RAIRO-Theor. Inf. Appl. **42** (2008) 417–450
DOI: 10.1051/ita:2008008

Available online at:
www.rairo-ita.org

A HIERARCHY OF AUTOMATIC ω-WORDS HAVING A DECIDABLE MSO THEORY

Vince Bárány[1]

Abstract. We investigate automatic presentations of ω-words. Starting points of our study are the works of Rigo and Maes, Caucal, and Carton and Thomas concerning lexicographic presentation, MSO-interpretability in algebraic trees, and the decidability of the MSO theory of morphic words. Refining their techniques we observe that the lexicographic presentation of a (morphic) word is in a certain sense canonical. We then generalize our techniques to a hierarchy of classes of ω-words enjoying the above mentioned definability and decidability properties. We introduce k-lexicographic presentations, and morphisms of level k stacks and show that these are inter-translatable, thus giving rise to the same classes of k-lexicographic or level k morphic words. We prove that these presentations are also canonical, which implies decidability of the MSO theory of every k-lexicographic word as well as closure of these classes under MSO-definable recolorings, *e.g.* closure under deterministic sequential mappings. The classes of k-lexicographic words are shown to constitute an infinite hierarchy.

Mathematics Subject Classification. 03D05, 68Q42, 68Q45, 68R15.

1. Introduction

This paper is concerned with infinite words of type ω, which are finitely presentable using automata and have a decidable monadic second-order theory. As such it is connected to two not so distant lines of research around the theme of using automata to decide logical theories.

Keywords and phrases. Morphic words, monadic second-order logic, automatic structures, automatic sequences.

[1] Mathematische Grundlagen der Informatik, RWTH Aachen, Aachen, Germany; vbarany@logic.rwth-aachen.de

Article published by EDP Sciences

The first of these approaches, initiated by Büchi, is limited to structures such as infinite words, ranked or unranked trees, or certain linear orderings for which an appropriate automaton model is definable, which has effective closure and decision properties and is expressively equivalent to monadic second-order logic over these structures. In this approach formulas are transformed into equivalent automata operating on the structure itself. The second approach consists of using finite automata to describe the operations and relations of a given structure with respect to a chosen representation of elements by words (or by trees). One thus speaks of an *automatic presentation* of the structure. Closure properties of automata yield in this case a straightforward procedure for representing not only the atomic relations but all first-order-definable relations of the represented structure. The first-order theory of the represented structure is thus "automaton decidable" [29]. Note that in this case the automata do not operate on the structure but rather encode it in a finite way.

In this paper we combine both of these approaches to decide the monadic second-order theory of certain automatically presentable ω-words. We provide a construction for transforming an automatic presentation of an ω-word w and a deterministic Muller automaton $\mathcal{A}$ into an automatic presentation of the behavior or run of $\mathcal{A}$ on w, which is itself an ω-word.

We define a hierarchy of classes of ω-words depending on the complexity of the automatic presentation involved, investigate basic closure properties of these classes, show how they generalize the notion of morphic words, and prove that they indeed form an infinite hierarchy.

Monadic second-order logic (MSO) on ω-words. Büchi originated the use of automata, ω-automata invented for this purpose, in deciding the MSO theory of the naturals with successor $(\mathbb{N}, succ)$. The fundamental result underlying this method is the convertibility of MSO formulas into Büchi automata and *vice versa*, which are equivalent in a natural way [28,53]. The same correspondence holds for extensions of $(\mathbb{N}, <)$ by unary predicates P_a $(a \in \Sigma)$, which can be assumed to partition $\mathbb{N}$. Deciding the MSO theory of such extensions is, by the above, equivalent to the problem of deciding acceptance of the corresponding ω-word by any given Büchi automaton. Elgot and Rabin [25] have invented the *method of contractions* to reduce this problem, for suitable ω-words, to the case of ultimately periodic ones, for which it is trivially solvable. However, little is known as to the applicability of the contraction method. Elgot and Rabin have illustrated their technique by proving the MSO decidability of the characteristic sequences of the factorial predicate, of k-th powers, and of powers of a fixed k. Fairly recently, Carton and Thomas [17] have used a very similar technique to prove the MSO decidability of *morphic words*. These results are elegantly rounded up in [43] and [44].

Morphic words. The study of morphic words, having applications in combinatorics of words, goes back to Thue. An ω-word is said to be *morphic* if it is the homomorphic image of a fixed point of a morphism of finite words. It is thus associated to an HD0L system in a natural way, namely, as the limit of the sequence of words generated by it, *i.e.* by iterated application of the latter morphism (*cf.* [30,32,47]). Morphic words have been intensively studied in both of these contexts.

In this paper we define *morphisms of k-stacks* (words of words of ... of words, k-fold) and classes of ω-words arising in a similar way by iterating such a morphism. We extend the result of Carton and Thomas to these classes of ω-words. Additionally, we prove that for each k the class of k-morphic ω-words is closed under MSO-definable re-colorings, *e.g.* under d.g.s.m. mappings. This result generalizes that of Pansiot on various subclasses of morphic words (*cf.* [41]) and underlines the robustness of these notions.

Automatic presentations. We use the formalism of *automatic presentations* of ω-word structures. An (injective) automatic presentation of $(\mathbb{N}, <)$ consists of a regular set D of names, a synchronized rational relation $\prec$, and a bijective valuation function $\nu : D \to \mathbb{N}$ such that $n < m$ iff $\nu^{-1}(n) \prec \nu^{-1}(m)$ for every $n, m \in \mathbb{N}$. In the otherwise rich field of *generalized numeration systems* (*cf. e.g.* [12]) the length-lexicographic ordering is the only natural choice for $\prec$. We consider automatic presentations of $(\mathbb{N}, <)$ where the ordering is a generalization of the length-lexicographical one. Given an ordered alphabet, we define the *k-lexicographic ordering* as a kind of k times nested length-lexicographical ordering.

An ω-word structure is an extension $(\mathbb{N}, <, \{P_a\}_{a\in\Sigma})$ of the above by a unary predicate for each letter according to the positions in the word where each letter occurs. We generalize the result of Rigo and Maes [45] by showing that ω-words representable using the k-lexicographic ordering are precisely those generated by iterated applications of a morphism of k-stacks followed by a homomorphic mapping. Thus, we obtain a hierarchy of k-lexicographic or k-morphic ω-words. Indeed, we show that these classes form a strictly increasing infinite hierarchy.

Automatic structures are known to have a decidable first-order theory and to be closed under first-order interpretations *via* straightforward application of basic automata techniques [10,29,33]. We use "higher-order" automata constructions to extend these properties to monadic second-order logic over ω-word structures having a k-lexicographic presentation. The construction follows the factorization of the ω-word provided by the k-lexicographic presentation. The key step consists of showing that the *contraction* of an ω-word (defined with respect to a given $(k+1)$-lexicographic presentation and a given finite monoid – *cf.* Sect. 5.2) is itself k-lexicographic. In other words, the automatic presentation guides us in applying the contraction method. This allows us to argue inductively, or, conversely and more intuitively, to reduce the MSO theory of a k-lexicographic ω-word in k contraction steps to questions about ultimately periodic words.

Related work. Given our still unsatisfactory understanding of the pushdown hierarchy of graphs having decidable MSO theories [20,54] it is natural to ask, which ω-words inhabit the individual levels of this hierarchy. Caucal has shown that morphic words are found on the second level [20] and it is suspected that they in fact exhaust the second level. More generally, Theorem 8.1 below states that

k-lexicographic words are on the $2k$th level of the pushdown hierarchy. However, already on the third level one will find words, which, for reasons of growth, are not automatic [36]. An extension of the notion of morphic words based on simply typed derivation rules has been introduced in order to capture words on respective levels of the pushdown hierarchy [11].

Also related to our work, though more in spirit and in terms of some techniques involved and less as far as the actual classes of sequences are concerned, is the recent work of Fratani and Sénizergues on sequences of integers, rationals and transductions computed by higher-order pushdown automata [26,50].

Outline of the paper. The rest of the paper is structured as follows.

In Section 2 we recall the most fundamental notions of finite synchronous multitape automata, primarily in connection with first-order and monadic second-order logic and in order to fix notation. We also introduce here the less widely known concept of *automatic presentations*, the central objects of our investigation, and summarize the most fundamental facts concerning first-order logic on structures allowing an automatic presentation.

Section 3 is a continuation of the Preliminaries section focusing on ω-words. We recall the classical notion of *morphic* ω*-words*, define what we mean by a *canonical automatic presentation of an* ω*-word* and in preparation for the main result we discuss some general properties of such canonical presentations.

Section 4 introduces the hierarchy of classes of k*-lexicographic* ω*-words* as advertised in the title. The definition requires the existence of a specific kind of automatic presentation of the ω-word involving the k-fold nested lexicographic ordering as introduced in the beginning of the section. We identify some of the basic properties of these presentations and the corresponding classes of ω-words. In particular, we highlight the fact that 0-lexicographic ω-words are the ultimately periodic ones whereas the 1-lexicographic ones are precisely those morphic.

Section 5 is devoted to establishing the Main Theorem 5.3 stating that all relations definable in monadic second-order logic over a k-lexicographic word are in the given presentation recognizable by finite automata. This is surprising because in general only first-order definable relations have this property. A number of consequences of the main result are stated as corollaries.

In Section 6 we prove that the classes of k-lexicographic ω-words actually do form a *proper hierarchy* parametrized by k. We give concrete examples of ω-words separating consecutive levels of this hierarchy.

Section 7 introduces a generalization of the notion of morphic ω-words based on *morphisms of higher-order stacks*, a very limited form of higher-order parallel rewriting. In Theorem 7.5 we prove that ω-words that can be generated *via* morphisms of level k stacks are precisely those allowing a k-lexicographic automatic presentation.

In Section 8 we recall the pushdown hierarchy of graphs and trees defined in terms of alternating applications of tree unfolding and monadic second-order interpretations [20,54], and we demonstrate that for each k the k-lexicographic ω-words can be constructed on the $2k$th level of the *pushdown hierarchy*.

Section 9 closes our discussion with a handful of remarks and open questions.

2. Preliminaries

Words. Let Σ be a finite alphabet. Σ^* denotes the set of finite words over Σ. The length of a word $w \in \Sigma^*$ is written $|w|$, the empty word is ε, for every $0 \leq i < |w|$ the ith symbol of w is written as $w[i]$, and when I denotes some interval of positions then wI (*e.g.* $w[n,m)$) is the factor of w on these positions. Note that we start indexing with 0. Accordingly, for every $n \in \mathbb{N}$, we let $[n] = \{0, \ldots, n-1\}$.

Morphisms. We denote by $\mathsf{Hom}(M, N)$ the set of homomorphisms from the monoid M to N. Each $\varphi \in \mathsf{Hom}(\Sigma^*, \Sigma^*)$ can be specified by the images $\varphi(a)$ of individual symbols $a \in \Sigma$. The length of φ, denoted $|\varphi|$, is the maximum of all the $|\varphi(a)|$, and φ is *uniform*, when $|\varphi(a)| = |\varphi|$ for every $a \in \Sigma$.

Automata. A finite labelled transition system (TS) is a tuple $\mathcal{T} = (Q, \Sigma, \Delta)$, where Q is a finite, nonempty set of states, Σ is a finite set of labels, and $\Delta \subseteq Q \times \Sigma \times Q$ is the transition relation. $\mathcal{T}$ is deterministic (DTS), when Δ is a function of type $Q \times \Sigma \to Q$, in this case we write δ instead of Δ, and δ^* for the unique homomorphic extension of δ to all words over Σ. Alternatively, each deterministic transition system can be represented as a pair (φ, M) where $M = (Q \to Q, \circ, \mathsf{id})$ is the monoid of (partial) functions from Q to Q with composition as product and $\varphi \in \mathsf{Hom}(\Sigma^*, M)$ is such that $\varphi(a)(q) = \delta(q, a)$ for every $a \in \Sigma$ and $q \in Q$. From (φ, M) one can again obtain the presentation (Q, Σ, δ). A *finite automaton* (FA) is a finite transition system together with sets of initial and final states $\mathcal{A} = (\mathcal{T}, I, F) = (Q, \Sigma, \Delta, I, F)$. $\mathcal{A}$ is deterministic (DFA) when $\mathcal{T}$ is and when I contains a single initial state q_0. The unfolding of a DFA $\mathcal{A}$ from its initial state is a Σ-branching Q-labelled regular tree, *i.e.* one having only finitely many subtrees up to isomorphism. Conversely, each such regular tree determines a DFA having the subtree-types as its states. The *completion* of a DFA $\mathcal{A}$ is the DFA $\overline{\mathcal{A}}$ obtained by introducing a new state $\bot$ and setting it the target of all yet undefined transitions. Thus, the transition function $\overline{\delta}$ of $\overline{\mathcal{A}}$ is defined for all pairs (q, a) with $q \in Q \cup \{\bot\}$.

Multi-tape automata. Let Σ be a finite alphabet. We consider relations on words, *i.e.* subsets of $(\Sigma^*)^n$ for some $n > 0$. *Asynchronous* n-tape automata accept precisely the *rational relations*, *i.e.*, rational subsets of the product monoid $(\Sigma^*)^n$. Finite *transducers*, recognizing *rational transductions* [5], are asynchronous 2-tape automata. A relation $R \subseteq (\Sigma^*)^n$ is *synchronized rational* [27] or *regular* [34] if it is accepted by a *synchronous* n-tape automaton. Finally, $R \subseteq (\Sigma^*)^n$ is *semi-synchronous rational* [3] if it is accepted by an n-tape automaton reading each of its tapes at a fixed speed. A *deterministic generalized sequential machine* (d.g.s.m.) $\mathcal{S} = (\mathcal{T}, q_0, \mathcal{O})$ consists of a DTS, an initial state, and an output function $\mathcal{O} : Q \times \Sigma \to \Gamma^*$ and computes, in a natural way, a function $S : \Sigma^* \to \Gamma^*$ that also extends to ω-words.

Automatic structures. The idea to use automata to represent structures goes back to Büchi. The general notion was first introduced and studied by Hodgson [29] and was then rediscovered by Khoussainov and Nerode [33]. Since then it has been subject of some theses and numerous publications, see *e.g.* [4,8,10,34,48,49] for an overview. Note that a great deal of attention has been given to natural automatic presentations of specific structure classes including Cayley-graphs of groups [14] and semigroups, as well as automatic sequences [1,12,13]. We shall take all structures to be relational with functions represented by their graphs.

Definition 2.1 (Automatic structures [33])**.** An automatic presentation of a structure $\mathfrak{A} = (A, \{R_i\}_i)$ consists of a collection of synchronous automata $\mathfrak{d} = (\mathcal{A}_D, \mathcal{A}_\approx, \{\mathcal{A}_{R_i}\}_i)$ and a naming, or coordinate function $\nu : L(\mathcal{A}_D) \to A$, such that $\approx = L(\mathcal{A}_\approx)$ is the equivalence relation $\{(x, y) \mid \nu(x) = \nu(y)\}$ and ν is a homomorphism from $(L(\mathcal{A}_D), \{L(\mathcal{A}_{R_i})\}_i)$ onto $\mathfrak{A}$, hence $\mathfrak{A} \cong (L(\mathcal{A}_D), \{L(\mathcal{A}_{R_i})\}_i)/_\approx$. AutStr designates the class of automatic structures.

One effectively obtains an injective presentation from any given automatic presentation by restricting $L(\mathcal{A}_D)$ to a set of unique (*e.g.* length-lexicographically least) representants of each $\approx$-class. In this paper we will only consider injective presentations, omitting $\approx$, and quite often tacitly consider a tuple of regular relations $(D, \{R_i\}_i)$ as an automatic presentation.

