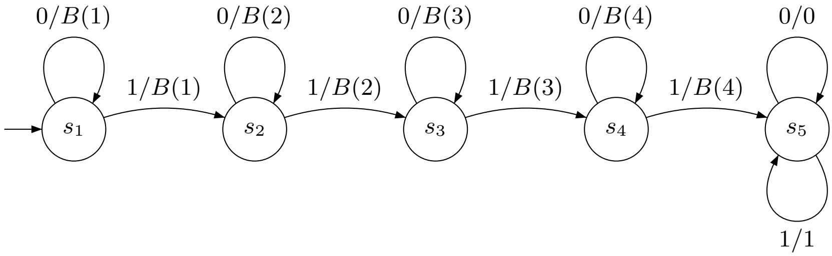

Fig. 1. Block representation of the transducer T_4

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

and

$|\sigma_n| + |p_1 \cdots p_{n-1} \cdot p'_j \cdot w'|$

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

This shows that for every prefix w of $\mathbf{b}_f$ presented in the form (2) as

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

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 $\lim_{n\to\infty} C_S(\mathbf{x} \upharpoonright n)/|n| = 0$. □

In the proof of Theorem 15 we have used an arbitrary function f satisfying (3). Of course, there exist computable and incomputable such functions.

Corollary 16. *For every perfect enumeration S there are normal and C_S-compressible computable and incomputable sequences.*

One might also consider transducers which satisfy that $|\mu(q, a)| \leq m$ 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. For these one can then also consider the variant $C_S^{(m)}$ of C_S which looks at complexity using m-bounded transducers. The following result is a sample result for this area.

Theorem 17. *For every enumeration S of all 2-bounded admissible transducers, there are normal sequences* $\mathbf{x}$ *such that* $\lim_{n\to\infty} C_S^{(2)}(\mathbf{x} \upharpoonright n)/n = 1/2$.

Proof. We start from the transducers T_n defined in the proof of Theorem 15 and we split every long output $B(i)$ of T_n into 2^{i-1} pieces of length 2. Formally, we replace the states $s_i, i \le n$, 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 $B(i) = u_{i,1} \cdots u_{i,2^{i-1}}$ 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 $f: \mathbf{N} \to \mathbf{N}$, $f(n) \ge \max\{|\sigma_{n+1}'|, n^n, 2^{n+2}\}$, $f(i) \ge 2 \cdot f(i-1)$, is as in the proof of Theorem 15. 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. (2) we have:

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

We then have:

$$\begin{aligned} |T_n'(\bar{p}_1 \cdots \bar{p}_{n-1}(0^{j-1})1 \cdot w')| &= \textstyle\sum_{i=1}^{n-1} 2^i \cdot f(i) + 2^n j + |w'| \\ &\ge \textstyle\sum_{i=1}^{n-1} 2^i \cdot f(i) + 2^n j, \end{aligned}$$

and

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

finally obtaining

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

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

Theorem 17 can be easily generalised to m-bounded complexity thereby yielding the bound $\lim_{n\to\infty} C_S^{(m)}(\mathbf{x} \upharpoonright n)/n = 1/m$. Moreover, the results of Theorems 15 and 17 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 15, we can ask whether the finite state complexity of each sequence $\mathbf{x}$ representing a Liouville number satisfies the inequality $\limsup_{n\to\infty} C_S(\mathbf{x} \upharpoonright n)/n < 1$. The answer is negative: Example 12 of [30] shows that there are sequences $\mathbf{x}$ representing Liouville numbers having $\limsup_{n\to\infty} K(\mathbf{x} \upharpoonright n)/n = 1$, hence by Theorem 5, $\limsup_{n\to\infty} C_S(\mathbf{x} \upharpoonright n)/n = 1$.

The following result complements Theorem 15: the construction is valid for every enumeration, but the degree of incompressibility is slightly smaller.

Theorem 18. *There exists an infinite, normal and computable sequence* $\mathbf{x}$ *which satisfies the condition* $\liminf_{n\to\infty} C_S(\mathbf{x} \upharpoonright n)/n = 0$*, for all 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 [12]. 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 of all (e, n, m) such that $e < n \wedge \varphi_e(n) = m$. 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 $|\alpha_{n_k}| \leq \sqrt{s_k}$. 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 $\beta = 0\sigma(0)0\sigma(1)\ldots0\sigma(u-1)1^{t^2-t}$ 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

$\beta_0 \dots \beta_s$ an extension of $\alpha_{f(0)} \dots \alpha_{f(s)}$. Note that the image of $\beta_0 \dots \beta_s$ is at most $m-1$ symbols longer than $\alpha_{f(0)} \dots \alpha_{f(s)}$. Let $\sigma = \alpha_{f(0)} \dots \alpha_{f(t \cdot n)}$. 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 $\alpha_{f(0)} \dots \alpha_{f(t \cdot n)}$, 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 13, it is normal. □

The proof method in Theorem 18 can be adapted to obtain the following result.

Theorem 19. *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 18; furthermore, it is assumed that the listing of the T_n is one-one. However, f has 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 $\alpha_{f(0)}\alpha_{f(1)} \dots \alpha_{f(n \cdot t)}$ 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. □

8 Conclusion and Open Questions

Enumerations are — in the context of this paper — computable listings of all admissible transducers and have a prefix-free domain. We have investigated two main notions of enumerations, the perfect ones (which have a decidable domain, are one-one, are surjective and have a computable inverse) and the universal ones (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 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.

