Level Two of the quantifier
alternation hierarchy over
infinite words

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Citation: KUFLEITNER, M. and WALTER, T., 2017. Level Two of the quantifier alternation hierarchy over infinite words. Theory of Computing Systems, 62 (3), pp.467-480

Additional Information:

- This journal article is part of the following topical collections: Computer Science Symposium in Russia. This is a post-peer-review, pre-copyedit version of an article published in Theory of Computing Systems. The final authenticated version is available online at: https://doi.org/10.1007/s00224-017-9801-x

Metadata Record: https://dspace.lboro.ac.uk/2134/31947

Version: Accepted for publication

Publisher: Springer

Rights: This work is made available according to the conditions of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) licence. Full details of this licence are available at: https://creativecommons.org/licenses/by-nc-nd/4.0/

Please cite the published version.
Level Two of the Quantifier Alternation Hierarchy over Infinite Words

Manfred Kufleitner · Tobias Walter

Received: date / Accepted: date

Abstract The study of various decision problems for logic fragments has a long history in computer science. This paper is on the membership problem for a fragment of first-order logic over infinite words; the membership problem asks for a given language whether it is definable in some fixed fragment. The alphabetic topology was introduced as part of an effective characterization of the fragment $\Sigma_2$ over infinite words. Here, $\Sigma_2$ consists of the first-order formulas with two blocks of quantifiers, starting with an existential quantifier. Its Boolean closure is $B\Sigma_2$. Our first main result is an effective characterization of the Boolean closure of the alphabetic topology, that is, given an $\omega$-regular language $L$, it is decidable whether $L$ is a Boolean combination of open sets in the alphabetic topology. This is then used for transferring Place and Zeitoun’s recent decidability result for $B\Sigma_2$ from finite to infinite words.

1 Introduction

Over finite words, the connection between finite monoids and regular languages is highly successful for studying logic fragments, see e.g. [2,18]. Over infinite words, the algebraic approach uses infinite repetitions. Not every logic fragment can express whether some definable property $P$ occurs infinitely often. For instance, the usual approach for saying that $P$ occurs infinitely often is as follows: for every position $x$ there is a position $y > x$ satisfying $P(y)$. Similarly, $P$ occurs only finitely often if there is a position $x$ such that all positions $y > x$ satisfy $\neg P(y)$. Each of these formulas requires (at least) one additional change of quantifiers, which not all fragments can provide. It turns out that

This work was supported by the German Research Foundation (DFG) under grants DI 435/5-2 and DI 435/6-1.

FMI, Universität Stuttgart, Germany
{kufleitner, walter}@fmi.uni-stuttgart.de
topology is a very useful tool for restricting the infinite behaviour of the algebraic approach accordingly, see e.g. [3, 5, 9, 21]. In particular, the combination of algebra and topology is convenient for the study of languages in $\Gamma^\infty$, the set of finite and infinite words over the alphabet $\Gamma$. In this paper, an $\omega$-regular language is a regular subset of $\Gamma^\infty$.

Topological ideas have a long history in the study of $\omega$-regular languages. The Cantor topology is the most famous example in this context. We write $G$ for the Cantor-open sets, and $B_X$ for the Boolean closure of $X$. Let $FO^k$ be the fragment of first-order logic which uses (and reuses) at most $k$ variables. By $\Sigma^k_m$ we denote the formulas with $m$ quantifier blocks, starting with a block of existential quantifiers. Here, we assume that $x < y$ is the only binary predicate. Let us consider $FO^1$ as a toy example. With only one variable, we cannot make use of the binary predicate $x < y$. Therefore, in $FO^1$ we can say nothing but which letters occur, that is, a language is definable in $FO^1$ if and only if it is a Boolean combination of languages of the form $\Gamma^*a\Gamma^\infty$ for $a \in \Gamma$. Thus $FO^1 \subseteq BG$. It is an easy exercise to show that an $\omega$-regular language is in $FO^1$ if and only if it is in $BG$ and its syntactic monoid is both idempotent and commutative. The algebraic condition without the topology is too powerful since this would also include the language $\{a, b\}^*a^\omega$, which is not definable in $FO^1$. For the fragment $B\Sigma_1$, the same topology $BG$ with a different algebraic condition works, cf. [9, Theorems VI.3.7, VI.7.4 and VIII.4.5].

In the fragment $\Sigma_2$, we can define the language $\{a, b\}^*ab^\infty$ which is not in $BG$ (it is not even deterministic). As shown by Diekert and the first author [3], the alphabetic topology is suitable for the fragment $\Sigma_2$. The open sets in this topology are arbitrary unions of languages of the form $uA^\infty$ for $u \in \Gamma^*$ and $A \subseteq \Gamma$. An $\omega$-regular language is definable in $\Sigma_2$ if and only if it satisfies some particular algebraic property and if it is open in the alphabetic topology [3]. Therefore, the canonical ingredient for an effective characterization of $B\Sigma_2$ is the Boolean closure of the open sets in the alphabetic topology. Our first main result, Theorem 2, shows that, for a given $\omega$-regular language $L$, it is decidable whether $L$ is a Boolean combination of open sets in the alphabetic topology. As a by-product, we see that every $\omega$-regular language which is a Boolean combination of arbitrary open sets in the alphabetic topology can be written as a Boolean combination of $\omega$-regular open sets. This resembles a similar result for the Cantor topology [16].

A major breakthrough in the theory of regular languages over finite words is due to Place and Zeitoun [13]. They showed that, for a given regular language $L \subseteq \Gamma^*$, it is decidable whether $L$ is definable in $B\Sigma_2$. This solved a longstanding open problem; see e.g. [12, Section 8] for an overview. Our second main result, Theorem 4, is to show that this decidability result transfers to languages in $\Gamma^\infty$. If $V_2$ is the algebraic counterpart of $B\Sigma_2$ over finite words, then we show that $V_2$ combined with the Boolean closure of the alphabetic topology yields a characterization of $B\Sigma_2$ over $\Gamma^\infty$. Combining the decidability of $V_2$ with our first main result, the latter characterization is effective. The proof that $B\Sigma_2$ satisfies both the algebraic and the topological restrictions follows a rather straightforward approach. The main difficulty is to show
the converse: every language satisfying both the algebraic and the topological conditions is definable in $\mathcal{B}\Sigma_2$.

