Monadic Second-Order Theories of Self-Similar Words

Valérie Berthé
Université Paris Cité, IRIF, CNRS
Paris, France
berthe@irif.fr

Toghrul Karimov
Mihir Vahanwala
Max Planck Institute for Software Systems
Saarland Informatics Campus
Saarbrücken, Germany
{toghs,mvahanwa}@mpi-sws.org

***Abstract*—A fundamental question in logic and verification, whose origins lie in the seminal works of Büchi, Elgot, and Rabin, is the following: for which unary predicates $P_1, \dots, P_k$ is the monadic second-order theory of $\langle \mathbb{N}; <, P_1, \dots, P_k \rangle$ decidable? Equivalently, for which infinite words α can we decide whether a given automaton $\mathcal{A}$ accepts α? Carton and Thomas showed decidability in case α is a fixed point of a letter-to-word substitution σ, i.e., $\sigma(\alpha) = \alpha$. However, abundantly more words, e.g., Sturmian words, are characterised by a broader notion of self-similarity that uses a set S of substitutions. A word α is said to be directed by a sequence $s = (\sigma_n)_{n \in \mathbb{N}}$ over S if there is a sequence of words $(\alpha_n)_{n \in \mathbb{N}}$ such that $\alpha_0 = \alpha$ and $\alpha_n = \sigma_n(\alpha_{n+1})$ for all n. We study the automaton acceptance problem for such words and prove, among others, the following. Given S and an automaton $\mathcal{A}$, we can compute an automaton $\mathcal{B}$ that accepts $s \in S^\omega$ if and only if s directs a word α accepted by $\mathcal{A}$.**

I. Introduction

In 1962, Büchi proved that the monadic second-order (MSO) theory of the structure $\langle \mathbb{N}; < \rangle$ is decidable [1], and, in doing so, laid the foundations of automata theory over infinite words. Subsequently in 1966, Elgot and Rabin [2] adopted automata-theoretic techniques to show how to decide the MSO theory of $\langle \mathbb{N}; <, P \rangle$ for various interesting unary predicates P including $\{n!\colon n \in \mathbb{N}\}$ and $\{2^n\colon n \in \mathbb{N}\}$. By then, it was already known that unary predicates lay at the frontiers of decidability: expanding $\langle \mathbb{N}; < \rangle$ with most natural functions (e.g., addition or doubling) or non-unary predicates yields undecidable MSO theories [3], [4], [5]. The question thus arose: for which unary predicates $P_1, \dots, P_k$ is the MSO theory of $\langle \mathbb{N}; <, P_1, \dots, P_k \rangle$ decidable? Equivalently,[1] for which infinite words α is the *automaton acceptance problem*, which asks whether a given automaton $\mathcal{A}$ accepts α, decidable?

The automaton acceptance problem under various assumptions on α has been studied, among others, by Semënov, Carton and Thomas, and Rabinovich [6], [7], [8]. Semënov [6] showed decidability for α that are *effectively almost-periodic*. These include, for example, the Thue-Morse word and *toric words*, which are obtained from certain compact dynamical systems [9]. Recently, almost-periodic words and Semënov's decidability result have been identified as a powerful tool for analysing linear while loops in program verification as well as linear dynamical systems [10]. Carton and Thomas [7], on the other hand, used algebraic methods to show decidability for *morphic* α, which include α that can be constructed by infinitely iterating a letter-to-word morphism σ on a starting letter a. Their result implies, in one fell swoop, the decidability of the MSO theory of $\langle \mathbb{N}; <, \{p(n)a^n\colon n \in \mathbb{N}\} \rangle$ where $a \geq 1$ and p is a polynomial with integer coefficients satisfying $p(\mathbb{N}) \subseteq \mathbb{N}$. Finally, Semënov [6] (also see [11]) as well as Rabinovich and Thomas [12] gave two characterisations of P such that the MSO theory of $\langle \mathbb{N}; <, P \rangle$ is decidable, with the caveat that it is not possible to effectively verify either of their criteria. In fact, it has been shown that determining whether the MSO theory of $\langle \mathbb{N}; <, P \rangle$ is decidable for a given unary predicate P is complete for the class Σ_3 of the arithmetical hierarchy [13].

In order to apply these results to solve a verification problem, one needs to link the setting under consideration to the generation of some combinatorially special word. As a recent example, [14] showed decidability of the MSO theory of $\langle \mathbb{N}; <, P_1, \dots, P_k \rangle$ where $P_1, \dots, P_k$ belong to a large class of predicates of arithmetic origin. In particular, the MSO theory of $\langle \mathbb{N}; <, a^{\mathbb{N}}, b^{\mathbb{N}} \rangle$ owes its decidability to the fact that the order in which powers of coprime a and b occur is effectively almost-periodic by dint of being a *Sturmian word* (Sec. III-A). Sturmian words form an extremely well-studied and fundamental class of "special words" that appear naturally in a range of fields including number theory, computer graphics, astronomy, and music [15, Chap. 9.6], [16, Chap. 6]. They are toric and contain a subset of morphic words. We refer the reader to [9] for a more detailed survey of the role of Sturmian words, and word combinatorics in general, in logic and verification.

In this paper, we adopt the *S-adic perspective*, which is a powerful tool for elucidating a vast array of combinatorial properties of infinite words. Akin to a continued fraction expansion of a real number, or even a Fourier decomposition of a signal, we seek to write an infinite word α as an infinite composition of (possibly different) *substitutions*. A substitution σ over an alphabet Σ gives rules to replace each

[1] We can encode the structure $\langle \mathbb{N}; <, P_1, \cdots, P_k \rangle$ as a word α over the alphabet $\{0, 1, \cdots, 2^{k-1}\}$ where the k-bit binary representation of $\alpha(n)$ indicates the predicates to which n belongs. An MSO formula φ is then translated into an automaton $\mathcal{A}$ that accepts α if and only if φ holds in the structure. Conversely, we can encode a word $\alpha \in \{1, \cdots, k\}^\omega$ as the structure $\langle \mathbb{N}; <, P_1, \dots, P_k \rangle$ where $n \in P_i$ if and only if $\alpha(n) = i$.

letter $a \in \Sigma$ with a non-empty word $\sigma(a) \in \Sigma^+$. For example, the Fibonacci substitution σ_{fib} over $\{0,1\}$ replaces 0 with 01 and 1 with 0; the Fibonacci word $\alpha_{\mathsf{fib}} = 01001010\cdots$ is obtained as the limit of iterating σ_{fib} infinitely on the letter 0 (or, alternatively, the letter 1). Hence we have the infinite decomposition $\alpha_{\mathsf{fib}} = \sigma_{\mathsf{fib}} \circ \cdots \circ \sigma_{\mathsf{fib}} \cdots$. In general, we have a set S of substitutions, and say that an infinite sequence s over S *directs* a word α if there there exists a sequence of words $\left(\alpha^{(n)}\right)_{n\in\mathbb{N}}$ such that $\alpha^{(0)} = \alpha$, and $\alpha^{(n)} = \sigma_n\left(\alpha^{(n+1)}\right)$ for all n. This gives us the S-adic decomposition $\alpha = \sigma_0 \circ \sigma_1 \circ \cdots$.[2] We refer to any $s \in S^\omega$ as a *directive sequence*. As hinted earlier, this transformation immediately reveals many desirable properties of words from the perspective of logical decidability. For example, if s is *weakly primitive*, then any α it directs is *uniformly recurrent* (Sec. III), which implies that α is almost-periodic (Sec. II-C).

Entire classes of words such as Sturmian (Sec. III-A) and Arnoux-Rauzy (Sec. III-B) words can be defined in terms of directive sequences over specific S; we call such a class $\mathcal{W}$ *S-adic*. Many interesting S-adic families, including the ones we consider in this paper are defined by *weakly primitive* directive sequences, which implies that every $\alpha \in \mathcal{W}$ is uniformly recurrent and hence almost-periodic. By the result of Semënov discussed above, under an effectiveness assumption, such words have a decidable automaton acceptance problem and MSO theory. It is thus natural to consider the decision problem of the *common* MSO theory of such a class $\mathcal{W}$ of words: given an automaton $\mathcal{A}$, what is the set of all $\alpha \in \mathcal{W}$ that are accepted by $\mathcal{A}$? Does there exist at least one such α? Equivalently, given a formula φ, can we decide whether there exists $\alpha \in \mathcal{W}$ that induces a structure in which φ holds?

We remark that this question is similar in spirit to the problem solved in [17]: given a formula φ, it is decidable whether there exists a Sturmian word α such that φ holds in the induced first-order structure $\langle \mathbb{N}; <, +, P\rangle$. The key ingredient in this case is that Sturmian words have descriptions in certain non-standard (compared to the usual binary or decimal number systems) number systems that are amenable to automata-theoretic tools. We refer the reader to [18] for more on *automatic structures* that are used to decide various extensions of Presburger arithmetic like the one above.

Our main contribution is showing that ω-regular specifications on words in the Σ-space translate into ω-regular constraints on their directive sequences in the S-space. Consequently, the common MSO theory of an S-adic class of words is decidable. This generalises the recent result of [19] that, given a finite set S and an automaton $\mathcal{A}$ whose language is closed (i.e., a non-deterministic Büchi automaton whose states are all accepting), it is decidable whether $\mathcal{A}$ accepts a word α directed by some S. Our results yield algorithms that thoroughly answer questions of the kind "Which Sturmian words are accepted by a given automaton $\mathcal{A}$?"

[2] Of course, we are not interested in trivial decompositions that, for example, just permute the letters back and fort, as these do not tell us anything new about α.

Outline and contributions of the paper

In Sec. II we recall preliminaries from topology, algebra, automata theory, word combinatorics, and number theory. In Sec. III we formally define what it means for a directive sequence to (i) generate a word, (ii) direct a word, and (iii) generate a *shift*. Briefly, a sequence of substitutions $(\sigma_n)_{n\in\mathbb{N}}$ generates α if there exists a sequence of letters $(a_n)_{n\in\mathbb{N}}$ such that $\alpha = \lim_{n\to\infty} \sigma_0 \cdots \sigma_n(a_n)$; generating a word is a strong form of directing it. We then recall well-known properties of directive sequences, the most important of them being *weak primitivity*, and then describe Sturmian, Arnoux-Rauzy, as well as dendric words (and shifts), which are all generated by weakly primitive directive sequences.

In Sec. IV we study the structure of words directed or generated by directive sequences. Our key new insight is the augmentation of a directive sequence s over[3] S into a *congenial* expansion $\hat{s}$ over $S \times \Sigma$ (Def. 17), which generates a word incrementally and predictably (Lem. 18). We prove the following pivotal results.

- If s generates α, then it also directs α (Lem. 16).
- If s directs a word α, then α is a concatenation of words generated congenially by s (Lem. 20).
- If s is weakly primitive, then s congenially generates α if and only if s directs α (Lem. 22)

In Sec. V, we identify that substitutions over Σ, when considered relative to an automaton $\mathcal{A}$ over Σ, coalesce into finitely many equivalence classes: we denote the set of classes as $\Xi_{\mathcal{A}}$. Given a sequence s over S, or $\hat{s}$ over $S \times \Sigma$, we naturally define its trace to be a word over $\Xi_{\mathcal{A}}$ or over $\Xi_{\mathcal{A}} \times \Sigma$.

The series of developments in Sec. IV and Sec. V converge in Sec. VI and Sec. VII, where we prove our main results.

Morphic Words Let $\mathcal{A}$ be an automaton over Σ, and let σ, π be substitutions. Using only their respective equivalence classes $\xi, \zeta \in \Xi_{\mathcal{A}}$, we can compute a regular language $L \subseteq \Sigma^+$ such that the word $\pi(\sigma^\omega(u))$ is well-defined and accepted by $\mathcal{A}$ if and only if $u \in L$ (Thm. 30). Hence we can also characterise all such π, σ, u, which substantially generalises the result of Carton and Thomas [7].

Generated Words Given an automaton $\mathcal{A}$ over Σ, we can construct an automaton $\mathcal{B}$ over $\Xi_{\mathcal{A}} \times \Sigma$ such that $\mathcal{B}$ accepts the trace of $\hat{s}$ if and only if $\hat{s}$ is congenial and generates a word accepted by $\mathcal{A}$ (Thm. 32).

Directed Words Given an automaton $\mathcal{A}$ over Σ, we can construct an automaton $\mathcal{B}$ over $\Xi_{\mathcal{A}}$ such that $\mathcal{B}$ accepts the trace of s if and only if s directs a word accepted by $\mathcal{A}$ (Thm. 34).

Generated Shifts Given an automaton $\mathcal{A}$ over Σ that recognises $\mathcal{L}(\mathcal{A})$, consider an automaton $\mathcal{A}_{\mathsf{suf}}$ that recognises $\Sigma^* \cdot \mathcal{L}(\mathcal{A})$. We can construct an automaton $\mathcal{B}$ over $\Xi_{\mathcal{A}_{\mathsf{suf}}}$ such that $\mathcal{B}$ accepts the trace of s if and only if s is weakly primitive and generates a shift that contains a word accepted by $\mathcal{A}$ (Thm. 37).

[3] Our set S of substitutions could possibly be infinite, which is the case with, for example, dendric words.

In Sec. VIII, we refine our main results for Sturmian, Arnoux-Rauzy, and (ternary) dendric words, which are generated by weakly primitive directive sequences, and have *a priori* known *factor complexity* (Sec. II-C). We show that for such classes, acceptance by $\mathcal{A}$ is completely determined by the first $N(\mathcal{A})$ *partial quotients* of the directive sequence (Thm. 40). In the case of Sturmian words, this has a nice geometric interpretation: an automaton can only resolve the slope and intercept associated with a Sturmian word up to a "pre-determined" finite precision.

II. Preliminaries

An alphabet Σ is a finite and non-empty set of symbols. We write ε for the empty word. A substitution σ is a non-erasing morphism from Σ^* to Σ^*, i.e., $\sigma(v) = \varepsilon$ if and only if $v = \varepsilon$. We denote the set of all such substitutions by $S(\Sigma)$. For substitutions μ, σ, we write $\mu\sigma$ for $\mu \circ \sigma$. A substitution is positive if every $b \in \Sigma$ appears in $\sigma(a)$ for all $a \in \Sigma$, and left-proper if there exists $b \in \Sigma$ such that $\sigma(a)$ begins with b for all $a \in \Sigma$.

For a word α, $\alpha(j)$ denotes the letter at the jth position of α, $\alpha[i, j)$ denotes the finite word $\alpha(i) \cdots \alpha(j-1)$, and $\alpha[j, \infty)$ denotes the infinite word $\alpha(j)\alpha(j+1)\cdots$. A finite word u is a factor of a word v if there exist indices i, j such that $v[i, j) = u$. When we say that an object is effectively computable, we mean that a representation in a scheme (that will be clear from the context) is effectively computable.

A. Topology of finite and infinite words

We equip $\Sigma^* \cup \Sigma^\omega$ with the product topology, and define the distance between words u, v to be 2^{-n}, where n is the first position in which they differ. E.g., distinct $a, b \in \Sigma$ are a distance of $2^0 = 1$ apart. A notion of convergence of sequences of words follows naturally.

Definition 1. *Let $(u_n)_{n\in\mathbb{N}}$ be a sequence of finite words. We define $\alpha = \lim_{n\to\infty} u_n \in \Sigma^\infty = \Sigma^* \cup \Sigma^\omega \cup \{\bot\}$ as follows.*

- *If there exists $v \in \Sigma^*$ and N such that for all $n \geq N$, $u_n = v$, then $\alpha = v$.*
- *If there exists $\beta \in \Sigma^\omega$ such that for all j, $u_n[0, j) = \beta[0, j)$ for all sufficiently large n, then $\alpha = \beta$.*
- *Otherwise, $\alpha = \bot$, which denotes lack of convergence.*

The space $X = \Sigma^* \cup \Sigma^\omega$ is compact. In particular, every infinite sequence of words from X has some infinite subsequence that converges to an element of X. The cylinder sets defined by fixing first N letters are both closed and open.