The results of Sections 6 and 7 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 [28].

There are various interesting open questions. Here are three more: 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 $\mathbf{x}$? For which enumerations S is it true that every sequence satisfying $C_S(\mathbf{x} \upharpoonright n) \geq n - c$ is Martin-Löf random? What is the relation between C_S–incompressible sequences and ε–random sequences, [8]? Note that some ε–random sequences can be finite-state predictable by not having a certain substring, cf. [31], hence they can be compressed by a single transducer; this is, however, not true for all ε–random sequences.

Acknowledgments. The authors would like to thank Sanjay Jain and the anonymous referees of TAMC 2014 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. In: Calude, C.S., Dinneen, M.J., Vajnovszki, V. (eds.) DMTCS 2003. LNCS, vol. 2731, pp. 97–108. Springer, Heidelberg (2003)
3. Ambos-Spies, K., Busse, E.: Computational aspects of disjunctive sequences. In: Fiala, J., Koubek, V., Kratochvíl, J. (eds.) MFCS 2004. LNCS, vol. 3153, pp. 711–722. Springer, Heidelberg (2004)
4. Berstel, J.: Transductions and Context-free Languages. Teubner (1979)
5. Bourke, C., Hitchcock, J.M., Vinodchandran, N.V.: Entropy rates and finite-state dimension. Theoretical Computer Science 349(3), 392–406 (2005)
6. Buhrman, H., Fortnow, L.: Resource-bounded Kolmogorov complexity revisited. In: Reischuk, R., Morvan, M. (eds.) STACS 1997. LNCS, vol. 1200, pp. 105–116. Springer, Heidelberg (1997)
7. Calude, C.S.: Information and Randomness. An Algorithmic Perspective, 2nd edn. Springer, Berlin (2002)
8. 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)

9. Calude, C.S., Salomaa, K., Roblot, T.K.: Finite state complexity. Theoretical Computer Science 412, 5668–5677 (2011)
10. Calude, C.S., Salomaa, K., Roblot, T.K.: State-size hierarchy for FS-complexity. International Journal of Foundations of Computer Science 25(1), 37–50 (2012)
11. Chaitin, G.J.: A theory of program size formally identical to information theory. Journal of the Association for Computing Machinery 22, 329–340 (1975)
12. Champernowne, D.G.: The construction of decimals normal in the scale of ten. Journal of the London Mathematical Society 8, 254–260 (1933)
13. 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. In: Proceedings of STOC 2002, pp. 792–801. ACM Press (2002)
14. de Bruijn, N.: A combinatorial problem. Proceedings of the Koninklijke Nederlandse Akademie van Wetenschappen 49, 758–764 (1946)
15. Dai, J.J., Lathrop, J.I., Lutz, J.H., Mayordomo, E.: Finite-state dimension. Theoretical Computer Science 310, 1–33 (2004)
16. Doty, D., Moser, P.: Finite-state dimension and lossy compressors, arxiv:cs/0609096v2 (2006)
17. Doty, D., Moser, P.: Feasible Depth. In: Cooper, S.B., Löwe, B., Sorbi, A. (eds.) CiE 2007. LNCS, vol. 4497, pp. 228–237. Springer, Heidelberg (2007)
18. Downey, R., Hirschfeldt, D.: Algorithmic Randomness and Complexity. Springer, Heidelberg (2010)
19. Katseff, H.P.: Complexity dips in random infinite binary sequences. Information and Control 38(3), 258–263 (1978)
20. Lehman, E.: Approximation Algorithms for Grammar-based Compression, PhD Thesis. MIT (2002)
21. Lehman, E., Shelat, A.: Approximation algorithms for grammar-based compression. In: Proceedings of SODA 2002, pp. 205–212. SIAM Press (2002)
22. Li, M., Vitányi, P.: An Introduction to Kolmogorov Complexity and Its Applications, 3rd edn. Springer (2007)
23. Martin-Löf, P.: The definition of random sequences. Information and Control 9, 602–619 (1966)
24. Martin-Löf, P.: Complexity oscillations in infinite binary sequences. Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete 19, 225–230 (1971)
25. Nies, A.: Computability and Randomness. Clarendon Press, Oxford (2009)
26. Nandakumar, S., Vangapelli, S.K.: Normality and finite-state dimension of Liouville numbers, arxiv:1204.4104v1 [cs.IT] (2012)
27. Rytter, W.: Application of Lempel-Ziv factorization to the approximation of grammar-based compression. Theoretical Computer Science 302, 211–222 (2002)
28. Schnorr, C.P., Stimm, H.: Endliche Automaten und Zufallsfolgen. Acta Informatica 1, 345–359 (1972)
29. Schnorr, C.P.: Process complexity and effective randomness tests. Journal of Comput. System Sciences 7, 376–388 (1973)
30. Staiger, L.: The Kolmogorov complexity of real numbers. Theoretical Computer Science 284, 455–466 (2002)
31. Tadaki, K.: Phase Transition and Strong Predictability, CDMTCS Research Report 435 (2013)
32. van Lint, J.H., Wilson, R.M.: A Course in Combinatorics. Cambridge University Press (1993)
33. Ziv, J., Lempel, A.: Compression of individual sequences via variable-rate coding. IEEE Transactions on Information Theory 24, 530–536 (1978)