2 Preliminaries

Words

Let $\Gamma$ be a finite alphabet. By $\Gamma^*$ we denote the set of finite words over $\Gamma$; we write $1$ for the empty word. The set of infinite words is $\Gamma^\omega$ and the set of finite and infinite words is $\Gamma^\infty = \Gamma^* \cup \Gamma^\omega$. By $u, v, w$ we denote finite words and by $\alpha, \beta, \gamma$ we denote words in $\Gamma^\infty$. In this paper a language is a subset of $\Gamma^\infty$. Let $L \subseteq \Gamma^*$ and $K \subseteq \Gamma^\infty$. As usual $L^*$ is the union of powers of $L$ and $LK = \{\alpha \in \Gamma^\infty \mid \alpha \subseteq L, \alpha \in K\}$ is the concatenation of $L$ and $K$. By $L^\omega$ we denote the set of infinite words which are an infinite concatenation of words in $L$ and the infinite concatenation $uu\cdots$ of the word $u$ is written $u^\omega$. A word $u = a_1 \ldots a_n$ is a scattered subword of $v$ if $v \in \Gamma^*_{a_1} \Gamma^*_{a_2} \cdots a_n$. The alphabet of a word is the set of all letters which appear in the word. The imaginary alphabet $\text{im}(\alpha)$ of a word $\alpha \in \Gamma^\infty$ is the set of letters which appear infinitely often in $\alpha$. Let $A_{\text{im}} = \{\alpha \in \Gamma^\infty \mid \text{im}(\alpha) = A\}$ be the set of words with imaginary alphabet $A$. In the following, we restrict ourselves to the study of $\omega$-regular languages. A language $L \subseteq \Gamma^*$ is regular if it is recognized by a (deterministic) finite automaton. A language $K \subseteq \Gamma^\infty$ is $\omega$-regular if it is recognized by a Büchi automaton. A language $L \subseteq \Gamma^*$ is regular and $L \cap \Gamma^\omega$ is $\omega$-regular.

First-Order logic

We consider first order logic FO over $\Gamma^\infty$. Variables range over positions of the word. The atomic formulas in this logic are $\top$ for true, $x < y$ to compare two positions $x$ and $y$ and $\lambda(x) = a$ which is true if the word has an $a$ at position $x$. One may combine those atomic formulas with the boolean connectives $\neg, \land$ and $\lor$ and quantifiers $\forall$ and $\exists$. A sentence $\varphi$ is an FO formula without free variables. We write $\alpha \models \varphi$ if $\alpha \in \Gamma^\infty$ satisfies the sentence $\varphi$. The language defined by $\varphi$ is $L(\varphi) = \{\alpha \in \Gamma^\infty \mid \alpha \models \varphi\}$. We classify the formulas of FO by counting the number of quantifier alternations, that is the number of alternations of $\exists$ and $\forall$. The fragment $\Sigma_i$ of FO contains all FO-formulas in prenex normal form with $i$ blocks of quantifiers $\exists$ or $\forall$, starting with a block of existential quantifiers. The fragment $\mathcal{B}\Sigma_i$ contains all Boolean combinations of formulas in $\Sigma_i$. We are particularly interested in the fragment $\Sigma_2$ and the Boolean combinations of formulas in $\Sigma_2$. A language $L$ is definable in a fragment $F$ (e.g. $F$ is $\Sigma_2$ or $\mathcal{B}\Sigma_2$) if there exists a formula $\varphi \in F$ such that $L = L(\varphi)$, i.e., if $L$ is definable by some $\varphi \in F$. The classes of languages defined by $\Sigma_i$ and $\mathcal{B}\Sigma_i$ form a hierarchy, the quantifier alternation hierarchy.

This hierarchy is strict, i.e., $\Sigma_i \subsetneq \mathcal{B}\Sigma_i \subsetneq \Sigma_{i+1}$ holds for all $i$, cf. [1,19].
Monomials

A monomial is a language of the form \( A_0^n a_1 A_1^n a_2 \cdots A_{n-1}^n a_n A_n^\infty \) for \( n \geq 0 \), \( a_i \in \Gamma \) and \( A_i \subseteq \Gamma \). The number \( n \) is called the degree. In particular, \( A_0^\infty \) is a monomial of degree 0. A monomial is called \( k \)-monomial if it has degree at most \( k \). In [3] it is shown that a language \( L \subseteq \Gamma^\infty \) is in \( \Sigma_2 \) if and only if it is a finite union of monomials. We are interested in \( \mathbb{B}\Sigma_2 \) and thus in finite Boolean combination of monomials. For this, let \( \equiv_k^\infty \) be the equivalence relation on \( \Gamma^\infty \) such that \( \alpha \equiv_k^\infty \beta \) if \( \alpha \) and \( \beta \) are contained in exactly the same \( k \)-monomials. Thus, \( \equiv_k^\infty \)-classes are Boolean combinations of monomials and every language in \( \mathbb{B}\Sigma_2 \) is a union of \( \equiv_k^\infty \)-classes for some \( k \). Further, since there are only finitely many monomials of degree \( k \), there are only finitely many \( \equiv_k^\infty \)-classes. The equivalence class of some word \( \alpha \) in \( \equiv_k^\infty \) is denoted by \( [\alpha]_k^\infty \). Note, that such a characterization of \( \mathbb{B}\Sigma_2 \) in terms of monomials does not yield a decidable characterization.

Our characterization of languages \( L \subseteq \Gamma^\infty \) in \( \mathbb{B}\Sigma_2 \) is based on the characterization of languages in \( \mathbb{B}\Sigma_2 \) over finite words. For this, we also introduce monomials over \( \Gamma^* \). A monomial over \( \Gamma^* \) is a language of the form \( A_0^n a_1 A_1^n a_2 \cdots A_{n-1}^n a_n A_n^k \) for \( n \geq 1 \), \( a_i \in \Gamma \) and \( A_i \subseteq \Gamma \). The degree is defined as above. Let \( \equiv_k \) be the congruence on \( \Gamma^* \) which is defined by \( u \equiv_k v \) if and only if \( u \) and \( v \) are contained in the same \( k \)-monomials over \( \Gamma^* \). Again, a language \( L \subseteq \Gamma^* \) is in \( \mathbb{B}\Sigma_2 \) if and only if it is a union of \( \equiv_k \)-classes for some \( k \).

