On the Topological Complexity of MSO+U and Related Automata Models

Szczepan Hummel, Michał Skrzypczak, and Szymon Toruńczyk*

University of Warsaw

Abstract. We show that Monadic Second Order Logic on ω-words extended with the unbounding quantifier (MSO+U) can define non-Borel sets. We conclude that there is no model of nondeterministic automata with a Borel acceptance condition which captures all of MSO+U. We also give an exact topological complexity of the classes of languages recognized by nondeterministic ωB-, ωS- and ωBS-automata studied by Bojańczyk and Colcombet in [BC06]. Furthermore, we show that corresponding alternating automata have higher topological complexity than nondeterministic ones — they inhabit all finite levels of the Borel hierarchy.

Introduction

Motivation and background. The notion of an ω-regular language is well established in the theory of automata. This class of languages carries over to ω-words many of the good properties of regular languages of finite words. It can be described using automata, namely by nondeterministic Büchi automata, or the equivalent deterministic Muller automata, and also alternating automata. In terms of logic, they are equivalent to both Monadic Second Order Logic (MSO) and Weak Monadic Second Order Logic (Weak MSO) – the fragment of MSO where quantifiers may only bind finite sets. Such connections between logic and automata are extremely important in the field of verification and specification.

Recently, in [Boj10, Boj09, BT09] it has been suggested that there are other robust classes of languages of ω-words, extending the canonical notion. It has been advocated that natural examples of languages that *might* be seen as regular (for instance, because of a finite Myhill-Nerode index) are languages such as

$$L_B = \{a^{n_1}ba^{n_2}b\ldots \,|\, \limsup n_i < \infty\},$$
$$L_S = \{a^{n_1}ba^{n_2}b\ldots \,|\, \liminf n_i = \infty\}.$$

which are not ω-regular in the usual sense.

These papers describe several such classes of languages, but none of them is known to have all the robust properties of ω-regular languages. On one hand, automata models often allow deciding emptiness, on the other hand, a class

* All authors have been supported by the Polish government grant no. N206 008 32/0810.

P. Hliněný and A. Kučera (Eds.): MFCS 2010, LNCS 6281, pp. 429–440, 2010.

described in terms of logic usually has good closure properties. Ideally, one would like to have a class of languages which can be described both in terms of automata and in terms of logic, as in the case of ω-regular languages.

The connection between automata and logic is better understood when restricted to deterministic automata and weak logics. Deterministic max-automata, introduced in [Boj09], have an alternative description in terms of logic, namely Weak MSO extended with the unbounding quantifier U, which is defined so that the formula $\mathsf{U}X.\varphi(X)$ is equivalent to writing:

"$\varphi(X)$ is satisfied by arbitrarily large finite sets X of positions"

Thus U is suited to capture (the complement of) L_B.

As shown in [BT09], the correspondence between deterministic automata and weak logics extends to various other classes of languages, for instance deterministic min-max-automata are equivalent to a logic called Weak MSO $+\mathsf{U}+\mathsf{R}$, and embrace both L_B and L_S.

Problems arise when we look for classes closed under set-theoretic projection, which corresponds to full existential quantification in logic or to nondeterminism on the automata side. In [BC06], ωBS-automata were defined as automata equipped with counters which can be incremented or reset, but not read. The acceptance condition may require a counter to be bounded (the B-condition) or convergent to ∞ (the S-condition). Although nondeterministic ωBS-automata are not closed under complementation, there is a partial fix to this problem. The main technical result of [BC06] shows that the complement of a language defined by an ωB-automaton is accepted by an ωS-automaton and vice versa, where the two are subclasses of ωBS-automata using only the B-condition or the S-condition, respectively. However, boolean combinations of ωB-automata are not closed under existential quantification.In consequence, it seems unlikely to find any sensible logic corresponding to either ωBS-automata or boolean combinations of ωB-automata. To try to overcome these issues, one might consider alternating ωBS-automata. So far, it was not known whether in the ωBS-setting there is any advantage of alternation over nondeterminism.

