Logical Methods in Computer Science
Volume 20, Issue 3, 2024, pp. 15:1–15:8
https://lmcs.episciences.org/

Submitted May 27, 2022
Published Aug. 12, 2024

DETERMINISTIC PUSHDOWN AUTOMATA CAN COMPRESS SOME NORMAL SEQUENCES

OLIVIER CARTON [a,b] AND SYLVAIN PERIFEL [a]

[a] Université Paris Cité, CNRS, IRIF, F-75013, Paris, France
e-mail address: Olivier.Carton@irif.fr, Sylvain.Perifel@irif.fr

[b] Institut Universitaire de France

Abstract. In this paper, we give a deterministic one-to-one pushdown transducer and a normal sequence of digits compressed by it. This solves positively a question left open in a previous paper by V. Becher, P. A. Heiber and the first author.

1. Introduction

A real number is normal to an integer base if, in its infinite expansion in that base, all blocks of digits of the same length have the same limiting frequency. Émile Borel [Bor09] defined normality more than one hundred years ago to formalize the most basic form of randomness for real numbers. Many of his questions are still open, such as whether any of π, e or $\sqrt{2}$ is normal in some base, as well as his conjecture that the irrational algebraic numbers are normal to each base [Bor50]. This motivates the search for new characterizations of the concept of normality.

One characterization is based on finite state machines. A sequence of digits is normal if and only if it cannot be compressed by lossless finite transducers (also known as finite-state compressors). These are deterministic finite automata augmented with an output tape with injective input-output behavior. The compression ratio of an infinite run of a transducer is defined as the lim inf, over all its finite prefixes, of the ratio between the number of symbols written and the number of symbols read so far. A given sequence is said to be compressed by a given transducer if the compression ratio it achieves is less than 1.

A direct proof of the incompressibility characterization of normal sequences can be found in [BH13]. However, the result was already known, although by indirect and more involved arguments. For instance, combining results of Schnorr and Stimm [SS72] and Dai, Lathrop, Lutz and Mayordomo [DLLM04] yields an earlier proof: the characterization of normality given in [SS72] is based on martingales and the equivalence between martingales and compressibility is shown in [DLLM04]. It is also proved in [DM06, SLZ95] that compression ratio and decompression ratio coincide.

The notion of incompressibility by finite state machines is quite robust: adding some feature to one-to-one transducers does not allow them to compress normal sequences. It is

Key words and phrases: normality, pushdown automata, compression.

LOGICAL METHODS IN COMPUTER SCIENCE
DOI:10.46298/LMCS-20(3:15)2024

Finite-state transducer	det.	non-det.
No extra memory	N	N
One counter	N	N
One stack	**?**	Y
More than one counter	Y (T)	Y (T)
One stack and one counter	Y (T)	Y (T)

Table 1: Compressibility by different kinds of transducers.

proved in [BCH15] that non-deterministic non-real-time transducers, with no extra memory or just a single counter, cannot compress any normal sequence. Non-real-time means here that the value of the counter can be incremented and decremented without consuming any input symbol. It is also shown in [CH15] that two-way transducers cannot compress normal sequences. Adding too much memory yields compressibility results: it is clear that Turing complete machines can compress computable normal sequences like the Champernowne sequence [Cha33]. This includes non-real-time transducers with at least two counters. Note however that Turing completeness is not necessary. For instance, it is shown in [LS97] that some normal sequences are compressed by Lempel-Ziv algorithm. Combining non-determinism with a single stack also yields compressibility of some normal sequence. Results given in [BCH15] are summarized in Table 1. One question left open was whether a deterministic pushdown transducer can compress a normal sequence, that is the question mark in Table 1, where (T) means Turing-complete. In this paper, we answer this question positively.

Theorem 1.1. *There is a deterministic one-to-one pushdown transducer that can compress some normal sequence.*

A more precise statement is given in Proposition 2.1 where the pushdown transducer and the normal sequence compressed by it are made explicit.