Logics. We use the abbreviation FO and MSO for first-order and for monadic second-order logic, respectively, and $\mathsf{FO}^{\infty,\mathrm{mod}}$ for the extension of FO by infinity ($\exists^\infty$) and modulo-counting quantifiers ($\exists^{(r,m)}$). The meaning of the formulas $\exists^\infty x\, \theta$ and $\exists^{(r,m)} x\, \theta$ is that there are infinitely many elements x, respectively r many elements x modulo m, such that θ holds. We shall make extensive use, often without direct reference, of the well-known relationships of automata and logics (*cf.* [28,53]) as well as of the following facts.

Theorem 2.2 (Consult [8,10] and [35,48])**.**

i) *Let* $(\mathfrak{d}, \nu)$ *be an automatic presentation of* $\mathfrak{A} \in$ AutStr*. Then for each* $\mathsf{FO}^{\infty,\mathrm{mod}}$*-formula* $\varphi(\vec{a}, \vec{x})$ *with parameters* $\vec{a}$ *from* $\mathfrak{A}$*, defining a* k*-ary relation* R *over* $\mathfrak{A}$*, one can construct a* k*-tape synchronous automaton recognizing* $\nu^{-1}(R)$*.*

ii) *The* $\mathsf{FO}^{\infty,\mathrm{mod}}$*-theory of every automatic structure is decidable.*

iii) AutStr *is effectively closed under* $\mathsf{FO}^{\infty,\mathrm{mod}}$*-interpretations.*

In this paper we extend these results to MSO over word structures having automatic presentations of a certain kind, *cf.* Definition 4.1. It will be convenient to consider automatic presentations up to equivalence.

Definition 2.3 (Equivalence of automatic presentations)**.**
Two presentations $(\mathfrak{d}_1, \nu_1)$ and $(\mathfrak{d}_2, \nu_2)$ of some $\mathfrak{A} \in$ AutStr are *equivalent* if for every relation R over $\mathfrak{A}$, $\nu_1^{-1}(R)$ is regular iff $\nu_2^{-1}(R)$ is regular.

In other words, two automatic presentations are equivalent if there is no difference between them in terms of representability of relations *via* automata, *i.e.*

if they are expressively equivalent. In [3] we have shown that two presentations are equivalent iff the transduction translating names of elements from one presentation to the other is computable by a semi-synchronous transducer: a two-tape finite automaton whose transitions, with the possible exception of a final one, are labelled by elements of $\Sigma^k \times \Gamma^l$ uniformly for some fixed positive k and l. Note that, except in trivial cases, k/l is uniquely determined [3].

Theorem 2.4 ([3])**.** *Two presentations* $(\mathfrak{d}_1, \nu_1)$ *and* $(\mathfrak{d}_2, \nu_2)$ *of some* $\mathfrak{A} \in \textsc{AutStr}$ *are equivalent if and only if the transduction* $T = \{(x, y) \in D \times D' \mid \nu_1(x) = \nu_2(y)\}$*, translating names of elements from one presentation to the other, is semi-synchronous rational.*

So equivalent presentations are truly identical modulo such a simple coding, in other words “expressive equivalence” coincides with “computational equivalence”.

To give a simple example, the translation from the base 4 representation of naturals into binary numerals (assuming both to be least-significant-digit-first fashion) is the uniform morphism mapping $0 \mapsto 00$, $1 \mapsto 10$, $2 \mapsto 01$ and $3 \mapsto 11$. The two automatic presentations based on these numerals are thus equivalent. Recall, that, on the other hand, the celebrated theorem of Cobham and Semenov (see *e.g.* [7,13]) implies that for p and q having no common power the base p and base q numeration systems are as far from being equivalent as they can be.

3. Word structures

An ω-word over Σ is a function $w : \mathbb{N} \to \Sigma$. The set of ω-words over Σ is denoted Σ^ω. To every $w \in \Sigma^\omega$ we associate its *word structure* $W_w = (\mathbb{N}, <, \{P_a\}_{a\in\Sigma})$, where $P_a = w^{-1}(a)$ for each $a \in \Sigma$. Word structures of finite words are defined similarly. Note that we consider the ordering, as opposed to the successor relation, as given in our word structures. When one is working with monadic second-order logic, there is of course no difference in terms of expressiveness. However, as we are engaging in an investigation of automatically presentable word structures, the presence of the ordering is not without significance.

Morphic words. A particularly well understood class of ω-words is that of the so called *morphic words*. The basic idea, successfully applied by Thue, is to obtain an infinite word *via* iteration of a suitable morphism $\tau : \Sigma^* \to \Sigma^*$. Suitability is expressed by the condition that $\tau(a)[0] = a$ for some $a \in \Sigma$. In this case τ is said to be *prolongable on* a. This ensures that the sequence $(\tau^n(a))_{n\in\mathbb{N}}$ converges to either a finite or infinite word, which is a fixed point of τ, denoted $\tau^\omega(a)$. An ω-word $w \in \Gamma^\omega$ is morphic, if $w = \sigma(\tau^\omega(a))$ for some τ prolongable on a and some $\sigma \in \mathsf{Hom}(\Sigma^*, \Gamma^*)$ extended in the obvious way to ω-words.

Example 3.1. Consider $\tau : a \mapsto ab, b \mapsto ccb, c \mapsto c$ and $\sigma : a, b \mapsto 1, c \mapsto 0$ both homomorphically extended to $\{a, b, c\}^*$. The fixed point of τ starting with a is the word $abccbccccbc^6b\ldots$, and its image under σ, $110010000100^61\ldots$, is the characteristic sequence of the set of squares.

In general, as was shown in [17], the characteristic sequence of every set of the form $\{\sum_{k=0}^{n} s_k \mid n \in \mathbb{N}\}$, where $0 < (s_k)$ is an $\mathbb{N}$-rational sequence is morphic. This result follows trivially from the characterization of [45], *cf.* Proposition 4.3.

Example 3.2. Let $\phi : a \mapsto ab, b \mapsto a$. Its fixed point $\phi^\omega(a)$ is the *Fibonacci word* $f = abaababaabaababaababa\ldots$, so called for the recursive dependence $\phi^{n+2}(a) = \phi^{n+1}(a) \cdot \phi^n(a)$ implying that $|\phi^n(a)|$ is the nth Fibonacci number.

Automatic presentations

In accordance with Definition 2.1 an automatic presentation $(D, R, \{P_a\}_{a\in\Sigma})$ of W_w as above comprises a regular set D partitioned by the regular sets P_a for each $a \in \Sigma$ over some alphabet Γ, together with a regular relation R, which is a linear ordering of type ω over D such that the i-th word in this ordering belongs to P_a iff the i-th symbol of w is a. Elements of D can be seen as numerals, each $x \in D$ representing the number $\nu(x)$ where ν is the coordinate map of the presentation. To enhance readability we identify x with $\nu(x)$ and tacitly write *e.g.* $w[x]$ in place of $w[\nu(x)]$ when indexing symbols or factors of w.

The most frequently, if not exclusively, used regular ordering of type ω is the *length-lexicographic* ordering, also called military-, radix-, or genealogical ordering by some and shortlex by others. Starting point of our investigation is the observation that those ω-words admitting an automatic presentation using the length-lexicographic ordering are precisely the *morphic* ones (*cf.* [45]). Nevertheless there are other choices of ordering worth investigating. Indeed, as we shall see, increasing the complexity of the ordering widens the class of words thus presentable. First we define the key concept of canonicity and derive extensions of Theorem 2.2 to MSO over word structures having a canonical presentation.

Definition 3.3 (Canonical presentations)**.** An automatic presentation $\mathfrak{d} = (D, <, \{P_a\}_{a\in\Sigma})$ of some infinite word $w \in \Sigma^\omega$ is *canonical* if there is an algorithm, which constructs for every homomorphism $\psi \in \mathsf{Hom}(\Sigma^*, M)$ into a finite monoid M and for every monoid element $m \in M$ a synchronous two-tape automaton recognizing the relation

$$B_m = \{(x, y) \in D^2 \mid x < y \wedge \psi(w[x, y]) = m\}.$$

Thus, canonicity means that membership of finite factors of w in a regular language can be decided by an effectively constructable automaton reading the representations of the two endpoints of the factor. It is very easy to derive decidability of the monadic second-order theory of words having canonical presentations.

Lemma 3.4. Let $\mathfrak{d} = (D, <, \{P_a\}_{a\in\Sigma})$ and ν constitute a canonical presentation of $w \in \Sigma^\omega$. Then for every deterministic Muller automaton $\mathcal{A}$ an automaton recognizing the following set can be effectively constructed.

$$E_{\mathcal{A}} = \{x \in D \mid w[x, \infty) \in L(\mathcal{A})\}.$$

Proof. Consider $\mathcal{A}$ as a pair (ψ, M) with $M = (Q \to Q, \circ)$ and $\psi \in \mathsf{Hom}(\Sigma^*, M)$. Canonicity of $\mathfrak{d}$ yields automata recognizing $X_q = \{(x, y) \in D^2 \mid x < y \wedge$

$\psi(w[x,y])(q_0) = q\}$ for each $q \in Q$. Using Theorem 2.2 we can construct automata recognizing $Y_F = \{x \in D \mid \bigwedge_{q\in F} \exists^\infty y\, X_q(x,y) \wedge \bigwedge_{q\notin F} \neg\exists^\infty y\, X_q(x,y)\}$ for all $F \subseteq Q$. Finally, $E_{\mathcal{A}}$ is the union of those Y_F such that a run of $\mathcal{A}$ is accepting with F being the set of infinitely often occurring states. The claim follows. □

Corollary 3.5. *Let w be an ω-word having a canonical automatic presentation. Then the* MSO*-theory of W_w is decidable.*

Proof. In line with the well known correspondence between automata and MSO on ω-words deciding the MSO-theory of a word structure amounts to deciding acceptance of the word by any given deterministic Muller automaton $\mathcal{A}$. Given a canonical presentation this can be done by checking membership of $\nu^{-1}(0)$ in $E_{\mathcal{A}}$ constructed as in the above lemma. □

Canonicity yields more than just decidability as we shall see next. Let φ be an MSO sentence in a language of word structures and let x, y be first-order variables not occurring in any subformula of φ. We define three kinds of *relativizations of* φ: $\varphi^{[0,x]}$, $\varphi^{[x,y]}$, and $\varphi^{[x,\infty)}$ obtained by relativizing all first- and second-order quantifications to the noted intervals. For instance $(\exists z\vartheta)^{[x,y]} = \exists z(x \leq z \wedge z \leq y \wedge \vartheta^{[x,y]})$, and $(\forall Z\vartheta)^{[x,\infty)} = \forall Z(\forall z(z \in Z \rightarrow x \leq z) \rightarrow \vartheta^{[x,\infty)})$. The relevance of relativization is expressed by the equivalence $W_w \models \varphi^I \iff W_{wI} \models \varphi$, where I is an interval of any of the three kinds.

Lemma 3.6 (Normal form of MSO formulas over word structures)**.** Every MSO formula $\varphi(\vec{x})$ having free first-order variables $x_0, \ldots, x_{n-1}$ and no free second-order variables is equivalent to a boolean combination of formulas $x_i < x_j$ and relativized MSO sentences[1] $\vartheta^{[0,x_i]}$, $\vartheta^{[x_i,x_j]}$, and $\vartheta^{[x_i,\infty)}$ with $i, j \in [n]$.

Proof. A similar normalform applies to MSO formulas over trees [21] and is a typical application of the composition method as introduced by Shelah [51]. Here we sketch a proof through automata.

Via standard construction [53], there is a deterministic Muller automaton $\mathcal{A}$ over the alphabet $\Sigma \times \{0,1\}^n$ such that $W_w \models \varphi(\vec{k})$ iff $w \otimes \xi_{\vec{k}} \in L(\mathcal{A})$ for all $\vec{k} \in \mathbb{N}^n$, where $\xi_{\vec{k}} \in (\{0,1\}^n)^\omega$ is the characteristic word of the tuple $\vec{k}$, *i.e.* $\xi_{\vec{k}}[i]_j = 1$ iff $k_j = i$. We collect for each pair of states (p,q) of $\mathcal{A}$ the regular language $L_{p,q} = \{u \in \Sigma^* \mid \delta^*(p, u \otimes (0^n)^{|u|}) = q\}$. Additionally, we let $L_q = \{u \in \Sigma^\omega \mid \mathcal{A}$ accepts $u \otimes (0^n)^\omega$ from state $q\}$. Again, by standard constructions, we find MSO sentences $\vartheta_{p,q}$ respectively ϑ_q defining these languages.

Each infinite word $w \otimes \xi_{\vec{k}}$ is naturally factored into segments in between consecutive k_i's, some of which can be equal. Accordingly, each run of $\mathcal{A}$ can be factored into finite number of finite segments and an infinite segment by those positions where in at least one of the last n components of the symbol read a 1 is encountered. The intermediate segments and the last infinite segment are models of the appropriate sentences $\vartheta_{p,q}$ and of ϑ_q respectively.

[1] For a sentence ϑ its relativization $\vartheta^{[x_i,x_j]}$, for instance, will have x_i and x_j, and only these, as its free variables.

By summing up all possible factorizations of accepting runs we obtain in a first attempt a boolean combination of formulas of type $x_i < x_j$, $x_i = x_j$, $P_a x_i$ and of relativized sentences of the form $\vartheta^{[0,x_i)}_{q_0,q}$, $\vartheta^{(x_i,x_j)}_{p,q}$ and $\vartheta^{(x_i,\infty)}_{q}$. Equality can be expressed using $<$, and integrating the $P_a x_i$ into the neighboring openly relativized segment formulas we finally arrive at a normal form as promised. □

Theorem 3.7 (MSO definability)**.** *Let w be an ω-word having a canonical presentation $\mathfrak{d}$ having domain D and bijective coordinate function $\nu : D \to \mathbb{N}$. Then there is an algorithm transforming every* MSO *formula $\varphi(\vec{x})$ having n free first-order variables (and no free set variables) into an n-tape synchronous automaton $\mathcal{A}$ such that for every $u_1, \ldots, u_n \in D$*

$$W_w \models \varphi[\nu(\vec{u})] \iff \vec{u} \in L(\mathcal{A}).$$

Proof. Using Lemma 3.6, we transform φ into a boolean combination of relativized sentences and comparison formulas $x_i < x_j$. Canonicity and Lemma 3.4 yield automata recognizing the relations defined by relativized sentences $\vartheta^{[0,x_i]}$, $\vartheta^{[x_i,x_j]}$, respectively $\vartheta^{[x_i,\infty)}$. Recall that synchronized rational relations form an effective boolean algebra [27]. Thus, by the appropriate combination of the automaton recognizing $<$ and of the automata recognizing the relativized subformulas of the normal form we obtain $\mathcal{A}$ as required. □

Note that a set $X \subseteq \mathbb{N}$ is definable by an MSO formula $\psi(X)$ in W_w iff it is point-wise definable by one of the form $\varphi(x)$. Thus, (W_w, X) is automatic for every canonically presentable W_w and for every X MSO-definable in W_w.