From the logic side, in order to capture nondeterminism, it seems natural to consider the logic MSO$+\mathsf{U}$, which extends the class of languages recognized by ωBS-automata. However, we face the essential question, whether this logic is decidable. In this paper we analyze large classes of automata to seek for a model capable of capturing MSO$+\mathsf{U}$.

Topological complexity. Our approach is to investigate from the topological viewpoint the classes of languages mentioned above and also explore other large classes of automata. Such an analysis can guide in constructing a suitable model of automata for MSO$+\mathsf{U}$, or show that such a model cannot exist. For instance, we prove that a large class of models of nondeterministic automata cannot capture MSO$+\mathsf{U}$. We also discover that alternating ωBS-automata are strictly more expressive than boolean combinations of nondeterministic ωBS-automata.

Let us illustrate these techniques here with some elementary examples. The language L_S corresponds to a property of a sequence of numbers $n_1, n_2, \ldots$ – namely, being convergent to ∞ – which is equivalent to:

$$\bigwedge_k \bigvee_m \bigwedge_{i \geq m} (n_i > k).$$

Being able to define L_S with a formula with three alternations of logical conectives directly translates into its topological complexity — we say that L_S is (at most) *in the third level of the Borel hierarchy.* Similarly, it is easy to see that L_B is in the second level of the Borel hierarchy.

A run of a *deterministic* automaton is a *continuous* function which maps an input word into a sequence of states. The recognized language is the inverse image of the set of accepting runs under this mapping. A basic property of continuous mappings proves that the language is topologically not more complex than this set of accepting runs.

This immediately yields several results: ω-regular languages occupy at most the first two levels of the Borel hierarchy, since such is the complexity of the Muller acceptance condition; max-automata (equivalently, Weak MSO+U) also fall into the first two levels of the hierarchy, as their acceptance condition is L_B.

As a sample impossibility result (stated in [Boj09]), observe that deterministic max-automata do not recognize the language L_S since it is in the third level of the Borel hierarchy and provably not lower.

In Section 2 we exhibit an example of a language M definable in MSO+U which is analytic-complete, i.e. lays beyond the infinite Borel hierarchy. This instantly proves that there can be no deterministic model of automata with a Borel acceptance condition which captures all of MSO+U.

The above method does not give upper bounds for the complexity of languages defined by *nondeterministic* automata. Such bounds require some nontrivial combinatorial results, usually determinization (as for ω-regular languages). We, in turn, use two difficult results to give upper bounds concerning nondeterministic automata.

In Section 2, Corollary 1, we use a strong topological result of Souslin to conclude that *no model of nondeterministic automata* with a Borel acceptance condition can capture all of MSO+U.

In Section 3, we use the combinatorial complementation result of [BC06] which allows us to compute the topological complexity of languages defined by nondeterministic ωB-, ωS- and ωBS-automata. In particular, ωBS-automata, which form the largest among all the subclasses of MSO+U that we know to have decidable emptiness, reach only the fourth level of the Borel hierarchy.

However, there is still some hope in alternating automata. In Section 4 we define the alternating variant of ωBS-automata and we prove that they inhabit at least all finite levels of the Borel hierarchy. This implies that alternating ωBS-automata are strictly more powerful than (boolean combinations of) nondeterministic ωBS-automata. However, we do not know whether they can recognize analytic languages, such as M. This leaves open the question whether this model captures MSO+U.

1 Basic Notions

Logic. We assume familiarity with the *Monadic Second Order Logic* (MSO). Fix an alphabet A. We denote positions of ω-words using symbols $x, y, \ldots$ and sets of positions with symbols $X, Y, \ldots$. For $a \in A$, the unary predicate P_a holds in all positions of the word where an a stands. It is well known that languages that can be described by this logic are exactly ω-regular languages.