2. Precise statement

Before giving a more precise statement, we recall a few definitions. Let A be a finite alphabet. Let A^* and $A^{\mathbb{N}}$ be respectively the set of finite words and the set of (infinite) sequences over A. The positions of words and sequences are numbered starting at 1. To denote the symbol at position i of a word (respectively sequence) w we write $w[i]$ and to denote the substring of w from position i to j we write $w[i{:}j]$. The length of a finite word w is denoted by $|w|$. The empty word is denoted by ε. For a word $w = a_1 \cdots a_n$, let $\widetilde{w}$ be the *reverse* of w defined by $\widetilde{w} = a_n \cdots a_1$. We write $\#E$ for the cardinality of a finite set E. For w and u two words, let us denote by $|w|_u$ the number of possibly overlapping *occurrences* of u in w. A sequence $x \in A^{\mathbb{N}}$ over alphabet A is *normal* if

$$\lim_{n\to\infty} \frac{|x[1{:}n]|_w}{n} = \frac{1}{(\#A)^{|w|}}$$

holds for each word $w \in A^*$.

A *pushdown transducer* is made of input and output alphabets A and B, a stack alphabet Z containing the starting symbol z_0, a finite state set Q containing the initial state q_0 and a finite set of transitions of the form $p, z \xrightarrow{a|v} q, h$ where $p, q \in Q$, $a \in A$, $v \in B^*$,

$z \in Z$ and $h \in Z^*$. The states p and q are the starting and ending states of the transition. The symbol a and the word v are its input and output labels. The stack symbol z and the word h are respectively the symbol popped from the stack and the word pushed to the stack. Note that the transition $p, z \xrightarrow{a|v} q, h$ replaces the top symbol z by the word h. If h is empty, it just pops the symbol z. The transducer is *deterministic* if for each triple (p, z, a), there exists at most one triple (q, h, v) such that $p, z \xrightarrow{a|v} q, h$ is one of its transitions. Note that pushdown transducers sometimes include transitions of the form $p, z \xrightarrow{\varepsilon|v} q, h$, called ε-transitions, that consume no input symbol. Such transitions are not needed for our compressor, but are needed for the decompressor as we shall see.

A configuration C of the transducer is a pair $\langle q, h\rangle$ where $q \in Q$ is its state and $h \in Z^*$ is its stack content. Note that the stack content is written bottom up: the top symbol is the last symbol of h. The starting configuration is the pair $\langle q_0, z_0\rangle$ where q_0 is the initial state and z_0 the starting symbol.

A *run step* is a pair of configuration $\langle C, C'\rangle$ denoted $C \xrightarrow{a|v} C'$ such that $C = \langle p, wz\rangle$, $C' = \langle q, wh\rangle$ for some word $w \in Z^*$ and $p, z \xrightarrow{a|v} q, h$ is a transition of the transducer. A finite (respectively infinite) *run* is a finite (respectively infinite) sequence of consecutive run steps

$$C_0 \xrightarrow{a_1|v_1} C_1 \xrightarrow{a_2|v_2} \cdots \xrightarrow{a_n|v_n} C_n.$$

The input and output labels of the run are respectively $a_1 \cdots a_n$ and $v_1 \cdots v_n$. Note that a transition $p, z \xrightarrow{a|v} q, h$ can be seen as a run step whose starting stack content is reduced to a single symbol z. Conversely, each run step is obtained from a transition $p, z \xrightarrow{a|v} q, h$ by adding a stack content w below the top symbol z.

Let A be the alphabet $\{0, \dots, k-1\}$ for some positive integer k and let B be the alphabet $A \uplus \{\triangle, \square\}$ where $\triangle$ and $\square$ are two new symbols not in A. Now we give the deterministic pushdown transducer $\mathcal{T}_k$ with input alphabet A and output alphabet B. We first describe it informally and second we give a more formal description of its transitions. The transducer $\mathcal{T}_k$ proceeds as follows whenever it reads a symbol $a \in A$ from the input tape. If the symbol a is different from the top symbol of the stack, the symbol a is pushed onto the stack and it is also written to the output tape. If the symbol a is equal to the top symbol of the stack, this top symbol is popped. Every two symbols consecutively popped from the stack, a symbol $\square$ is written to the output tape. An additional symbol $\triangle$ is also written to the output tape if the whole sequence of consecutive popped symbols is of odd length. In other words, after a maximal sequence of n consecutive pops, is written to the output tape either the word $\square^{n/2}$ if n is even or the word $\square^{(n-1)/2}\triangle$ if n is odd. This coding of the length n is far from being optimal but it is sufficient to get compression. More formally the state set of $\mathcal{T}_k$ is $Q = \{0, 1\}$ and the initial state is $q_0 = 0$. Its stack alphabet is $A \uplus \{\perp\}$ and the start symbol z_0 is the new symbol $\perp$. As the symbol $\perp$ is different from any input symbol, it is never popped from the stack. Therefore, the symbol $\perp$ always remains at the bottom of the stack and it is used to mark it. The transitions set E of $\mathcal{T}_k$ is defined as follows.