4. k-lexicographic presentations

Let Γ be a finite non-empty alphabet. To each word $u = a_0 a_1 \ldots a_{n-1} \in \Gamma^*$ of length n and to each $0 < k$ we associate its *k-split* $(u^{(1)}, u^{(2)}, \ldots, u^{(k)})$ defined as follows. Let t be such that $tk \leq n < (t+1)k$. Then the ith word of the k-split is $u^{(i+1)} = a_i a_{k+i} a_{2k+i} \ldots a_{tk+i}$ for each $i < k$. In other words, the $i+1$st component of the k-split, $u^{(i+1)}$, is the subword of u restricted to letters in a position equal to i modulo k. We call *k-merge* the operation $\otimes_k$ producing the original word $u = \otimes_k(u^{(1)}, \ldots, u^{(k)})$ from the components $(u^{(1)}, \ldots, u^{(k)})$. The $rk+i$th letter of $\otimes_k(u^{(1)}, \ldots, u^{(k)})$ is thus the rth letter of $u^{(i)}$. The k-merge will only be applied to words obtainable as components in a k-split. *E.g.* the 2-merge of *abaa* and *cddc* is *acbdadac*. Additionally, we define $u^{(0)} = |u| \in \mathbb{N}$ or in unary presentation as $1^{|u|} \in 1^*$, whichever is more convenient. For $0 \leq i < k$ we define the equivalence

$$u =_i v \overset{\text{def}}{\iff} \forall j \leq i \;\; u^{(j)} = v^{(j)}.$$

This implies, in particular, $|u| = |v|$. Let now $<$ be a linear ordering of Γ, and let $<_{\text{lex}}$ denote the induced lexicographic ordering. For each $0 \leq k$ we define the

k-length-lexicographic ordering ($<_{k\text{-llex}}$) of Γ^* as

$$u <_{k\text{-llex}} v \overset{\text{def}}{\iff} |u| < |v| \vee \exists i < k: \; u =_i v \wedge u^{(i+1)} <_{\text{lex}} v^{(i+1)}.$$

To give an example let $\{0 < 1 < a < b < c\}$ be an ordered alphabet. In the induced 2-lexicographic ordering we have, *e.g.*,

$$a0a0a0a0 <_{2\text{-llex}} a1a1b1a1 <_{2\text{-llex}} a0a0b0b1 <_{2\text{-llex}} a0a1b1b0 <_{2\text{-llex}} b0a0b1b1$$

and $a0a0b0b1 =_1 a0a1b1b0$ holds.

Definition 4.1 (k-lexicographic words)**.** An ω-word $w \in \Sigma^\omega$ is *k-lexicographic* (short: *k-lex*) if there is an automatic presentation $(D, <_{k\text{-llex}}, \{P_a\}_{a\in\Sigma})$ of the associated word structure W_w. For each k, the class of k-lexicographic words is denoted $\mathcal{W}_k$, and we also let $\mathcal{W} = \bigcup_k \mathcal{W}_k$.

Observe that the 0-lexicographic ordering is just the ordering of words according to their length. Therefore, by definition, the domain of a 0-lex presentation has to be *thin*, *i.e.* containing at most one word of each length. All such presentations are easily seen to be equivalent to one over a unary alphabet (the length-preserving morphism mapping each letter to 1 is a rational projection to 1^* that is length preserving, hence synchronous rational [27]). Thus, $\mathcal{W}_0$ is precisely the class of ultimately periodic words.

Proposition 4.2. $\mathcal{W}_0$ is the class of ultimately periodic words.

Further, it is not hard to see, that an ω-word is 1-lex iff it is morphic. For the sake of completeness and to illustrate the techniques of Section 5 in this simple case we present a compact proof of this fact, which has appeared in [45].

To each morphism $\varphi \in \mathsf{Hom}(\Sigma^*, \Sigma^*)$ with $|\varphi| = l$ we associate its *index transition system* $\mathcal{I}_\varphi = (\Sigma, [l], \delta)$ where $\delta(a, i) = \varphi(a)[i]$ for every $i < |\varphi(a)|$ and undefined otherwise. For each $a \in \Sigma$ considered as the initial state, the DFA $(\mathcal{I}_\varphi, a, \Sigma)$ accepts the set $I(a) = I_\varphi(a)$ of valid sequences of indices starting from a. Applying φ n times to a gives the word

$$\varphi^n(a) = \prod_{x\in I(a)\cap[l]^n}^{lex} \delta^*(a, x) \tag{1}$$

where x is meant to run through all valid sequences of indices of length n in lexicographic order. Thus $\varphi^n(a)$ is the sequence of labels of the nth level of the tree unfolding of $\mathcal{I}_\varphi$ from a.

Conversely, given a linear ordering $a_0 < a_1 < \ldots < a_s$ of Σ we associate to each DTS $\mathcal{T} = (Q, \Sigma, \delta)$ its *transition morphism* $\tau = \tau_{\mathcal{T}} \in \mathsf{Hom}(Q^*, Q^*)$ defined as $\tau(q) = \delta(q, a_{i_1})\delta(q, a_{i_2}) \ldots \delta(q, a_{i_k})$ where $a_{i_1} < a_{i_2} < \ldots < a_{i_k}$ are precisely those symbols for which a transition from q is defined. Just as in (1) applying τ n times to some q results in $\tau^n(q) = \prod_{w\in L(\mathcal{T},q,Q)\cap\Sigma^n}^{lex} \delta^*(q, w)$, where w runs, in lexicographic order, through all words of length n, which are labels of some path in $\mathcal{T}$ starting from q. Thus $\tau^n_{\mathcal{T}}(q)$ is the sequence of labels of the nth level of the tree unfolding of $\mathcal{T}$ from q.

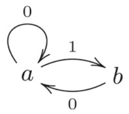
Proposition 4.3 ([45])**.** $\mathcal{W}_1$ is the class of morphic words.

Proof. Let $\tau \in \mathsf{Hom}(\Gamma^*, \Gamma^*)$ be prolongable on a and consider its index transition system $\mathcal{I} = \mathcal{I}_\tau$. It is clear from our previous observations that the language $L(\mathcal{I}, a, \Gamma)$ recognized by $\mathcal{I}$ with all states final and a as its initial state provides, equipped with the prefix-ordering, an automatic presentation of the tree unfolding $\mathcal{T} = \mathcal{T}_{\mathcal{I},a}$ of $\mathcal{I}$ from the initial state a. As also remarked, $\tau^n(a)$ is precisely the word one obtains by reading the nth level of $\mathcal{T}$ from "left to right", *i.e.* in lexicographic order. Also note that τ being prolongable on a, $\mathcal{I}_\tau$ contains a transition $a\overset{0}{\mapsto}a$, therefore the subtree of $\mathcal{T}$ rooted at 0 is isomorphic to the whole tree. Let $\tau(a) = au$ for some $u = u_1 \dots u_t \in \Gamma^*$ and let $\mathcal{U}_i$ be the subtree rooted at $0 < i \leq t$. Then $\tau^{n+1}(a) = au\tau(u)\cdots\tau^n(u) = \tau^n(a) \cdot \tau^n(u)$ and $\mathcal{T} \cong a(\mathcal{T}, \mathcal{U}_1, \dots, \mathcal{U}_t)$. To obtain a length-lexicographic presentation of $\tau^\omega(a)$ we dispense with the subtree rooted at 0 so that the levels of the remaining regular tree $a(\mathcal{U}_1, \dots, \mathcal{U}_t)$ correspond to the increments $\tau^n(u)$ between iterations of τ. We have thus shown that $D = L(\mathcal{I}_\tau, a, \Gamma) \setminus 0[|\tau|]^*$ and $P_c = L(\mathcal{I}_\tau, a, c) \setminus 0[|\tau|]^*$ for each $c \in \Gamma$ together with the natural length-lexicographic ordering provide an automatic presentation of $\tau^\omega(a)$. To give a lexicographic presentation of $w = \sigma(\tau^\omega(a))$ where $\sigma \in \mathsf{Hom}(\Gamma^*, \Sigma^*)$ we set $D' = \{xi \mid c \in \Gamma, x \in P_c, i < |\sigma(c)|\}$ and $P_b = \{xi \mid c \in \Gamma, x \in P_c, \sigma(c)[i] = b\}$ for each $b \in \Sigma$.

Conversely, given a lexicographic presentation $(\mathcal{A}_D, <_{\mathrm{lex}}, \{\mathcal{A}_{P_a}\}_{a\in\Sigma})$ of some w consider the product automaton $\mathcal{A} = \prod_{a\in\Sigma} \mathcal{A}_{P_a}$. Let $\tau = \tau_{\mathcal{A}}$ be the associated transition morphism, and let us define $\sigma \in \mathsf{Hom}(Q(\mathcal{A})^*, \Sigma^*)$ by stipulating that $\sigma(\vec{q}) = a$ whenever the ath component of $\vec{q}$ is an accepting state of $\mathcal{A}_{P_a}$ (assuming the $\mathcal{A}_{P_a}$ are trim a is clearly uniquely determined) and $\sigma(\vec{q}) = \varepsilon$ when no such a exists. To ensure that τ is prolongable, we introduce a new symbol $q_{\mathrm{init}} \notin Q(\mathcal{A})$ and set $\tau(q_{\mathrm{init}}) = q_{\mathrm{init}}\tau(\vec{q}_0)$ and $\sigma(q_{\mathrm{init}}) = \sigma(\vec{q}_0)$, where $\vec{q}_0$ is the initial state of $\mathcal{A}$. We leave it to the reader to check that $w = \sigma(\tau^\omega(\vec{q}_0))$. □

Example 4.4. Recall the Fibonacci word generated by the morphism $\phi : a \mapsto ab, b \mapsto a$ of Example 3.2. The index transition system of ϕ,

$\mathcal{I}$:



accepts, with a being initial and both states being final, the language $\{0,1\}^*\setminus\{0,1\}^* 11\{0,1\}^*$ of Fibonacci numerals *with* leading zeros. The construction of the proof of Proposition 4.3 dispenses precisely with those numerals starting with a zero, thus producing an injective presentation. Length-lexicographically ordered, the first few numerals are $\varepsilon, 1, 10$, $100, 101, 1000$, 1001, 1010, 10000, 10001, $\dots$ with 10^n representing the nth Fibonacci number.

Let us now give an example of a 2-lexicographic word, which is not morphic.

Example 4.5. Consider the Champernowne word $s = 12345678910111213\ldots$ (also called Smarandache sequence) obtained by concatenating all decimal numerals (without leading zeros) in ascending, *i.e.* length-lexicographic order. To give a natural 2-lex presentation of W_s we use words $\otimes_2(x^{(1)}, x^{(2)})$ such that $x^{(1)}$ is a decimal numeral (not starting with a zero) and $x^{(2)} \in 1^*01^*$. We use the single 0 in $x^{(2)}$ to mark a position within $x^{(1)}$. For each digit $d \in [10]$ we can thus define the unary predicate P_d as $([10]1)^*d0([10]1)^*\backslash 0[10]^*$.

We close this section with two simple but useful observations.

Lemma 4.6 (Normal form lemma)**.** Let $1 < k \in \mathbb{N}$. Each k-lexicographic presentation $\mathfrak{d} = (D, <_{k\text{-llex}})$ of $(\mathbb{N}, <)$ over an alphabet Σ is equivalent to one $\mathfrak{d}' = (D', <_{k\text{-llex}})$ over some Γ such that $D' \subseteq (\Gamma^k)^*$. In fact, one can choose $\Gamma = \{0, 1\}$ in the above.

Proof. Let first $\Gamma = \Sigma \uplus \{\widehat{0}, \ldots, \widehat{k-1}, \diamond\}$ endowed with the ordering $\diamond < \widehat{0} < \ldots < \widehat{k-1} < a_1 < \ldots < a_s$ where $a_1 < \ldots < a_s$ is the ordering of Σ used in the presentation $\mathfrak{d}$. We define the translation $t : \Sigma^* \to (\Gamma^k)^*$ padding each word x to $t(x) = \widehat{l}\diamond^{k-1} x \diamond^{k-l}$ where $l = |x| \bmod k$. Observe that the moduli of the positions of symbols of x are preserved in the process of this coding, *i.e.* $t(x)^{(i)} = \alpha x^{(i)} \diamond$ with α being $\widehat{l}$ for $i = 0$ and $\diamond$ otherwise. Consequently $x <_{k\text{-llex}} y$ iff $t(x) <_{k\text{-llex}} t(y)$ in the orderings induced by that of the symbols. Since t is a synchronized rational bijection $\mathfrak{d}' = (t(D), <_{k\text{-llex}})$ is, by Theorem 2.4, equivalent to $\mathfrak{d}$.

Finally, to obtain an equivalent presentation over $\{0, 1\}$ take any binary coding $a \mapsto b_0 \ldots b_{l-1}$ of the symbols $a \in \Gamma$ uniformly of length l and such that $a < a'$ iff $b_0 \ldots b_{l-1} <_{1\text{-llex}} b'_0 \ldots b'_{l-1}$. Extend this into a coding of blocks of k consecutive symbols as $a^0 \ldots a^{k-1} \mapsto b_0^0 \ldots b_0^{k-1} \ldots b_{l-1}^0 \ldots b_{l-1}^{k-1}$, and extend this homomorphically to $(\Gamma^k)^*$. Due to the uniformity requirement, this translation is semi-synchronous, and further it respects the k-lexicographic ordering. Hence, by Theorem 2.4, it yields an equivalent k-lexicographic presentation. □

The important implication of the presentation in normalform being equivalent to the original one is that one is canonical if and only if the other one is. Therefore, in order to establish canonicity of all k-lexicographic presentations it will be sufficient to treat k-lexicographic presentations in normalform, as we shall do.

Proposition 4.7 (Closure under homomorphic mappings)**.** The class of automatically presentable ω-words is closed under homomorphic mappings. In particular, if w is k-lexicographic, then so is $h(w)$ for every homomorphism h.

Proof. The idea is to append each word $x \in P_a$ of a given presentation of w indexing a symbol a by $|h(a)|$ many appropriately chosen suffixes $u_{a,i}$ with $i < |h(a)|$. For instance, one can take $u_{a,i} = \#i$ where $\#$ is a new symbol. Then $P'_c = \{x\#i \mid \bigvee_a x \in P_a \wedge h(a)[i] = c\}$ for all letters c in the image alphabet of h. For k-lexicographic presentations, wlog. in normalform, we can simply choose $|u_{a,i}| = 0^{k-1}i$, though this will typically not yield a presentation in normalform. □

5. Canonicity, closure and decidability

This section is devoted to proving our main result, Theorem 5.3, stating that all k-lexicographic presentations are canonical. Recall that this asserts that for every k-lexicographically presented ω-word w and every regular language L a synchronous automaton can be constructed that recognizes pairs of words representing endpoints of precisely those factors of w, which belong to L. The proof of this theorem proceeds by induction on k and is segmented into three layers. As the basic machinery at the bottom layer stands a "higher-order" automaton transformation on which the entire construction rests. We present it in Section 5.1. On the intermediate layer we have the Contraction Lemma 5.2, which is the heart of the inductive argument. At the top of the construction the pieces are put together in the proof of Theorem 5.3.

5.1. Technical tools: automata transformations

Consider a finite deterministic and complete transition system $\mathcal{T} = (Q, \Sigma, \delta)$ and the associated pair (M, φ) consisting of the finite monoid $M = (Q \to Q, \circ)$ and the homomorphism $\varphi \in \mathsf{Hom}(\Sigma^*, M)$ induced by δ. We call $\mathsf{Hom}(\Sigma^*, M)$ the *derived state space* and denote it by $Q^{(\Sigma)}$. Furthermore, we call $M^{(\Sigma)} = Q^{(\Sigma)} \to Q^{(\Sigma)}$ the monoid of *automata transformations*. Note that both $Q^{(\Sigma)}$ and $M^{(\Sigma)}$ are finite. This terminology is justified by the fact that $Q^{(\Sigma)} = \mathsf{Hom}(\Sigma^*, M)$ is in essence the set of all Σ-labelled DTS's over the state space Q, hence $M^{(\Sigma)}$ is indeed the monoid of all transformations of such transition systems.