Algebra

In this paper all monoids are either finite or free. Finite monoids are a common way for defining regular and \( \omega \)-regular languages. A monoid element \( e \) is idempotent if \( e^2 = e \). An ordered monoid \( (M, \leq) \) is a monoid equipped with a partial order which is compatible with the monoid multiplication, i.e., \( s \leq t \) and \( s' \leq t' \) implies \( ss' \leq tt' \). Every monoid can be ordered using the identity as partial order. A homomorphism \( h : (N, \leq) \to (M, \leq) \) between two ordered monoids must satisfy \( s \leq t \Rightarrow h(s) \leq h(t) \) for all \( s, t \in N \). A divisor is the homomorphic image of a submonoid. A class of monoids which is closed under division and finite direct products is a pseudovariety. Eilenberg showed a correspondence between certain classes of languages (of finite words) and pseudovarieties [4]. A pseudovariety of ordered monoids is defined the same way as with unordered monoids, using homomorphisms of ordered monoids. The Eilenberg correspondence also holds for ordered monoids [11]. Let \( V_{3/2} \) be the pseudovariety of ordered monoids which corresponds to \( \Sigma_2 \) and \( V_2 \) be the pseudovariety of monoids which corresponds to languages in \( \mathbb{B}\Sigma_2 \). Since \( \Sigma_2 \subseteq \mathbb{B}\Sigma_2 \), we obtain \( V_{3/2} \subseteq V_2 \) when ignoring the order. The connection between monoids and languages is given by the notion of recognizability. A language \( L \subseteq \Gamma^* \) is recognized by an ordered monoid \( (M, \leq) \) if there is a monoid homomorphism \( h : \Gamma^* \to M \) such that \( L = \bigcup \{ h^{-1}(t) \mid s \leq t \text{ for some } s \in h(L) \} \). If \( M \) is not
ordered, then this means that \( L \) is an arbitrary union of languages of the form \( h^{-1}(t) \).

For \( \omega \)-languages \( L \subseteq \Gamma^\infty \) the notion of recognizability is slightly more technical. For simplicity, we only consider recognition by unordered monoids. Let \( h : \Gamma^* \rightarrow M \) be a monoid homomorphism. If the homomorphism \( h \) is understood, we write \([s]\) for the language \( h^{-1}(s) \). We call \((s, e) \in M \times M\) a linked pair if \( e^2 = e \) and \( se = s \). By Ramsey's Theorem [14] for every word \( \alpha \in \Gamma^\infty \) there exists a linked pair \((s, e)\) such that \( \alpha \in [s][e]^{\omega} \). A language \( L \subseteq \Gamma^\infty \) is recognized by \( h \) if

\[
L = \bigcup \{ [s][e]^{\omega} \ | \ (s, e) \text{ is a linked pair with } [s][e]^{\omega} \cap L \neq \emptyset \}.
\]

Since \( 1^\omega = 1 \), the language \([1]^{\omega}\) also contains finite words. We thus obtain recognizability of languages of finite words as a special case.

Next, we define syntactic homomorphisms and syntactic monoids; these are the minimal recognizers of an \( \omega \)-regular language. Let \( L \subseteq \Gamma^\infty \) be an \( \omega \)-regular language. The syntactic monoid of \( L \) is defined as the quotient \( \text{Synt}(L) = \Gamma^*/\approx_L \), where \( u \approx_L v \) holds if and only if for all \( x, y, z \in \Gamma^* \) we have both \( xuyz^{\omega} \in L \iff xvyz^{\omega} \in L \) and \( (uy)^{\omega} \in L \iff (vy)^{\omega} \in L \). The syntactic monoid can be ordered by the partial order \( \preceq_L \) defined by \( u \preceq_L v \) if for all \( x, y, z \in \Gamma^* \) we have \( xuyz^{\omega} \in L \Rightarrow xvyz^{\omega} \in L \) and \( (uy)^{\omega} \in L \Rightarrow (vy)^{\omega} \in L \). The syntactic homomorphism \( h_L : \Gamma^* \rightarrow \text{Synt}(L) \) is given by \( h_L(u) = [u]_{\approx_L} \). One can effectively compute the syntactic homomorphism of \( L \). The syntactic monoid \( \text{Synt}(L) \) satisfies the property that \( L \) is \( \omega \)-regular if and only if \( \text{Synt}(L) \) is finite and the syntactic homomorphism \( h_L \) recognizes \( L \), see e.g. [9, 20]. Every pseudovariety is generated by its syntactic monoids [4], i.e., every monoid in a given pseudovariety is a divisor of a direct product of syntactic monoids. The importance of the syntactic monoid of a language \( L \subseteq \Gamma^\infty \) is that it is the smallest monoid recognizing \( L \).

**Lemma 1** Let \( L \subseteq \Gamma^\infty \) be a language which is recognized by a homomorphism \( h : \Gamma^* \rightarrow (M, \leq) \). Then, \((\text{Synt}(L), \preceq_L)\) is a divisor of \((M, \leq)\).

**Proof** We assume that \( h \) is surjective and show that \( \text{Synt}(L) \) is a homomorphic image of \( M \). If \( h \) is not surjective, we can therefore conclude that \( \text{Synt}(L) \) is a divisor of \( M \). We show that \( h(u) \leq h(v) \Rightarrow u \preceq_L v \). Let \( u, v \) be words with \( h(u) \leq h(v) \) and denote \( h^{-1}(h(u)) = [h(w)] \) for words \( w \). Assume \( xuyz^{\omega} \in L \), then there exists an index \( i \) such that \( (h(xuyz^i), h(z)^{\omega}) \) is a linked pair. Thus, \( [h(xuyz^i)][h(z)]^{\omega} \subseteq L \) and by \( h(u) \leq h(v) \) also \( [h(xuyz^i)][h(z)]^{\omega} \subseteq L \). This implies \( xuyz^{\omega} \in L \). The proof that \( x(uy)^{\omega} \in L \Rightarrow x(vy)^{\omega} \in L \) is similar. Thus, \( u \preceq_L v \) holds which shows the claim. \( \square \)

### 3 Alphabetic Topology

As mentioned in the introduction, combining algebraic and topological conditions is a successful approach for characterizations of language classes over
\( \Gamma^\infty \). A topology on a set \( X \) is given by a family of subsets of \( X \) (called open) which are closed under finite intersections and arbitrary unions. We define the alphabetic topology on \( \Gamma^\infty \) by its basis \( \{ uA^\infty \mid u \in \Gamma^* \} \). Hence, an open set is given by \( \bigcup A W_A A^\infty \) with \( W_A \subseteq \Gamma^\infty \). The alphabetic topology has been introduced in [3], where it is used as a part of the characterization of \( \Sigma_2 \).

**Theorem 1** ([3]) Let \( L \subseteq \Gamma^\infty \) be an \( \omega \)-regular language. Then \( L \in \Sigma_2 \) if and only if \( \text{Syt}(L) \in V_{3/2} \) and \( L \) is open in the alphabetic topology.