$$E = \{0, z \xrightarrow{a|a} 0, za : z \neq a\} \qquad \text{Pushing } a \text{ and outputting } a$$
$$\{0, z \xrightarrow{a|\varepsilon} 1, \varepsilon : z = a\} \qquad \text{Popping } z = a \text{ and outputting } \varepsilon$$
$$\{1, z \xrightarrow{a|\Box} 0, \varepsilon : z = a\} \qquad \text{Popping } z = a \text{ and outputting } \Box$$
$$\{1, z \xrightarrow{a|\triangle a} 0, za : z \neq a\} \qquad \text{Pushing } a \text{ and outputting } \triangle a$$

The function realized by this transducer is one-to-one and the inverse function can even be computed by the following deterministic pushdown transducer. This transducer works as follows. Each symbol $a \in A$ is pushed to the stack and output. When $\triangle$ is read, one symbol from the stack is popped and output. When $\Box$ is read, two symbols from the stack are popped and output (the topmost first).

$$E' = \{0, z \xrightarrow{a|a} 0, za : z, a \in A\} \qquad \text{Pushing } a \text{ and outputting } a$$
$$\{0, z \xrightarrow{\triangle|z} 0, \varepsilon : z \in A\} \qquad \text{Reading } \triangle\text{, popping and outputting the top stack symbol}$$
$$\{0, z \xrightarrow{\Box|z} 1, \varepsilon : z \in A\} \qquad \text{Reading } \Box\text{, popping and outputting the top stack symbol}$$
$$\{1, z \xrightarrow{\varepsilon|z} 0, \varepsilon : z \in A\} \qquad \text{Popping and outputting the top stack symbol again}$$

Let $\mathcal{T}$ be a pushdown transducer with input alphabet A and output alphabet B. The *compression ratio* ρ of an infinite run

$$C_0 \xrightarrow{a_1|v_1} C_1 \xrightarrow{a_2|v_2} C_2 \xrightarrow{a_3|v_3} \cdots$$

is

$$\rho = \liminf_{n\to\infty} \frac{|v_1 \cdots v_n| \log \#B}{n \log \#A}$$

The factors $\log \#A$ and $\log \#B$ take into account the alphabet sizes. Without them, it would be to easy to compress by taking a larger alphabet B. The transducer $\mathcal{T}$ is said to *compress* a sequence x if it realizes a one-to-one function and if the compression ratio ρ of the infinite run of $\mathcal{T}$ on input x satisfies $\rho < 1$.

The following proposition is a more precise reformulation of Theorem 1.1.

Proposition 2.1. *Let A be the alphabet $\{0, \dots, k-1\}$ for some large enough integer k. Let w_n be, for each integer $n \geqslant 1$, the concatenation in lexicographic order of all words of length n over A. The deterministic pushdown transducer $\mathcal{T}_k$ given above compresses the normal sequence $x = w_1\widetilde{w}_1 w_2\widetilde{w}_2 w_3\widetilde{w}_3 \cdots$.*

Before proving the proposition, we make some comments. The proof that the sequence x is normal is an easy adaptation that the Champernowne sequence is normal [BC18, Thm 7.7.1].

The proposition states the result for k large enough. The proof below shows that the condition $k \geqslant 7$ is sufficient but numerical experiments show that $k \geqslant 5$ is actually sufficient.

Some other normal sequences are compressible by the same transducer. For each integer $n \geqslant 1$, let $u_1, \dots, u_{\ell_n}$ be an enumeration in some order of all words of length n over A. This means that $\ell_n = (\#A)^n$. Let w_n be the word $u_1\widetilde{u}_1u_2\widetilde{u}_2 \cdots u_{\ell_n}\widetilde{u}_{\ell_n}$ for each integer $n \geqslant 1$. The sequence $x = w_1w_2w_3\cdots$ is also compressible by the same transducer $\mathcal{T}_k$. It seems that this result can be proved using the same techniques. However, our numerical experiments suggest that the compression ratio of this latter sequence is worse than the one given in the proposition.