A particular submonoid of $M^{(\Sigma)}$ that interests us is that of *inverse homomorphic transformations* $H^{(\Sigma)}$ defined as follows. Consider a homomorphism $h \in \mathsf{Hom}(\Sigma^*, \Sigma^*)$. We can associate to h the element $\Phi(h)$ of $M^{(\Sigma)}$ defined as $(Q^{(\Sigma)} \ni \chi \mapsto \chi \circ h)$. It can be readily seen that Φ is a monoid homomorphism from $\mathsf{Hom}(\Sigma^*, \Sigma^*)$ to $M^{(\Sigma)}$, therefore $H^{(\Sigma)} \overset{\text{def}}{=} \Phi(\mathsf{Hom}(\Sigma^*, \Sigma^*))$ is a submonoid of $M^{(\Sigma)}$. In terms of automata transformations this amounts to mapping a transition function δ to δ' such that $\delta'(q, a) = q'$ whenever $\delta^*(q, h(a))$, where δ^* denotes as usual the extension of δ to all words over Σ. We let $h^{-1}(\mathcal{T})$ denote the transition system (Q, Σ, δ'). Thus, for every $q, q' \in Q$ and $w \in \Sigma^*$ there is a path in $h^{-1}(\mathcal{T})$ labelled w from q to q' iff there is a path in $\mathcal{T}$ labelled $h(w)$ from q to q'.

Consider a finite alphabet Θ and a mapping $\vartheta : \Theta \to \mathsf{Hom}(\Sigma^*, \Sigma^*)$. We extend ϑ to Θ^* according to the rule

$$\vartheta(x \cdot x') = \vartheta(x') \circ \vartheta(x) \tag{2}$$

which ensures that $\Phi_\vartheta = \Phi \circ \vartheta$ is a homomorphism from Θ^* to $H^{(\Sigma)}$:

$$\begin{aligned}
\Phi(\vartheta(x \cdot x'))(\chi) &= \Phi(\vartheta(x') \circ \vartheta(x))(\chi) = \chi \circ \vartheta(x') \circ \vartheta(x) \\
&= \Phi(\vartheta(x'))(\chi) \circ \vartheta(x) = \Phi(\vartheta(x))(\Phi(\vartheta(x'))(\chi)) \\
&= (\Phi(\vartheta(x)) \circ \Phi(\vartheta(x')))(\chi).
\end{aligned}$$

Therefore, the pair $(H^{(\Sigma)}, \Phi_\vartheta)$ represents, in accordance with our initial correspondence, a Θ-labelled finite transition system with state space $Q^{(\Sigma)}$. Elements of Θ^* can thus be seen as words over Θ, or, *via* ϑ as homomorphisms from Σ^* to Σ^*, or, *via* Φ_ϑ, as transformations of Σ-labelled transition systems. Given a word $w \in \Sigma^*$ and a monoid element $m \in M$, we are interested in the following subset of Θ^*.

$$L_{\mathcal{T},w,m,\vartheta} = \{x \in \Theta^* \mid \text{the state transformation induced by } w \text{ in } \vartheta(x)^{-1}(\mathcal{T}) \text{ is } m\}.$$

Lemma 5.1 (Higher-Order Regularity (HOR) Lemma)**.** For every $\mathcal{T} = (Q, \Sigma, \delta)$ with associated (M, φ) and for every $w \in \Sigma^*$, $m \in M$, and every Θ and ϑ as above we can construct an automaton recognizing $L_{\mathcal{T},w,m,\vartheta}$.

Proof. Observe that we can write $L_{\mathcal{T},w,m,\vartheta}$ equivalently as

$$\begin{aligned}
L_{\mathcal{T},w,m,\vartheta} &\overset{\text{def}}{=} \{x \in \Theta^* \mid \text{the state transformation induced by } w \text{ in } \vartheta(x)^{-1}(\mathcal{T}) \text{ is } m\}\\
&= \{x \in \Theta^* \mid \text{the state transformation induced by } \vartheta(x)(w) \text{ in } \mathcal{T} \text{ is } m\}\\
&= \{x \in \Theta^* \mid \varphi(\vartheta(x)(w)) = m\}\\
&= \{x \in \Theta^* \mid \Phi(\vartheta(x))(\varphi)(w) = m\}\\
&= \{x \in \Theta^* \mid \Phi(\vartheta(x)) \in H_{m,\varphi,w}\}\\
&= \Phi_\vartheta^{-1}(H_{m,\varphi,w})
\end{aligned}$$

where $H_{m,\varphi,w} = \{\xi = \Phi(h) \in H^{(\Sigma)} \mid \xi(\varphi)(w) = \varphi(h(w)) = m\}$. Hence, $L_{\mathcal{T},w,m,\vartheta}$ is recognized by the subset $H_{m,\varphi,w}$ of the finite monoid $H^{(\Sigma)}$ under the morphism Φ_ϑ.

Alternatively, we can describe the automaton recognizing $L_{\mathcal{T},w,m,\vartheta}$ as one having as its states all deterministic transition systems having the same set of states as $\mathcal{T}$ and for every $h \in \Theta$ and DTS $\mathcal{T}'$ a transition labelled h from $\mathcal{T}'$ to $h^{-1}(\mathcal{T}')$. The initial state is $\mathcal{T}$ and a transition system $\mathcal{T}'$ is accepting if with the corresponding homomorphism $\varphi_{\mathcal{T}'}$ we have $\varphi_{\mathcal{T}'}(w) = m$. □

Recall that an *HDT0L system* is a collection $H = (h, h_1, \ldots, h_r, w)$, where $h_1, \ldots, h_r \in \mathsf{Hom}(A^*, A^*)$, $h \in \mathsf{Hom}(A^*, B^*)$ and $w \in A^*$ [32,47]. Given H let $\Theta = \{h_1, \ldots, h_r\}$ and ϑ be defined as above. The *HDT0L mapping* $t_H : \Theta^* \to A^*$ associated to H is defined, just like above, by $t_H(x) = h(\vartheta(x)(w)) = \Phi_\vartheta(x)(h)(w)$ for all $x \in \Theta^*$. A mapping of this kind is said to be (effectively) *continuous* if $t_H^{-1}(L)$ is rational for every rational language L (and an automaton recognizing $t_H^{-1}(L)$ can be constructed) [42]. Now if L is recognized by the morphism φ then an automaton/morphism recognizing $t_H^{-1}(L)$ can be constructed as in the HOR Lemma. Hence, in essence, our HOR Lemma asserts that HDT0L mappings are effectively continuous.

5.2. Canonicity of k-lexicographic presentations

Let a $(k+1)$-lex presentation $\mathfrak{d} = (D, <_{(k+1)\text{-llex}}, \{\mathcal{A}_a\}_{a\in\Sigma})$ of $w \in \Sigma^\omega$ in normal form over the alphabet Γ together with the bijective coordinate function $\nu : D \to \mathbb{N}$ as well as a homomorphism $\psi \in \mathsf{Hom}(\Sigma^*, M)$ into a finite monoid

M be given. We associate to $\mathfrak{d}$ the DFA $\mathcal{A}_{\mathfrak{d}} = \prod_{a\in\Sigma} \overline{\mathcal{A}_a}$ consisting of the DTS $\mathcal{T}_{\mathfrak{d}} = (Q_{\mathfrak{d}}, \Gamma, \delta_{\mathfrak{d}})$ and having initial state $\vec{q}_0$. Recall that $\overline{\mathcal{A}}$ denotes the completion of the automaton $\mathcal{A}$ as defined on page 421. Further let $\sigma_{\mathfrak{d}} \in \mathsf{Hom}(Q_{\mathfrak{d}}^*, \Sigma^*)$ be such that $\sigma_{\mathfrak{d}}(\vec{q}) = a$ whenever the ath component of $\vec{q}$ is in an accepting state (in which case a is uniquely determined) and $\sigma_{\mathfrak{d}}(\vec{q}) = \varepsilon$ otherwise. Finally, we set $w_{\mathfrak{d}} = \prod_{x\in\Gamma^*}^{<_{(k+1)\text{-llex}}} \delta_{\mathfrak{d}}^*(\vec{q}_0, x) \in Q_{\mathfrak{d}}^{\omega}$, which makes sense since $\mathcal{A}_{\mathfrak{d}}$ is complete. Clearly, $w = \sigma_{\mathfrak{d}}(w_{\mathfrak{d}})$

For every $x = \otimes_{k+1}(x^{(1)}, \ldots, x^{(k+1)})$ let $x' = \otimes_k(x^{(1)}, \ldots, x^{(k)})$ be the projection of x onto its first k splitting components when $k > 0$ and let $x' = x^{(0)} = 1^{|x|}$ when $k = 0$. We define $D' = \{x' \mid x \in D\}$ as the point-wise projection of D. The equivalence $=_k$ partitions the set D of indices into consecutive intervals. Let $c(x')$ denote the interval containing x, *i.e.* $c(x') = \{y \in D \mid y' = x'\}$, and consider the factorization of w according to such intervals.

$$w = \prod_{x'\in D'}^{<_{k\text{-llex}}} w[c(x')].$$

The *contraction* (compare with that of [25]) of w wrt. $\mathfrak{d}$ and ψ is the ω-word

$$c_{\mathfrak{d}}^{\psi}(w) = \prod_{x'\in D'}^{<_{k\text{-llex}}} \psi(w[c(x')]) \quad \in M^{\omega}$$

indexed by elements of D' ordered according to $<_{k\text{-llex}}$. We can prove that $c_{\mathfrak{d}}^{\psi}(w)$ is in fact automatically presentable over $(D', <_{k\text{-llex}})$.

Lemma 5.2 (Contraction Lemma)**.** Let $\mathfrak{d} = (D, <_{k+1\text{-llex}}, \{\mathcal{A}_a\}_{a\in\Sigma})$ be a $(k+1)$-lex presentation with coordinate function ν of the word structure of an ω-word $w \in \Sigma^{\omega}$. Then for every finite monoid M, every $\psi \in \mathsf{Hom}(\Sigma^*, M)$ and for each $m \in M$ the following relations are regular.

$$\begin{array}{rcl} B'_m & = & \{(x,y) \in D^2 \mid x \leq_{k+1\text{-llex}} y \wedge x =_k y \wedge \psi(w[x,y]) = m\} \\ P'_m & = & \{x' \in D' \mid \psi(w[c(x')]) = m\} \end{array}$$

whence, $(D', <_{k\text{-llex}}, \{P'_m\}_{m\in M})$ is a k-lexicographic presentation of $c_{\mathfrak{d}}^{\psi}(w)$.

Proof. We are going to employ the machinery introduced in Section 5.1. The crucial observation hereto is that the transduction $x' \mapsto w[c(x')]$ is an HDT0L mapping. In order to make this more precise and to apply the HOR Lemma 5.1 we first generalize the notion of transition morphisms (*cf.* Prop. 4.3).

Wlog. the ordered alphabet Γ of the presentation $\mathfrak{d}$ is $[t] = 0 < 1 < \ldots < t-1$. Let $Q = \{q, \ulcorner q, q\urcorner, \ulcorner q\urcorner \mid q \in Q_{\mathfrak{d}}\}$ and $\pi : Q \to Q_{\mathfrak{d}}$ be the projection forgetting the markers and just keeping the states. We define the mapping $\beta : ([t]^k([t] \times [t]))^* \to$

$\mathsf{Hom}(Q^*, Q^*)$ *via* homomorphic extension as in (2) while stipulating that

$$\begin{array}{lcl}
\beta_{u(i,j)}(q) & = & \delta_{\mathfrak{d}}^*(q, u0)\delta_{\mathfrak{d}}^*(q, u1)\dots\delta_{\mathfrak{d}}^*(q, u(t-1)) \\
\beta_{u(i,j)}(\ulcorner q) & = & \ulcorner\delta_{\mathfrak{d}}^*(q, ui)\,\delta_{\mathfrak{d}}^*(q, u(i+1))\dots\delta_{\mathfrak{d}}^*(q, u(t-1)) \\
\beta_{u(i,j)}(q\urcorner) & = & \delta_{\mathfrak{d}}^*(q, u0)\dots\delta_{\mathfrak{d}}^*(q, u(j-1))\,\delta_{\mathfrak{d}}^*(q, uj)\urcorner \\
\beta_{u(i,j)}(\ulcorner q\urcorner) & = & \begin{cases} \varepsilon & (i > j) \\ \ulcorner\delta_{\mathfrak{d}}^*(q, ui)\urcorner & (i = j) \\ \ulcorner\delta_{\mathfrak{d}}^*(q, ui)\,\delta_{\mathfrak{d}}^*(q, u(i+1))\dots\delta_{\mathfrak{d}}^*(q, u(j-1))\,\delta_{\mathfrak{d}}^*(q, uj)\urcorner & (i < j) \end{cases}
\end{array}$$

where u ranges over Γ^k and $i, j < t$. Note that β does not introduce any new markers, rather it is merely keeping track of them by applying $\ulcorner$ and $\urcorner$, if at all, only to the first and last symbols, respectively, on the right-hand side. Also note that i and j are only taken into account as delimiters in connection with $\ulcorner$ and $\urcorner$, respectively. We regard β as a mapping from pairs of $=_k$-equivalent words $x, y \in D$. Indeed, each pair (x, y) of words with $x' = y'$ determines a sequence $u_1(i_1, j_1)\dots u_n(i_n, j_n)$, and *vice versa*, such that $x^{(k+1)} = i_1 \dots i_n$, $y^{(k+1)} = j_1 \dots j_n$ and $x' = y' = u_1 \dots u_n$. In accordance with (2) we can thus define $\beta_{x,y}$ as the composition $\beta_{u_n(i_n,j_n)} \circ \cdots \circ \beta_{u_1(i_1,j_1)}$. We further let $\tau_u = \beta_{u(0,t-1)}$. Note that, for $k = 0$, τ_ε is essentially the transition morphism τ associated to $\mathcal{T}_{\mathfrak{d}}$ as defined on page 427. To allow for uniform treatment we set $\tau_{1^n} = {\tau_\varepsilon}^n$ when $k = 0$.

Claim. *For all $k \in \mathbb{N}$ and $x, y \in (\Gamma^{k+1})^*$ such that $x' = y'$ and $x \leq_{k+1\text{-}llex} y$:*

$$\pi(\beta_{x,y}(\ulcorner\vec{q}\urcorner) \quad = \quad \textstyle\prod_{z=x}^{y} \delta_{\mathfrak{d}}^*(\vec{q}, z)$$

where the concatenation product is taken over the values of z in the $(k+1)$-lexicographic ordering. Consequently, when in addition $x, y \in D$ then we have

$$\begin{array}{lcl}
\sigma_{\mathfrak{d}}\pi(\beta_{x,y}(\ulcorner\vec{q}_0\urcorner)) & = & w[x, y] \\
\sigma_{\mathfrak{d}}\pi(\tau_{x'}(\ulcorner\vec{q}_0\urcorner)) & = & w[c(x')].
\end{array}$$

By the above claim we know that $\psi(w[x, y]) = \psi(\sigma_{\mathfrak{d}}(\pi(\beta_{x,y}(\ulcorner\vec{q}_0\urcorner)))$ and that $\psi(w[c(x')]) = \psi(\sigma_{\mathfrak{d}}(\pi(\tau_{x'}(\ulcorner\vec{q}_0\urcorner)))$. Recall that $\beta_{x,y}$ was defined as $\beta_{u_n(i_n,j_n)} \circ \cdots \circ \beta_{u_1(i_1,j_1)}$ for all $x' = y' = u_1 \dots u_n$ with $u_i \in [t]^k$ and $x^{(k+1)} = i_1 \dots i_n$, $y^{(k+1)} = j_1 \dots j_n$. Similarly, $\tau_{x'} = \tau_{u_n} \circ \cdots \circ \tau_{u_1}$. The results are established by applying the HOR Lemma 5.1 with $\varphi = \psi \circ \sigma_{\mathfrak{d}} \circ \pi$ and $\Theta = [t]^k([t] \times [t])$, $\vartheta_{x \otimes y} = \beta_{x,y}$ in the first case, respectively with $\Theta = [t]^k$, $\vartheta_{x'} = \tau_{x'}$ in the second case. □

In particular, the contraction of a morphic word wrt. any given lexicographic presentation and any given morphism into a finite monoid is an ultimately periodic sequence. This is already sufficient to yield MSO decidability of morphic words, and is essentially the proof given in [17]. Obviously, by iterating this contraction process starting from any given k-lex presentation of an ω-word we arrive after (at most) k contractions, at an ultimately periodic sequence. It is now easy to use this fact to prove MSO decidability of k-lexicographic words. However, we aim for the stronger canonicity result.