The alphabetic topology has by itself been the subject of further study [15]. We are particularly interested in Boolean combinations of open sets. An effective characterization of a language \( L \) being a Boolean combination of open sets in the alphabetic topology is given in the theorem below.

**Theorem 2** Let \( L \subseteq \Gamma^\infty \) be an \( \omega \)-regular language which is recognized by \( h : \Gamma^* \to M \). Then the following are equivalent:

1. \( L \) is a Boolean combination of open sets in the alphabetic topology where each open set is \( \omega \)-regular.
2. \( L \) is a Boolean combination of open sets in the alphabetic topology.
3. For all linked pairs \((s,e),(t,f)\) it holds that if there exists an alphabet \( C \) and words \( \hat{e}, \hat{f} \) with \( h(\hat{e}) = e, h(\hat{f}) = f \), \( \text{alph}(\hat{e}) = \text{alph}(\hat{f}) = C \) and \( s \cdot h(C^*) = t \cdot h(C^*) \), then \([s][e]^\omega \subseteq L \iff [t][f]^\omega \subseteq L \).

**Proof** “1 \( \Rightarrow \) 2”: This is immediate.

“2 \( \Rightarrow \) 3”: Let \( L \) be a Boolean combination of open sets in the alphabetic topology. Note that for \( P,Q \subseteq \Gamma^* \) and \( A,B \subseteq \Gamma \) it holds \( PA^\infty \cap QB^\infty = (PA^* \cap QB^*)(A \cap B)^\infty \). Therefore, we may assume

\[
L = \bigcup_{i=1}^{n} \left( (P_i A_i^\infty) \setminus \left( \bigcup_{j=1}^{m} Q_{i,j} B_{i,j}^\infty \right) \right)
\]

for some \( P_i, Q_{i,j} \subseteq \Gamma^* \) and alphabets \( A_i, B_{i,j} \subseteq \Gamma \).

Let \((s,e)\) and \((t,f)\) be some linked pairs, \( C \subseteq \Gamma \) be an alphabet such that \( s \cdot h(C^*) = t \cdot h(C^*) \) holds and there exist words \( \hat{e}, \hat{f} \) with \( h(\hat{e}) = e, h(\hat{f}) = f \) and \( \text{alph}(\hat{e}) = \text{alph}(\hat{f}) = C \). Assume \([s][e]^\omega \subseteq L \), but \([t][f]^\omega \nsubseteq L \). Since \( h \) recognizes \( L \), it suffices to show that \([t][f]^\omega \cap L \) is nonempty to obtain a contradiction. Let \( \tilde{w}^\omega \subseteq [s][e]^\omega \subseteq L \) for some \( \tilde{w} \in [s] \). Since \( s \cdot h(C^*) = t \cdot h(C^*) \), we may choose \( x,y \in C^* \) such that \( s \cdot h(x) = t \) and \( t \cdot h(y) = s \).

The idea is to find an increasing sequence of words \( u_\ell \in [s] \) and sets \( I_\ell \subseteq \{1, \ldots, n\} \) such that \( u_\ell C^\infty \cap \left( P_i A_i^\infty \setminus \left( \bigcup_{j=1}^{m} Q_{i,j} B_{i,j}^\infty \right) \right) = \emptyset \) for all \( \ell \in I_\ell \).

We can set \( u_0 = u \) and \( I_0 = \emptyset \). Consider the word \( u_\ell \hat{e}^\omega \in L \). There exists an index \( i \in \{1, \ldots, n\} \setminus I_\ell \) such that \( u_\ell \hat{e}^\omega \in P_i A_i^\infty \setminus \left( \bigcup_{j=1}^{m} Q_{i,j} B_{i,j}^\infty \right) \). Choose a number \( k \), such that \( u_\ell \hat{e}^k \in P_i A_i^\infty \). Since \( C = \text{alph}(\hat{e}) \subseteq A_i \), we conclude \( \beta_\ell = u_\ell \hat{e}^k x \hat{f}^\omega \in P_i A_i^\infty \). By construction we have \( \beta_\ell \in [t][f]^\omega \) and therefore, assuming \([t][f]^\omega \cap L = \emptyset \), there exists an index \( j \) such that \( \beta_\ell \in Q_{i,j} B_{i,j}^\infty \).
Analogously, there exists $k'$ such that $u_{l+1} = u_{l}v^{k}x^{k'}yC^\infty \subseteq Q_{i,j}B_{i,j}^\infty$. Hence we can choose $u_{l+1} = u_{l}v^{k}x^{k'}y$ and $I_{l+1} = I_{l} \cup \{i\}$. Figure 1 gives an overview of the construction.

Since $u_{l}[e]^\omega \subseteq L \cap u_{l}C^\infty$, this construction has to fail at an index $\ell \leq n$. Therefore, the assumption is not justified and we have $[t][f]^\omega \cap L \neq \emptyset$, proving the claim.

"3 $\Rightarrow$ 1": Let $\alpha \in [s][e]^\omega \subseteq L$ for a linked pair $(s,e)$ and let $C = \text{im}(\alpha)$ denote the imaginary alphabet of $\alpha$. By $\alpha \in [s][e]^\omega$ and the definition of $C$, there exists an $\hat{e} \in C^*$ with alph($\hat{e}$) = $C$ and $h(\hat{e}) = e$. Define

$$L(s, C) = [s][C^\infty \setminus \left( \bigcup_{D \subseteq C} \Gamma^* D^\infty \cup \bigcup_{s \in [t] \cdot h(C^*)} [t][C^\infty] \right).$$

We have $\alpha \in L(s, C)$ and $L(s, C)$ is a Boolean combination of open sets in the alphabetic topology where each open set is $\omega$-regular. There are only finitely many sets of the type $L(s, C)$. The idea is to saturate $L$ with sets of this type, i.e., it suffices to show $L(s, C) \subseteq L$. For $C = \emptyset$, we have $L(s, C) = [s] \subseteq L$. Thus, we may assume $C \neq \emptyset$. Let $\beta \in L(s, C)$ be an arbitrary element and let $(t,f)$ be a linked pair such that $\beta \in [t][f]^\omega$. Since $\beta$ is in $L(s, C)$, there exists a prefix $u$ of $\beta$ such that $\beta \in uC^\omega$ and $u \in [s]$.