B. Monoids and automata theory

We consider infinite words from both the algebraic and combinatorial perspectives, and hence use monoids and automata respectively

Recall that a monoid M is a set equipped with an associative binary operation $M \times M \to M$ (usually denoted by $\cdot$ and written in infix notation) and a neutral element 1_M with respect to $\cdot$: for all $m \in M$, $1_M \cdot m = m \cdot 1_M = m$. In this paper, monoids serve as an alternate perspective through which we view ω-regular languages.[4] This is realised via morphisms h from the free monoid Σ^* (generated by letters of the alphabet Σ with concatenation as the binary operation and the empty word as the neutral element) into a finite monoid M. Throughout this paper we shall work with monoids in which $m \cdot m' = 1_M$ if and only if $m = m' = 1_M$: this is justified by the fact that the concatenation of two words is the empty word ε if and only if both words are ε: for convenience of exposition we shall say the monoid has no identity divisors.

Since we work with infinite words, it is necessary to have a well-defined notion of an infinite product of elements of a monoid M. This is obtained by extending M to an *ω-monoid* $\overline{M}$ [20, Chap. 7], [21], whose elements fall into two partitions, M and M_ω. We additionally have that:

- $\overline{M}$ is a monoid.
- For an infinite sequence $(m_j)_{j\in\mathbb{N}}$ over M, the infinite product $m_0 \cdot m_1 \cdots \in M_\omega$, and is associative.
- If $m \in M_\omega$, then for all m' we have that $m \cdot m' = m$ and $m' \cdot m \in M_\omega$.

We extend the free monoid Σ^* into $\Sigma^\infty = \Sigma^* \cup \Sigma^\omega \cup \{\bot\}$, where $\bot$ as before is a construct to denote non-convergence. We extend a substitution σ to also act on Σ^ω in the natural way, and define $\sigma(\bot) = \bot$.

We next present the techniques for extending M to $\overline{M}$ for completeness, since we slightly adapt them for our purposes.

Definition 2. *Let M be a finite monoid. A linked pair $(\eta, e) \in M \times M$ is such that e is idempotent (i.e., $e = e \cdot e$) and $\eta \cdot e = \eta$. Linked pairs (η, e) and (η', e') are conjugate if there exist $x, y \in M$ such that $e = x \cdot y$, $e' = y \cdot x$, and $\eta' = \eta \cdot x$ (which together imply $\eta = \eta' \cdot y$).*

We use $[\eta, e]$ to denote the conjugacy class of the linked pair (η, e). The definition of the infinite product hinges on the following result.

Theorem 3. *Let Σ be a finite alphabet, M be a finite monoid, and $h : \Sigma^* \to M$ be a morphism. For every $\alpha \in \Sigma^\omega$ there exists a linked pair $(\eta, e) \in M \times M$ such that α can be factorised into a sequence of words $u_0, u_1, \ldots \in \Sigma^+$ satisfying $h(u_0) = \eta$, and $h(u_j) = e$ for all $j \geq 1$. Furthermore, the linked pair (η, e) is unique up to conjugacy, i.e., if a different linked pair (η', e') satisfies the above requirement, then it is conjugate to (η, e).*

The proof of Thm. 3 is an application of Ramsey's theorem; we provide it in App. A-A for completeness. Observe that this theorem enables us to define the infinite product of elements of M as a conjugacy class of linked pairs. Indeed, we can apply the above theorem by taking $\Sigma = M$ and h that maps a word $m_1 \cdots m_k$ to the monoid element $m_1 \cdots m_k$.

Definition 4. *We extend a finite monoid M into an ω-monoid $\overline{M}$ as follows.*

- *The elements of M_ω consist of conjugacy classes of linked pairs of M, along with a special error element $\bot$.*

[4] Semigroups are more common; we prefer monoids because of a technical necessity to distinguish the empty word.

- *Infinite products of elements of M are equal to a conjugacy class of linked pairs as per Thm. 3. As a single exception, we define $(1_M)^\omega = 1_M$, which reflects the convention that infinite concatenation of the empty word yields the empty word.*
- *For all $m \in M$, we have that $m \cdot \bot = \bot$, and $m \cdot [\eta, e] = [m \cdot \eta, e]$.*

We remark that $m^\omega \in M_\omega$ can readily be computed for any $m \in M$ that is not the identity. Recall that for every $m \in M$, there is an exponent p and an idempotent e such that $m^p = e$. Then $m^\omega = [e, e]$.

The definition above allows us to extend a morphism $h : \Sigma^* \to M$ to $\overline{h} : \Sigma^\infty \to \overline{M}$ in the obvious way, which maps $\alpha \in \Sigma^\omega$ to a conjugacy class of linked pairs per Thm. 3, and $\bot$ to $\bot$. We next note a compositional property.

Lemma 5. *Let Σ be a finite alphabet, σ be a substitution, M be a finite monoid, and $h : \Sigma^* \to M$ be a morphism. We have that $\overline{h} \circ \sigma = \overline{h \circ \sigma}$.*

Proof. The two morphisms are trivially equal when applied to finite words and $\bot$. We thus only consider their applications to $\alpha \in \Sigma^\omega$. Let $\overline{h \circ \sigma}(\alpha) = [\eta, e]$, where $\alpha = u_0 u_1 \cdots$ with $h \circ \sigma(u_0) = \eta$, and $h \circ \sigma(u_j) = e$ for $j \geq 1$. Now, we factorise $\sigma(\alpha) = \sigma(u_0)\sigma(u_1)\cdots$, giving us $\overline{h} \circ \sigma(\alpha) = [\eta, e]$, in fact with the same witness linked pair (η, e). □

For the combinatorial perspective, we work with deterministic parity automata because some technical tools, particularly those involving Semënov's theorem, require the automaton to be deterministic.

Definition 6. *A deterministic parity automaton $\mathcal{A}$ over infinite words is given by the tuple $(\Sigma, Q, q_{\mathsf{init}}, \delta, \mathsf{index})$, where Σ is the finite input alphabet, Q is the finite set of states, $q_{\mathsf{init}} \in Q$ is the initial state, $\delta : Q \times \Sigma \to Q$ is the transition function, and $\mathsf{index} : Q \to \mathbb{N}$ is a function that associates each state with a natural number. The run $r_\alpha \in Q^\omega$ of $\mathcal{A}$ on $\alpha \in \Sigma^\omega$ satisfies $r_\alpha(0) = q_{\mathsf{init}}$, and for all n, $\delta(r_\alpha(n), \alpha(n)) = r_\alpha(n+1)$. The word α is accepted if $\limsup_n \mathsf{index}(r_\alpha(n))$ is even. We write $\mathcal{L}(\mathcal{A})$ for the set of all words accepted by $\mathcal{A}$.*

By an automaton we mean a deterministic parity automaton, unless specified otherwise; in some proofs we will construct nondeterministic (Büchi or parity) automata for technical convenience. The translation between these models is effective. Deterministic parity automata recognise precisely the class of ω-regular languages.

The following standard lemmata establish that monoids and automata are equivalent perspectives to recognise ω-regular languages; for completeness, we describe the relevant constructions in App. A-B.

Lemma 7. *Let $\mathcal{A}$ be an automaton over an alphabet Σ. We can construct a finite monoid $M_\mathcal{A}$ (with no identity divisors) called the journey monoid of $\mathcal{A}$, morphism $h_\mathcal{A}$ from Σ^* into $M_\mathcal{A}$ (with the trivial kernel), respectively extend them into an ω-monoid $\overline{M_\mathcal{A}}$, $\overline{h_\mathcal{A}}$, and construct $F \subseteq \overline{M_\mathcal{A}}$ such that for all $\alpha \in \Sigma^* \cup \Sigma^\omega$, α is (infinite and) accepted by $\mathcal{A}$ if and only if $\overline{h_\mathcal{A}}(\alpha) \in F$.*

Lemma 8. *Let M be a finite monoid and $x \in \overline{M}$. We can construct an automaton $\mathcal{A}_x$ that accepts an infinite sequence $(x_n)_{n\in\mathbb{N}}$ over M if and only if $x = \prod_{n=0}^\infty x_n$.*

Proof. If $x \in M$, we check that there are only finitely many terms that are not 1_M, and that the product of these terms is x: this can easily be done with a Büchi automaton. Suppose therefore $x \notin M$. Write $x = [\eta, e]$, and let $(\eta_1, e_1), \ldots, (\eta_k, e_k)$ be the linked pairs in this conjugacy class. It is straightforward to construct a Büchi automaton that checks that the input sequence admits one of these linked-pair factorisations. □

C. Uniformly recurrent words

For a word $\alpha \in \Sigma^\omega$, the set $\mathcal{L}(\alpha) = \{u : u \text{ is a factor of } \alpha\}$ is called the *factor language* (or when the context is clear, simply *language*) of α. The factor complexity function p_α takes as input a number n and returns the number of factors of α that have length n. For example, if α is Sturmian, then $p_\alpha(n) = n + 1$ for all n; if α is an Arnoux-Rauzy word over a d-letter alphabet, then $p_\alpha(n) = n(d-1) + 1$ for all n.

For an infinite word α and $l \geq 0$, denote by $R_\alpha(l)$ the smallest $r \in \mathbb{N} \cup \{\infty\}$ such that every factor of α of length l is a factor of every factor of α of length r. We call R_α the *recurrence function* of α. A word $\alpha \in \Sigma^\omega$ is said to be *uniformly recurrent* if $R_\alpha(l) \in \mathbb{N}$ for every $l \in \mathbb{N}$. That is, every $u \in \Sigma^l$ either does not occur in α, or occurs infinitely often with bounded gaps. It is clear from the definition that for uniformly recurrent α, the value of $R_\alpha(l)$ only depends on $\mathcal{L}(\alpha)$. We record the following, which follows by brute enumeration.

Lemma 9. *Let $\alpha \in \Sigma^\omega$ be uniformly recurrent. Suppose we have access to an oracle that, given $u \in \Sigma^*$, checks whether $u \in \mathcal{L}(\alpha)$. Then we can effectively compute $R_\alpha(l)$.*

Semënov [6] gave an algorithm[5] for determining whether a given automaton $\mathcal{A}$ accepts a given uniformly recurrent word α, which is represented by (A) an oracle computing $\alpha(n)$ on input n and (B) an oracle computing an upper bound $\overline{R}_\alpha(l)$ on $R_\alpha(l)$ given l.

Theorem 10 (Semënov). *Let $\mathcal{A}$ be an automaton and α be a uniformly recurrent word represented by the oracles (A-B). We can effectively compute $M \in \mathbb{N}$ such that a state q of $\mathcal{A}$ appears infinitely often in $\mathcal{A}(\alpha)$ if and only if it appears in $\mathcal{A}(\alpha)[M, 2M)$.*

In other words, to check whether $\mathcal{A}$ accepts α, we simply need to run $\mathcal{A}$ on α for $2M$ steps, and observe the states that are visited. See [23, Chap. 3.1] for effective bounds on M, from which we can deduce the following.

Proposition 11. *Let $\mathcal{A}$ be an automaton and α be a uniformly recurrent word represented by the oracles (A-B). We can*

[5] Semënov's result applies to the more general family of *effectively almost-periodic words* [22].

compute l, M such that for any uniformly recurrent β with $\beta[0,M) = \alpha[0,M)$ and $R_\beta(n) = R_\alpha(n)$ for all $n \le l$, we have that $\mathcal{A}$ accepts α if and only if it accepts β.

The shift generated by α. The *(one-sided) shift operator* $T\colon \Sigma^\omega \to \Sigma^\omega$ is defined by $T((a_n)_{n\in\mathbb{N}}) = (a_{n+1})_{n\in\mathbb{N}}$. A shift is a closed subset X of Σ^ω that satisfies $T(X) \subseteq X$. A minimal shift is one that does not contain any proper subshifts. The language $\mathcal{L}(X)$ of a shift X is the set of words u that appear as factors of some $\alpha \in X$. The factor complexity function p_X of a shift takes as input a number n and returns that number of length-n words in $\mathcal{L}(X)$.

The shift generated by a word $\alpha \in \Sigma^\omega$ is the minimal shift containing α. The following is well-known.

Lemma 12. *The shift generated by a uniformly recurrent word $\alpha \in \Sigma^\omega$ is the set of all uniformly recurrent $\beta \in \Sigma^\omega$ satisfying $\mathcal{L}(\alpha) = \mathcal{L}(\beta)$.*

For an automaton $\mathcal{A}$ we denote by $\mathcal{A}_{\mathsf{suf}}$ the automaton that recognises the language $\Sigma^* \cdot \mathcal{L}(\mathcal{A})$. Applying Lem. 9 and Semënov's theorem to uniformly recurrent words, we obtain the following.

Lemma 13. *Let $\mathcal{A}$ be an automaton and $\alpha \in \Sigma^\omega$ be a uniformly recurrent word. Define*

$$X_\alpha = \{\beta : \beta \text{ is uniformly recurrent with } \mathcal{L}(\alpha) = \mathcal{L}(\beta)\}.$$

Then $\mathcal{L}(\mathcal{A}) \cap X_\alpha \neq \varnothing$ if and only if α is accepted by $\mathcal{A}_{\mathsf{suf}}$. Consequently, $\mathcal{L}(\mathcal{A}_{\mathsf{suf}}) \cap X_\alpha$ is either X_α or $\varnothing$.

Proof. Suppose $\mathcal{A}_{\mathsf{suf}}$ accepts α. Then $\mathcal{A}$ accepts $\alpha[n,\infty)$ for some $n \in \mathbb{N}$. It remains to observe that $\alpha[n,\infty) \in X_\alpha$.

Now suppose there exists uniformly recurrent $\beta \in X_\alpha$ that is accepted by $\mathcal{A}$. Recall that $\mathcal{L}(\alpha) = \mathcal{L}(\beta)$. By uniform recurrence, for any suffix γ of α we have $\mathcal{L}(\gamma) = \mathcal{L}(\beta)$ and hence $R_\gamma(l) = R_\beta(l)$ for all l. Invoking Prop. 11, let M be such that for any such γ, if $\gamma[0,M) = \beta[0,M)$ then $\mathcal{A}$ accepts γ. Because $\beta \in X_a$, there must exist a suffix γ of α such that $\gamma[0,M) = \beta[0,M)$ and hence $\mathcal{A}$ accepts γ. Thus γ is a witness that $\mathcal{A}_{\mathsf{suf}}$ accepts α.

The consequence follows from the fact that if $\beta \in X_\alpha$, then $X_\beta = X_\alpha$. Hence $X_\alpha \cap \mathcal{L}(\alpha) \neq \varnothing \Leftrightarrow X_\beta \cap \mathcal{L}(\alpha) \neq \varnothing$. ☐

D. Ostrowski numeration systems

Let $\eta \in (0,1) \setminus \mathbb{Q}$. The *continued fraction expansion* of η is the unique sequence $(a_n)_{n\ge1}$ of positive integers such that

$$\eta = \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cdots}}$$

We write $\eta = [0; a_1, a_2, \ldots]$. The *convergents* $(p_n/q_n)_{n\in\mathbb{N}}$ of η are obtained by truncating the expansion at the respective n-th levels. The numerators and denominators satisfy the recurrences $p_0 = 0$, $q_0 = 1$, $p_1 = 1$, $q_1 = a_1$, and $(p_{n+2}, q_{n+2}) = a_{n+2} \cdot (p_{n+1}, q_{n+1}) + (p_n, q_n)$ for all $n \ge 0$. The convergents are the *locally best approximants* of η: for every $n \in \mathbb{N}$, $p \in \mathbb{Z}$, and $0 < q < q_n$,

$$|q_n\eta - p_n| < \min_{p\in\mathbb{N}} |q\eta - p|$$

which implies that

$$\left|\eta - \frac{p_n}{q_n}\right| < \min_{p\in\mathbb{N}} \left|\eta - \frac{p}{q}\right|.$$

The *Ostrowski numeration system in base η* is based on[6] the sequence $\theta_n = q_n\eta - p_n$. For any $\chi \in [-\eta, 1-\eta]$, there exists a sequence $(b_n)_{n\ge1}$ over $\mathbb{N}$ such that (i) $0 \le b_1 < a_1$, (ii) $0 \le b_n \le a_n$ for all $n \ge 2$, (iii) for all n, $b_n = 0$ if $b_{n+1} = a_{n+1}$, and

$$\chi = \sum_{n=1}^{\infty} b_n \theta_{n-1}.$$

We refer to $(b_n)_{n\ge1}$ as an *Ostrowski expansion of χ in base η*. Conversely, every $(b_n)_{n\in\mathbb{N}}$ satisfying (i-iii) is an Ostrowski expansion of some χ in base η, i.e., the infinite sum converges to a value in $[-\eta, 1-\eta]$.[7] If $\xi \notin \mathbb{Z} + \eta\mathbb{Z}$, then it has a unique Ostrowski expansion in base η. Otherwise χ can have two expansions in base η, one of which is ultimately zero.