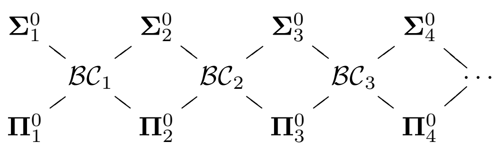
MSO+U allows building formulas using MSO constructs and an additional quantifier U, called the *unbounding quantifier*, defined as follows. The formula $\mathsf{U}X.\varphi(X)$ holds in a word w if $\varphi(X)$ is satisfied for arbitrarily large finite sets X of positions. Formally, $\mathsf{U}X.\varphi(X)$ is equivalent to:

$$\bigwedge_{n\in\mathbb{N}} \exists X.\ (\varphi(X)\ \wedge\ n < |X| < \infty)$$

The canonical examples of languages that can be described are the languages L_B and L_S defined in the introduction.

Topology. For a fixed alphabet A, we treat A^ω as a topological space. A basic open set is determined by a prefix $s \in A^*$ and is of the form $s \cdot A^\omega$. Other open sets are obtained by taking unions of basic open sets. If A is finite, this topological space is homeomorphic (i.e. topologically isomorphic) to the Cantor space.

The Borel hierarchy. The Borel hierarchy is defined inductively. We assist the definition with the following diagram[1].



$\mathbf{\Sigma}^0_1$ denotes the class of open sets and $\mathbf{\Pi}^0_1$ denotes the class of closed sets, i.e. complements of open sets. Having defined $\mathbf{\Sigma}^0_n$ and $\mathbf{\Pi}^0_n$, we define $\mathcal{BC}_n$ as (finite) boolean combinations of $\mathbf{\Sigma}^0_n$-sets and $\mathbf{\Pi}^0_n$-sets. In the next step, we define $\mathbf{\Sigma}^0_{n+1}$ as unions of countable families of $\mathcal{BC}_n$-sets and $\mathbf{\Pi}^0_{n+1}$ as intersections of countable families of $\mathcal{BC}_n$-sets. Note that for each n, $\mathbf{\Pi}^0_n$ consists of complements of $\mathbf{\Sigma}^0_n$-sets, and vice versa.

This way we define all *finite* levels of the Borel hierarchy, which is all we will need in this paper. Note that for each n, both $\mathbf{\Sigma}^0_n$ and $\mathbf{\Pi}^0_n$ are strictly contained in both $\mathbf{\Sigma}^0_{n+1}$ and $\mathbf{\Pi}^0_{n+1}$. In fact, in order to obtain a class which is closed under both complements and countable unions, one should continue the construction using transfinite induction up to level ω_1, where we arrive at the class of *Borel sets.* For these and other facts concerning the Borel hierarchy see e.g. [Sri98, Chapter 3.6].

[1] This diagram is more commonly presented with the larger class $\mathbf{\Delta}^0_{n+1} = \mathbf{\Sigma}^0_{n+1} \cap \mathbf{\Pi}^0_{n+1}$ in place of $\mathcal{BC}_n$. However, we will not use the classes $\mathbf{\Delta}^0_n$.

Analytic sets. The direct image of a Borel set under a continuous mapping may no longer be a Borel set. We call such sets *analytic*, and the class of all analytic sets is denoted $\mathbf{\Sigma}^1_1$. Complements of analytic sets are called *coanalytic* and form the class $\mathbf{\Pi}^1_1$. An important result in the theory, the theorem of Souslin (see e.g. [Kec95, Chapter 14.C]) states that if both a set and its complement are analytic then they are in fact Borel. It is worth mentioning that the Borel hierarchy and analytic sets are part of a bigger hierarchy of classes, called the projective hierarchy.

Topological complexity. A *topological complexity class* $\mathbf{C}$, for the needs of this paper is any of the classes $\mathbf{\Sigma}^0_n$, $\mathbf{\Pi}^0_n$ where n is a finite number (although the full Borel hierarchy has levels above ω), and the classes $\mathbf{\Sigma}^1_1$ and $\mathbf{\Pi}^1_1$. Analogously to complexity theory, we have the notions of *reductions* and *completeness*. Let A, B be two alphabets and let $K \subseteq A^\omega$ and $L \subseteq B^\omega$. We say that a continuous mapping $f\colon A^\omega \to B^\omega$ is a *reduction* of K to L if $K = f^{-1}(L)$. It is a simple property of continuous mappings that if L belongs to a topological complexity class $\mathbf{C}$ then so does K. The language L is called $\mathbf{C}$-*hard* iff any set $K \in \mathbf{C}$ can be reduced to L. We say that L is $\mathbf{C}$-*complete* if additionally $L \in \mathbf{C}$.