By $\beta \in [t][f]^\omega$, one gets $\beta = v\beta'$ with $v \in [t]$, $\beta' \in [f]^\omega$. Using $tf = t$ and $C \neq \emptyset$, we may assume that $u$ is a prefix of $v$, which implies $\beta' \in C^\omega$. Hence we have $t = h(v) \in h(uC^*) = s \cdot h(C^*)$. By construction $\beta \not\in \bigcup_{s \in [t] \cdot h(C^*)} [t][C^\infty]$ and therefore $s \in t \cdot h(C^*)$. It follows $s \cdot h(C^*) = t \cdot h(C^*)$. Since $\beta \not\in \bigcup_{D \subseteq C} \Gamma^* D^\infty$, there must be a preimage of $f$ of full alphabet $C$. Therefore, $\beta \in [t][f]^\omega \subseteq L$.

The alphabetic topology above is a refinement of the well-known Cantor topology. The Cantor topology is given by the basis $u\Gamma^\infty$ for $u \in \Gamma^*$. An $\omega$-regular language $L$ is a Boolean combination of open sets in the Cantor topology if and only if $[s][e]^\omega \subseteq L$ if and only if $[t][f]^\omega \subseteq L$ for all linked pairs $(s,e)$ and $(t,f)$ of the syntactic monoid of $L$ with $s \mathcal{R} t$; cf. [3,9,20]. Here $s \mathcal{R} t$ denotes one of Green’s relations: $s \mathcal{R} t$ if and only if $s \cdot \text{Synt}(L) = t \cdot \text{Synt}(L)$. Theorem 2 is a similar result, but one had to consider the alphabetic information of the linked pairs. Hence, one does not have $s \mathcal{R} t$ as condition, but rather $\mathcal{R}$-equivalence within a certain alphabet $C$. 

![Figure 1](image-url)
Remark 1 The strict alphabetic topology on \( \Gamma^\infty \), which is introduced in [3],
is given by the basis \( \{ u A^\infty \cap A^m \mid u \in \Gamma^*, A \subseteq \Gamma \} \) and the open sets are of
the form \( \bigcup_A W_A A^\infty \cap A^m \) with \( W_A \subseteq \Gamma^* \). Reusing the proof of Theorem 2
it turns out, that it is equivalent to be a Boolean combination of open sets in the alphabetic topology and in the strictly alphabetic topology. Since \( uA^\infty = \bigcup_{B \subseteq A} u A^\infty B^\infty \cap B^m \), every open set in the alphabetic topology is also open in the strict alphabetic topology. Further, one can adapt the proof of “2 \( \Rightarrow \) 3” of Theorem 2 to show that if \( L \) is a Boolean combination of open sets in the strict alphabetic topology, then item 3 of Theorem 2 holds.

Using Theorem 2, we show that the problem whether a given \( \omega \)-regular language is a Boolean combination of open sets in the alphabetic topology is PSPACE-complete.

Proposition 1 Given a Büchi automaton \( A \), deciding whether \( L(A) \) is a Boolean combination of open sets in the alphabetic topology is in PSPACE.

Proof We use condition 3 of Theorem 2. A canonical recognizing monoid \( M \) is the three-valued transition matrix monoid as described in [9, Chapter II.7]. Elements of this monoid can be stored in space \( O(n^3) \) where \( n \) is the size of the state set of \( A \). Let \( h \) be the homomorphism \( h : \Gamma^* \to M \). Since PSPACE is closed under complementation, it suffices to find linked pairs \((s, e), (t, f)\) such that there exists an alphabet \( C \) and words \( \hat{e}, \hat{f} \) with \( h(\hat{e}) = e, h(\hat{f}) = f \), \( \text{alph}(\hat{e}) = \text{alph}(\hat{f}) = C \), \( s \cdot h(C^*) = t \cdot h(C^*) \) and \([s][e]^\infty \subseteq L \), but \([t][f]^\infty \not\subseteq L \).

We guess the elements \( s, t, e, f \), the alphabet \( C \) and check non-deterministically the existence of the words \( \hat{e} \) and \( \hat{f} \) such that \( h(\hat{e}) = e, h(\hat{f}) = f \) and \( \text{alph}(\hat{e}) = \text{alph}(\hat{f}) = C \). Note that the check \( x \in t \cdot h(C^*) \) is possible in PSPACE by guessing the word in \( C^* \) letter by letter. Thus, by closure under complementation, one can check in PSPACE whether \( x \not\in t \cdot h(C^*) \). This implies, again by closure under complementation, that the question whether \( s \cdot h(C^*) = t \cdot h(C^*) \) is also in PSPACE. Consequently, the problem whether \( L(A) \) is a Boolean combination of open sets in the alphabetic topology is in PSPACE. \( \Box \)

Proposition 2 Given a Büchi automaton \( A \), it is PSPACE-hard to decide whether \( L(A) \) is a Boolean combination of open sets in the alphabetic topology.

Proof We reduce the universality problem of non-deterministic finite automaton over the alphabet \( \Gamma = \{a, b\} \) to this problem [6]. Let \( A' \) be an NFA with \( n \) states. We require \( A' \) to accept one word for every possible subalphabet; thus, we check whether \( 1, a, b, ab \in L(A') \). If one of these checks fails, we directly know that \( A' \) is not universal. Let \( c \not\in \Gamma \) be a new letter. Let \( A \) be a new Büchi automaton such that

\[
L(A) = \{cw_1cw_2cw_3 \cdots \mid \text{there are infinitely many } i \text{ with } w_i \in L(A') \}.
\]

The automaton \( A \) can be constructed in polynomial time. If \( L(A') = \Gamma^* \), we have \( L(A) = (\Gamma \cup \{c\})^\infty \setminus (\Gamma \cup \{c\})^* \Gamma^\infty \) and therefore \( L(A) \) is a Boolean combination of open sets. If \( L(A') \neq \Gamma^* \), let \( w \notin L(A') \) and \( h : (\Gamma \cup \{c\})^* \to M \).
be the syntactic homomorphism of \( L(A) \). Note that \( w \neq 1 \) since \( 1 \in L(A') \).

Choose a word \( v \in \{a, b, ab\} \) such that \( \text{alph}(v) = \text{alph}(w) \) and consider the linked pairs \([cv][cw]^{\omega} \subseteq L(A)\) and \([cv][cw]^{\omega} \subseteq L(A)\). The elements \( h(cv) \) and \( h(cv) \) are idempotent, and \( \text{alph}(cv) = \text{alph}(cw) = C \) and \( s = t = h(cv) \) implies \( s \cdot h(C^{\omega}) = t \cdot h(C^{\omega}) \). Hence, by Theorem 2, the language \( L(A) \) cannot be a Boolean combination of open sets in the alphabetic topology. \( \square \)

4 The fragment \( \mathbb{B}\Sigma_2 \)

Place and Zeitoun have shown that \( \mathbb{B}\Sigma_2 \) is decidable over finite words. In particular, they have shown that, given the syntactic homomorphism of a language \( L \subseteq \Gamma^{\omega} \), it is decidable if \( L \in \mathbb{B}\Sigma_2 \). Since every pseudovariety is generated by its syntactic monoids, the result of Place and Zeitoun can be stated as follows:

**Theorem 3** ([13]) The pseudovariety \( V_2 \) corresponding to the \( \mathbb{B}\Sigma_2 \)-definable languages in \( \Gamma^{\omega} \) is decidable.