III. S-ADIC WORDS AND SHIFTS

Let us begin to develop the technical tools we will need to solve the automaton acceptance problem for Sturmian and other families of words. Let Σ be an alphabet and $S \subseteq S(\Sigma)$ be a possibly infinite set of substitutions.

We refer to $\alpha \in \Sigma^* \cup \Sigma^\omega$ as *S-directed* if there exists a sequence $(\sigma_n)_{n\in\mathbb{N}}$ over S and a sequence of infinite words $(\alpha^{(n)})_{n\in\mathbb{N}}$ such that $\alpha^{(0)} = \alpha$ and $\sigma_n(\alpha^{(n+1)}) = \alpha^{(n)}$ for all $n \in \mathbb{N}$. We say that $(\sigma_n)_{n\in\mathbb{N}}$ *directs* α. A word $\alpha \in \Sigma^* \cup \Sigma^\omega$ is called *S-generated* if there exists a sequence $(\sigma_n, a_n)_{n\in\mathbb{N}}$ over $S(\Sigma) \times \Sigma$ such that[8]

$$\alpha = \lim_{n\to\infty} \sigma_0 \cdots \sigma_n(a_n). \tag{1}$$

We refer to $(\sigma_n, a_n)_{n\in\mathbb{N}}$ as an *S-adic expansion* of α. For finite or infinite α, whenever (1) holds we say that $(\sigma_n, a_n)_{n\in\mathbb{N}}$ *generates* α. In both the S-directed and the S-generated settings (to which we collectively refer as *S-adic*), the sequence $(\sigma_n)_{n\in\mathbb{N}}$ is called a *directive sequence*.

Directive sequences have also been defined to generate shifts (which we also call S-adic) as follows. For $s = (\sigma_n)_{n\in\mathbb{N}}$, let $\mathcal{L}(s)$ to be the set of all $u \in \Sigma^+$ such that u is a factor of $\sigma_0 \cdots \sigma_n(a)$ for some n and $a \in \Sigma$. The shift X_s is defined as $\{\alpha \in \Sigma^\omega : \mathcal{L}(\alpha) \subseteq \mathcal{L}(s)\}$. By definition, any word generated by s is also in the shift X_s. We next study properties of various special classes of S-adic words with which we will work.

[6] The numeration system can also be defined in terms of $|\theta_n|$; in this paper we follow [24, Sec. 2.4] and work with the θ_n-based system.

[7] To check this, observe that θ_n alternates between positive and negative; $\theta_0 = \eta$, $\theta_1 = a_1\eta - 1$; $\theta_{n+2} = a_{n+2}\theta_{n+1} + \theta_n$.

[8] In the most general definition, each σ_n is allowed to operate on a different alphabet, i.e., $\sigma_n : \Sigma_{n+1}^* \to \Sigma_n^*$ and $a_n \in \Sigma_{n+1}$ for a sequence of alphabets $(\Sigma_n)_{n\in\mathbb{N}}$.

Definition 14. *A sequence $(\sigma_n)_{n\in\mathbb{N}}$ over $S(\Sigma)$ is everywhere growing if for any $(a_n)_{n\in\mathbb{N}}$ over Σ we have that*

$$\lim_{n\to\infty} |\sigma_0 \cdots \sigma_n(a_n)| = \infty.$$

It is weakly primitive if for every n there exists $m \geq n$ such that $\sigma_n \cdots \sigma_m$ is positive, i.e., for every $b, c \in \Sigma$, b appears in $\sigma_n \cdots \sigma_m(c)$.

In order to justify our interest in weakly primitive expansions, we state two standard facts, whose proofs we provide in App. B for completeness.

Lemma 15. *If $s = (\sigma_n)_{n\in\mathbb{N}}$ is weakly primitive, then it is everywhere growing. The generated shift*

$$X_s = \{\alpha \in \Sigma^\omega : \alpha \text{ is uniformly recurrent and } \mathcal{L}(\alpha) = \mathcal{L}(s)\}.$$

We now illustrate the above concepts through example classes of S-adic words.

A. Sturmian words

Let $\Sigma = \{0, 1\}$ and $\eta \in (0, 1) \setminus \mathbb{Q}$. The *characteristic* Sturmian word with slope η is defined by

$$\alpha_\eta(n) = \lfloor (n+2)\eta \rfloor - \lfloor (n+1)\eta \rfloor$$

for $n \in \mathbb{N}$. For instance, the Fibonacci word has $\eta = 1/\phi^2$, where ϕ is the golden ratio. Characteristic Sturmian words are S-directed, where S is taken to be the set $\{\lambda_0, \lambda_1\}$. The substitution λ_i maps i to i, and the other letter j to ij, i.e., it inserts i to the left.

The word α_η is intimately connected to the continued fraction expansion of η. Suppose $\eta = [0; 1 + a_1, a_2, \ldots]$. We then have that α_η is the unique word directed by the sequence

$$\underbrace{\lambda_0, \ldots, \lambda_0}_{a_1 \text{ times}}, \underbrace{\lambda_1, \ldots, \lambda_1}_{a_2 \text{ times}}, \underbrace{\lambda_0, \ldots, \lambda_0}_{a_3 \text{ times}}, \underbrace{\lambda_1, \ldots, \lambda_1}_{a_4 \text{ times}}, \ldots$$

which we denote by s_η. For example, $1/\phi^2$ has the continued fraction expansion $[0; 2, 1, 1, \ldots]$, and hence the Fibonacci word is directed by $(\lambda_0\lambda_1)^\omega$. Observe that λ_0, λ_1 are left-proper. Moreover, for every $k, m > 0$, $\lambda_0^k \lambda_1^m$ is positive, and hence s_η is weakly primitive. For such sequences, the notions of direction and generation coincide, and the word generated is unique (Lem. 23).

A (general) Sturmian word α of slope $\eta \in (0, 1) \setminus \mathbb{Q}$ and intercept $\chi \in [-\eta, 1-\eta]$ is given by one of the following:

$$\alpha(n) = \lfloor (n+2)\eta + \chi \rfloor - \lfloor (n+1)\eta + \chi \rfloor, \quad (2)$$
$$\alpha(n) = \lceil (n+2)\eta + \chi \rceil - \lceil (n+1)\eta + \chi \rceil. \quad (3)$$

Sturmian words are uniformly recurrent, and are equivalently characterised by their factor complexity $p(n) = n + 1$, which is the lowest among non-periodic words. A Sturmian word α with slope η and intercept χ satisfies $\mathcal{L}(\alpha) = \mathcal{L}(\alpha_\eta)$. Hence the shift generated by s_η, denoted by X_η, is the set of all Sturmian words of slope η.

Sturmian words are not necessarily S-adic for S defined above. However, they are S-adic for $S = \{\lambda_0, \lambda_1, \rho_0, \rho_1\}$ where the substitution ρ_i inserts i to the right, i.e, maps i to i and j to ji. Let α be a Sturmian word with slope $\eta = [0; a_1 + 1, a_2, \ldots]$ and intercept χ. Then there exists [24, Prop. 2.7, also see remark after Thm. 2.10] an Ostrowski expansion $(b_n)_{n\in\mathbb{N}}$ of χ in base η such that α is directed by a sequence $(\tau)_{n\in\mathbb{N}}$ where

$$\tau_n = \lambda_0^{b_{2n+1}} \rho_0^{a_{2n+1} - b_{2n+1}} \lambda_1^{b_{2n+2}} \rho_1^{a_{2n+2} - b_{2n+2}}. \quad (4)$$

Conversely, the rules of Ostrowski expansion guarantee that each such sequence obtained from the expansions $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$ is weakly primitive and directs a Sturmian word. Observe that we can unpack each τ_n to obtain a *bona fide* directive sequence over S, and every morphism in S is left-proper.

B. Arnoux-Rauzy and episturmian words

We generalise Sturmian words by considering the alphabet $\Sigma = \{0, \ldots, d-1\}$, and taking the set S of substitutions to be $\{\lambda_0, \rho_0, \ldots, \lambda_{d-1}, \rho_{d-1}\}$. As before, $\lambda_i(i) = \rho_i(i) = i$, and for $j \neq i$, $\lambda_i(j) = ij$, and $\rho_i(j) = ji$. Observe that each λ_i is left-proper.

Similarly to the Sturmian case, *characteristic* Arnoux-Rauzy words [25, Sec. 2.3] are those that are directed by sequences over $\{\lambda_0, \ldots, \lambda_{d-1}\}$ in which each λ_i occurs infinitely often. Such directive sequences are weakly primitive and by Lem. 22, direct and generate a unique word. For instance, the Tribonacci word $\alpha_{\mathsf{trib}} = 0102010 \cdots \in \{0, 1, 2\}^\omega$ is a characteristic Arnoux-Rauzy word, and is directed by the sequence $(\lambda_0\lambda_1\lambda_2)^\omega$.

General Arnoux-Rauzy words [25, Sec. 5] are those directed by sequences s in which each $i \in \{0, \ldots, d-1\}$ is infinitely represented. i.e., for any i, there are infinitely many λ_i or infinitely many ρ_i in s. Such sequences are weakly primitive, and Arnoux-Rauzy words are thus uniformly recurrent. In order to ensure that each directive sequence directs a unique word by the virtue of left-properness [25, Prop. 4.3], we can further require that there be at least one i such that there are infinitely many λ_i in s [25, Thm. 4.12]. Arnoux-Rauzy words over a d-letter alphabet have factor complexity $p(n) = n(d-1) + 1$. Observe that Sturmian words are exactly the two-letter Arnoux-Rauzy words.

An Arnoux-Rauzy word α directed by a sequence $s \in S^\omega$ is factor-equivalent to (i.e., has the same language as) the word α' directed by the sequence s' obtained by replacing each ρ_i in s with the corresponding λ_i [25, Rmk. 3.5]. Thus, any Arnoux-Rauzy word is in the shift generated by a characteristic Arnoux-Rauzy word, and a directive sequence over $\{\lambda_0, \ldots, \lambda_{d-1}\}$.

Episturmian words generalise Arnoux-Rauzy words as follows [25, Sec. 5]: any episturmian $\alpha \in \{0, 1, \cdots, d-1\}^\omega$ is the image of some Arnoux-Rauzy word $\beta \in \{0, 1, \cdots, c-1\}^\omega$ $(c \leq d)$ under a morphism $\sigma_0 \cdots \sigma_m$, where each σ_m is either in S, or is a permutation θ_{ij} of letters that exchanges i, j and fixes the other letters. Thus, episturmian words are also uniformly recurrent. Any sequence s over S directs some episturmian word, and this word is unique if s contains infinitely many λ_i for some i [25, Prop. 4.3]. Conversely, any episturmian word has a directive sequence with infinitely many left-inserting substitutions [25, Thm. 4.12].

C. Dendric words

Dendric words over $\Sigma = \{0, \dots, d-1\}$ generalise Arnoux-Rauzy words from the perspective of word combinatorics. They have striking combinatorial, algebraic, and ergodic properties, and their study has recently attracted sustained interest.

Consider a uniformly recurrent word α, and its factor language $\mathcal{L}(\alpha)$. We can define the *extension graph* $G(u)$ of every $u \in \mathcal{L}(\alpha)$: this is a bipartite graph, whose left partition $V_L(u)$ consists of vertices corresponding to letters a such that $au \in \mathcal{L}(\alpha)$, and right partition $V_R(u)$ consists of vertices corresponding to letters b such that $ub \in \mathcal{L}(\alpha)$. We draw an edge between $a \in V_L(u)$ and $b \in V_R(u)$ if $aub \in \mathcal{L}(\alpha)$. We say that the factor u is dendric in $\mathcal{L}(\alpha)$ if $G(u)$ is a tree, and the word α is dendric if all its factors are dendric in $\mathcal{L}(\alpha)$.

Given a uniformly recurrent shift X and a factor $u \in \mathcal{L}(X)$, we can similarly construct an extension graph $G(u)$. The shift X is dendric if $G(u)$ is a tree for every $u \in \mathcal{L}(X)$. Dendric words over a d-letter alphabet have factor complexity function $p(n) = n(d-1)+1$. Arnoux-Rauzy words, in particular, are dendric [26, Prop. 2.1].

Fix an alphabet Σ. There exists a finite set $S_e \subseteq S(\Sigma)$ "elementary" substitutions such that every dendric word has an S_e-adic expansion, which is obtained as a decomposition of *dendric return substitutions* [27, Thm. 6]. Not every S_e-adic expansion, however, generates a dendric word: in fact, even a sequence of dendric return substitutions is subject to constraints in order to generate a dendric shift [28, Thm. 1].

On the other hand, over the ternary alphabet, *dendric shifts* have been characterised explicitly [26, Thm. 1.1]: we can compute a set S (which is infinite but has a finite representation) of left-proper substitutions, an effective finite partition T_{den} of S [26, Figs. 5, 6], and an automaton $\mathcal{B}_{\mathsf{den}}$ over T_{den}, such that a shift $X \subset \{0,1,2\}^\omega$ is minimal dendric if and only if it is generated by a weakly primitive $s = (\sigma_n)_{n\in\mathbb{N}}$ over S whose trace[9] $t \in T_{\mathsf{den}}^\omega$ is accepted by $\mathcal{B}_{\mathsf{den}}$. We remark that $\mathcal{B}_{\mathsf{den}}$, as given by [26, Fig. 7], recognises a closed language, and clarify that $\mathcal{B}_{\mathsf{den}}$ itself does not check for weak primitivity.

IV. Structure theorems for S-directed words

Let s be a directive sequence over $S(\Sigma)$ for an alphabet Σ. In this section we will show that (i) a word α generated by s is also directed by s, and (ii) a word directed by s can be written as a product of *congenial* words generated by s. Intuitively, such words are generated by $(\sigma_n, a_n)_{n\in\mathbb{N}}$ such that $\sigma_0 \cdots \sigma_n(a_n)$ monotonically converges to the limit. Statement (ii) will play a key role in our analysis of the automaton acceptance problem for S-adic and S-directed words. We begin with (i).

Lemma 16. *If $\alpha \in \Sigma^* \cup \Sigma^\omega$ is generated by s over $S(\Sigma)$, then it is also directed by s. Furthermore, s directs at least one non-empty word $\beta \in \Sigma^+ \cup \Sigma^\omega$.*

[9] The letter $t(n)$ records which partition σ_n belongs to.

Proof. We will first prove a slightly more general version of the first statement. Let $(a_n)_{n\in\mathbb{N}}$ be a sequence of letters and $U_m = \{\sigma_m \cdots \sigma_n(a_n) : n \geq m\}$ for all $m \in \mathbb{N}$. Suppose α is an accumulation point of U_0, which subsumes the case of α being generated by $(\sigma_n, a_n)_{n\in\mathbb{N}}$. We shall inductively prove the existence of $(\alpha^{(m)})_{m\in\mathbb{N}}$ such that $\alpha^{(0)} = \alpha$, $\alpha^{(m)} = \sigma_m\left(\alpha^{(m+1)}\right)$, $\alpha^{(m)}$ is an accumulation point of U_m for all m. The base case is immediate.

For the inductive step, suppose we have constructed $\alpha^{(0)}, \dots, \alpha^{(m)}$ with the properties above. Write $u_{m,n} = \sigma_m \cdots \sigma_n(a_n)$, and observe that by the induction hypothesis, $\alpha^{(m)}$ is the limit of some sequence $(\sigma_m(u_{m+1,n_i}))_{i\in\mathbb{N}}$. By compactness, the sequence $(u_{m+1,n_i})_{i\in\mathbb{N}}$ itself has an infinite subsequence $(u_{m+1,k_j})_{j\in\mathbb{N}}$ that converges. We choose $\alpha^{(m+1)}$ to be the limit, which is an accumulation point of U_{m+1}. By the continuity of σ_m,

$$\lim_{j\to\infty} \sigma_m(u_{m+1,k_j}) = \sigma_m\left(\lim_{j\to\infty} u_{m+1,k_j}\right)$$

which implies that $\alpha^{(m)} = \sigma_m(\alpha^{(m+1)})$.