Our numerical experiments show that the compression ratio converges to $3/4$ when the alphabet size k goes to infinity. It seems that the same ideas used in the proof of the proposition can achieve this result, but we preferred simplicity in our presentation.

3. Proof

Now we introduce a congruence $\sim$ on A^* which is used to characterize stack contents of the pushdown transducer $\mathcal{T}_k$. Let $\to$ be the relation defined on A^* as follows. Two words w and w' satisfy $w \to w'$ if there are two words u and v and a symbol $a \in A$ such that $w = uaav$ and $w' = uv$. The word w' is thus obtained from w by deleting two consecutive identical symbols. A word w is *irreducible* for $\to$ if it contains no consecutive occurrences of the same symbol. Let $\xrightarrow{*}$ be the reflexive-transitive closure of the relation $\to$. Let us recall that the relation $\to$ is *Noetherian* if there is no infinite chain $w_0 \to w_1 \to w_2 \to \cdots$ and that it is *confluent* if the relations $w \xrightarrow{*} w_1$ and $w \xrightarrow{*} w_2$ imply that there exists another word w' such that $w_1 \xrightarrow{*} w'$ and $w_2 \xrightarrow{*} w'$.

Lemma 3.1. *The relation $\to$ is Noetherian and confluent.*

Proof. Since $w \to w'$ implies $|w| > |w'|$, the relation $\to$ is obviously Noetherian. Hence, by Newman's lemma, it is sufficient for confluence to prove that $\to$ is locally confluent. This means that relations $w \to w_1$ and $w \to w_2$ imply that there exists w' such that $w_1 \xrightarrow{*} w'$ and $w_2 \xrightarrow{*} w'$. Suppose that $w \to w_1$ and $w \to w_2$ where w_1 and w_2 are obtained from w by deleting respectively the blocks aa and bb of two identical symbols. Either the two blocks overlaps and $a = b$ or they are disjoint. In the former case, w is equal to u_1aaau_3 for $u_1, u_3 \in A^*$ and in the latter case, w is equal to $u_1aau_2bbu_3$ for $u_1, u_2, u_3 \in A^*$ if it is assumed, by symmetry, that aa occurs before bb. In the former case $w' = w_1 = w_2 = u_1au_3$ and in the latter case $w_1 = u_1u_2bbu_3$ and $w_2 = u_1aau_2u_3$ and then $w_1 \to w'$ and $w_2 \to w'$ where $w' = u_1u_2u_3$. □

The fact that $\to$ is Noetherian and confluent implies that for each word w, there is a unique irreducible word $\widehat{w}$ such that $w \xrightarrow{*} \widehat{w}$. Let us define the equivalence relation $\sim$ on A^* by $w \sim w'$ if and only if $\widehat{w} = \widehat{w}'$. It can be checked that the relation $\sim$ is the reflexive-symmetric-transitive closure of $\to$, that is, the relation $(\to \cup \leftarrow)^*$: the equality $\widehat{w} = \widehat{w}'$ implies the relations $w \xrightarrow{*} \widehat{w} = \widehat{w}' \xleftarrow{*} w'$ and the converse is due to confluence which allows us to replace each pattern $w_1 \xleftarrow{*} w \xrightarrow{*} w_2$ by the pattern $w_1 \xrightarrow{*} w' \xleftarrow{*} w_2$ for some word w'. The equivalence relation $\sim$ is actually a congruence: if $u \sim u'$ and $v \sim v'$, then $uv \sim u'v'$. Note that each palindrome of even length, that is, each word of the form $w\widetilde{w}$, satisfies $w\widetilde{w} \sim \varepsilon$. The following lemma is easily proved by induction on the length of w.

Lemma 3.2. *After reading a word w, the stack content of $\mathcal{T}_k$ is $\bot\widehat{w}$ where $\widehat{w}$ is the unique irreducible word such that $w \xrightarrow{*} \widehat{w}$.*

Let us recall that the input sequence is $w_1\widetilde{w}_1w_2\widetilde{w}_2w_3\widetilde{w}_3\cdots$. The lemma just stated above implies that the stack only contains the bottom symbol $\bot$ after reading the prefix $w_1\widetilde{w}_1\cdots w_n\widetilde{w}_n$ because $w_i\widetilde{w}_i \sim \varepsilon$ for each integer $i \geqslant 1$.