Over infinite words, definability in \( \mathbb{B}\Sigma_2 \) does not only depend on the membership of the syntactic monoid in \( V_2 \). Intuitively, the main reason is that one needs to restrict the ‘infinite behaviour’. The following example shows that this restriction also depends on the ‘finite behaviour’.

**Example 1** Let \( L_1 = (a^+ \{b, bb\})^* a^{\omega}, L_2 = (a^+ b)^{\omega}, \) and \( L_3 = (a^+ \{b, bb\})^* (a^+ b)^{\omega} \).

The syntactic monoids of all these languages are in \( V_2 \). However, only \( L_1 \) and \( L_2 \) are definable in \( \mathbb{B}\Sigma_2 \); the language \( L_3 \) is not definable in \( \mathbb{B}\Sigma_2 \).

Our second main result characterizes the \( \mathbb{B}\Sigma_2 \)-definable \( \omega \)-regular languages. We use Theorem 3 as a black-box result.

**Theorem 4** Let \( L \subseteq \Gamma^{\infty} \) be \( \omega \)-regular. Then the following are equivalent:

1. \( L \) is a finite Boolean combination of monomials.
2. \( L \) is definable in \( \mathbb{B}\Sigma_2 \).
3. The syntactic homomorphism \( h \) of \( L \) satisfies:
   a. \( \text{Synt}(L) \in V_2 \) and
   b. for all linked pairs \((s, e), (t, f)\) it holds that if there exists an alphabet \( C \) and words \( \hat{e}, \hat{f} \) with \( h(\hat{e}) = e, h(\hat{f}) = f \), \( \text{alph}(\hat{e}) = \text{alph}(\hat{f}) = C \) and \( s \cdot h(C^{\omega}) = t \cdot h(C^{\omega}) \), then \( [s][e]^{\omega} \subseteq L \Leftrightarrow [t][f]^{\omega} \subseteq L \).

Note that condition 3 of Theorem 4 is decidable: 3a is decidable by Theorem 3 and 3b is decidable by Proposition 1. \(^1\) We start with the difficult direction “3 \( \Rightarrow \) 1” in the proof of Theorem 4. This is Proposition 3 below, together with Theorem 2. The following lemma is an auxiliary result for Proposition 3.

\(^1\) During the preparation of this submission, we learned that Pierron, Place and Zeitoun [10] independently found another proof for the decidability of \( \mathbb{B}\Sigma_2 \) over infinite words. For documenting the independency of the two proofs, we also include the technical report [7] of this paper in the list of references.
Lemma 2 For all \( k \) there exists a number \( \ell \) such that for every set \( \{ M_1, \ldots, M_d \} \) of \( k \)-monomials over \( \Gamma^* \) and every \( w \) with \( w \in M_i \) for all \( i \in \{1, \ldots, d\} \), there exists an \( \ell \)-monomial \( N \) over \( \Gamma^* \) with \( w \in N \) and \( N \subseteq \bigcap M_i \).

Proof Since the number of \( k \)-monomials over \( \Gamma^* \) is bounded, this induces a bound on \( d \) and one can iterate the statement. Therefore, it suffices to show the case \( d = 2 \). Consider two \( k \)-monomials \( M_1 = A_0a_1A_1^*a_2 \cdots A_{n-1}^*a_nA_n^* \) and \( M_2 = B_0b_1B_1^*b_2 \cdots B_{m-1}^*b_mB_m^* \). Since \( w \in M_1 \) and \( w \in M_2 \), it admits factorizations \( w = u_0a_1u_1a_2 \cdots u_{n-1}a_nu_n \) and \( w = v_0b_1v_1b_2 \cdots v_{m-1}b_mv_m \), such that \( u_i \in A_i^* \) and \( v_j \in B_j^* \). The factorizations mark the positions of the \( a_i \)s and the \( b_j \)s and pose an alphabetic condition for the factors in between. Thus, there exists a factorization \( w = w_0c_1w_1c_2 \cdots w_{\ell-1}c_\ell w_\ell \), such that the positions of \( c_i \) are exactly those that are marked by \( a_i \) or \( b_j \), i.e., \( c_i = a_j \) or \( c_i = b_j \) for some \( j \). The words \( w_i \) are over some alphabet \( C_i \) such that \( C_i = A_j \cap B_k \) for some \( j \) and \( k \) induced by the factorizations. In the case of consecutive marked positions, one can set \( C_i = \emptyset \). Thus, we obtain a monomial \( N = C_0^*c_1C_1^*c_2 \cdots C_{\ell-1}^*c_\ell C_\ell^* \) with \( C_\ell = A_n \cap B_m \). An illustration of this construction can be found in Figure 2. By construction \( N \subseteq M_1 \), \( N \subseteq M_2 \) and \( w \in N \) holds. Since there are only finitely many monomials of degree \( k \), the size of the number \( \ell \) is bounded. \( \square \)

An analysis of the proof of Lemma 2 yields the bound \( \ell \leq n_k \cdot k \), where \( n_k \) is the number of distinct \( k \)-monomials over \( \Gamma^* \). Next, we show that a language which is in \( V_2 \) and is a Boolean combination of alphabetic open sets is a finite Boolean combination of monomials. One ingredient of the proof is Lemma 2: we are able to compress the information of a set of \( k \)-monomials which contain a fixed word into the information of a single \( \ell \)-monomial that contains this fixed word.

Proposition 3 Let \( L \subseteq \Gamma^\infty \) be a Boolean combination of alphabetic open sets such that \( \text{Synt}(L) \in V_2 \). Then \( L \) is a finite Boolean combination of monomials.