To prove the second claim, choose an arbitrary sequence $(a_n)_{n\in\mathbb{N}}$ of letters and let β be an accumulation point of $\{\sigma_0 \cdots \sigma_n(a_n) \colon n \in \mathbb{N}\}$. Apply the preceding argument. □

We next introduce congenial expansions, which form the cornerstone of most of our technical results.

Definition 17. *Let Σ be an alphabet. A sequence $((\sigma_n, a_n))_{n\in\mathbb{N}}$ over $S(\Sigma) \times \Sigma$ is congenial if $\sigma_{n+1}(a_{n+1})$ begins with a_n for all n. A word $\alpha \in \Sigma^+ \cup \Sigma^\omega$ is s-congenial for a directive sequence s if $\alpha = \lim_{n\to\infty} \sigma_0 \cdots \sigma_n(a_n)$ for a congenial sequence $((\sigma_n, a_n))_{n\in\mathbb{N}}$.*

The most desirable property of congenial $(\sigma_n, a_n)_{n\in\mathbb{N}}$ is that $\lim_{n\to\infty} \sigma_0 \cdots \sigma_n(a_n)$ is guaranteed to exist, and has every $\sigma_0 \cdots \sigma_n(a_n)$ as a prefix. The following lemma captures this property, and is proven via a straightforward induction.

Lemma 18. *Let $((\sigma_n, a_n))_{n\in\mathbb{N}}$ be congenial, and for $n \geq 1$, $v_n \in \Sigma^*$ be such that $\sigma_n(a_n) = a_{n-1}v_n$. For all n,*

$$\sigma_0 \cdots \sigma_n(a_n) = \sigma_0(a_0) \cdot \sigma_0(v_1) \cdot \sigma_0\sigma_1(v_2) \cdots (\sigma_0 \cdots \sigma_{n-1}(v_n)).$$

We will show in Sec. VII that, for any automaton $\mathcal{A}$ and finite set S of substitutions, the set of all congenial S-adic expansions that define a word accepted by $\mathcal{A}$ is ω-regular. To apply this result to S-directed and general S-adic words we need to argue that it suffices to consider congenial expansions. For a directive sequence s, denote by $\mathsf{congenials}_s$ the set of all s-congenial $\alpha \in \Sigma^\omega$. We next show that this set is finite.

Lemma 19. *Let Σ be an alphabet and $s = (\sigma_n)_{n\in\mathbb{N}}$ be a directive sequence over $S(\Sigma)$. There exist at least one and at most $|\Sigma|$ congenial expansions of the form $(\sigma_n, a_n)_{n\in\mathbb{N}}$, and hence $1 \leq |\mathsf{congenials}_s| \leq |\Sigma|$.*

Proof. By Lem. 16, there exists $(\alpha^{(n)})_{n\in\mathbb{N}}$ over Σ^ω such that $\sigma_0 \cdots \sigma_n(\alpha^{(n+1)}) = \alpha^{(0)}$ for all n. Let a_n be the first letter of $\alpha^{(n)}$. We have that (σ_n, a_{n+1}) is congenial, and hence $|\mathsf{congenials}_s| \geq 1$.

Now suppose there exist $m \geq |\Sigma| + 1$ congenial sequences $(\sigma_n, a_n^{(i)})_{n\in\mathbb{N}}$. By a pigeonhole argument, there must exist $i \neq j$ such that $a_n^{(i)} = a_n^{(j)}$ for infinitely many n. From congeniality it follows that $a_n^{(i)} = a_n^{(j)}$ for all n. □

Congenial words constitute the building blocks of directed words. The "if" part of the following lemma follows by definition; the "only if" part holds because a directed word can naturally be factorised into a congenial prefix and a directed suffix (if the former is finite). A detailed technical proof is provided in App. C.

Lemma 20. *Let Σ be an alphabet and $s = (\sigma_n)_{n\in\mathbb{N}}$ be a directive sequence over $S(\Sigma)$. A word $\alpha \in \Sigma^* \cup \Sigma^\omega$ is directed by s if and only if it can be expressed as a (possibly infinite) concatenation $u_0 u_1 \cdots$ of s-congenial words.*

If s is everywhere growing (and hence weakly primitive), then have the following strengthening of Lem 20.

Lemma 21. *Suppose $s = (\sigma_n)_{n\in\mathbb{N}}$ is everywhere growing and directs $\alpha \in \Sigma^\omega$. Then α has a congenial expansion $(\sigma_n, a_n)_{n\in\mathbb{N}}$.*

Proof. Let $(\alpha^{(n)})_{n\in\mathbb{N}}$ be such that $\alpha^{(0)} = \alpha$ and $\sigma_n(\alpha^{(n+1)}) = \alpha^{(n)}$. Choose a_n to be the first letter of $\alpha^{(n+1)}$. Then $(\sigma_n, a_n)_{n\in\mathbb{N}}$ is congenial, $\sigma_0 \cdots \sigma_n(a_n)$ is a prefix of α for all n by construction, and $\lim_{n\to\infty} \sigma_0 \cdots \sigma_n = \infty$ by the growth assumption. Therefore, $(\sigma_n, a_n)_{n\in\mathbb{N}}$ generates α. □

Combining Lem. 16 and Lem. 21, we obtain the following.

Lemma 22. *Let s be an everywhere growing directive sequence. A word α is directed by s if and only if it is congenially generated by s.*

Finally, we consider weakly primitive left-proper directive sequences, which capture all special word classes we consider.

Lemma 23. *Let $s = (\sigma_n)_{n\in\mathbb{N}}$ be a weakly primitive sequence of left-proper substitutions over $S(\Sigma)$. Then there exists unique $\alpha \in \Sigma^\omega$ that is directed (and by Lem. 22, generated) by s.*

Proof. By left-properness, there exist exactly $|\Sigma|$ congenial sequences of the form $(\sigma_n, a_n)_{n\in\mathbb{N}}$; these differ only on the first letter and generate the same word. Apply Lem. 22. □

V. Equivalence of substitutions modulo $\mathcal{A}$

The idea of the monoid-based approach to the automaton acceptance problem is that, even though there are infinitely many morphisms in $S(\Sigma)$, from the perspective of an automaton $\mathcal{A}$ they can be divided into finitely many equivalence classes, which we now proceed to define. Let $\mathcal{A}$ be a deterministic parity automaton over the alphabet Σ, $M_\mathcal{A}$ be its journey monoid, and $h_\mathcal{A} : \Sigma^* \to M_\mathcal{A}$ be the characteristic morphism that maps each word to its set of journeys. Denote by $\mathsf{morphisms}_\mathcal{A}$ the finite set of monoid homomorphisms h from Σ^* into $M_\mathcal{A}$ such that $h(v) = 1_{M_\mathcal{A}}$ if and only if $v = \varepsilon$; these are exactly the monoid homomorphisms of type $\Sigma^* \to M_\mathcal{A}$ that have the trivial kernel. Note that because $\mathcal{A}$ is deterministic, $h_\mathcal{A} \in M_\mathcal{A}$.

We define an equivalence relation on the set of non-erasing substitutions $\sigma\colon \Sigma^* \to \Sigma^*$. Let $\mathsf{segments}_\sigma$ be the function that takes a letter $a \in \Sigma$ and $h \in \mathsf{morphisms}_\mathcal{A}$, and returns a finite sequence of pairs from $\Sigma \times M_\mathcal{A}$, determined as follows. Write

$$\sigma(a) = b_1 v_1 \cdots b_d v_d$$

where $b_1, \ldots, b_d$ are distinct letters and $v_i \in \{b_1, \ldots, b_i\}^*$ for all $1 \leq i \leq d$. That is, we consider the factorisation of $\sigma(a)$ into segments based on the first occurrence of each letter. Then

$$\mathsf{segments}_\sigma(a, h) = \langle (b_1, h(v_1)), \ldots, (b_d, h(v_d)) \rangle.$$

Note that there are only finitely many possibilities $\mathsf{segments}_\sigma$. For non-erasing $\sigma, \mu\colon \Sigma^* \to \Sigma^*$, define

$$\sigma \equiv_\mathcal{A} \mu \Leftrightarrow \mathsf{segments}_\sigma = \mathsf{segments}_\mu.$$

We denote the class of σ by $[\sigma]_\mathcal{A}$, and the finite set of the equivalence classes by $\Xi_\mathcal{A}$; we refer the reader to App. D for a proof of their effectiveness.

Lemma 24. *Given a deterministic automaton $\mathcal{A}$, we can compute $\sigma_1, \ldots, \sigma_m$ such that $[\sigma_i]_\mathcal{A} \neq [\sigma_j]_\mathcal{A}$ for all $i \neq j$ and $\Xi_\mathcal{A} = \{[\sigma_i]_\mathcal{A} \colon 1 \leq i \leq m\}$.*

For technical convenience, we define the following auxiliary functions that can be derived from $\mathsf{segments}_\sigma$; the first four of them are independent of $\mathcal{A}$.

(a) $\mathsf{expanding}_\sigma$ records for each letter a whether $|\sigma(a)| > 1$. It evaluates to false if and only if $\mathsf{segments}_\sigma(a, h) = \langle (b, 1_{M_\mathcal{A}}) \rangle$ for all h.
(b) $\mathsf{introduces}_\sigma$ maps each letter to a finite sequence of pairs of letters with Boolean flags: if $\sigma(a) = b_1 v_1 \cdots b_d v_d$ when factorised as in the definition of $\mathsf{segments}_\sigma$, then

$$\mathsf{introduces}_\sigma(a) = \langle (b_1, f_1), \ldots, (b_d, f_d) \rangle$$

where $f_i = 1$ if and only if $v_i \neq \varepsilon$.
(c) head_σ maps each letter a to the first letter in $\sigma(a)$.
(d) $\mathsf{letters}_\sigma$ returns for each letter a the set of letters appearing in $\sigma(a)$.
(e) tail_σ takes as input a letter a and a morphism h. Write $\sigma(a) = \mathsf{head}_\sigma(a) \cdot v$. We define $\mathsf{tail}_\sigma(a, h) = h(v)$.
(f) $\mathsf{compose}_\sigma$ takes $h \in \mathsf{morphisms}_\mathcal{A} \to \mathsf{morphisms}_\mathcal{A}$ and returns $h \circ \sigma$.

We next argue that composition of morphisms can be defined on equivalence classes. Let $\sigma, \mu\colon \Sigma^* \to \Sigma^*$ be non-erasing. We show how to determine $\mathsf{segments}_{\sigma\circ\mu}(a, h)$. Suppose $\mu(a) = b_1 v_1 \cdots b_m v_m$, where $b_i \in \Sigma$ and $v_i \in \{b_1, \ldots, b_i\}^*$ for all i. Then

$$\sigma(\mu(a)) = \sigma(b_1)\sigma(v_1) \cdots \sigma(b_m)\sigma(v_m)$$

and each letter of $\sigma(v_i)$ already appears in one of $\sigma(b_1), \ldots, \sigma(b_i)$. Write $\mathsf{segments}_\sigma(b_i, h) = \langle (c_{i,1}, h(w_{i,1})), \ldots, (c_{i,k_i}, h(w_{i,k_i})) \rangle$. Then $\sigma(\mu(a))$ is equal to

$$c_{1,1} w_{1,1} \cdots c_{1,k_1} w_{1,k_1} \sigma(v_1) \cdots c_{m,1} w_{m,1} \cdots c_{m,k_m} w_{m,k_m} \sigma(v_m).$$

Write $t_i = w_{i,k_i}\,\sigma(v_i)$. Observe that each letter of $w_{i,j}$ appears as a factor $c_{e,l}$ before $w_{i,j}$ in the factorisation above; the same applies to every t_i. To compute $\mathsf{segments}_{\sigma\circ\mu}(a,h)$ we begin with the finite sequence

$$(c_{1,1}, h(w_{1,1})), \ldots, (c_{1,k_1-1}, h(w_{1,k_1-1})), (c_{1,k_1}, h(t_1)) \cdots$$
$$(c_{m,1}, h(w_{m,1})), \ldots, (c_{m,k_m-1}, h(w_{m,k_m-1})), (c_{m,k_m}, h(t_m)).$$

Note that the above can be effectively computed from $\mathsf{segments}_\sigma$ and $\mathsf{segments}_\mu$:

$$h(t_i) = h(w_{i,k_i})\, h(\sigma(v_i))$$

and the two factors can be gleaned from $\mathsf{segments}_\sigma(b_i, h)$ and $\mathsf{segments}_\mu(a, \mathsf{compose}_\sigma(h))$, respectively. Rename the indices in the sequence above to obtain $\langle (c_1, h(w_1), \ldots, (c_M, h(w_m)\rangle$ where $c_i \in \Sigma$ and $w_i \in \Sigma^*$ for all i. Recall that $w_i \in \{c_1, \ldots, c_i\}^*$ for all i. But it is possible that $c_i = c_j$ for some i, j. To eliminate these, we repeat the following process for as long as possible. Find the smallest j such that $c_i = c_j$ for some $i < j$. Replace the two consecutive terms $(c_{j-1}, h(w_{j-1}), (c_j, h(w_j))$ with $(c_{j-1}, h(w_{j-1})\, h(c_j)\, h(w_j))$. In the end we are left with $\mathsf{segments}_{\sigma\circ\mu}(a,h)$.

To summarise, we have the following.

Lemma 25. *The set $\Xi_\mathcal{A}$ is a finite monoid with the binary operation $[\sigma]_\mathcal{A} \cdot [\mu]_\mathcal{A} = [\sigma \circ \mu]_\mathcal{A}$ and the identity element $[\mathsf{id}]_\mathcal{A}$, where $\mathsf{id}(w) = w$ for all $w \in \Sigma^*$.*

Proof. Observe that substitutions form a free monoid with composition being the binary operation. The map from substitutions to their equivalence classes respects the binary operation, by construction. Using this fact, it is straightforward to check that the binary operation on equivalence classes is associative, and that $[\mathsf{id}]_\mathcal{A}$ is indeed the identity element. □

We will represent an element $\xi \in \Xi_\mathcal{A}$ with a substitution σ such that $[\sigma]_\mathcal{A} = \xi$. For $\xi \in \Xi_\mathcal{A}$, we denote by $\mathsf{segments}_\xi$ the function $\mathsf{segments}_\sigma$ for some $[\sigma]_\mathcal{A} = \xi$; note that the choice of σ does not matter. We define $\mathsf{expanding}_\xi$ etc. similarly. Next, we show that representatives of the equivalence classes can be effectively computed.

Let $(\sigma_n)_{n\in I}$ be a sequence of non-erasing substitutions of type $\Sigma^* \to \Sigma^*$ and $(a_n)_{n\in I}$ be a sequence of letters from Σ, where I can be finite or infinite. Let $\xi_n = [\sigma_n]_\mathcal{A}$. We define $\mathsf{trace}_\mathcal{A}((\sigma_n)_{n\in I}) = (\xi_n)_{n\in I}$ and $\mathsf{trace}_\mathcal{A}((\sigma_n, a_n)_{n\in I}) = (\xi_n, a_n)_{n\in I}$. We can extend the definition of $[\cdot]_\mathcal{A}$ to sequences in the natural way: $[(\sigma_n)_{n\in I}]_\mathcal{A}$ is the (possibly infinite) product $\prod_{n\in I} \xi_n \in \overline{\Xi_\mathcal{A}}$.

We end this section by showing that for any infinite word α and substitution σ, whether $\mathcal{A}$ accepts $\sigma(\alpha)$ can be determined from $[\sigma]_\mathcal{A}$. Recall that for a morphism $h\colon \Sigma^* \to M$, where M is a finite monoid, $\overline{h}$ denotes the extension of h to Σ^ω.