2 Non-borel Sets in MSO+U

In this section we show that the set B of trees on $\mathbb{N}$ (i.e. prefix closed subsets of $\mathbb{N}^*$) with an infinite branch is MSO+U-definable modulo some encoding. The set B is well known to be $\mathbf{\Sigma}^1_1$-complete (see [Kec95, Theorem 27.1]).

To simplify notation, we consider B as a subset of $\mathcal{T}$ — the set of infinite trees. Since $B \subseteq \mathcal{T}$ and $\mathcal{T} \subseteq 2^{\mathbb{N}^*}$ is in $\mathbf{\Pi}^0_2$, this restriction doesn't affect the topological complexity of B.

Let $\prec$ be some fixed order of type ω on $\mathbb{N}^*$. We continuously embed $\mathcal{T}$ into A^ω, where $A = \{a, b, c\}$. For a given vertex $v = (n_1, n_2, \ldots, n_m) \in \mathbb{N}^*$, let $K(v) = a^{n_1} b a^{n_2} b \ldots b a^{n_m} \in A^*$. For a given tree $T \in \mathcal{T}$, let $K(T) = K(v_1) c K(v_2) c \ldots \in A^\omega$, where v_i is the i'th vertex of T in the order $\prec$. It is easy to see that $K\colon \mathcal{T} \to A^\omega$ defined above is a homeomorphism onto its image.

Proposition 1. *There exists an MSO+U formula φ such that*

$$T \in B \quad \Leftrightarrow \quad K(T) \models \varphi.$$

Proof. Below, $\mathrm{pre}(G)$ denotes the set of prefixes of elements of G.

Lemma 1. *Let $T \in \mathcal{T}$ be a tree. The following conditions are equivalent:*

1. *T has an infinite branch,*
2. *T has an infinite set of vertices G such that for any subset S of* $\mathrm{pre}(G)$*, if S has bounded height then S is finite.*

The proof of the lemma is an easy application of König's Lemma.

To prove Proposition 1 it suffices to show that the second condition can be verified by a formula of MSO+U on $K(T)$. The construction of such a formula is described below.

For fixed $w \in A^\omega$, $S \subseteq \omega$, let $\psi(S)$ express the following properties:

– for each *block* of the form $c(a^*b)^*a^*c$ in w, S contains some initial segment of its positions,
– there is a bound r such that within every block the number of b's contained in S is bounded by r. Let r_S denote the minimal bound.

Let $\gamma(S)$ express that all a-blocks inside S are jointly bounded in length.

Let φ be an MSO+U formula expressing that there exists an infinite set $G \subseteq \omega$ containing only whole blocks of the form $c(a^*b)^*a^*c$, such that

$$\forall S \subseteq G.\ \psi(S) \Rightarrow \gamma(S) \tag{1}$$

The formula φ verifies the second condition of the lemma. □

Let us denote $M = L(\varphi) = \{w \in A^\omega : w \models \varphi\}$. Therefore we have shown that K is a reduction of B to M.

Proposition 2. *$M \subseteq A^\omega$ is a $\mathbf{\Sigma}^1_1$-set.*

Proof. Let Z be a set of pairs $(w, G) \subseteq A^\omega \times 2^\omega$ with G being a witness for φ and (w, G) satisfying the formula (1).