Let $P = \{1, \ldots, |w_n\widetilde{w}_n|\}$ be the set of positions of symbols in $w_n\widetilde{w}_n$. Each symbol of $w_n\widetilde{w}_n$ is consumed by either a pushing transition or a popping transition. In the former case, the consumed symbol is pushed to the stack. In the latter case, the same symbol as the one consumed is popped from the stack. This dichotomy induces the partition $P = P_0 \uplus P_1$ where P_0 is the set of positions of symbols being pushed and P_1 is the set of positions of symbols popping. Since the stack only contains the bottom symbol $\bot$ before and after reading $w_n\widetilde{w}_n$, each pushed symbol is popped later. Then, the run of $\mathcal{T}_k$ also induces a function f from P_0 to P_1 which maps each position of a pushed symbol to the position of the symbol that pops it. This function f is of course, one-to-one and onto because each pushed symbol is popped by exactly one symbol. By definition, the function f satisfies that $i < f(i)$ for each i in P_0 and that the symbols at positions i and $f(i)$ are the same. The stack policy implies that if two positions i and j in P_0 satisfy $i < j$, then $f(i) > f(j)$. Let us call an *edge* a pair $(i, f(i))$. An edge is *short* if $f(i) - i = 1$ and is *long* if $f(i) - i > 1$.

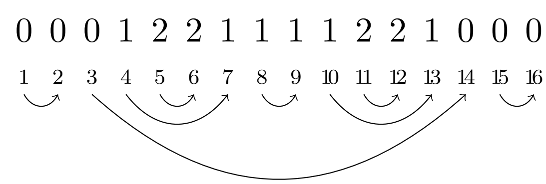

Figure 1: Example of a function f: $f(3) = 14$.

Let us call a *block* a maximal set $\{i, i+1, \ldots, j\}$ of consecutive positions with the same symbol at each position. Maximal means here that the set cannot be expanded to the left because either $i = 1$ or symbols at positions $i-1$ and i are different and that it cannot be expanded to the right because either $j = |w_n\widetilde{w}_n|$ or symbols at positions j and $j+1$ are different. The following lemma states a link between the number of long edges and the length of the output of $\mathcal{T}_k$.

Lemma 3.3. *While reading $w_n\widetilde{w}_n$, the transducer $\mathcal{T}_k$ writes at most $|w_n\widetilde{w}_n| - h/6$ symbols where h is the number of blocks of length 1 in $w_n\widetilde{w}_n$.*

Proof. Let d be the difference between the length of $w_n\widetilde{w}_n$ and the number of symbols written by $\mathcal{T}_k$ while reading $w_n\widetilde{w}_n$. We have to prove that $d \geqslant h/6$. Each symbol pushed to the stack by $\mathcal{T}_k$ is also written to the output tape. A maximal sequence of n consecutive popping transitions of $\mathcal{T}_k$ writes $\square^{n/2}$ if n is even and $\square^{(n-1)/2}\triangle$ if n is odd. This shows that such a maximal sequence of length $n \geqslant 2$ contributes $\lfloor n/2 \rfloor \geqslant n/3$ to d.

Let N be the number of popping transitions belonging to a sequence of at least two popping transitions. From the previous reasoning $d \geqslant N/3$.

For each block of length 1, there is a long edge $(i, f(i))$ such that either i or $f(i)$ belongs to the block. This shows that the number of long edges is at least $h/2$. Due to the nesting of edges, the position $f(i) - 1$ is also the arrival of another edge. These two edges contribute at least 1 to N. This shows that $N \geqslant h/2$ and hence $d \geqslant h/6$. ∎

Lemma 3.4. *For $n \geqslant 3$, the number of blocks of length 1 in $w_n\widetilde{w}_n$ is exactly*

$$\frac{(k-1)^2}{k^2}|w_n\widetilde{w}_n|.$$

Proof. Note that w_n starts with n occurrences of the symbol 0 and ends with n occurrences of the symbol $k-1$. It follows that a block of length 1 in $w_n\widetilde{w}_n$ can occur neither at the beginning nor at the end of w_n and $\widetilde{w}_n$. The number of blocks of length 1 in $w_n\widetilde{w}_n$ is twice the number of blocks of length 1 in w_n.