Proof Let \( h : \Gamma^* \to \text{Synt}(L) \) be the syntactic homomorphism of \( L \) and consider the languages \( h^{-1}(p) \) for \( p \in \text{Synt}(L) \). By Theorem 3 we obtain \( h^{-1}(p) \in
\[ \alpha = \begin{array}{c|c} u' & \alpha' \\ \hline \end{array} \quad \beta = \begin{array}{c|c} \hat{v}' & \beta' \\ \hline x & \end{array} \]

Figure 3 Factorization of \( \alpha \) and \( \beta \) in the proof of Proposition 3

exists a factorisation \( \exists t \in \{0, 1\}^n \) and there exists linked pairs \((s, e)\) and \((t, f)\) such that \( s = h(u), t = h(v), \alpha' \in [e]^{\omega} \) and \( \beta' \in [f]^{\omega} \).

Note that, by the choice of \( u', v' \), we have \( \alpha', \beta' \in C^\infty \). We show that \( s \cdot h(C') = t \cdot h(C') \), which implies \( \beta \in L \) by Theorem 2. By symmetry, it suffices to show \( t \in s \cdot h(C') \). Consider the set of \( k \)-monomials \( N_0 = N_i \cap C^\infty \) which hold at \( u \), i.e., such that \( u \in N_0 \) and \( \alpha' \in C^\infty \). By the choice of \( \ell \), there exists an \( \ell \)-monomial \( N' \) such that \( u \in N' \) and \( N' \subseteq \bigcap N_0 \). Since \( u \in N' \), we obtain \( \alpha \in N := N' C^\infty \) and by \( \alpha \equiv \beta \) the membership \( \beta \in N \) holds. By construction of \( v \), there exists a word \( \hat{v} \) with \( \hat{v} \leq v' \leq v, \hat{v} \in N' \) and \( \beta \in C^\infty \) with \( \beta \) being defined by \( \beta = \hat{v} \beta \). Let \( v = \hat{v}x \), then \( x \in C^\infty \). The situation is depicted in Figure 3. We show that \( ux \equiv v \) which then implies \( t \in sh(C') \).

Let \( M \) be a \( k \)-monomial. If \( ux \in M \), then there exists a factorization \( M = M_1 M_2 \) where \( M_1, M_2 \) are \( k \)-monomials with \( u \in M_1 \) and \( x \in M_2 \). Since \( u \beta' \in M_1 C^\infty \), we obtain \( \hat{v} \in N' \subseteq M_1 \) by the definition of \( N' \). We conclude that \( v = \hat{v}x \in M_1 M_2 = M \).

If \( v = \hat{v}x \in M \), then there exists a factorization of the monomial \( M = M_1 M_2 \) where \( M_1, M_2 \) are \( k \)-monomials with \( \hat{v} \in M_1 \) and \( x \in M_2 \). Since \( (c_1 c_2 \cdots c_m)^k \) is a scattered subword of \( x \), there must be some \( A_i \) in the monomial \( M_2 \) such that \( C \subseteq A_i \) by the pigeonhole principle. Thus, there exists a factorisation \( M_2 = M_2_1 M_2_2 \) in \( k \)-monomials \( M_2_1, M_2_2 \) such that \( M_2_1 \cdot C^* = M_2_1 \). Let \( x = x' x'' \) such that \( x' \in M_2 \) and \( x'' \in M_2_2 \) and
consider $\beta = \beta x' \in M_1 M_{21} \cdot C^\infty$. Since $\alpha \equiv^\infty \beta$, we obtain $\alpha \in M_1 M_{21} \cdot C^\infty$. By construction, some prefix of $u$ is in $M_1 M_{21}$ and by $M_{21} \cdot C^* = M_{21}$ and $x' \in C^*$, we obtain $ux' \in M_1 M_{21}$. Thus, $ux = ux' \cdot x'' \in M_1 M_{21} \cdot M_{22} = M$ holds. We conclude $ux \equiv_k v$ and thus $t = h(v) = h(ux) \in s \cdot h(C^*)$. \hfill \Box

It is well-known, that the direct product $(g \times h) : \Gamma^* \to M \times N, w \mapsto (g(w), h(w))$ of the homomorphisms $g : \Gamma^* \to M$ and $h : \Gamma^* \to N$ recognizes Boolean combinations:

Lemma 3 Let $L$ and $K$ be languages such that $L$ is recognized by $g : A^* \to M$ and $K$ is recognized by $h : A^* \to N$. Then, any Boolean combination of $L$ and $K$ is recognized by $(g \times h)$.

Proof Since $L \cap [s][e]^\omega \neq \emptyset$ implies $[s][e]^\omega \subseteq L$ for some linked pair $(s, e)$, we obtain $L = \cup \{[s][e]^\omega : [s][e]^\omega \cap L \neq \emptyset\}$ for the complement of $L$. Thus, it suffices to show that $L \cup K$ is recognized by $(g \times h)$. Obviously, $L$ is covered by $[(s, t)][(e, f)]^\omega$, where $(s, e)$ is a linked pair of $M$ with $[s][e]^\omega \subseteq L$ and $(t, f)$ is any linked pair of $N$. Similarly one can cover $K$ and thus $M \times N$ recognizes $L \cup K$. \hfill \Box

Next, we show that the algebraic characterisation $V_2$ of $\mathbb{B} \Sigma_2$ over finite words also holds over finite and infinite words simultaneously. The proof of this is based on the fact that the algebraic part of the characterisation of $\Sigma_2$ over finite words and finite and infinite words is the same [3] and on the fact that every language of $\Sigma_2$ is in $\mathbb{B} \Sigma_2$, i.e., $V_{3/2} \subseteq V_2$.

Lemma 4 If $L \subseteq \Gamma^\infty$ is definable in $\mathbb{B} \Sigma_2$, then $\operatorname{Synt}(L) \in V_2$.

Proof By definition, $L \in \mathbb{B} \Sigma_2$ implies that $L$ is a Boolean combination of languages $L_i \in \Sigma_2$. We have $\operatorname{Synt}(L_i) \in V_{3/2} \subseteq V_2$ by [3]. Since $L$ is a Boolean combination of $L_i$, the direct product of all $\operatorname{Synt}(L_i)$ recognizes $L$ by Lemma 3. In particular, $\operatorname{Synt}(L)$ is a divisor of the direct product of $\operatorname{Synt}(L_i)$ by Lemma 1. Hence, we obtain $\operatorname{Synt}(L) \in V_2$. \hfill \Box

The proof that monomials are definable in $\Sigma_2$ is straightforward which yields:

Lemma 5 Every monomial $L \subseteq \Gamma^\infty$ is definable in $\Sigma_2$.