Lemma 26. *Let $\mathcal{A}$ be an automaton over alphabet Σ and $\alpha \in \Sigma^\omega$. There exists $\Phi \subseteq \Xi_\mathcal{A}$ such that for any $\sigma\colon \Sigma^* \to \Sigma^*$, $\mathcal{A}$ accepts $\sigma(\alpha)$ if and only if $[\sigma]_\mathcal{A} \in \Phi$. Furthermore, Φ can be effectively computed if we know $\overline{h}(\alpha)$ for all $h \in \mathsf{morphisms}_\mathcal{A}$.*

Proof. By Lem. 7, we can compute a set $F \subseteq \overline{M_\mathcal{A}}$ such that, for all α, $\mathcal{A}$ accepts α if and only if $\overline{h_\mathcal{A}}(\alpha) \in F$. Applying Lem. 5,

$$\overline{h_\mathcal{A}}(\sigma(\alpha)) = \overline{(h_\mathcal{A} \circ \sigma)}(\alpha) = \overline{\mathsf{compose}_{[\sigma]_\mathcal{A}}(h_\mathcal{A})}(\alpha).$$

We thus define $\Phi = \{\xi \in \Xi \colon \overline{\mathsf{compose}_\xi(h_\mathcal{A})}(\alpha) \in F\}$. The effectiveness claim follows from the fact that $\mathsf{compose}_\xi(h_\mathcal{A}) \in \mathsf{morphisms}_\mathcal{A}$ for all ξ. □

VI. Morphic words accepted by $\mathcal{A}$

In this section, we shall use the algebraic machinery developed in Sec. V to characterise the set of morphic words accepted by $\mathcal{A}$, which generalises the main result of [7]. A word $\alpha \in \Sigma^\omega$ is *substitutive* if it is a fixed point of a non-trivial substitution, i.e., if there exists $\sigma \in S(\Sigma)$ not the identity such that $\sigma(\alpha) = \alpha$. A word $\alpha \in \Sigma^\omega$ is *morphic* if it is of the form $\pi(\alpha)$ for $\pi \in S(\Sigma)$ and α a substitutive word. For $u \in \Sigma^*$ and substitutions σ, π, define $\sigma^\omega(u) = \lim_{n\to\infty} \sigma^n(u)$ and $\pi \circ \sigma^\omega(u) = \pi(\sigma^\omega(u))$. We say that $\alpha \in \Sigma^\omega$ is *constructibly morphic* if $\alpha = \pi \circ \sigma^\omega(u)$ for substitutions σ, π and $u \in \Sigma^*$. For such words, given an automaton $\mathcal{A}$, we will show that the respective equivalence classes $\xi, \zeta \in \Xi_\mathcal{A}$ of substitutions σ, π determine:

- Whether σ has a fixed point accepted by $\mathcal{A}$ (Thm. 31).
- A regular language $\mathcal{L}(\xi, \zeta) \subseteq \Sigma^+$ such that $\pi \circ \sigma^\omega(u)$ is accepted by $\mathcal{A}$ if and only if $u \in \mathcal{L}(\xi, \zeta)$ (Thm. 30).

We have the following properties, whose proofs are in App. E.

Lemma 27. *Let Σ be a finite alphabet and σ be a substitution. The following properties hold.*

Saturation *For $u \in \Sigma^*$, we have that $\sigma^\omega(u) = \sigma^\omega(\sigma(u))$.*
Distributivity *For $u, v \in \Sigma^*$, $\sigma^\omega(uv) = \sigma^\omega(u) \cdot \sigma^\omega(v)$.*
Left-expansion *For $a \in \Sigma, u \in \Sigma^+, v \in \Sigma^*$, if $\sigma^\omega(a) = u \cdot \sigma^\omega(a) \cdot \sigma^\omega(v)$, then $\sigma^\omega(a) = u^\omega$.*
Right-expansion *For $a \in \Sigma, u \in \Sigma^*$, if $\sigma(a) = au$, then $\sigma^\omega(a) = a \cdot u \cdot \sigma(u) \cdots \sigma^n(u) \cdots$.*
Cycle of Contradiction *Let $a_0, \ldots, a_{p-1}$, with $p > 1$ be distinct letters such that $\mathsf{head}_{\sigma^p}(a_0) = a_0$, and for any $r \in \{1, \ldots, p-1\}$, $\mathsf{head}_{\sigma^r}(a_0) = a_r$. We have that $\sigma^\omega(a_0) = \cdots = \sigma^\omega(a_{p-1}) = \bot$.*
Terminal Letters *If, for a letter a, $\sigma^\omega(a) = u \in \Sigma^+$, then for all $n \geq |\Sigma|$, $\sigma^n(a) = u$.*

The following result, along with distributivity, implies that whether $\sigma^\omega(u)$ for a finite word u is an infinite word accepted by $\mathcal{A}$ is determined by the equivalence class $[\sigma]_\mathcal{A}$; this is the main technical novelty of this section.

Theorem 28. *Let $\mathcal{A}$ be an automaton. For any $h \in \mathsf{morphisms}_\mathcal{A}$, substitutions σ, τ with $\sigma \equiv_\mathcal{A} \tau$, and letter a, we have that $\overline{h} \circ \sigma^\omega(a) = \overline{h} \circ \tau^\omega(a)$. Moreover, this element in $\overline{M_\mathcal{A}}$ can be computed given only the equivalence class $\xi \in \Xi_\mathcal{A}$ of σ, τ along with the values of h, a.*

Proof sketch. By Lem. 27, we have a dynamic programming algorithm to compute $\overline{h} \circ \sigma^\omega(a)$ in a "depth first" manner.

We defer the technical details to App. E. The key insight that structures this approach is that we can write

$$\begin{aligned}\overline{h}\circ\sigma^{\omega}(b) &= \overline{h}\circ\sigma^{\omega}(c_1)\cdot\overline{h}\circ\sigma^{\omega}(v_1)\cdots\overline{h}\circ\sigma^{\omega}(c_k)\cdot\overline{h}\circ\sigma^{\omega}(v_k)\\ &= \overline{h}\circ\sigma^{\omega}(c_1)\cdot g(v_1)\cdots\overline{h}\circ\sigma^{\omega}(c_k)\cdot g(v_k),\end{aligned}$$

where $g(v_i) = h\circ\sigma^{|\Sigma|}(v_i)$, and can be obtained from $\mathsf{segments}_{\xi}(b, g) = \mathsf{segments}_{\xi}(b, \mathsf{compose}_{\xi^{|\Sigma|}}(h))$. ☐

As a corollary, we obtain that whether a substitution π, when applied to the word obtained by iterating σ infinitely on u, produces a word accepted by $\mathcal{A}$, is also determined by the equivalence classes $[\sigma]_{\mathcal{A}}$ and $[\pi]_{\mathcal{A}}$: indeed, acceptance only depends on $\overline{h}\circ\pi\circ\sigma^{\omega}(u)$ where $h = h_{\mathcal{A}}$, which by Lem. 26 is equivalent to $\overline{h\circ\pi}\circ\sigma^{\omega}(u)$. The latter, in turn, is the same as $\overline{\mathsf{compose}_{[\pi]_{\mathcal{A}}}(h)}$ applied to $\sigma^{\omega}(u)$.

Corollary 29. *Let $\mathcal{A}$ be an automaton. For any letter a and $h\in\mathsf{morphisms}_{\mathcal{A}}$ we can compute $\overline{h}\circ\pi\circ\sigma^{\omega}(a)$ given only $h, a, [\sigma]_{\mathcal{A}}, [\pi]_{\mathcal{A}}$.*

We thus arrive at the following.

Theorem 30. *Let $\mathcal{A}$ be an automaton over Σ and σ,π be substitutions with respective equivalence classes $\xi,\zeta\in\Xi_{\mathcal{A}}$. We can compute a regular language $\mathcal{L}(\xi,\zeta)\subseteq\Sigma^{+}$ such that $\mathcal{A}$ accepts $\pi\circ\sigma^{\omega}(u)$ if and only if $u\in\mathcal{L}(\xi,\zeta)$.*

Proof. We construct a deterministic finite-word automaton recognising $\mathcal{L}(\xi,\zeta)$. The set of states are the elements of $M_{\mathcal{A}}$, the initial state is $1_{M_{\mathcal{A}}}$, the set of accepting states is F (as in Lem. 7), and the transition function maps (m, a) to $m\cdot(\overline{h_{\mathcal{A}}}\circ\pi\circ\sigma^{\omega})(a)$. The latter is effective by Cor. 29. ☐

Finally, the following result implies that whether σ has a fixed point accepted by $\mathcal{A}$ is effectively determined by $[\sigma]_{\mathcal{A}}$.

Theorem 31. *Let $\mathcal{A}$ be an automaton and σ be a substitution. Given $[\sigma]_{\mathcal{A}}$ and $h\in\mathsf{morphisms}_{\mathcal{A}}$, we can compute the set $\{\overline{h}(\alpha)\colon\sigma(\alpha)=\alpha\}\subseteq\overline{M_{\mathcal{A}}}$.*

Proof. We observe that any fixed point of σ must be a concatenation of words $\sigma^{\omega}(a)$, where $\sigma(a) = au$ for $u\in\Sigma^*$. Write A for the set of all such letters a, which can be extrracted from $\mathsf{introduces}_{\sigma}$. The required set is then the submonoid of $\overline{M_{\mathcal{A}}}$ generated by $\{\overline{h}\circ\sigma^{\omega}(a)\colon a\in A\}$, whose elements in turn can be computed using Thm. 28. ☐

VII. The automaton acceptance problem

In this section we present our main results: solutions to the automaton acceptance problem for S-adic words and shifts. Thm. 32 addresses generated words, Thm. 34 addresses directed words, and finally Thm. 37 addresses generated shifts.

We make the following remark. Assuming that the set $S\subseteq S(\Sigma)$ is such that for every automaton $\mathcal{A}$ we can compute the finite set $\Xi_{\mathcal{A},S} = \{[\sigma]_{\mathcal{A}}\colon\sigma\in S\}$, we can characterise the set of all α accepted by $\mathcal{A}$ that have a congenial S-adic expansion: in this case we restrict the automata we construct to run over the alphabet $\Xi_{\mathcal{A},S}\times\Sigma$. Note that in particular, finite S given explicitly satisfy the above effectiveness requirement.

Already, these results can directly be applied to Sturmian words (Sec. III-A), Arnoux-Rauzy words (Sec. III-B), and ternary dendric shifts (Sec. III-C); we shall obtain case-specific refinements in Sec. VIII. For an automaton $\mathcal{A}$, recall that $M_{\mathcal{A}}$ is the journey monoid of $\mathcal{A}$, $S(\Sigma)$ denotes the set of all non-erasing substitutions of type $\Sigma^*\to\Sigma^*$, and $\Xi_{\mathcal{A}}$ denotes the equivalence classes on $S(\Sigma)$ constructed from $\mathcal{A}$.

A. Generated and directed words

We will prove that for congenial $\hat{s}$ over $S(\Sigma)\times\Sigma$, defining a word accepted by $\mathcal{A}$ is a property of $\mathsf{trace}_{\mathcal{A}}(s)$, and the set of all such traces is ω-regular.

Theorem 32 (Main Result for Generated Words)**.** *Given an automaton $\mathcal{A}$ over Σ, we can compute an automaton $\mathcal{B}$ over $\Xi_{\mathcal{A}}\times\Sigma$ such that for all infinite sequences $\hat{s}$ over $S(\Sigma)\times\Sigma$, $\mathcal{B}$ accepts $\mathsf{trace}_{\mathcal{A}}(\hat{s})$ if and only if $\hat{s}$ is congenial and generates a word accepted by $\mathcal{A}$.*

To prove Thm. 32, let $F\subseteq\overline{M_{\mathcal{A}}}$ be as in Lem. 7. The automaton $\mathcal{B}$ accepts $t\in(\Xi_{\mathcal{A}}\times\Sigma)^{\omega}$ if and only if t is accepted by $\mathcal{B}_x$ (as constructed by Thm. 33) for some $x\in F$.

Theorem 33. *Let $\mathcal{A}$ be an automaton over an alphabet Σ and $x\in\overline{M_{\mathcal{A}}}$. We can construct an automaton $\mathcal{B}_x$ over $\Xi_{\mathcal{A}}\times\Sigma$ such that for all infinite sequences $\hat{s}$ over $S(\Sigma)\times\Sigma$, $\mathcal{B}_x$ accepts $\mathsf{trace}_{\mathcal{A}}(\hat{s})$ if and only if $\hat{s}$ is congenial and $\overline{h_{\mathcal{A}}}(\alpha_{\hat{s}}) = x$, where $\alpha_{\hat{s}}$ is the word generated by $\hat{s}$.*

Proof. Recall that a sequence $s = (\sigma_n, a_n)_{n\in\mathbb{N}}$ over $\Sigma\times\Xi_{\mathcal{A}}$ is congenial if and only if $\mathsf{head}_{\xi_n}(a_n) = a_{n-1}$ for all $n\geq 1$, where $\xi_n = [\sigma_n]_{\mathcal{A}}$. Observe that this property only depends on $\mathsf{trace}_{\mathcal{A}}(s)$. The automaton $\mathcal{B}_x$, first and foremost, checks this condition for all $n\geq 1$, and permanently transitions into a rejecting state if it observes violating ξ_n, a_n, a_{n-1}.

Now suppose $s = (\sigma_n, a_n)_{n\in\mathbb{N}}$ is congenial, and define ξ_n as above. As shown in Lem. 18,

$$\sigma_0\cdots\sigma_n(a_n) = \sigma_0(a_0)\cdot\sigma_0(v_1)\cdot\sigma_0\sigma_1(v_2)\cdots(\sigma_0\cdots\sigma_{n-1}(v_n))$$

for all n, where v_n satisfies $\sigma_n(a_n) = a_{n-1}v_n$. Let $u_0 = \sigma_0(a_0)$ and $u_n = \sigma_0\cdots\sigma_{n-1}(v_n)$ for $n\geq 1$. By the properties of infinite products in $\overline{M_{\mathcal{A}}}$ (see Sec. II-B) we have that $\overline{h_{\mathcal{A}}}(\alpha_s) = x$ if and only if $\prod_{n=0}^{\infty} h_{\mathcal{A}}(u_n) = x$. The automaton $\mathcal{B}_x$ simulates the run of the automaton $\mathcal{A}_x$ of Lem. 8 on the sequence $(h_{\mathcal{A}}(u_n))_{n\in\mathbb{N}}$. It remains to show how the automaton keeps track of $h_{\mathcal{A}}(u_n)$ as it reads $(\xi_n, a_n)_{n\in\mathbb{N}}$. We have that

$$\begin{aligned}h_{\mathcal{A}}(u_0) &= (\mathsf{compose}_{\xi_0}(h_{\mathcal{A}}))(a_0),\\ h_{\mathcal{A}}(u_n) &= \mathsf{tail}_{\xi_n}(a_n, \mathsf{compose}_{\xi_0\cdots\xi_{n-1}}(h_{\mathcal{A}}))\end{aligned}$$

for $n\geq 1$. The state of the automaton $\mathcal{B}_x$ keeps track of one piece of information $\xi\in\Xi_{\mathcal{A}}$, in addition to the state required for simulating a run of $\mathcal{A}_x$ on $(h_{\mathcal{A}}(u_n))_{n\in\mathbb{N}}$. For all n, before reading (ξ_n, a_n) the value of ξ is $\xi_0\cdots\xi_{n-1}$, where the empty product (corresponding to the initial value of ξ) is the identity element of $\Xi_{\mathcal{A}}$. Upon reading (ξ_n, a_n), the automaton $\mathcal{B}_x$ first computes $h_{\mathcal{A}}(u_n)$, in which the value of ξ is used, then feeds the computed value to $\mathcal{A}_x$, and finally updates ξ to $\xi\cdot\xi_n$.

Finally, $\mathcal{B}_x$ accepts $(\sigma_n, a_n)_{n\in\mathbb{N}}$ if and only if $\mathcal{A}_x$ accepts $(h_{\mathcal{A}}(u_n))_{n\in\mathbb{N}}$. □

We arrive at the representative result from the abstract.