Main Theorem 5.3 (Canonicity of k-lex presentations)**.** *All k-lexicographic presentations are canonical.*

Proof. The proof is by induction on k, the base case being clear. For the induction step, we consider a $k+1$-lex presentation. Observe that if two $k+1$-lex presentations of the same ω-word are equivalent, then one is canonical iff the other one is. Therefore, by the Normal Form Lemma, it is sufficient to provide a proof for $k+1$-lex presentations in normal form. So let $\mathfrak{d} = (D, <_{k+1\text{-llex}}, \{P_a\}_{a\in\Sigma})$ be a $k+1$-lex presentation in normal form of an ω-word $w \in \Sigma^\omega$. Let a morphism $\psi \in \mathsf{Hom}(\Sigma^*, M)$ into a finite monoid M be given. We need to construct automata deciding, given words $x, y \in D$ with $x \leq_{k+1\text{-llex}} y$, whether $\psi(w[x,y]) = m$. There are two cases. If $x' = y'$ then we simply verify $(x,y) \in B'_m$ as in the Contraction Lemma. When on the other hand $x' <_{k\text{-llex}} y'$ then we partition the interval $x \leq_{k+1\text{-llex}} z \leq_{k+1\text{-llex}} y$ into three segments according to whether $x' = z'$, $x' <_{k\text{-llex}} z' <_{k\text{-llex}} y'$ or $z' = y'$, *i.e.* consider the factors $w[x,\hat{x}]$, $w[\{z \in D \mid x' <_{k\text{-llex}} z' <_{k\text{-llex}} y'\}]$ and $w[\hat{y}, y]$, where $\hat{x}$ is the greatest element of $c(x')$ with respect to $<_{k+1\text{-llex}}$ and similarly $\hat{y}$ is the least element of $c(y')$. Note that both $\hat{x}$ and $\hat{y}$ are first-order definable using $<_{k\text{-llex}}$ and $=_k$, hence automaton computable from x, respectively from y. We can therefore compute $\psi(w[x,\hat{x}])$ as well as $\psi(w[\hat{y},y])$ by an automaton simultaneously verifying B'_m for both pairs $(x,\hat{x})$ and $(\hat{y},y)$ for all $m \in M$.

It remains to show that the value of the central segment is also automaton computable. By the Contraction Lemma we know that $\mathfrak{d}' = (D', <_{k\text{-llex}}, \{P'_m\}_{m\in M})$ is a k-lex presentation of $c^{\psi}_{\mathfrak{d}}(w)$. Thus, by the induction hypothesis, $\mathfrak{d}'$ is canonical. We use this fact to compute the value of the central segment. To this end, we employ the *multiplier morphism* $\mu_M \in \mathsf{Hom}(M^*, M)$ defined by stipulating that $\mu_M(m) = m$ for all $m \in M$. Let ν' denote the coordinate mapping associated to $\mathfrak{d}'$. By definition of a contraction $\psi(w[\nu(c(z'))]) = c^{\psi}_{\mathfrak{d}}(w)[\nu'(z')]$, therefore the value of the central segment $\psi(w[\{z \in D \mid x' <_{k\text{-llex}} z' <_{k\text{-llex}} y'\}])$ can be written as $\mu_M(c^{\psi}_{\mathfrak{d}}(w)(x',y'))$, which is by canonicity of $\mathfrak{d}'$ automaton computable. □

Corollary 5.4 (MSO decidability)**.** *The* MSO *theory of the word structure W_w associated to a k-lex word $w \in \mathcal{W}$ is decidable.*

MSO interpretations are usually understood to be one-dimensional. We use the notation $\leq^{\mathcal{I}}_{\mathsf{mdMSO}}$ to stress that $\mathcal{I}$ might be multi-dimensional. Further, we say that a tuple $(\varphi(x), \{\varphi_b(x)\}_{b\in\Gamma})$ of MSO formulas, together with the formula $\varphi_<(x,y) = x < y$, form a *restricted* MSO *interpretation* $\mathcal{I}$ (the restriction being that $\mathcal{I}$ can only redefine the coloring, but not $<$) of a finite or infinite word structure $W_{w'} \leq^{\mathcal{I}}_{\mathsf{rMSO}} W_w$. From Theorem 5.3 and Theorem 2.2 we conclude the next corollaries.

Corollary 5.5 (Closure under MSO interpretations)**.** *Let w be a k-lexicographic word. For every structure $\mathfrak{A}$ and word w' we have*

1. $\mathfrak{A} \leq_{\mathrm{md}\mathsf{MSO}} W_w \Longrightarrow \mathfrak{A}$ *is automatic,*
2. $W_{w'} \leq_{\mathsf{rMSO}} W_w \Longrightarrow W_{w'}$ *is k-lexicographic.*

Corollary 5.6 (Closure under d.g.s.m. mappings)**.** *For each $k \in \mathbb{N}$ the class $\mathcal{W}_k$ is closed under deterministic generalized sequential mappings.*

Proof. Let S be a deterministic sequential transducer. Wlog. we may assume that S stores in its every state the last symbol read. With this assumption the image $S(w)$ of a word w under S can be obtained by a homomorphic mapping of the run of S over w. The homomorphism corresponding to the output function of the sequential transducer S. The run of S on w is of course rMSO interpretable in W_w. Thus for each $w \in \mathcal{W}_k$ the run of S over w is in $\mathcal{W}_k$ by Corollary 5.5 and therefore also $S(w) \in \mathcal{W}_k$ by Proposition 4.7. □

As an example of what can be interpreted in a word consider the following.

Theorem 5.7 (Automatic equivalence structures)**.** *Consider $\mathfrak{A} = (A, E)$ with E an equivalence relation on a countably infinite set A having only finite equivalence classes. Assume further that for each n there are $f(n) \in \mathbb{N}$ many equivalence classes of size n.*
Then $\mathfrak{A} \in$ AUTSTR if and only if there is a 2-lex word $w = 0^{m_0}10^{m_1}10^{m_2}1\ldots$ such that $f(n) = |\{i \mid m_i = n\}|$, in which case $\mathfrak{A} \leq^{\mathcal{I}}_{\mathsf{FO}} W_w$ for a fixed one-dimensional FO-interpretation $\mathcal{I}$, also implying that $\mathrm{Th}_{\mathsf{MSO}}(\mathfrak{A})$ is decidable.

Proof. For the "if" direction, the interpretation in question is $\mathcal{I} = (\varphi_A(x), \varphi_E(x,y))$ with $\varphi_A(x) = P_0(x)$ and $\varphi_E(x,y) = \varphi_A(x) \wedge \varphi_A(y) \wedge \forall z(x < z < y \vee y < z < x \rightarrow P_0(z))$. It is now easy to check that $\mathcal{I}(W_w)$ is indeed isomorphic to $\mathfrak{A}$ and is thus, by Theorem 2.2 or by Corollary 5.5, automatic.

For the "only if" direction we construct, given an automatic presentation (L_A, L_E) of $\mathfrak{A}$, an automatic presentation of a binary word with the claimed property.

First observe that since all equivalence classes of $\mathfrak{A}$ are finite, in other words E is a locally finite relation, there is a constant C such that $||x| - |y|| < C$ for all $x, y \in L_A$ with $(x,y) \in L_E$. This can be verified using a standard pumping argument (*cf. e.g.* [10], Prop. 6.1) and does not require the classes to be globally bounded. We can therefore easily construct by padding an equivalent presentation of $\mathfrak{A}$ in which $|x| = |y|$ holds for all x and y representing equivalent elements. We shall now assume this holds.

Let Γ be the alphabet of the presentation of $\mathfrak{A}$. Wlog. $\Gamma = \{0, \ldots, s-1\}$. The alphabet of the presentation of w will be $\Gamma' = \{0, \ldots, s-1, s\}$ ordered naturally. We set $P_0 = \{\otimes_2(x,y) \mid (x,y) \in L_E \wedge \forall (x,z) \in L_E\ x \leq_{\mathrm{lex}} z\}$, $P_1 = \{\otimes_2(x, s^{|x|}) \mid \forall (x,z) \in L_E\ x <_{\mathrm{lex}} z\}$, and $D = P_0 \cup P_1$. It is now clear that $(D, <_{\text{2-llex}}, P_0, P_1)$ is an a.p. as promised. □

6. Hierarchy theorem

It is readily seen, that $\mathcal{W}_k$ is included in $\mathcal{W}_{k+1}$ for each k. Next we show that each $\mathcal{W}_k$ is properly included in the next one by exhibiting ω-words $s_{k+1} \in$

$\mathcal{W}_{k+1} \setminus \mathcal{W}_k$. We call the s_k *stuttering words.* Each s_k is a word over the $(k+1)$-letter alphabet $\{a_0, \ldots, a_k\}$ and is defined as the infinite concatenation product $s_k = \prod_{n=0}^{\infty} s_{k,n}$, where $s_{0,n} = a_0$ and $s_{k+1,n} = (s_{k,n})^{2^n} a_{k+1}$ for every k and n. That is

$$s_k = \prod_{n=0}^{\infty} (\cdots(((a_0^{2^n})a_1)^{2^n})\cdots)^{2^n} a_k.$$

To give an illustration, we write for convenience $a, b, c, d\ldots$ instead of $a_0, a_1, a_2, a_3\ldots$ for small k. The first few stuttering words are

$$\begin{array}{rcl} s_0 & = & a^\omega \\ s_1 & = & abaabaaaaba^8ba^{16}b\ldots \\ s_2 & = & abcaabaabc(aaaab)^4c(a^8b)^8c\ldots \\ s_3 & = & abcd(aabaabc)^2d((aaaab)^4c)^4d((a^8b)^8c)^8d\ldots \\ & \vdots & \end{array}$$

As to the complexity of these stuttering words let us note that s_2 is not a fixed point of any d.g.s.m. mapping [2].

Theorem 6.1 (Hierarchy Theorem)**.** *For each $k \in \mathbb{N}$ we have $s_{k+1} \in \mathcal{W}_{k+1} \backslash \mathcal{W}_k$.*

Proof. We leave it to the reader to give a k-lex presentation of s_k for every k. To show that $s_{k+1} \notin \mathcal{W}_k$ we argue indirectly as follows. Assume that there is a k-lex presentation $(D, <_{k\text{-llex}}, \{P_{a_i}\}_{i \leq k+1})$ of s_{k+1}, and assume it to be in normal form, *i.e.* $D \subseteq (\{0,1\}^k)^*$. Consider for each $i \leq k+1$ the (regular) relations $S_i(x,y)$ consisting of pairs of consecutive words $x, y \in P_{a_i}$, *i.e.* such that there are no occurrences of a_i on intermediate positions. Let automata be given for D, P_{a_i}, and S_i for every $i \leq k+1$ and let C be greater than the maximum of the number of states of any of these automata.

Claim. *For every $i = 1, \ldots, k$ there is a t_i such that for all $N \in \mathbb{N}$ there are $x = \otimes_k(x^{(1)}, \ldots, x^{(k)})$, and $y = \otimes_k(y^{(1)}, \ldots, y^{(k)})$ with $|x| = |y| > N$ and such that $S_i(x,y)$ and $x =_{k-i} y$ (i.e. $x^{(j)} = y^{(j)}$ for all $j \leq k-i$) and that $x^{(k-i+1)}$ and $y^{(k-i+1)}$ differ only on their last t_i bits.*

For $i = 1$ we immediately get a contradiction since for large enough N there are more than 2^{t_1} many a_0's between consecutive a_1's represented by words x and y of length N contrary to the above claim that x and y differ only on the least significant t_1 bits of their least significant components leaving room for at most 2^{t_1} many intermediate positions in the k-lexicographic ordering.

Proof of claim. We start with $i = k$ and proceed inductively in descending order. Values of the t_i will be implicitly given during the proof.

First note that $|v| < |u| + C$ for every $S_{k+1}(u,v)$ because the tail of a longer v could otherwise be pumped up to produce infinitely many would-be S_{k+1}-successors of u when there is but one. Let $n > C \log C$ and let u represent the position of the nth occurrence of a_{k+1} in s_{k+1} and chose v with $S_{k+1}(u,v)$. Then there are 2^n many a_k's distributed evenly between u and v, therefore there must be some $|u| \leq L \leq |v|$ such that there are at least $2^n/C > 2^C$ many $u <_{k\text{-llex}} x <_{k\text{-llex}} v$, $|x| = L$, and $x \in P_{a_k}$.

Consider the ascending sequence of all such x ordered according to $<_{k\text{-llex}}$. Then the first C bits of their first components (these are the most significant bits) are lexicographically non-decreasing. Since there are more than 2^C such x we must find two consecutive ones agreeing on the first C bits of their first components.

Let x and y be such a pair, *i.e.* $|x| = |y| = L$, $S_k(x,y)$ and such that $x^{(1)}$ and $y^{(1)}$ agree on their first C bits. Set $t_k = L - C$. By pumping into the initial segment of length kC of the pair (x,y) (this involves the first C symbols of each component of both x and y) we can obtain arbitrary long x', y' with $S_k(x',y')$ and whose first components may only differ on their last t_i bits. Thus we have established the case $i = k$.

To advance from $i+1$ to i we do the same as above. By the induction hypothesis we have for arbitrary large L two words $u = \otimes_k(u^{(1)},\ldots,u^{(k)})$ and $v = \otimes_k(v^{(1)},\ldots,v^{(k)})$ both of length L such that $S_{i+1}(u,v)$ and having $u^{(j)} = v^{(j)}$ for all $j < k-i$ and $u^{(k-i)}$ and $v^{(k-i)}$ differing only on their last t_{i+1} bits. Choose L large enough to ensure that there are more than 2^{C+t_i} many occurrences of a_i in between these two positions. As x runs through, in the k-length-lexicographic order, all the words representing positions of consecutive a_i's from u to v the last t_i bits of the $(k-i)$th component together with the first C bits of the $(k-i+1)$th component of x is lexicographically non-decreasing. As there are more than 2^{C+t_i} many such x for large enough L we must have two consecutive a_i's on positions represented by some x and y agreeing on their first $(k-i)$ components and on the first C bits of their $(k-i+1)$th components. Thus, by pumping into the initial segment of length kC of the pair (x,y) we obtain arbitrary long x', y' fulfilling the conditions of our claim for i. □

7. Equivalent characterizations

In the previous sections we have been concerned with k-lexicographic presentations of ω-words. Each automatic presentation provides a finite description of an ω-word that is *internal* in the sense that positions within the ω-word are individually named and that their properties and relationships are given in terms of these names chosen. The description of a morphic word *via* two morphisms generating it can, in contrast, be seen as being *external* or generative in nature. Proposition 4.3 demonstrated how to transform a length-lexicographic presentation of a morphic ω-word into an equivalent description in terms of morphisms and *vice versa*.

In this section we generalize both the notion of morphic ω-words and the technique of Proposition 4.3 to each level k providing equivalent external descriptions of k-lexicographic ω-words in the form of iterating morphisms of higher-order stacks of level k. In addition we give equivalent characterizations of k-lexicographic ω-words in terms of MSO-interpretations and deterministic generalized sequential mappings restricted in a certain sense and applied to a fixed ω-word of level $k+1$.

A number of generalizations of morphic words, given as so-called HD0L TAG-systems, have been introduced. These include DGSM TAG-systems, double or triple D0L TAG-systems, etc. [31]. To the knowledge of the author the notion of higher-order morphic words introduced here is new. It generalizes HD0L TAG-systems in a new direction close to the spirit of [50].