Proof Let $L = A_0 a_1 A_1^* a_2 \cdots A_{n-1} a_n A_n^\infty$. The $\Sigma_2$-formula

$$\exists x_1 \ldots \exists x_n \forall y : \bigwedge_{i=1}^n \lambda(x_i) = a_i \land \bigwedge_{i=1}^{n-1} x_i < y < x_{i+1} \Rightarrow \lambda(y) \in A_i \land (y > x_n \Rightarrow \lambda(y) \in A_n) \land (y < x_1 \Rightarrow \lambda(y) \in A_0).$$

defines $L$. \hfill \Box

Combining our results we are ready to prove Theorem 4.
Proof (Theorem 4) “1 ⇒ 2”: Since \( L \) is closed under Boolean combinations, it suffices to find a \( \Sigma_2 \)-formula for a single monomial. This is provided by Lemma 5.

“2 ⇒ 3”: 3a is proved by Lemma 4. Since \( A_n^0 a_1 A_1^1 a_2 \cdots A_{n-1}^1 a_{n-1} A_1^\infty \) is a set of finite words, a monomial \( A_n^0 a_1 A_1^1 a_2 \cdots A_{n-1}^1 a_{n-1} A_1^\infty \) is open in the alphabetic topology. The languages in \( \Sigma_2 \) are unions of such monomials [3] and thus languages in \( \mathcal{B} \Sigma_2 \) are Boolean combinations of open sets. This implies 3b by Theorem 2.

“3 ⇒ 1”: This is Proposition 3 and Theorem 2. \( \square \)

Example 2 In this example we show that \( \text{Synt}(L) \in \mathcal{V}_2 \) for some language \( L \subseteq \Gamma^\infty \) does not imply \( L \in \mathcal{B} \Sigma_2 \), i.e., the topological property 3b of Theorem 4 is necessary. For this define \( L = \{ (a, b)^+ \} \). We show that \( \text{Synt}(L) \in \mathcal{V}_2 \), but \( L \) is not a Boolean combination of open sets of the alphabetic topology. Computing the syntactic monoid of \( L \) yields \( \text{Synt}(L) = \{ 1, a, b, aa, ab, ba \} \). The equations \( b^2 = b, xaa = aax = aa \) and \( bab = b \) hold in \( \text{Synt}(L) \). In particular, \( (ab)^2 = ab \) and \( (aa)^2 = aa \). Thus, \( (s, e) = (aa, aa) \) and \( (t, f) = (aa, ab) \) are linked pairs. Let \( h \) denote the syntactic homomorphism of \( L \). Choosing \( aab \) as a preimage for \( aa \in \text{Synt}(L) \) yields the alphabetic condition \( \text{alph}(aab) = \text{alph}(ab) = C \) on the idempotents. Since \( s = t \), we trivially have \( s \cdot h(C^*) = t \cdot h(C^*) \). However, \( [aa][ab]^\omega \cap L = \emptyset \) but \( [aa][aa]^\omega \subseteq L \). Thus, \( L \) does not satisfy the topological condition 3b of Theorem 4. It remains to check \( \text{Synt}(L) \in \mathcal{V}_2 \).

It is enough to show that the preimages are in \( \mathcal{B} \Sigma_2 \).

\[
\begin{align*}
  - [1] &= 1 & - [a] = (ab^+)^*a & - [b] = (b^+a)^+ & - [ba] = (b^+a)^+ \\
  - [a] &= (ab^+)^*a & - [ab] = (ab^+)^+ & - [aa] &= \{a, b\}^+ \{a, b\}^*
\end{align*}
\]

One can find \( \mathcal{B} \Sigma_2 \) formulas for these languages, e.g., \( [ab] = L(\varphi) \) with

\[
\varphi \equiv (\exists x \forall y: x \leq y \land \lambda(x) = a) \land (\exists x \forall y: x \geq y \land \lambda(x) = b) \land \\
(\forall x \forall y: x \geq y \lor (\exists z: x < z < y) \lor (\lambda(x) \neq \lambda(y)))
\]

and thus \( \text{Synt}(L) \in \mathcal{V}_2 \).

\( \diamond \)

5 Summary and Open Problems

The alphabetic topology is an essential ingredient in the study of the fragment \( \Sigma_2 \). Thus, in order to study Boolean combinations of \( \Sigma_2 \) formulas, i.e., the fragment \( \mathcal{B} \Sigma_2 \) over infinite words, we considered Boolean combinations of open sets in the alphabetic topology. It turns out that it is decidable whether an \( \omega \)-regular language is a Boolean combination of open sets. This does not follow immediately from the decidability of the open sets. We used linked pairs of the syntactic homomorphism (which are effectively computable) to get decidability of the topological condition. Combining this result with the decidability of
we obtained an effective characterization of $B\Sigma_2$ over $\Gamma^\infty$, the finite and infinite words over the alphabet $\Gamma$.

In this paper we dealt with $B\Sigma_2$, which is the second level of the Straubing-Th´erien hierarchy. Another well-known hierarchy is the dot-depth hierarchy. On the level of logic, the difference between the Straubing-Thérien hierarchy and the dot-depth hierarchy is that formulas for the dot-depth hierarchy may also use the successor predicate $y = x + 1$. A deep result of Straubing is that over finite words each level of the Straubing-Thérien hierarchy is decidable if and only if it is decidable in the dot-depth hierarchy [17]. Thus, the decidability result for $B\Sigma_2$ by Place and Zeitoun also yields decidability of $B\Sigma_2[\text{\textless},+1]$. The fragment $\Sigma_2[\text{\textless},+1]$ is decidable for $\omega$-regular languages [5]. This result also uses topological ideas, namely the factor topology. The open sets in this topology describe which factors of a certain length $k$ may appear in the “infinite part” of the words. The study of Boolean combinations of open sets in the factor topology is an interesting line of future work, and it may yield a decidability result for $B\Sigma_2[\text{\textless},+1]$ over infinite words. We believe that the topological approach could also be used for transfer results form finite to infinite words on the other levels of the Straubing-Thérien or the dot-depth hierarchy. However, the alphabetic or the factor topology does not seem to be the right approach for describing the ‘infinite’ behaviour on the levels greater than 2; thus, more refined topologies need to be considered.

Another interesting class of predicates are modular predicates. In [8] the authors have studied $\Sigma_2[\text{\textless},\text{MOD}]$ over finite words. The results of [8] can be generalized to infinite words by adapting the alphabetic topology to the modular setting. As for successor predicates, we believe that an appropriate effective characterization of this topology might help in deciding $B\Sigma_2[\text{\textless},\text{MOD}]$ over infinite words. To the best of our knowledge, however, modular predicates have not yet been considered over infinite words.

References