Theorem 34 (Main Result for Directed Words)**.** *Given an automaton $\mathcal{A}$ over Σ, we can construct an automaton $\mathcal{B}$ over $\Xi_{\mathcal{A}}$ such that for all $s \in S(\Sigma)^\omega$, $\mathcal{B}$ accepts $\mathsf{trace}_{\mathcal{A}}(s)$ if and only if $\mathcal{A}$ accepts some $\alpha \in \Sigma^\omega$ that is directed by s.*

Proof. We use $X_s \subset \overline{M_{\mathcal{A}}}$ to denote the image of $\mathsf{congenials}_s$ under $\overline{h_{\mathcal{A}}}$. Recall (Lem. 20) that directed words are obtained by concatenating congenial words. To prove Thm. 34, we need to recognise the set of trace_s for which the submonoid generated by X_s intersects the accepting set $F \subseteq \overline{M_{\mathcal{A}}}$ (from Lem. 7). We can precompute a set $\mathcal{X}$ of sets X that generate submonoids intersecting F. Our automaton $\mathcal{B}$ needs to check that X_s contains at least one such set X. We denote by $\mathcal{C}_x$ the projection of $\mathcal{B}_x$ from Thm. 33 to $\Xi_{\mathcal{A}}$, and observe: $s \in \mathcal{L}(\mathcal{B})$ if and only if $\bigvee_{X\in\mathcal{X}} \bigwedge_{x\in X} x \in \mathcal{L}(\mathcal{C}_x)$. □

Recall that for everywhere growing directive sequences, the notions of direction and congenial generation coincide (Lem. 21). We can therefore give a characterisation, similarly to Thm. 34, of all α accepted by $\mathcal{A}$ that are *generated* by some everywhere growing $s \in S(\Sigma)$. We only additionally need the following lemma, proven in Sec. F.

Lemma 35. *Let $\mathcal{A}$ be an automaton. We can construct an automaton $\mathcal{B}_{\mathsf{eg}}$ over $\Xi_{\mathcal{A}}$ such that for every $s \in S(\Sigma)^\omega$, $\mathcal{B}$ accepts $\mathsf{trace}_{\mathcal{A}}(s)$ if and only if s is everywhere growing.*

B. Generated shifts

We now show how to compute an effective representation of the set of all weakly primitive directive sequences s that generate a shift X_s intersecting $\mathcal{L}(\mathcal{A})$.

Theorem 36. *Let $\mathcal{A}$ be an automaton. The shift X_s generated by a weakly primitive directive sequence $s = (\sigma_n)_{n\in\mathbb{N}}$ intersects $\mathcal{L}(\mathcal{A})$ if and only if s directs (equivalently, congenially generates) a word $\alpha \in \Sigma^* \cdot \mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{A}_{\mathsf{suf}})$.*

Proof. Since s is weakly primitive, by Lem. 15 that it is everywhere growing, and hence by Lem. 22 the notions of being directed and congenially generated coincide. Recall that by definition, any word α generated by s is in X_s, and moreover by Lem. 15, we have that

$$X_s = \{\beta \in \Sigma^\omega : \beta \text{ is uniformly recurrent and } \mathcal{L}(\beta) = \mathcal{L}(s)\}.$$

Applying Lem. 13 completes the proof. □

Theorem 37 (Main Result for Generated Shifts)**.** *Let $\mathcal{A}$ be an automaton over an alphabet Σ, and let $\mathcal{A}_{\mathsf{suf}}$ be an automaton that recognises the language $\Sigma^*\mathcal{L}(\mathcal{A})$. We can compute an automaton $\mathcal{B}$ over $\Xi_{\mathcal{A}_{\mathsf{suf}}}$ such that for all infinite sequences s over $S(\Sigma)$, $\mathcal{B}$ accepts $\mathsf{trace}_{\mathcal{A}}(s)$ if and only if s is weakly primitive and generates a shift X_s that intersects $\mathcal{L}(\mathcal{A})$.*

Proof. Recall Thm. 36: given s is weakly primitive (which we use Lem. 38 to check), it suffices to check that s directs a word that is accepted by $\mathcal{A}_{\mathsf{suf}}$ (whose language is $\Sigma^*\mathcal{L}(\mathcal{A})$). It remains to apply Thm. 34 to $\mathcal{A}_{\mathsf{suf}}$. □

Lemma 38. *Let $\mathcal{A}$ be an automaton over Σ. We can construct an automaton $\mathcal{B}_{\mathsf{wp}}$ over $\Xi_{\mathcal{A}}$ such that for every $s \in S(\Sigma)^\omega$, $\mathcal{B}_{\mathsf{wp}}$ accepts $\mathsf{trace}_{\mathcal{A}}(s)$ if and only if s is weakly primitive.*

Proof. We will construct an automaton that recognises exactly the traces of sequences that are not weakly primitive. On input $(\xi_n)_{n\in\mathbb{N}} \in \Xi_{\mathcal{A}}^\omega$, it simply guesses n and checks whether for all $\mu_n = \xi_m \cdots \xi_n$, $\mathsf{letters}_{\mu_n}(a) \neq \Sigma$ for some $a \in \Sigma$. □

Note that whether s is weakly primitive or not does not depend on $\mathcal{A}$; our formulation merely has the advantage of being convenient.

VIII. Partial Quotients

In this section, we refine our main results for the classes presented in Sec. III through a synergy of word combinatorics, algebra, and topology. We shall first demonstrate our key ideas for Sturmian (Sec. III-A) and Arnoux-Rauzy words (Sec. III-B), and subsequently discuss how the techniques can be generalised to handle the case of ternary dendric shifts (Sec. III-C).

A. Partial quotients for Sturmian and Arnoux-Rauzy words

We work with a class $\mathcal{W}$ of words over Σ and a set $S \subseteq S(\Sigma)$ such that

(A) $\alpha \in \mathcal{W}$ if and only if α is directed by some weakly primitive s over S^ω, and
(B) there exists effectively computable function p such that the factor complexity $p_\alpha(n) = p(n)$ for all $\alpha \in \mathcal{W}$ and $n \in \mathbb{N}$.

Characteristic Sturmian, Sturmian, and d-letter Arnoux-Rauzy words are captured, respectively, with $S = \{\lambda_0, \rho_0\}$, $p(n) = n + 1$; $S = \{\lambda_0, \rho_0, \lambda_1, \rho_1\}$, $p(n) = n + 1$; and $S = \{\lambda_0, \ldots, \lambda_{d-1}, \rho_0, \ldots, \rho_{d-1}\}$, $p(n) = n(d-1) + 1$.

The main idea is that every uniformly recurrent $\alpha \in \mathcal{W}$ with factor-complexity function p has a prefix u that is p-*saturated* with respect to a given automaton $\mathcal{A}$: any uniformly recurrent extension β of u with factor-complexity function p agrees with α upon acceptance by $\mathcal{A}$ (Lem. 39). The proof involves a careful consideration of Semënov's algorithm for deciding whether an automaton accepts a uniformly recurrent word (App. F).

Lemma 39. *Let $\mathcal{A}$ be an automaton and α be a uniformly recurrent word such that $\alpha(n)$ and $p_\alpha(n)$ can be effectively computed for all n. We can compute N such that any uniformly recurrent β with $\beta[0, N) = \alpha[0, N)$ and $p_\beta(n) = p_\alpha(n)$ for all n is accepted by $\mathcal{A}$ if and only if α is accepted by $\mathcal{A}$.*

Recall that a directive sequence $(\sigma_n)_{n\in\mathbb{N}}$ is weakly primitive if for every n, there exists r such that $\sigma_n \cdots \sigma_{n+r}$ is positive: for every letter a, the image $\sigma_n \cdots \sigma_{n+r}(a)$ contains all letters. Therefore, given a congenial expansion $(\sigma_n, a_n)_{n\in\mathbb{N}}$ of α, we can compute an increasing sequence $(k_n)_{n\in\mathbb{N}}$ with $k_0 = 0$ such that, writing $l_n = k_{n+1} - 1$, $\tau_n = \sigma_{k_n} \cdots \sigma_{l_n}$ and

$b_n = a_{l_n}$, each τ_n is positive, all strict prefixes of the composition $\sigma_{k_n} \cdots \sigma_{l_n}$ are not positive, and $(\tau_n, b_n)_{n\in\mathbb{N}}$ is also a congenial expansion of α. We refer to $(\tau_n, b_n)_{n\in\mathbb{N}}$ as the sequence of *partial quotients* of $(\sigma_n, a_n)_{n\in\mathbb{N}}$. By construction, the sequence of partial quotients is weakly primitive. The main result of this section is the following.

Theorem 40. *Let $\mathcal{W}, S$ be as above. Given an automaton $\mathcal{A}$, we can compute N with the following property. Let $\alpha, \alpha' \in \mathcal{W}$ with congenial S-adic expansions $(\tau_n, a_n)_{n\in\mathbb{N}}, (\tau'_n, a'_n)_{n\in\mathbb{N}}$ and partial quotients $(\pi_n, b_n)_{n\in\mathbb{N}}$ and $(\pi'_n, b'_n)_{n\in\mathbb{N}}$, respectively. If $\pi_n \equiv_{\mathcal{A}} \pi'_n$ and $b_n = b'_n$ for all $n \le N$ then $\mathcal{A}$ accepts α if and only if it accepts α'.*

Proof. Let P be the set of all positive $\sigma_1 \cdots \sigma_r$ such that $\sigma_i \in S$ for all i and $\sigma_1 \cdots \sigma_i$ is not positive for all $i < r$. Construct finite $\Pi \subseteq P$ such that $\{[\pi]_{\mathcal{A}} : \pi \in P\} = \{[\pi]_{\mathcal{A}} : \pi \in \Pi\}$, and let Ω be the set of all congenial $(\pi_n, b_n)_{n\in\mathbb{N}}$ over $\Pi \times \Sigma$. Because congeniality is a local condition, Ω is a compact subset of $(\Pi \times \Sigma)^\omega$. Next, consider α generated by some $\hat{s} = (\pi_n, b_n)_{n\in\mathbb{N}} \in \Omega$. By Lem. 39 there exists M such that $\alpha[0, M)$ is p-saturated. Write Ω_M for the set of all $\hat{s} \in \Omega$ whose first M terms generate a p-saturated finite word, observing that each Ω_M is open. From Lem. 39 it follows that $\{\Omega_M : M \in \mathbb{N}\}$ is an open cover of Ω., which, by compactness, admits a finite sub-cover. That is, there exists N (which we can effectively computed by enumeration) such that for every $(\pi_n, b_n)_{n\in\mathbb{N}} \in \Omega$, $(\pi_n, b_n)_{n=0}^N$ generates a finite word that is p-saturated.

Now consider $\alpha \in \mathcal{W}$ with a weakly primitive and congenial S-adic expansion $(\mu_n, a_n)_{n\in\mathbb{N}}$ and partial quotients sequence $\hat{s} := (\tau_n, b_n)_{n\in\mathbb{N}}$. Let $(\pi_n)_{n\in\mathbb{N}}$ over Π be such that $\tau_n \equiv_{\mathcal{A}} \pi_n$ for all n, and observe that $\hat{s} := (\pi_n, b_n)_{n\in\mathbb{N}}$ is also congenial. Let β be the word generated by $\hat{s}$. By Thm. 33 and Lem. 7, $\mathcal{A}$ accepts α if and only if it accepts β. In particular, acceptance by $\mathcal{A}$ only depends on the trace $([\tau_n]_{\mathcal{A}}, b_n)_{n\in\mathbb{N}}$. By the earlier argument, whether $\mathcal{A}$ accepts β only depends on $([\pi_n]_{\mathcal{A}}, b_n)_{n\in\mathbb{N}}$. It follows that whether $\mathcal{A}$ accepts α only depends on $([\tau_n]_{\mathcal{A}}, b_n)_{n=0}^N$. ☐

We illustrate this result on Sturmian words. Given an automaton $\mathcal{A}$, apply Thm. 40 with the class of Sturmian words, $S = \{\lambda_0, \lambda_1, \rho_0, \rho_1\}$, and $p(n) = n + 1$ to compute N. Let $\eta \in (0, 1) \setminus \mathbb{Q}$, $\chi \in [-\eta, 1 - \eta]$, $[0; a_1 + 1, a_2, \ldots]$ be the continued fraction expansion of η, $(b_n)_{n\in\mathbb{N}}$ be an Ostrowski expansion of χ in base η, and α be the corresponding Sturmian word with slope η and intercept χ. Recall that α is directed (and, thanks to weak primitivity and left-properness, congenially generated; see Lem. 23) by the sequence

$$(\sigma_n)_{n\in\mathbb{N}} = \langle \lambda_0^{b_1}, \rho_0^{c_1}, \lambda_1^{b_2}, \rho_1^{c_2}, \lambda_0^{b_3}, \rho_0^{c_3}, \lambda_1^{b_4}, \rho_1^{c_4} \ldots \rangle$$

where $c_i = a_i - b_i$. By the rules of Ostrowski expansion, at least one of c_n, c_{n+1} is non-zero for all n. Moreover, every composition of morphisms from $\{\lambda_0, \lambda_1, \rho_0, \rho_1\}$ that includes two morphisms with differing indices is positive. Since at least one of b_n, c_n is non-zero for all $n > 1$ (we could, however, have $a_1 = b_1 = c_1 = 0$), we have that for every n, $\sigma_n \cdots \sigma_{n+5}$ is positive. Applying Thm. 40 we conclude that whether α is accepted by $\mathcal{A}$ can be determined by looking at the first $6N$ digits of the expansions of η and χ, since these are guaranteed to generate at least the first N partial quotients of the unique S-adic expansion of α. In fact, because $([\sigma^n]_{\mathcal{A}})_{n\in\mathbb{N}}$ is ultimately periodic, the acceptance of α by $\mathcal{A}$ can be formulated as a Boolean combination of formulas in modular arithmetic involving the first N digits of the expansions of η and χ.

B. Partial quotients for ternary dendric shifts

Recall from Sec. III-C that ternary dendric shifts have factor complexity $p(n) = 2n + 1$, and that we can compute a set S (which is infinite but has a finite representation) of left-proper substitutions, an effective finite partition [26, Figs. 5, 6] T_{den} of S, and a closed language $\mathcal{L}(\mathsf{den}) \subset T_{\mathsf{den}}^\omega$ (recognised by an automaton constructed from [26, Fig. 7]), such that a shift $X \subset \{0, 1, 2\}^\omega$ is minimal dendric if and only if it is generated by a weakly primitive $s \in S^\omega$ whose trace [10] $t \in \mathcal{L}(\mathsf{den})$.

We can generalise the techniques of Sec. VIII-A to the setting of a class $\mathcal{X}$ of shifts, a set $S \subseteq S(\Sigma)$ of left-proper substitutions that is finitely and effectively partitioned into $T_{\mathcal{X}}$, and a closed ω-regular language $\mathcal{L}(\mathcal{X}) \subseteq T_{\mathcal{X}}^\omega$ such that:

(A) A shift $X \in \mathcal{X}$ if and only if it is generated by a weakly primitive sequence s over S whose trace $t \in \mathcal{L}(\mathcal{X})$.
(B) There exists effectively computable function p such that for all $X \in \mathcal{X}$, the factor complexity function $p_X(n) = p(n)$ for all n.

We seek to characterise the shifts $X \in \mathcal{X}$ which intersect the language of a given automaton $\mathcal{A}'$. These shifts consist of uniformly recurrent words. Hence by Thm. 36 we have to determine whether the directive sequence generates a word accepted by $\mathcal{A} = \mathcal{A}'_{\mathsf{suf}}$. The directive sequences that we work with are left-proper and weakly primitive, and hence generate a unique congenial word. By the assumptions on $\mathcal{X}$, a finite prefix of a valid directive sequence s will generate a prefix[11] of $\alpha \in X_s$ that is p-saturated with respect to $\mathcal{A}$ (Lem. 39). We next sketch that we again need only $N(\mathcal{A})$ partial quotients.

In this setting, we additionally rely upon the morphism $h_{\mathcal{X}}$ from $T_{\mathcal{X}}$ into the monoid $\Xi_{\mathcal{X}}$ recognising the language $\mathcal{L}(\mathcal{X})$ to represent our partial quotients. Given an automaton $\mathcal{A} = \mathcal{A}'_{\mathsf{suf}}$, a positive subsequence $\sigma_n \cdots \sigma_{n+r}$ is said to be $(\mathcal{X}, \mathcal{A})$*-represented* by π if: (i) $[\pi]_{\mathcal{A}} = [\sigma_n \cdots \sigma_{n+r}]_{\mathcal{A}}$; (ii) $\pi = \tau_1 \cdots \tau_k$, where $\tau_1, \ldots, \tau_k \in S$; (iii) the trace of $\tau_1 \cdots \tau_k \in T_{\mathcal{X}}^+$ and the trace of $\sigma_n \cdots \sigma_{n+r} \in T_{\mathcal{X}}^+$ agree upon $h_{\mathcal{X}}$. It is clear that we can compute a finite set $\Pi_{\mathcal{X},\mathcal{A}} \subset S(\Sigma)$ of representatives.