We begin with the definition of higher-order stacks. Let Γ be a finite, non-empty stack alphabet. A (level 1) stack is a finite sequence of symbols of Γ, and level $k+1$ stacks are sequences of level k stacks. Additionally, we shall call individual symbols of Γ level 0 stacks. Formally

$$\begin{aligned} \mathsf{Stack}_\Gamma^{(0)} &= \Gamma \\ \mathsf{Stack}_\Gamma^{(k+1)} &= [(\mathsf{Stack}_\Gamma^{(k)})^*] \end{aligned}$$

where '[' and ']' are used to identify the boundaries of lower-level stacks within higher-level ones. Outer most brackets will most often be omitted.

Level k stacks can be viewed as trees of height k having an unbounded number of ordered branches and leaves labelled by elements of Γ. Each leaf, *i.e.* each level 0 element stored in a k-stack γ can be accessed by a vector of k indices $(i_0,\ldots,i_{k-1})$ leading to it. We denote the sequence of "leaves" of a $k+1$-stack γ, taken in the natural ordering, by $\mathsf{leaves}(\gamma)$. In other words, $\mathsf{leaves}(\gamma)$ is obtained from γ by forgetting the brackets.

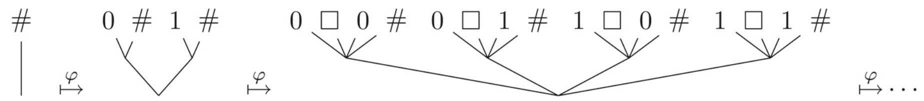
The *concatenation* of two $(k+1)$-stacks $\gamma^{(k+1)} = [\gamma_1^{(k)}\ldots\gamma_s^{(k)}]$ and $\xi^{(k+1)} = [\xi_1^{(k)}\ldots\xi_t^{(k)}]$ is the $(k+1)$-stack $\gamma^{(k+1)}\cdot\xi^{(k+1)} = [\gamma_1^{(k)}\ldots\gamma_s^{(k)}\xi_1^{(k)}\ldots\xi_t^{(k)}]$. Concatenation can also be regarded as operations on trees. For $k>0$ every k-stack $\gamma^{(k)} = [\gamma_0^{(k-1)}\ldots\gamma_{s-1}^{(k-1)}]$ can be written as the concatenation product $\prod_{i=0}^{s-1}[\gamma_i^{(k-1)}]$ and by propagating through all dimensions as

$$\gamma^{(k)} = \prod_{i_0}[\prod_{i_1}[\cdots\prod_{i_{k-1}}[\gamma^{(0)}_{(i_0,\ldots,i_{k-1})}]\cdots]] \tag{3}$$

where the index vector $(i_0,\ldots,i_{k-1})$ runs through all allowed tuples (all branches of length k) in lexicographic fashion.

Definition 7.1 (Morphisms of k-stacks)**.** *Morphisms of k-stacks* over Γ are just k-stacks of actions of Γ. That is, $\mathsf{Hom}_\Gamma^{(k)} = \mathsf{Stack}_{\Gamma\to\Gamma}^{(k)}$, *i.e.* $\mathsf{Hom}_\Gamma^{(0)} = \Gamma\to\Gamma$ and $\mathsf{Hom}_\Gamma^{(k+1)} = [(\mathsf{Hom}_\Gamma^{(k)})^*]$. *Application* is defined inductively as follows.

- $\varphi^{(0)}(\gamma^{(0)})$ is as given;
- for $\varphi^{(k+1)} = [\varphi_1^{(k)}\ldots\varphi_s^{(k)}] \in \mathsf{Hom}_\Gamma^{(k+1)}$ and $\gamma^{(k+1)} = [\gamma_1^{(k)}\ldots\gamma_t^{(k)}] \in \mathsf{Stack}_\Gamma^{(k+1)}$ let $\varphi^{(k+1)}(\gamma^{(k+1)}) = [\varphi_1^{(k)}(\gamma_1^{(k)})\ldots\varphi_s^{(k)}(\gamma_1^{(k)})\cdots\varphi_1^{(k)}(\gamma_t^{(k)})\ldots\varphi_s^{(k)}(\gamma_t^{(k)})]$.



#
0 # 1 #
0 □ 0 # 0 □ 1 # 1 □ 0 # 1 □ 1 #
$\overset{\varphi}{\mapsto}$
$\overset{\varphi}{\mapsto}$
$\overset{\varphi}{\mapsto}\dots$

FIGURE 1. Iteratively applying $\varphi = [\tau\sigma]$ of Example 7.3 to $\gamma = [[\#]]$.

Definition 7.2 (k-morphic words)**.** Let $k \in \mathbb{N}$. An infinite word $w \in \Sigma^\omega$ is k-*morphic* if there is a finite alphabet Γ, an initial k-stack $\gamma^{(k)} = [\cdots[\gamma_0^{(0)}]\cdots] \in \mathsf{Stack}_\Gamma^{(k)}$, a k-morphism $\varphi^{(k)} \in \mathsf{Hom}_\Gamma^{(k)}$ and a terminal homomorphism $h : \Gamma^* \to \Sigma^*$ such that

$$w = h\left(\prod_{n=0}^{\infty} \mathsf{leaves}(\varphi^n(\gamma))\right).$$

Note that our morphisms are *uniform*, *e.g.* $\mathsf{Hom}_\Gamma^{(1)}$ consists of the uniform homomorphisms of Γ^* [2]. To illustrate the workings of morphisms of higher-level stacks consider the following level 2 example generating a relative of the binary Champernowne word (*cf.* Ex. 4.5).

Example 7.3. Consider the initial 2-stack $\gamma = [[\#]]$ and the level 2 morphism $\varphi = [\tau\sigma]$ containing $\tau = [\tau_0\tau_1]$ and $\sigma = [\sigma_0\sigma_1]$ with

$$\tau : \left|\begin{array}{cccc} & & \tau_0 & \tau_1 \\ 0 & \mapsto & 0 & \square \\ 1 & \mapsto & 1 & \square \\ \# & \mapsto & 0 & \# \\ \square & \mapsto & \square & \square \end{array}\right. \qquad \sigma : \left|\begin{array}{cccc} & & \sigma_0 & \sigma_1 \\ 0 & \mapsto & 0 & \square \\ 1 & \mapsto & 1 & \square \\ \# & \mapsto & 1 & \# \\ \square & \mapsto & \square & \square \end{array}\right. .$$

Note that τ is just a complicated way of writing the morphism ($0 \mapsto 0$, $1 \mapsto 1$, $\# \mapsto 0\#$) in our framework as a sequence of 0-morphisms. Padding is needed to compensate for the inherent uniformity in our definition.

The stacks obtained in the first few iterations of φ on γ are depicted as trees in Figure 1. Let further h be the morphism erasing □'s and #'s. Then the 2-morphic word generated by φ on γ and filtered by h, is the concatenation of all finite binary sequences in length-lexicographic order:

$$0\,1\;00\,01\,10\,11\;000\,001\,010\,011\,100\,101\,110\,111\;\ldots$$

Clearly, an infinite word is 0-morphic iff it is *ultimately periodic*, and 1-morphic iff it is *morphic* in the customary sense despite the uniformity restriction on φ, which can be made up for by the choice of h. The next Lemma generalizes (1).

Lemma 7.4 (Iteration Lemma)**.** Consider a k-stack $\gamma = [\cdots[\gamma_0]\cdots] \in \mathsf{Stack}_\Gamma^{(k)}$ and a morphism $\varphi = \varphi^{(k)} = \prod_{j_0}\left[\prod_{j_1}\left[\cdots\prod_{j_{k-1}}\left[\varphi^{(0)}_{j_0\ldots j_{k-1}}\right]\cdots\right]\right] \in \mathsf{Hom}_\Gamma^{(k)}$. Let

[2] This could be remedied by defining $\mathsf{Hom}_\Gamma^{(1)}$ as the set of mappings $\Gamma \to \Gamma \cup \{\varepsilon\}$. Although this would not increase the expressive power of our formalism it could allow a simpler description of certain examples. However, for the sake of compacter proofs we opted for the definition as given. See also Remark 7.6.

B be the set of those words $w = j_0 \dots j_{k-1}$ of length k corresponding to branches of the tree associated to φ, and let $\varphi_u^{(0)} = \varphi_{w_n}^{(0)} \circ \dots \circ \varphi_{w_1}^{(0)}$ for all words $u = w_1 w_2 \cdots w_n \in B^*$. Then, applying φ n times to γ yields

$$\varphi^n(\gamma) = \underbrace{\prod_{u^{(1)}} \Big[\prod_{u^{(2)}} \Big[\cdots \prod_{u^{(k)}} \Big[}_{u=\otimes_k(u^{(1)},\dots,u^{(k)})\in B^n} \quad \varphi_u^{(0)}(\gamma_0) \quad \Big] \cdots \Big] \Big].$$

Consider a regular well-ordering $\prec$ of finite binary words and let $u_0 \prec u_1 \prec u_2 \prec \dots$ be the sequence of words in this ordering. We define the infinite word $w_\prec \in \{0, 1, \#\}^\omega$ as the concatenation of the u_i in ascending order separated by # symbols: $w_\prec = u_0 \# u_1 \# u_2 \# \cdots$. Let $w_{k-\mathrm{llex}}$ be the word thus associated to $<_{k\text{-llex}}$ (restricted to words of length divisible by k). For instance,

$$\begin{aligned} w_{1\text{–llex}} &= \#0\#1\#00\#01\#10\#11\#000\#001\#010\#011\#100\# \dots \\ w_{2\text{–llex}} &= \#00\#01\#10\#11\#0000\#0001\#0100\#0101\#0010\#0011\#0110\#0111 \dots \end{aligned}$$

Further, let $w_{0-\mathrm{llex}} = \#0\#00\#000\# \dots$. It is easy to see that $w_{k-\mathrm{llex}} \in \mathcal{W}_{k+1}$ for all $k \in \mathbb{N}$. We say that a sequential transducer S with input alphabet $\{0, 1, \#\}$ and output alphabet Σ is *#-driven* if it is deterministic and in each transition S produces either no output (*i.e.* the empty string ε) or a single letter output $a \in \Sigma$, but this only on reading a # on the input tape.

Theorem 7.5 (Equivalent Characterizations)**.** *Let Σ be a finite alphabet. For every $k \in \mathbb{N}$ and every ω-word $w \in \Sigma^\omega$ the following are equivalent.*

(1) w is k-morphic;
(2) w is k-lexicographic;
(3) $w = S(w_{k-\mathrm{llex}})$ for some #-driven sequential transduction S;
(4) $W_w \le^{\mathcal{I}}_{\mathsf{rMSO}} W_{w_{k\text{-}llex}}$ for some $\mathcal{I} = (\varphi_D, <, \{\varphi_a\}_{a\in\Sigma})$ such that $\models \forall x(\varphi_D(x) \to P_\#(x))$.

Moreover, there are effective translations among these representations.

Proof. (1)⇒(2) (for $k > 0$.). Let $w = h\left(\prod_{n=0}^{\infty} \mathsf{leaves}(\varphi^n(\gamma))\right)$ with $\gamma = [\cdots[\gamma_0]\cdots]$, φ and h as in the definition of k-morphic words. Consider the tree structure of φ, let l be the maximum of the number of children of any of the nodes, and let $B \subseteq [l]^k$ be the set of labels of ordered branches from the root to a leaf, using the natural ordering on $[l]$. We define the *index transition system* of φ as $\mathcal{I}_\varphi = (\Gamma, [l]^k, \delta)$ with $\delta(g, w) = \varphi_w^{(0)}(g)$ for each $g \in \Gamma$ and $w \in B$ and $\delta(g, w)$ undefined otherwise. Note that for uniform morphism of words this definition is identical to that used in the proof of Proposition 4.3. By the Iteration Lemma

$$\mathsf{leaves}(\varphi^n(\gamma)) = \prod_{u\in B^n}^{<_{k\text{-llex}}} \varphi_u^{(0)}(\gamma_0)$$

and, since for each $g \in \Gamma$ the set $P_g = \{u \in B^* \mid \varphi_u^{(0)}(\gamma_0) = g\}$ is obviously accepted by $\mathcal{I}_\varphi$ with initial state γ_0 and single final state g, we can conclude that $(B^*, <_{k\text{-llex}}, \{P_g\}_{g\in\Gamma})$ is a k-lex presentation (in normal form) of $\hat{w} = \prod_{n=0}^{\infty} \mathsf{leaves}(\varphi^n(\gamma)) \in \Gamma^\omega$. By Proposition 4.7, $w = h(\hat{w})$ is also k-lex.

(2)⇒(1) (for $k > 0$). By the Normal Form Lemma w has a k-lex presentation $(D, <_{k\text{-llex}}, \{P_a\}_{a\in\Sigma})$ in normal form over $\{0,1\}$, *i.e.* with D and each P_a being a regular subset of $(\{0,1\}^k)^*$. Recall $\mathcal{A}_\mathfrak{d}$, $\mathcal{T}_\mathfrak{d}$, $\sigma_\mathfrak{d}$, etc. from Section 5. To provide a proof, we only need to adapt the notion of transition morphisms to k-stacks. The stack alphabet will, of course, be $\Gamma = Q_\mathfrak{d}$. We define for each $l \le k$ and for every $u \in \{0,1\}^{k-l}$ a morphism $\tau_u^{(l)} \in \mathsf{Hom}_\Gamma^{(l)}$ recursively by setting $\tau_u^{(l+1)} = [\tau_{u0}^{(l)}\tau_{u1}^{(l)}]$ for each u of length $k-l-1$, $l < k$, and by setting $\tau_u^{(0)}(\vec{q}) = \delta_\mathfrak{d}^*(\vec{q}, u)$ for every $u \in \{0,1\}^k$. Finally, let $\varphi = \tau_\varepsilon^{(k)} = \prod_{j_0=0}^{1} \big[\prod_{j_1=0}^{1} \big[\cdots \prod_{j_{k-1}=0}^{1} \big[\tau_{j_0\dots j_{k-1}}^{(0)}\big]\cdots\big]\big]$ and $\gamma = [..[\vec{q}_0]..] \in \mathsf{Stack}_\Gamma^{(k)}$. Observe that the structure of φ is the complete binary tree of depth k. Noting that $\tau_{w_n}^{(0)}(\dots\tau_{w_2}^{(0)}(\tau_{w_1}^{(0)}(\vec{q}))\dots) = \delta^*(\vec{q}, w_1w_2\dots w_n)$ the Iteration Lemma yields

$$\varphi^n(\gamma) = \prod_{u^{(1)}=0^n}^{1^n} \Big[\prod_{u^{(2)}=0^n}^{1^n} \Big[\cdots \prod_{u^{(k)}=0^n}^{1^n} \big[\delta^*(\vec{q}_0, \otimes_k(u^{(1)}, \dots, u^{(k)}))\big]\cdots\Big]\Big]$$

and we can conclude that $w = \sigma_\mathfrak{d}(\prod_{n=0}^{\infty} \mathsf{leaves}(\varphi^n(\gamma)))$.