If each word of length n has exactly m cyclic occurrences in a word w, then each word u of length $1 \leqslant \ell \leqslant n$ has $mk^{n-\ell}$ occurrences since u is the prefix of $k^{n-\ell}$ words of length n. By Theorem 5 in [ABFY16], each word of length n has exactly n cyclic occurrences in w_n. Applying the previous remark for $\ell = 3$ yields that each word of length 3 has exactly nk^{n-3} cyclic occurrences in w_n. A block of length 1 corresponds to an occurrence of a word abc where the symbols $a, b, c \in A$ satisfy $a \neq b$ and $b \neq c$. Such a word abc cannot overlap the border of w_n. It follows that each word abc with $a \neq b$ and $b \neq c$ has exactly nk^{n-3} occurrences in w_n. Since the length of w_n is nk^n and there are $k(k-1)^2$ such words abc, the proof is complete. □

Proof of Proposition 2.1. Combining Lemmas 3.3 and 3.4 yields that, for $n \geqslant 3$, the number of symbols written by $\mathcal{T}_k$ while reading the word $w_n\widetilde{w}_n$ is at most $(1-(k-1)^2/6k^2)|w_n\widetilde{w}_n|$. Therefore the transducer $\mathcal{T}_k$ given above compresses the normal sequence $x = w_1\widetilde{w}_1w_2\widetilde{w}_2w_3\widetilde{w}_3\cdots$ as soon as the following inequality holds.

$$\left(1-\frac{(k-1)^2}{6k^2}\right)\frac{\log(k+2)}{\log k} < 1$$

The first term of the left hand side decreases to 5/6 and the second term decreases to 1. The inequality is satisfied for $k \geqslant 7$ since it boils down to $9^{43} < 7^{49}$. □

Acknowledgements. We would like to thank the anonymous reviewers for their valuable comments that improved the quality of the paper.

References

[ABFY16] N. Alvarez, V. Becher, P. A. Ferrari, and S. A. Yuhjtman. Perfect necklaces. *Advances of Applied Mathematics*, 80:48–61, 2016.

[BC18] V. Becher and O. Carton. Normal numbers and computer science. In V. Berthé and M. Rigo, editors, *Sequences, Groups, and Number Theory*, Trends in Mathematics Series, pages 233–269. Birkhäuser/Springer, 2018. doi:10.1007/978-3-319-69152-7.

[BCH15] V. Becher, O. Carton, and P. A. Heiber. Normality and automata. *Journal of Computer and System Sciences*, 81:1592–1613, 2015. doi:10.1016/j.jcss.2015.04.007.

[BH13] V. Becher and P. A. Heiber. Normal numbers and finite automata. *Theoretical Computer Science*, 477:109–116, 2013.

[Bor09] É. Borel. Les probabilités dénombrables et leurs applications arithmétiques. *Rendiconti Circ. Mat. Palermo*, 27:247–271, 1909.

[Bor50] É. Borel. Sur les chiffres décimaux de $\sqrt{2}$ et divers problèmes de probabilités en chaîne. *Comptes rendus de l'Académie des Sciences de Paris*, 230:591–593, 1950.

[CH15] O. Carton and P. A. Heiber. Normality and two-way automata. *Inf. Comput.*, 241:264–276, 2015. doi:10.1016/j.ic.2015.02.001.

[Cha33] D. G. Champernowne. The construction of decimals normal in the scale of ten. *J. London Math. Soc.*, 8:254–260, 1933.

[DLLM04] J. Dai, J. Lathrop, J. Lutz, and E. Mayordomo. Finite-state dimension. *Theoretical Computer Science*, 310:1–33, 2004.

[DM06] D. Doty and Ph. Moser. Finite-state dimension and lossy decompressors. *CoRR*, abs/cs/0609096, 2006. arXiv:cs/0609096.

[LS97] J. I. Lathrop and M. Strauss. A universal upper bound on the performance of the Lempel-Ziv algorithm on maliciously-constructed data. In *Compression and Complexity of SEQUENCES*, pages 123–135. IEEE, 1997. doi:10.1109/SEQUEN.1997.666909.

[SLZ95] D. Sheinwald, A. Lempel, and J. Ziv. On encoding and decoding with two-way head machines. *Inf. Comput.*, 116(1):128–133, 1995. doi:10.1006/inco.1995.1009.

[SS72] C. P. Schnorr and H. Stimm. Endliche Automaten und Zufallsfolgen. *Acta Informatica*, 1:345–359, 1972.