By construction, any directive sequence over $\Pi_{\mathcal{X},\mathcal{A}}$ is weakly primitive, and the set Ω of $\mathcal{X}$-valid sequences is closed in $\Pi_{\mathcal{X},\mathcal{A}}^\omega$, thus forming a compact space. Furthermore, a weakly primitive directive sequence over S can be replaced by its sequence of representatives, which results in a directive sequence over $\Pi_{\mathcal{X},\mathcal{A}}$. By construction, the shift generated by

[10] The letter $t(n)$ records the set in the partition to which σ_n belongs.

[11] Take $\sigma_0 \cdots \sigma_{n+1}$ to generate $\sigma_0 \cdots \sigma_n(a_{n+1})$, where each image of σ_{n+1} starts with a_{n+1}.

the new directive sequence agrees with the original upon membership in $\mathcal{X}$ and containment in $\mathcal{L}(\mathcal{A})$ (due to equivalence modulo both $\Xi_{\mathcal{X}}$ and $\Xi_{\mathcal{A}}$). It suffices to look at only the first $N(\mathcal{A})$ terms of this new sequence because the compactness arguments in the proof of Thm. 40 apply *mutatis mutandis*.

IX. Discussion

For an automaton $\mathcal{A}$ over Σ, we gave a non-trivial equivalence relation $\equiv_{\mathcal{A}}$ over substitutions of type $\Sigma^* \to \Sigma^*$ such that acceptance (by $\mathcal{A}$) properties of $\alpha \in \Sigma^\omega$ associated with a substitution σ or directive sequence $(\sigma_n)_{n\in\mathbb{N}}$ can be inferred from $[\sigma]_{\mathcal{A}}$ and $([\sigma_n]_{\mathcal{A}})_{n\in\mathbb{N}}$, respectively. We also completely characterised the set of S-directed words accepted by $\mathcal{A}$. The next step is to consider *billiard words*, which play a key role in decidability of the MSO theory of $\langle \mathbb{N}; <, a^{\mathbb{N}}, b^{\mathbb{N}}, c^{\mathbb{N}} \rangle$ and are not known to have good S-adic representations [9].

References

[1] J. R. Büchi, *On a Decision Method in Restricted Second Order Arithmetic*. New York, NY: Springer New York, 1990, pp. 425–435.

[2] C. C. Elgot and M. O. Rabin, "Decidability and undecidability of extensions of second (first) order theory of (generalized) successor," *The Journal of Symbolic Logic*, vol. 31, no. 2, pp. 169–181, 1966.

[3] R. Robinson, "Restricted set-theoretical definitions in arithmetic," *Proceedings of the American Mathematical Society*, vol. 9, no. 2, pp. 238–242, 1958.

[4] B. Trahtenbrot, "Finite automata and the logic of one-place predicates," *Siberian Mathematical Journal*, vol. 3, pp. 103–131, 1962.

[5] W. Thomas, "A note on undecidable extensions of monadic second order successor arithmetic," *Archiv für mathematische Logik und Grundlagenforschung*, vol. 17, no. 1, pp. 43–44, 1975.

[6] A. Semënov, "Logical theories of one-place functions on the set of natural numbers," *Mathematics of the USSR-Izvestiya*, vol. 22, no. 3, p. 587, 1984.

[7] O. Carton and W. Thomas, "The monadic theory of morphic infinite words and generalizations," *Information and Computation*, vol. 176, no. 1, pp. 51–65, 2002.

[8] A. Rabinovich, "On decidability of monadic logic of order over the naturals extended by monadic predicates," *Information and Computation*, vol. 205, no. 6, pp. 870–889, 2007.

[9] V. Berthé, T. Karimov, J. Nieuwveld, J. Ouaknine, M. Vahanwala, and J. Worrell, "The monadic theory of toric words," *Theoretical Computer Science*, vol. 1025, p. 114959, 2025.

[10] T. Karimov, E. Kelmendi, J. Ouaknine, and J. Worrell, "What's decidable about discrete linear dynamical systems?" in *Principles of Systems Design: Essays Dedicated to Thomas A. Henzinger on the Occasion of His 60th Birthday*. Springer, 2022, pp. 21–38.

[11] T. Colcombet, "Green's relations and their use in automata theory," in *Language and Automata Theory and Applications: 5th International Conference, LATA 2011, Tarragona, Spain, May 26-31, 2011. Proceedings 5*. Springer, 2011, pp. 1–21.

[12] A. Rabinovich and W. Thomas, "Decidable theories of the ordering of natural numbers with unary predicates," in *Computer Science Logic*, Z. Ésik, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 562–574.

[13] D. Kuske, J. Liu, and A. Moskvina, "Infinite and bi-infinite words with decidable monadic theories," in *24th EACSL Annual Conference on Computer Science Logic (CSL 2015)*, S. Kreutzer, Ed., vol. 41. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2015, pp. 472–486.

[14] V. Berthé, T. Karimov, J. Nieuwveld, J. Ouaknine, M. Vahanwala, and J. Worrell, "On the decidability of monadic second-order logic with arithmetic predicates." New York, NY, USA: Association for Computing Machinery, 2024.

[15] J.-P. Allouche and J. Shallit, *Automatic Sequences: Theory, Applications, Generalizations*. Cambridge University Press, 2003.

[16] N. P. Fogg, V. Berthé, S. Ferenczi, C. Mauduit, and A. Siegel, *Substitutions in dynamics, arithmetics and combinatorics*. Springer, 2002.

[17] P. Hieronymi, D. Ma, R. Oei, L. Schaeffer, C. Schulz, and J. Shallit, "Decidability for Sturmian words," *Logical Methods in Computer Science*, vol. 20, 2024.

[18] J. Shallit, *The logical approach to automatic sequences: Exploring combinatorics on words with Walnut*. Cambridge University Press, 2022, vol. 482.

[19] P. Béaur and B. Hellouin de Menibus, "Sturmian and infinitely desubstitutable words accepted by an ω-automaton," in *International Conference on Combinatorics on Words*. Springer, 2023, pp. 104–116.

[20] D. Perrin and J.-É. Pin, "Semigroups and automata on infinite words," *NATO ASI Series C Mathematical and Physical Sciences-Advanced Study Institute*, vol. 466, pp. 49–72, 1995.

[21] T. Wilke, "An Eilenberg theorem for ω-languages," in *International Colloquium on Automata, Languages, and Programming*. Springer, 1991, pp. 588–599.

[22] A. Muchnik, A. Sëmenov, and M. Ushakov, "Almost periodic sequences," *Theoretical Computer Science*, vol. 304, no. 1, pp. 1–33, 2003.

[23] T. Karimov, "Algorithmic verification of linear dynamical systems," PhD thesis, Saarland University, 2024.

[24] V. Berthé, C. Holton, and L. Zamboni, "Initial powers of Sturmian sequences," *Acta Arithmetica*, vol. 122, 01 2006.

[25] A. Glen and J. Justin, "Episturmian words: a survey," *RAIRO - Theoretical Informatics and Applications*, vol. 43, no. 3, p. 403–442, 2009.

[26] F. Gheeraert, M. Lejeune, and J. Leroy, "$\mathcal{S}$-adic characterization of minimal ternary dendric shifts," *Ergodic Theory and Dynamical Systems*, vol. 42, no. 11, p. 3393–3432, 2022.

[27] "Rigidity and substitutive dendric words," *International Journal of Foundations of Computer Science*, vol. 29, no. 05, pp. 705–720, 2018.

[28] F. Gheeraert and J. Leroy, "$\mathcal{S}$-adic characterization of minimal dendric shifts," 2024. [Online]. Available: https://arxiv.org/abs/2206.00333

Appendix A
Proofs from Sec. II

A. The linked pair construction

Proof of Thm. 3. We apply Ramsey's theorem.

Theorem 41 (Ramsey)**.** *Let V be an infinite set, M be a finite set of colours, and $h : [V]^2 \to M$ map each subset of V of size 2 to a colour. Then there exists some infinite subset U such that h assigns all size-2 subsets of U the same colour.*

Recall that $\alpha \in \Sigma^\omega$, and $h : \Sigma^* \to M$.

Existence. We first prove the existence of a linked pair with an appropriate factorisation. Let $V = \mathbb{N}$, and we assign a set $\{i, j\}$ (with $i < j$) the colour $h(\alpha(i, j)) \in M$. By Ramsey's theorem, there must exist a set of indices $i_0 < i_1 < \cdots$ such that for all $k < \ell$, $h(\alpha(i_k, i_\ell)) = e$. This e is indeed idempotent: observe that

$$e = h(\alpha(i_0, i_2)) = h(\alpha(i_0, i_1)) \cdot h(\alpha(i_1, i_2)) = e \cdot e.$$

Now, let $h(\alpha(0, i_0)) = m$. Let $m \cdot e = \eta$. Observe that $\eta \cdot e = m \cdot e \cdot e = m \cdot e = \eta$, establishing that (η, e) is a desired linked pair, and the factorisation is $\alpha(0, i_1), \alpha(i_1, i_2), \ldots$.

Uniqueness. We now prove that (η, e) is unique up to conjugacy. Suppose we have another linked pair (η', e') and its corresponding factorisation $\alpha(0, j_1), \alpha(j_1, j_2), \ldots$. We observe that for all $k \geq 1$, we have $h(\alpha(0, i_k)) = \eta$, and $h(\alpha(0, j_k)) = \eta'$.

Now, for each k, let ℓ be the smallest index such that $j_\ell \geq i_k$, and let $h(\alpha(i_k, j_\ell)) = x'_k$. We argue (similarly to how we obtained the component η to show the existence of a linked pair) that for each $p > \ell$, $h(\alpha(i_k, j_p)) = x_k = x'_k \cdot e'$. By pigeonhole principle, there exists an x such that $x_k = x$ for infinitely many k. In other words, for this x, there exist infinitely many k such that $h(\alpha(i_k, j_p)) = x$ for all but finitely many p. Let the set of such k be K. Reasoning symmetrically, there exists y for which there exist infinitely many ℓ such that $h(\alpha(j_\ell, i_q)) = y$ for all but finitely many q. Let the set of such ℓ be L.

Now, consider $k \in K$, and $\ell \in L$ such that $h(\alpha(i_k, j_\ell)) = x$. We immediately deduce that $\eta' = \eta \cdot x$. Furthermore, there exists $k' \in K$ such that $h(\alpha(j_\ell, i_{k'})) = y$, whence we deduce that $x \cdot y = e$. Reasoning symmetrically, we also deduce that $\eta = \eta' \cdot y$, and $y \cdot x = e'$, thus establishing conjugacy of (η, e) with (η', e'). □

B. Automata to Monoid construction

We largely follow [14]. In order to reason about a run of an automaton $\mathcal{A}$ on α, we introduce the notion of a *journey*, which is a tuple in $Q \times Q \times (\mathbb{N} \cup \{-1\})$. A word u of length ℓ can make a journey $(\rho(0), \rho(\ell), c)$ in $\mathcal{A}$ if $\delta(\rho(j), u(j)) = \rho(j+1)$ for all $j < \ell$, and $c = \max_{1 \leq j \leq \ell} \mathsf{index}(\rho(j))$. The empty word can only make journeys of the form $(q, q, -1)$. By construction, journeys aggregate a segment of a run. Journeys also compose: if u can make the journey (q_1, q_2, c_1) and v can make the journey (q_2, q_3, c_2), then uv can make the journey $(q_1, q_3, \max(c_1, c_2))$. We record the following, which will allow us to factor α as we wish.

Lemma 42. *Let $\mathcal{A}$ be an automaton over the alphabet Σ and $\alpha \in \Sigma^\omega$. Further let $(u_n)_{n\in\mathbb{N}}$ be a sequence of finite words such that $\alpha = u_0 u_1 \cdots$, and*

$$(q_0, q_1, c_0), (q_1, q_2, c_1), \ldots$$

be a sequence of journeys such that $q_0 = q_{\mathsf{init}}$, and the word u_p can make the journey (q_p, q_{p+1}, c_p) for all n. Then

$$\limsup_{k\to\infty} \mathsf{index}(r_\alpha(k)) = \limsup_{p\to\infty} c_p$$

where r_α is the run of $\mathcal{A}$ on α.

We define a congruence $\sim_{\mathcal{A}}$ on finite words: $u \sim_{\mathcal{A}} v$ if and only if the sets of journeys (which have exactly one element per starting state) that u and v can undertake in $\mathcal{A}$ are equal. Observe that if $u \sim_{\mathcal{A}} v$ and $x \sim_{\mathcal{A}} y$ then $ux \sim_{\mathcal{A}} vy$. Since there are only finitely many sets of possible journeys, there are finitely many congruence classes. We define the *journey monoid* $M_{\mathcal{A}}$ to be the monoid of congruence classes, and the morphism $h_{\mathcal{A}}$ to map each word to its congruence class. Observe that $M_{\mathcal{A}}$ is generated by $\{m : h_{\mathcal{A}}(a) = m, a \in \Sigma\}$ and has $1_{M_{\mathcal{A}}} = h_{\mathcal{A}}(\varepsilon)$ as its neutral element. In fact $h_{\mathcal{A}}(v) = 1_{M_{\mathcal{A}}}$ if and only if $v = \varepsilon$. An element m of $M_{\mathcal{A}}$ may be presented as a representative word u such that $h_{\mathcal{A}}(u) = m$. We extend $M_{\mathcal{A}}$ into $\overline{M_{\mathcal{A}}}$, and $h_{\mathcal{A}}$ into $\overline{h_{\mathcal{A}}}$ as defined in Sec. II-B.

Let $\alpha \in \Sigma^\omega$ with $\overline{h_{\mathcal{A}}}(\alpha) = [\eta, e]$. Consider a linked pair (η, e) such that $\alpha = u_0 u_1 \cdots$ where $h_{\mathcal{A}}(u_0) = \eta$, and $h_{\mathcal{A}}(u_j) = e$ for all $j \geq 1$. This linked pair *prescribes* a decomposition of the run of $\mathcal{A}$ on α into journeys $(q_0, q_1, c_0), (q_1, q_2, c_1), \ldots$ such that $q_0 = q_{\mathsf{init}}$, words in the congruence class η undertake the journey (q_0, q_1, c_0), and for all $j \geq 1$, words in the congruence class e undertake the journey (q_j, q_{j+1}, c_j). The following lemma shows that $\overline{h_{\mathcal{A}}}(\alpha)$ determines acceptance of α by $\mathcal{A}$.

Lemma 43. *Let $\mathcal{A}$ be an automaton over the alphabet Σ, and $\alpha, \alpha' \in \Sigma^\omega$ be such that $\overline{h_{\mathcal{A}}}(\alpha) = \overline{h_{\mathcal{A}}}(\alpha')$. We have that $\mathcal{A}$ accepts α if and only if it accepts α'. More specifically,*

$$\limsup_{k\to\infty} \mathsf{index}(r_\alpha(k)) = \limsup_{k\to\infty} \mathsf{index}(r_{\alpha'}(k)).$$

Proof. Let (η, e) and (η', e') be linked pairs respectively corresponding to α and α' and let x, y be witnesses of their conjugacy, i.e., $\eta' = \eta \cdot x$, $\eta = \eta' \cdot y$, $e = x \cdot y$, $e' = y \cdot x$. Let $(q_0, q_1, c_0), (q_1, q_2, c_1), \ldots$ and $(q_0, q'_1, c'_0), (q'_1, q'_2, c'_1), \ldots$ respectively be the sequences of journeys prescribed for α and α'. By Lem. 42, it suffices to prove that $\limsup_k c_k = \limsup_k c'_k$.

For each $k \geq 1$, let words in the congruence class x make journeys (q_k, q''_k, d_{2k-1}) and those in the congruence class y make journeys (q''_k, q_{k+1}, d_{2k}) such that $c_k = \max(d_{2k-1}, d_{2k})$. Indeed, this property must hold because $x \cdot y = e$. In particular, this implies that $\limsup_\ell d_\ell = \limsup_k c_k$.