(2)⇒(3): (Hint) $\mathcal{S}$ simulates $\mathcal{A}_\mathfrak{d}$, restarting on every #.
(3)⇒(4): (Hint) The run of $\mathcal{S}$ is obviously restricted MSO-interpretable.
(4)⇒(2): There is a $k+1$-lex presentation $(\mathfrak{d}, \nu)$ of $w_{k\text{-llex}}$, similar to that given in Example 4.5, such that each maximal factor u# with $u \in \{0,1\}^*$ is represented on words $x \in D$ satisfying $x' = u$ and with the $k+1^{st}$ component telling the position within u#. Let $\mathcal{I} = (\varphi_D, <, \{\varphi_a\}_{a\in\Sigma})$ be a restricted MSO- interpretation as in (4). By Theorem 3.7 each color-formula φ_a can be transformed into an equivalent automaton $\mathcal{A}_a$. Finally, to obtain a a k-lex presentation of $\mathcal{I}(W_{w_{k\text{-llex}}})$, we construct automata $\mathcal{A}'_a$ accepting those x' such that $x \in L(\mathcal{A}_a)$. □

Remark 7.6 (On the irrelevance of uniformity)**.** Let us point out, that in the proof of (2)⇒(1) of Theorem 7.5 we made use of the Normal Form Lemma 4.6 to first uniformize the k-lexicographic presentation in preparation for turning it into a k-morphism generating the same word. This step was necessary due to the above hinted uniformity of our morphisms. Thus, Lemma 4.6 shows that this uniformity is really no restriction in terms of generating power as long as we allow ourselves to apply an arbitrary homomorphism h in the final step.

Remark 7.7 (On morphic predicates, *cf.* [38])**.** Our definition of morphisms of k-stacks not only resembles that of k-dimensional "pictures", but is essentially identical with that, up to a natural coding. Indeed, k-dimensional pictures are k-stacks satisfying the uniformity condition that every level $l+1$ sub-stack consists of exactly the same number n_{l+1} of l-stacks, where $(n_1, \ldots, n_k)$ are the dimensions of the picture. Due to their above mentioned uniformity our morphisms preserve uniformity of stacks. Hence, morphisms of k-stacks and morphisms of k-dimensional pictures are easily seen to be one and the same, up to this coding.

8. Connection to the pushdown hierarchy

8.1. Caucal's pushdown hierarchy

Following Courcelle we say that a function T mapping structures of one signature σ to structures of another signature σ' is *(effectively)* MSO-*compatible* if there is an algorithm mapping each monadic formula φ of signature σ' to a monadic formula φ^T in the signature σ such that whenever $T(\mathfrak{A})$ is defined

$$\mathfrak{A} \models \varphi^T \quad \Longleftrightarrow \quad T(\mathfrak{A}) \models \varphi.$$

The fact that MSO-interpretations are MSO-compatible is straightforward. The more difficult result that the *unfolding* operation mapping graphs to trees is also MSO-compatible appeared in [24], see also [23] for an exposition and a treatment of the simpler case of deterministic graphs. We note that this result follows from Muchnik's theorem [6,55] and that it implies Rabin's theorem.

With the aid of the MSO-compatible operations of MSO-interpretations and unfolding a rich class of graphs of decidable monadic theories can be constructed [54]. In [20] Caucal proposed to consider, starting with finite graphs, the hierarchies of graphs and trees obtained by alternately applying unfoldings and MSO-interpretations as follows. Below we let $\mathfrak{T}_{\mathfrak{G},v}$ denote the tree resulting from unfolding the graph $\mathfrak{G}$ from its vertex v.

$$\begin{array}{rcl}
\mathcal{G}raphs_0 & = & \{\text{finite edge- and vertex-labelled graphs}\} \\
\mathcal{T}rees_{n+1} & = & \{\mathfrak{T}_{\mathfrak{G},v} \mid (\mathfrak{G}, v) \in \mathcal{G}raphs_n\} \\
\mathcal{G}raphs_{n+1} & = & \{\mathcal{I}(\mathfrak{T}) \mid \mathfrak{T} \in \mathcal{T}rees_{n+1}, \mathcal{I} \text{ is an MSO interpretation}\}.
\end{array}$$

If follows from the fact that both interpretations and unfolding are MSO-compatible that the MSO theory of each tree $\mathfrak{T} \in \mathcal{T}rees_n$ and of each graph $\mathfrak{G} \in \mathcal{G}raphs_n$ is decidable for every $n \in \mathbb{N}$.

This turns out to be a very rich and robust hierarchy: various weakenings and strengthenings of the above definition yield the exact same classes [16]. This hierarchy of graphs is also referred to as the pushdown hierarchy owing to the fact that for each n, $\mathcal{G}raphs_n$ contains, up to isomorphism, those graphs obtained as the ϵ-closure of the configuration graph of some higher-order pushdown automaton of level n [16]. This characterization was also used to show the strictness of the hierarchy [16].

The level-zero graphs are the finite graphs, trees of level one are the regular trees, the level-one graphs are those *prefix-recognizable* or equivalently *VR-equational* [9,18,19]. The deterministic level-two trees are known as *algebraic trees*.

However, from the second level onward we have no clear structural understanding of what kind of graphs inhabit the individual levels. For an exposition we recommend [54].

8.2. k-lex words are on the $2k$-th level: $\mathcal{W}_k \subset \mathcal{G}raphs_{2k}$

In this section we demonstrate that for each k all k-morphic words are on the $2k$-th level of the pushdown hierarchy of graphs. It is open whether this is tight.

Note that it only makes sense to try to locate infinite words in the hierarchy of graphs rather than of trees, for unless a word is ultimately periodic it is not the unfolding of anything simpler than itself. Therefore we wish to view infinite words as graphs. To this end we identify each ω-word $a_1a_2a_3\dots$ with the edge-labelled successor graph $\bullet\xrightarrow{a_1}\bullet\xrightarrow{a_2}\bullet\xrightarrow{a_3}\cdots$

Without doubt, the ω-words inhabiting the first level of the pushdown hierarchy are precisely the ultimately periodic ones. Indeed, the first level graphs are prefix-recognizable [19] and those among them of finite degree are context-free [9] and as such, by a classical result of Muller and Schupp [40], have only finitely many *ends* up to isomorphism (*cf.* also [37]). For our word graphs this means precisely that they are ultimately periodic. The converse containment is obvious.

On the next level, Caucal [20] has shown that morphic words, in the classical sense, are on the second level of the pushdown hierarchy. Whether they also exhaust the second level word graphs is, to the authors knowledge, not settled, though very plausible.

Starting with the third level, the pushdown hierarchy contains graphs of binary words of faster than exponential growth, which can hence not be automatic as can be verified by a standard pumping argument. An example of a fast growing sequence that is on the third level of the pushdown hierarchy is the characteristic sequence of the set of factorials, $01100010^{17}10^{95}10\dots$, also known as the Liouville word [11].

In order to place k-morphic words in the pushdown hierarchy, for each k we only need to locate a single tree $\mathfrak{T}_{<_{k\text{-llex}}}$, defined as follows. Let

$$T_{<_{k\text{-llex}}} = \{1^n\#w_1\#\otimes_2(w_1,w_2)\#\dots\#\otimes_k(w_1,w_2,\dots,w_k) \mid \forall i:\ w_i\in\{0,1\}^n\}$$

$P_{<_{k\text{-llex}}} = \mathsf{Pref}(T_{<_{k\text{-llex}}})$ the set of prefixes of words in $T_{<_{k\text{-llex}}}$ and $\mathfrak{T}_{<_{k\text{-llex}}}$ be the tree $(P_{<_{k\text{-llex}}}, succ_0, succ_1, succ_\#)$ illustrated in Figure 2. It has a single infinite branch 1^ω off of which at every position 1^n a finite subtree of depth $(n+1)k$ is hanging, the maximal paths of which are labelled by elements of $T_{<_{k\text{-llex}}}$. This set was designed so that the lexicographic ordering (for $\# < 0 < 1$) of these paths will correspond to the $<_{k\text{-llex}}$ ordering of their final segment below the last $\#$-edge.

We claim that an infinite word is k-lex iff its word graph is MSO-interpretable as a lexicographically ordered subset of the leaves of $\mathfrak{T}_{<_{k\text{-llex}}}$. Relying on the Normal Form Lemma 4.6 it is straightforward to give such an interpretation of any k-lex word. The converse implication (a proof of which can be found in [4], Sect. 6.2.1) is more involved as it yields the main results of Section 5.2 as corollaries.

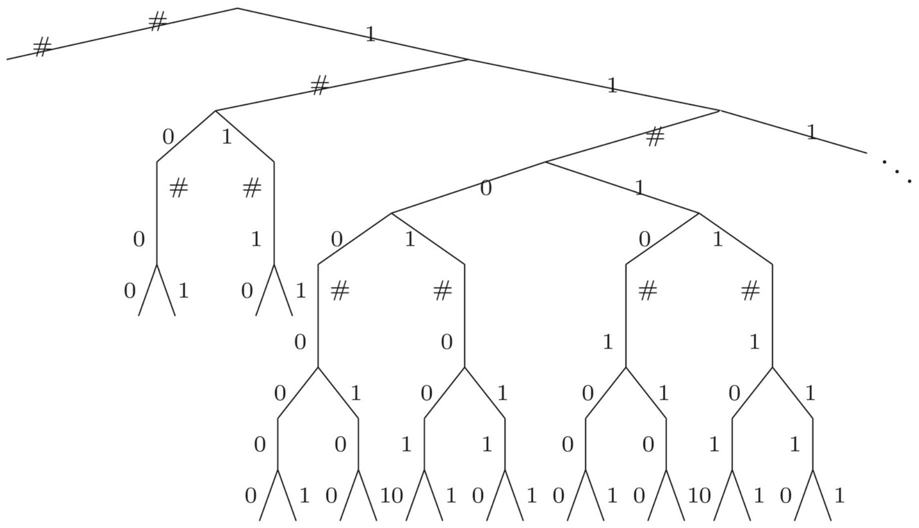


FIGURE 2. The tree $\mathfrak{T}_{<_{2\text{-llex}}}$ facilitating 2-lex words.

We show by induction that $\mathfrak{T}_{<_{k\text{-llex}}} \in \mathcal{G}\mathit{raphs}_{2k}$ for each $k > 0$, implying, by our previous observation, that k-morphic words are level $2k$ pushdown graphs.

Surely, $\mathfrak{T}_{<_{1\text{-llex}}}$ is an algebraic (level 2) tree as it is the unfolding of the graph of a one-counter automaton. This is essentially Caucal's argument [20] showing that morphic words are on the second level of the pushdown hierarchy.

To proceed with the induction we give MSO-interpretations $\mathcal{I}, \mathcal{J}, \mathcal{K}$, such that $\mathfrak{T}_{<_{k+1\text{-llex}}} = \mathcal{K}(\mathsf{Unfold}(\mathcal{J}(\mathsf{Unfold}(\mathcal{I}(\mathfrak{T}_{<_{k\text{-llex}}})))))$ for each $k > 0$. This approach was first suggested to the author by Thomas Colcombet, the construction presented below was conceived during discussions with Arnaud Carayol and owes a lot to his assistance.

The first interpretation, $\mathcal{I}$, preserves the original structure while also introducing two kinds of new edges: 1) reflexive #-edges on all leaves; 2) $\bar{\sigma}$-labelled reversals of σ-edges, for $\sigma = 0, 1$, but only in "final segments": between nodes which do not have a #-edge in the subtree below them. Obviously, these definitions are MSO expressible.

It should be clear that the unfolding of $\mathcal{I}(\mathfrak{T}_{<_{k\text{-llex}}})$, let us denote this tree by T' for now, contains all branches of the form

$$1^n \# w_1 \# \otimes_2 (w_1, w_2) \# \ldots \# \otimes_k (w_1, w_2, \ldots, w_k) \# \overline{\otimes_k (w_1, w_2, \ldots, w_k)}^{rev} \tag{4}$$

where $w_1, \ldots, w_k \in \{0,1\}^n$, and the last segment $\overline{\otimes_k(\ldots)}^{rev}$ denotes the reversal of $\otimes_k(\ldots)$ with barred symbols. This is precisely what we have intended. However, aside of these, the unfolding produces an abundance of unwanted "junk" paths obtained by alternately traversing forward and backward edges and/or by passing

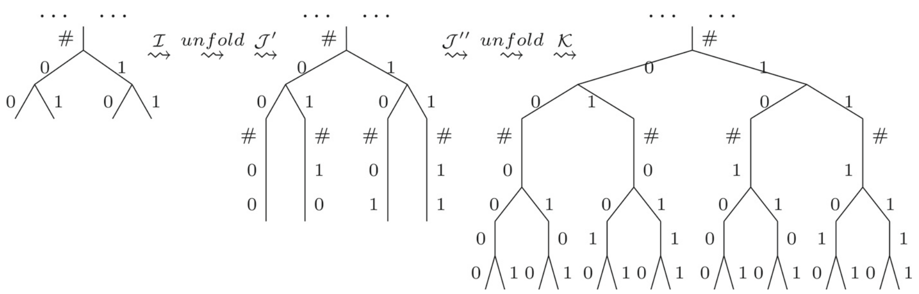

. . . . . .
$\mathcal{I}$ ⇝ unfold ⇝ $\mathcal{J}'$ ⇝
$\mathcal{J}''$ ⇝ unfold ⇝ $\mathcal{K}$ ⇝
#
0
1

FIGURE 3. Constructing $\mathfrak{T}_{<_{2\text{-llex}}}$ from $\mathfrak{T}_{<_{1\text{-llex}}}$: illustration on a finite subtree.

through a reflexive edge more than once. The interpretation $\mathcal{J}$ is defined in order to achieve the following tasks.

- Restrict T' to nodes appearing on branches of type (4) above.
  This is done by forbidding unintended patterns, *e.g.* repeated reflexive edges, etc, as implicated above, on branches leading to a node from the root.
- Reversing the final $\overline{\otimes_k(w_1, w_2, \ldots, w_k)}^{rev}$ segments of branches of type (4).
  This is a very simple operation, which can be done without producing any "junk": σ-labelled reversals of $\overline{\sigma}$-edges are added, while $\overline{\sigma}$-edges will be deleted, and those #-edges closest to a leaf are redirected to that leaf below them.
- Making room for w_{k+1} on every final segment: by introducing reflexive a- and b-labelled edges on nodes z from which the leaf below them is reachable on a #-free path of length divisible by k.

After the second unfolding we obtain a tree $T'' = \mathsf{Unfold}(\mathcal{J}(\mathsf{Unfold}(\mathcal{I}(\mathfrak{T}_{<_{k\text{-llex}}}))))$, which includes essentially $\mathfrak{T}_{<_{k+1\text{-llex}}}$ as an induced subtree (once a- and b-edges are renamed to 0 and 1 respectively), again, together with some unwanted branches arising from repeated traversals of reflexive a- or b-edges around the same node. The final clean-up needed is performed by the interpretation $\mathcal{K}$ by first restricting the domain to nodes reached from the root on a path avoiding immediate repetitions of a- or b-edges and finally renaming *e.g.* a-labels to 0 and b's to 1. The two-step construction is illustrated in Figure 3[3]. Thus we have established

Theorem 8.1. *For every k the word structure of every k-morphic word is on the $2k$-th level of the pushdown hierarchy: $\mathcal{W}_k \subset \mathcal{G}raphs_{2k}$.*

9. Closing remarks and open questions

In this section we hint at some possible further generalizations of the notion of k-lexicographic and k-morphic words and the results reported in this paper and

[3] Note that for the sake of a simpler illustration we decomposed $\mathcal{J}$ into two interpretations: $\mathcal{J}'$ purging unwanted branches produced by the previous unfolding and $\mathcal{J}''$ preparing ground for the second unfolding by the introduction of reverse edges and loops.

we also raise a handful of related questions.

Variations on k-lex. Although it is yet unclear whether and how our results can be extended to all automatically presentable ω-words, our techniques seem to extend easily to a mild generalization of k-lexicographic presentations. Let us sketch the idea here.