We now use the facts that $\eta' = \eta \cdot x$ and $e' = y \cdot x$. This means that words in the class η' can make the journey (q_0, q'_1, c'_0) as well as $(q_0, q''_1, \max(c_0, d_1))$, giving us $q'_1 = q''_1$ and $c'_0 = \max(c_0, d_1)$. Continuing inductively, we get that for all $k > 1$, $q'_k = q''_k$, and $c'_k = \max(d_{2k}, d_{2k+1})$ by equating journeys words in the class e' can undertake starting in q'_k. This gives us that $\limsup_k c'_k = \limsup_\ell d_\ell = \limsup_k c_k$, completing the proof. □

Consequently, we have that we can equivalently recognise the language of $\mathcal{A}$ through the ω-monoid $\overline{M_{\mathcal{A}}}$.

Appendix B
Proofs from Sec. III

Proof of Lem. 15. We first prove the directive sequence is everywhere growing. Let $(a_n)_{n\in\mathbb{N}}$ be a sequence of letters and $k \in \mathbb{N}$. Let $m \geq k$ be such that $\sigma_{k+1} \cdots \sigma_m$ is positive. Then for every $n \geq m$, $\sigma_{k+1} \cdots \sigma_n$ is positive and hence a_k appears in $\sigma_{k+1} \cdots \sigma_n(a_n)$. It follows that

$$\begin{aligned}|\sigma_0 \cdots \sigma_n(a_n)| &= |\sigma_0 \cdots \sigma_k(\sigma_{k+1} \cdots \sigma_n(a_n))| \\ &> |\sigma_0 \cdots \sigma_k(a_k)|,\end{aligned}$$

as required.

We now establish uniform recurrence. Suppose $w \in \mathcal{L}(s)$. We have that for some n, a, $\sigma_0 \cdots \sigma_n(a)$ has w as a factor. By weak primitivity, there exists r such that every image of $\sigma_{n+1} \cdots \sigma_{n+r}$ contains a, and hence every image of $\sigma_0 \cdots \sigma_{n+r}$ contains w as a factor. It follows that there exists ℓ such that every length-ℓ word in $\mathcal{L}(s)$ has w as a factor.

Finally, suppose for the sake of contradiction that $w \notin \mathcal{L}(\alpha)$ for some $\alpha \in X_s$. We thus have that α has a length-ℓ factor v that does not contain w, contradicting that $\mathcal{L}(\alpha) \subseteq \mathcal{L}(s)$. □

Appendix C
Proofs from Sec. IV

Proof of Lem. 20. Suppose $\alpha = u_0 u_1 \cdots$, where $u_i \in \mathsf{congenials}_s$ for all i. Let $(\sigma_n, a_n^{(i)})_{n\in\mathbb{N}}$ be a congenial sequence generating u_i, and $u_i^{(n)} = \lim_{k\to\infty} \sigma_n \cdots \sigma_k(a_k^{(i)})$ for all i. We have that $u_n^{(i)} = \sigma_n(u_{n+1}^{(i)})$ for all n. It remains to define $\alpha^{(n)} = u_0^{(n)} u_1^{(n)} \cdots$. Then $\alpha^{(0)} = \alpha$ and $\alpha_n^{(i)} = \sigma_n(\alpha_{n+1}^{(i)})$ for all n.

Now suppose α is s-directed, and let $(\alpha^{(n)})_{n\in\mathbb{N}}$ be the witnessing sequence of words with $\alpha^{(0)} = \alpha$. Write a_n for the first letter of $\alpha^{(n)}$. We construct the desired factorisation inductively. Let v be the word defined by the congenial sequence $(\sigma_n, a_{n+1})_{n\in\mathbb{N}}$. By the choice of $(a_n)_{n\in\mathbb{N}}$, v is a prefix of α. If $v = \alpha$, then we are done. Otherwise, v must be finite. Let $(v_n)_{n\in\mathbb{N}}$ be the unique sequence of finite words such that $v_0 = v$, v_n is a prefix of $\alpha^{(n)}$ for all n, and $\sigma_n(v_{n+1}) = v_n$ for all n. Write $\alpha^{(n)} = v_n \gamma^{(n)}$ for all n. Because $\sigma_n(\alpha^{(n+1)}) = \alpha^{(n)}$ for all n and $\sigma_n(v_{n+1}) = v_n$, we have that $\sigma_n(\gamma^{(n+1)}) = \gamma^{(n)}$. That is, $\gamma = \gamma^{(0)}$ is a suffix of α directed by s. Set $u_0 = v$, and repeat the process on γ. □

Appendix D
Proofs from Sec. V

Proof of Lem. 24. We simply iterate over each syntactic possibility for $\mathsf{segments}_\sigma$, and check if it is realised by a non-erasing substitution. In order to do so, for each letter a, we will find a word w_a such that assigning $\sigma(a) = w_a$ is consistent with $\mathsf{segments}_\sigma(a, h)$ for all h. For $a \in \Sigma$ and $h \in \mathsf{morphisms}_{\mathcal{A}}$ under consideration, let the purported $\mathsf{segments}_\sigma(a, h) = \langle (b_1, x_1), \ldots, (b_k, x_k) \rangle$ with $k \geq 1$ such that $b_i \neq b_j$ for all $i \neq j$ and $x_i \in M_{\mathcal{A}}$, $b_i \in \Sigma$ for all i. We can compute regular languages $L_1, \ldots, L_k \subseteq \Sigma^*$ such that for all i and $w \in \Sigma^*$, $w \in L_i$ if and only if $h(w_i) = x_i$ and $w_i \in \{1, \ldots, b_i\}^*$. Denote $L_{a,h} = b_1 L_1 \cdots b_k L_k$.

We can effectively check whether $L_a = \bigcap_{h \in \mathsf{morphisms}_{\mathcal{A}}} L_{a,h}$ is non-empty, and if yes, effectively compute $w_a \in L_a$. Such a word can be computed as an image for every letter (if and) only if the purported $\mathsf{segments}_\sigma$ is indeed realisable by assigning each $\sigma(a)$ to the corresponding w_a. □

Appendix E
Proofs from Sec. VI

Proof of Lem. 27. **Saturation.** The sequences $(\sigma^n(a))_n$ and $(\sigma^{n+1}(a))_{n+1}$ have the same limit.

Distributivity. We have that for all n, $\sigma^n(uv) = \sigma^n(u)\sigma^n(v)$. If $\sigma^\omega(u) = \mu \in \Sigma^*$, then for all large n, $\sigma^n(uv) = \mu \cdot \sigma^n(v)$. Taking the limit, we get $\sigma^\omega(uv) = \mu \cdot \sigma^\omega(v)$. If $\sigma^\omega(u) \in \Sigma^\omega$, then for every position j, there exists N such that for all $n \geq N$, $|\sigma^n(u)| > j$. In other words, every position j is eventually part of $\sigma^n(u)$. Thus, $\sigma^\omega(uv) = \sigma^\omega(u)$. However, if $\beta \in \Sigma^\omega$, then $\beta\mu = \beta$ for all $\mu \in \Sigma^\infty$. If $\sigma^\omega(u) = \bot$, there is some position j such that the letter $\sigma^n(u)(j)$ fluctuates with n. This means that the limit of $\sigma^n(u)\sigma^n(v)$ must also be mapped to $\bot$.

Left-expansion. Follows by repeatedly unrolling the equality.

Right-expansion. Follows by repeatedly applying σ.

Cycle of Contradiction. The limit must be $\bot$ as the first letter of $\sigma^n(a_r)$ keeps alternating between $a_0, \ldots, a_{p-1}$.

Terminal Letters. We find the set A of letters a such that $\sigma^\omega(a) = u \in \Sigma^+$ by saturation. The key idea is that if $\sigma^\omega(a)$ converges within n iterations, then $\sigma^\omega(\sigma(a))$ must converge within $n-1$ iterations, i.e., for every letter b in $\sigma(a)$, $\sigma^\omega(b)$ must converge within $n-1$ iterations.

We start with the set A_0 of letters a_0 such that $\sigma(a_0) = a_0$. We construct A_{n+1} as the union of A_n with the set of letters a_{n+1} such that $\sigma(a_{n+1})$ only contains letters from A_n. This construction will saturate within $|\Sigma|$ steps. The invariant is that A_j is the set of a such that $\sigma^\omega(a)$ converges within j iterations. We conclude that since for $n \geq |\Sigma|$, $A_n = A_{|\Sigma|}$, if $\sigma^\omega(a)$ converges in n steps then it must have already converged within $|\Sigma|$ steps. □

Detailed proof of Thm. 28. By Lem. 27, we have a dynamic programming algorithm to compute $\overline{h} \circ \sigma^\omega(a)$ in a "depth first" manner.

Indeed, assume that $\sigma(b) = c_1 v_1 \cdots c_k v_k$ (where the factorisation is based on the first occurrence of each letter; we get $c_1, \ldots, c_k$ from $\mathsf{introduces}_\xi$). By Saturation, we have that

$$\overline{h} \circ \sigma^\omega(b) = \overline{h} \circ \sigma^\omega(c_1) \cdot \overline{h} \circ \sigma^\omega(v_1) \cdots \overline{h} \circ \sigma^\omega(c_k) \cdot \overline{h} \circ \sigma^\omega(v_k).$$

We observe that the terms $\overline{h} \circ \sigma^\omega(v_i)$ will affect the result only if all $\overline{h} \circ \sigma^\omega(c_j)$ for $j \leq i$ are elements of $M_{\mathcal{A}}$. In this case, $\sigma^\omega(v_i)$ will also be a finite word, and by the Terminal Letters property, be $\sigma^{|\Sigma|}(v_i)$. We denote $h \circ \sigma^{|\Sigma|}$ by g; thus in this case, we have $\overline{h} \circ \sigma^\omega(v_i) = g(v_i)$. We can rewrite

$$\overline{h} \circ \sigma^\omega(b) = \overline{h} \circ \sigma^\omega(c_1) \cdot g(v_1) \cdots \overline{h} \circ \sigma^\omega(c_k) \cdot g(v_k). \quad (5)$$

The elements $g(v_1), \ldots, g(v_k)$ can be obtained from $\mathsf{segments}_\xi(b, g)$.

If $c_1 = b$, we are in the simpler case of Right-expansion where $\sigma(b) = bu$, and we have that

$$\overline{h} \circ \sigma^\omega(b) = h(b) \cdot h(u) \cdot h \circ \sigma(u) \cdot h \circ \sigma^2(u) \cdots .$$

The first two terms are easily obtained through tail_ξ. We observe that $h \circ \sigma^n(u) = \mathsf{tail}_\xi(b, \mathsf{compose}_{\xi^n}(h))$. Since the sequence of monoid elements ξ^n is effectively ultimately periodic, so is the sequence of factors in the above infinite product, allowing us to compute it.

If $c_1 \neq b$, we first check that there is no r such that $\sigma^r(c_1) = b$: this can be done with access to head_{ξ^r} for $r \leq |\Sigma|$. If this check fails, we have a Cycle of Contradiction, and have that $\overline{h} \circ \sigma^\omega(b) = \bot$.

We now evaluate expansion 5. Write $m_0 = 1_{M_{\mathcal{A}}}$, and

$$m_i = \overline{h} \circ \sigma^\omega(c_1) \cdot g(v_1) \cdots \overline{h} \circ \sigma^\omega(c_k) \cdot g(v_i).$$

Clearly, $m_{i+1} = m_i \cdot \overline{h} \circ \sigma^\omega(c_{i+1}) \cdot g(v_{i+1})$, and

$$\overline{h} \circ \sigma^\omega(b) = m_i \cdot \overline{h} \circ \sigma^\omega(c_{i+1}) \cdot g(v_{i+1}) \cdots \overline{h} \circ \sigma^\omega(c_k) \cdot g(v_k).$$

In particular when $m_i \notin M_{\mathcal{A}}$ for some i, then $\overline{h} \circ \sigma^\omega(b) = m_i$. For each i, if $c_i \neq b$, we evaluate $\overline{h} \circ \sigma^\omega(c_i)$ (we make a recursive call if it has not been evaluated before, otherwise we look up the memoized value). Otherwise, we are in the case of Left-expansion, and get that $\overline{h} \circ \sigma^\omega(b) = m_{i-1}^\omega$. In any case, we will eventually compute $m_k = \overline{h} \circ \sigma^\omega(b)$.

Our desired result follows by applying the above depth-first routine to compute $\overline{h} \circ \sigma^\omega(a)$. □

APPENDIX F
PROOF OF LEM. 35 FROM SEC. VII

We will need the following.

Lemma 44. *A sequence $(\sigma_n)_{n\in\mathbb{N}}$ of non-erasing substitutions of the type $\Sigma^* \to \Sigma^*$ is* not *everywhere growing if and only if there exist $m \geq 0$ and a sequence of letters $(b_n)_{n\geq m}$ such that for all $n \geq m$, $\sigma_{n+1}(b_{n+1}) = b_n$.*

Proof. **If.** This direction is trivial, as any sequence of letters $(a_n)_{n\in\mathbb{N}}$ satisfying $a_n = b_n$ for all $n \geq m$ is a witness for the sequence not being everywhere growing.

Only If. Suppose $(\sigma_n)_{n\in\mathbb{N}}$ is not everywhere growing. Let $\ell_n = \min_{a\in\Sigma} |\sigma_0 \cdots \sigma_n(a)|$ for $n \in \mathbb{N}$. By construction, $(\ell_n)_n$ is monotone, and hence either converges to an integer or becomes arbitrarily large. By the assumption on $(\sigma_n)_{n\in\mathbb{N}}$, there exist ℓ, m such that $\ell_n = \ell$ for all $n \geq m$.

Let $k > 0$ and $a \in \Sigma$ be such that $|\sigma_0 \cdots \sigma_{m+k}(a)| = \ell$. Because $\ell_m, \ldots, \ell_{m+k} = \ell$, for every $0 \leq s < k$ we have

$$\begin{aligned} |\sigma_0 \cdots \sigma_{m+k}(a)| &= |\sigma_0 \cdots \sigma_{m+s}(\sigma_{m+s+1} \cdots \sigma_{m+k}(a))| \\ &\geq \ell \cdot |(\sigma_{m+s+1} \cdots \sigma_{m+k}(a))|. \end{aligned}$$

It follows that $|(\sigma_{m+s+1} \cdots \sigma_{m+k}(a))| = 1$ for all $0 \leq s < k$. Thus for every $k > 0$ there exist $b_m, \ldots, b_{m+k} \in \Sigma$ such that $\sigma_{m+s+1}(b_{m+s+1}) = b_{m+s}$ for all $0 \leq s < k$. It then follows from König's lemma that there exists an infinite sequence $(b_n)_{n\geq m}$ of letters such that $\sigma_{n+1}(b_{n+1}) = b_n$ for all $n \geq m$. □

We can now finalise the proof. It suffices to construct an automaton $\mathcal{B}'$ over $\Xi_{\mathcal{A}}$ that recognises traces of sequences that are not everywhere growing. The automaton $\mathcal{B}'$, on input $(\xi_n)_{n\in\mathbb{N}}$, simply guesses and verifies m and the sequence $(b_n)_{n\geq m}$ in Lem. 44, by checking that $\mathsf{expanding}_{\xi_n}(b_n)$ evaluates to false and $\mathsf{head}_{\xi_n}(b_n) = b_{n-1}$ for all $n > m$.

APPENDIX G
PROOFS FROM SEC. VIII

Proof of Lem. 39. Since we have access to the factor complexity function of α (and β), we can evaluate $R_\alpha(n)$ for all n: given n, enumerate prefixes $\alpha(0, L)$ until finding M such that (i) $\gamma[0, L)$ contains $p_\alpha(M)$ distinct factors of length M, and (ii) all of these $p_\alpha(M)$ factors contain $p_\alpha(n)$ distinct factors of length n. Then $M = R_\alpha(n)$.

Applying Prop. 11, we obtain N_1 and an ℓ such that if $\alpha[0, N_1) = \beta[0, N_1)$ and $R_\alpha(n) = R_\beta(n)$ for $n \leq \ell$, then α and β agree upon acceptance by $\mathcal{A}$. By the above paragraph, we can compute N_2 such that if $\alpha[0, N_2) = \beta[0, N_2)$, then $R_\alpha(n) = R_\beta(n)$ for $n \leq \ell$. It remains to take $N = \max(N_1, N_2)$. □