Imagine a variant of the 2-lexicographic ordering, which still compares the length of words x and y first, then $x^{(1)}$ with $y^{(1)}$ lexicographically, but in case these coincide then $x^{(2)}$ and $y^{(2)}$ are compared in *reverse lexicographic order*. We may denote such an ordering by $<_{lr}$. In general, for every sequence $\alpha \in \{l,r\}^*$ one can define the family of α-lexicographic orderings in a similar fashion, and commonly denote them by $<_\alpha$, when the ordered alphabet is understood or irrelevant. Thus, $<_\alpha$ generalizes $<_{k\text{-llex}}$ in that those components with an r in the respective position in α are compared not lexicographically but rather in reverse lexicographic order.

Based on this notion of α-lexicographic ordering we can introduce the class $\mathcal{W}_\alpha$ of α-lexicographic ω-words as those automatically presentable using $<_\alpha$. The classes $\mathcal{W}_\alpha$ form an infinite and possibly richer hierarchy as the classes of k-lex words. Let $\overline{l} = r$ and $\overline{r} = l$ and further extended to $\{l,r\}$-sequences. It is easily seen that $\mathcal{W}_\alpha = \mathcal{W}_{\overline{\alpha}}$ for each α since the reversal of all numerals transforms an α-lex presentation into an $\overline{\alpha}$-lex presentation and *vice versa*.

Although an automatic presentation obtained by reversal is not equivalent to the original one, relying on a recent result of Colcombet it is easy to see that an automatic presentation of an ω-word is canonical iff its reversal is canonical. Indeed, in [21] Colcombet proved that MSO-definable relations on ω-words (and on trees in general) are in fact first-order definable from $<$ and from certain MSO-definable *unary* predicates[4] and it is well-known that regularity of unary predicates is preserved under reversal.

Notice that the proof of the Hierarchy Theorem can be adapted to show that the $(k+1)$-st stuttering word s_{k+1} is not α-lex presentable for any $\alpha \in \{l,r\}^{\leq k}$. Also, if α is a proper subword (not necessarily a factor) of α' then $\mathcal{W}_\alpha \subsetneq \mathcal{W}'_\alpha$. A comprehensive comparison of the $\mathcal{W}_\alpha$ classes remains open. It is for instance unclear how $\mathcal{W}_{lr}$ and $\mathcal{W}_{ll}$ are related.

We claim without giving a thorough proof that all α-lex presentations are canonical. This can be checked by adapting the proof of the Contraction Lemma 5.2 on which the inductive step in the proof of Theorem 5.3 is based. One can argue that if the last symbol of α is l, *i.e.* if the last components are lexicographically ordered, then the proof goes through without any necessary adjustments. Furthermore, the

[4] According to Lemma 3.6 (*cf.* [21], Lem. 1) all MSO-definable relations on ω-words are first-order definable from $<$ and from MSO-definable *binary* predicates. The fact that already *unary* predicates are hereto sufficient and the tools involved in proving this have some very interesting implications and applications [22]. For instance, it allows us to define canonicity equivalently by requiring only that all MSO-definable unary predicates be regularly represented. However, this does not seem to make our proof of Theorem 5.3 with the method of contractions significantly simpler.

Contraction Lemma is invariant under reversal of all numerals of a presentation. Therefore, the Contraction Lemma holds for α iff it also holds for $\overline{\alpha}$, and obviously one of them ends with l. The MSO-decidability and MSO–definability results thus extend to all α-lexicographic ω-words.

Our approach of embedding $\mathcal{W}_k$ into the pushdown hierarchy is equally simple to adapt for $\mathcal{W}_\alpha$. Assuming the Normal Form Lemma 4.6, we can associate to each α a tree $\mathfrak{T}_\alpha$ to be constructed inside the pushdown hierarchy. We know of no better way of defining $\mathfrak{T}_{\alpha r}$ then *via* unfolding from $\mathfrak{T}_{\overline{\alpha}l}$. Also note that a single unfolding and MSO-interpretations suffice to build $\mathfrak{T}_{\overline{\alpha}l}$ from $\mathfrak{T}_\alpha$.

In light of the above we would be eager to find answers to the following pressing questions. We conjecture that the answer to at least three of them is affirmative.

(1) Is every automatic presentation of every ω-word canonical?
(2) Is every automatic ω-word constructable in the pushdown hierarchy?
(3) Is every aut. pres. of an ω-word equivalent to an α-lex presentation?
(4) Does every automatic ω-word allow a k-lex presentation for some k?

Variations on morphisms of level k stacks. One way to overcome the inherent uniformity of morphisms of level k stacks would be to utilize derivation rules Δ of level $k+1$ of the form

$$\Delta \ : \ A^0_x \longrightarrow A^1_{\delta^1(x)} \cdots A^s_{\delta^s(x)}$$

where the A^i's are "non-terminals" of level $k+1$ and the δ^i's are derivation rules of level k and x is a variable of order k. If one defines level 1 rules as homomorphisms of finite words, then it is not hard to extend Theorem 7.5 to show that every finite system of level $\leq k$ rules of the above form generates, in the style of Definition 7.2, a k-lexicographic ω-word.

Although we have thus far not found a proper means of generating α-lexicographic ω-words in a similar fashion for arbitrary α, it seems that the extension of the above scheme allowing for rules of the form

$$\Delta \ : \ A^0_x \longrightarrow A^1_{\delta^{1,1}(x)\ldots\delta^{1,t_1}(x)} \cdots A^s_{\delta^{s,1}(x)\ldots\delta^{s,t_s}(x)}$$

would necessitate the use of both left- and right-ordered components to allow for an automatic presentation. The relationship of ω-words generated by rules of the latter form and between $\mathcal{W}_\alpha$ classes is unclear. The pursuit of these ideas is left open.

Finite factors and combinatorics. A key aspect of the theory of automatic ω-words we have not touched upon concerns combinatorics of finite factors. Let us now note some sporadic facts involving finite factors of automatic ω-words while leaving any kind of systematic study entirely open.

From [15,39,46] we know that the set of finite factors and of finite prefixes of every automatic ω-word is context sensitive. Clearly, the growths of distances of consecutive occurrences of infinitely often occurring factors in an automatic ω-word are bounded by an exponential function. It would be desirable to have much finer conditions of (non-)automaticity.

It is a classical result that the subword complexity of every morphic word is bounded by $\mathcal{O}(n^2)$ (see *e.g.* [1] for a finer classification). We have seen that the Champernowne word having all finite words as factors, hence an exponential subword complexity, is 2-morphic. How can the possible subword complexities of k-morphic words be classified?

Analyzing ω-regular sets using methods from descriptive set theory Staiger points out a key property of ω-words having all finite words as factors, called *rich* in [52]. Observe that the first-order theory of a rich ω-word cannot allow an elementary decision procedure, for it can interpret the finite satisfiability problem of $\mathsf{FO}[<]$ on word structures [28].

Isomorphism and lower bounds. Interesting and difficult questions not considered here concern deciding the exact level of a given ω-word in our hierarchy, and deciding isomorphism of ω-words on each level. Both of these problems have long been open for morphic ω-words, that is for level one, having known solutions in very special cases only (see for instance [30] and the references therein).

(5) Is isomorphism of k-lexicographic words decidable?

(6) Let $k > k'$. Is it decidable whether a k-lex word is k'-lexicographic? In particular, is eventual periodicity of k-lex words decidable?

Acknowledgements. I thank Didier Caucal for correspondence, Tobias Ganzow for careful corrections, Łukasz Kaiser for numerous stimulating discussions and suggestions considerably simplifying the notation, Thomas Colcombet and Arnaud Carayol for their ideas contributing to Theorem 8.1, Alexander Rabinovich for his [43], Sasha Rubin for pointing out [45], and Wolfgang Thomas for a preprint of [44]. I am thankful to one of the referees for numerous very helpful remarks.

References

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

[2] J.-M. Autebert and J. Gabarró, Iterated GSMs and Co-CFL. *Acta Informatica* **26**, 749–769 (1989).

[3] V. Bárány, Invariants of automatic presentations and semi-synchronous transductions. In *STACS '06*. Lect. Notes Comput. Sci. **3884**, 289 (2006).

[4] V. Bárány, *Automatic Presentations of Infinite Structures*. Ph.D. thesis, RWTH Aachen (2007).

[5] J. Berstel, *Transductions and Context-Free Languages*. Teubner, Stuttgart (1979).

[6] D. Berwanger and A. Blumensath, The monadic theory of tree-like structures. In *Automata, Logics, and Infinite Games*. Lect. Notes Comput. Sci. **2500**, 285–301 (2002).

[7] A. Bès, Undecidable extensions of Büchi arithmetic and Cobham-Semënov theorem. *Journal of Symbolic Logic* **62**, 1280–1296 (1997).

[8] A. Blumensath, Automatic Structures. Diploma thesis, RWTH-Aachen (1999).

[9] A. Blumensath, Axiomatising Tree-interpretable Structures. In *STACS*. Lect. Notes Comput. Sci. **2285**, 596–607 (2002).

[10] A. Blumensath and E. Grädel, Finite presentations of infinite structures: Automata and interpretations. *Theor. Comput. Syst.* **37**, 641–674 (2004).

[11] L. Braud, Higher-order schemes and morphic words. Journées Montoises, Rennes (2006).

[12] V. Bruyère and G. Hansel, Bertrand numeration systems and recognizability. *Theoretical Computer Science* **181**, 17–43 (1997).

[13] V. Bruyère, G. Hansel, Ch. Michaux and R. Villemaire, Logic and p-recognizable sets of integers. *Bull. Belg. Math. Soc. Simon Stevin* **1**, 191–238 (1994).

[14] J.W. Cannon, D.B.A. Epstein, D.F. Holt, S.V.F. Levy, M.S. Paterson and W.P. Thurston, *Word processing in groups*. Jones and Barlett Publ., Boston, MA (1992).

[15] A. Carayol and A. Meyer, Context-sensitive languages, rational graphs and determinism (2005).

[16] A. Carayol and S. Wöhrle, The Caucal hierarchy of infinite graphs in terms of logic and higher-order pushdown automata. In *FSTTCS*. Lect. Notes Comput. Sci. **2914**, 112–123 (2003).

[17] O. Carton and W. Thomas, The monadic theory of morphic infinite words and generalizations. *Information and Computation* **176**, 51–65 (2002).

[18] D. Caucal, Monadic theory of term rewritings. In *LICS*, pp. 266–273. IEEE Computer Society (1992).

[19] D. Caucal, On infinite transition graphs having a decidable monadic theory. In *ICALP'96*. Lect. Notes Comput. Sci. **1099**, 194–205 (1996).

[20] D. Caucal, On infinite terms having a decidable monadic theory. In *MFCS*, pp. 165–176 (2002).

[21] Th. Colcombet, A combinatorial theorem for trees. In *ICALP'07*. Lect. Notes Comput. Sci. **4596**, 901–912 (2007).

[22] Th. Colcombet, On factorisation forests and some applications. `arXiv:cs.LO/0701113v1` (2007).

[23] B. Courcelle, The monadic second-order logic of graphs ix: Machines and their behaviours. *Theoretical Computer Science* **151**, 125–162 (1995).

[24] B. Courcelle and I. Walukiewicz, Monadic second-order logic, graph coverings and unfoldings of transition systems. *Annals of Pure and Applied Logic* **92**, 35–62 (1998).

[25] C.C. Elgot and M.O. Rabin, Decidability and undecidability of extensions of second (first) order theory of (generalized) successor. *Journal of Symbolic Logic* **31**, 169–181 (1966).

[26] S. Fratani and G. Sénizergues, Iterated pushdown automata and sequences of rational numbers. *Annals of Pure and Applied Logic* **141**, 363–411, (2006).

[27] Ch. Frougny and J. Sakarovitch, Synchronized rational relations of finite and infinite words. *Theoretical Computer Science* **108**, 45–82 (1993).

[28] E. Grädel, W. Thomas and T. Wilke, Eds. *Automata, Logics, and Infinite Games*. Lect. Notes Comput. Sci. **2500**, (2002).

[29] B.R. Hodgson, Décidabilité par automate fini. *Ann. Sci. Math. Québec* **7**, 39–57 (1983).

[30] J. Honkala and M. Rigo, A note on decidability questions related to abstract numeration systems. *Discrete Math.* **285**, 329–333 (2004).

[31] K. Culik II and J. Karhumäki, Iterative devices generating infinite words. In *STACS '92*. Lect. Notes Comput. Sci. **577**, 529–543 (1992).

[32] L. Kari, G. Rozenberg and A. Salomaa, L systems. In *Handbook of Formal Languages*, G. Rozenberg and A. Salomaa Eds., *volume I*, pp. 253–328. Springer, New York (1997).

[33] B. Khoussainov and A. Nerode, Automatic presentations of structures. In *LCC '94*. Lect. Notes Comput. Sci. **960**, 367–392 (1995).

[34] B. Khoussainov and S. Rubin, Automatic structures: Overview and future directions. *J. Autom. Lang. Comb.* **8**, 287–301 (2003).

[35] B. Khoussainov, S. Rubin and F. Stephan, Definability and regularity in automatic structures. In *STACS '04*. Lect. Notes Comput. Sci. **2996**, 440–451 (2004).

[36] T. Lavergne, Prédicats algébriques d'entiers. Rapport de stage, IRISA: Galion (2005).

[37] O. Ly, Automatic Graph and D0L-Sequences of Finite Graphs. *Journal of Computer and System Sciences* **67**, 497–545 (2003).

[38] A. Maes, An automata theoretic decidability proof for first-order theory of $\langle \mathbb{N}, <, P \rangle$ with morphic predicate *P. J. Autom. Lang. Comb.* **4**, 229–245 (1999).

[39] Ch. Morvan and Ch. Rispal, Families of automata characterizing context-sensitive languages. *Acta Informatica* **41**, 293–314 (2005).

[40] D.E. Muller and P.E. Schupp, The theory of ends, pushdown automata, and second-order logic. *Theor. Comput. Sci.* **37**, 51–75 (1985).

[41] J.-J. Pansiot, On various classes of infinite words obtained by iterated mappings. In *Automata on Infinite Words*, pp. 188–197 (1984).

[42] J.-E. Pin and P.V. Silva, A topological approach to transductions. *Theoretical Computer Science* **340**, 443–456 (2005).

[43] A. Rabinovich, On decidability of monadic logic of order over the naturals extended by monadic predicates. Unpublished note (2005).

[44] A. Rabinovich and W. Thomas, Decidable theories of the ordering of natural numbers with unary predicates. In *CSL 2006*. Lect. Notes Comput. Sci. **4207**, 562–574 (2006).

[45] M. Rigo and A. Maes, More on generalized automatic sequences. *J. Autom. Lang. Comb.* **7**, 351–376 (2002).

[46] Ch. Rispal, The synchronized graphs trace the context-sensistive languages. *Elect. Notes Theoret. Comput. Sci.* **68** (2002).

[47] G. Rozenberg and A. Salomaa, *The Book of L*. Springer Verlag (1986).

[48] S. Rubin, *Automatic Structures*. Ph.D. thesis, University of Auckland, NZ (2004).

[49] S. Rubin, Automata presenting structures: A survey of the finite-string case. Manuscript.

[50] G. Sénizergues, Sequences of level 1, 2, 3,..., k,... In *CSR'07*. Lect. Notes Comput. Sci. **4649**, 24–32 (2007).

[51] S. Shelah, The monadic theory of order. *Annals of Mathematics* **102**, 379–419 (1975).

[52] L. Staiger, Rich omega-words and monadic second-order arithmetic. In *CSL*, pp. 478–490 (1997).

[53] W. Thomas, Languages, automata, and logic. In *Handbook of Formal Languages*, G. Rozenberg and A. Salomaa, Eds., *Vol. III*, pp. 389–455. Springer, New York (1997).

[54] W. Thomas, Constructing infinite graphs with a decidable mso-theory. In *MFCS'03*. Lect. Notes Comput. Sci. **2747**, 113–124 (2003).

[55] I. Walukiewicz, Monadic second-order logic on tree-like structures. *Theoretical Computer Science* **275**, 311–346 (2002).