Note that for each $r_S \in \mathbb{N}$ and $G \subseteq \omega$, there is a maximal set $S \subseteq G$, satisfying $\psi(S)$ with given r_S bound. Such S depends continuously on (w, G). If we take $S \subseteq S' \subseteq G$, then $\gamma(S') \Rightarrow \gamma(S)$. Therefore, to check the validity of (1) it is enough to consider only countably many maximal S's (one per each $r_S \in \mathbb{N}$). The formula $\gamma(S)$ defines a Borel set. So Z is a countable intersection of Borel sets. Since $M = \pi_1(Z)$, M is $\mathbf{\Sigma}^1_1$. □

Therefore, we have proved:

Theorem 1. *$M \subseteq A^\omega$ is a $\mathbf{\Sigma}^1_1$-complete set (in particular, non-Borel) and is definable in MSO+U.*

Corollary 1. *There is no model of nondeterministic automata with a Borel acceptance condition, capturing all of MSO+U.*

Proof. Assume that for each MSO+U formula ψ, there exists a nondeterministic automaton $\mathcal{A}$ with a Borel acceptance condition $F \subseteq Q^\omega$, such that $L(\mathcal{A}) = L(\psi)$. In this proof the set Q can be even infinite (but countable), taking into account values of counters or any other additional state information. Then, $L(\psi)$ is a $\mathbf{\Sigma}^1_1$-set — it is the projection of the Borel set of pairs (w, ρ) such that ρ is a run of $\mathcal{A}$ on w.

Therefore, both $L(\varphi) = M$ and $L(\neg\varphi) = A^\omega \setminus M$ are $\mathbf{\Sigma}^1_1$-sets. By the theorem of Souslin, M must be Borel, which is a contradiction. □

3 ωBS-Automata

We will now define ωBS-automata as described in [BC06, Boj10]. They define a strict subclass of MSO+U, but it is the greatest subclass of MSO+U of which we know to have decidable emptiness.

An *ωBS-automaton* $\mathcal{A}$, as other nondeterministic finite automata, has a finite input alphabet A, a finite set Q of states and an initial state q_I. Apart from that it is equipped with a finite set Γ of counters. The counters can only be updated and cannot be read during the run. They will be used by the acceptance condition. A transition of the automaton is a transformation of states, as in standard NFA's, and additionally a finite sequence of counter updates. A counter update can be either an increment or a reset of a counter $c \in \Gamma$.

The value of a counter c is initially set to 0 and is incremented or reset according to the transitions in a run. For $c \in \Gamma$ we define a sequence $val_\rho(c)$, where $val_\rho(c)_i$ is the value of counter c right before its i-th reset in the run ρ. Note that if a counter c is reset only finitely many times then the sequence $val_\rho(c)$ is finite.

The acceptance condition of ωBS-automaton is a boolean combination of constraints that can be of one of the forms:

$$\limsup_i val_\rho(c)_i < \infty \qquad\qquad \liminf_i val_\rho(c)_i = \infty$$

The first constraint is called the *B-condition* (bounded), the second — the *S-condition* (strongly unbounded). In order that lim inf and lim sup make sense, the constraints implicitly require the corresponding sequences to be infinite.

It is a simple observation that the negation of a B-condition can be simulated using an S-condition and nondeterminism, and vice versa. Thanks to this fact we can consider automata with acceptance conditions that are *positive* boolean combinations of S- and B-conditions, without loss of expressive power.

We will use the notation $B(c)$ for the B-condition and $S(c)$ for the S-condition imposed on a counter c.

If the acceptance condition of an automaton is a positive boolean combination of B-conditions, the automaton is called an *ωB-automaton*. We similarly define *ωS-automata*.

Languages recognized by ωBS-automata (ωB-automata, ωS-automata) are called *ωBS-regular* (resp. *ωB-regular*, *ωS-regular*). An important result of [BC06] is that the complement of an ωB-regular language is an ωS-regular language and vice versa. Both classes are extensions of the class of ω-regular languages since the Büchi condition can be simulated by either a B-condition or an S-condition.

Example 1. The language L_S defined in the introduction can be recognized by an ωS-automaton. The automaton has one state and uses one counter that is increased when reading a letter a and is reset after each b. The acceptance condition is simply an S-condition on the only counter.

3.1 Complexity of ωB- and ωS-Regular Languages

Theorem 2. *Each ωB-regular language is in* $\mathbf{\Sigma}^0_3$.

Proof. Fix an ωB-automaton $\mathcal{A}$ recognizing a language L, and let us first assume that its accepting condition is a conjunction of B-conditions, i.e. is of the form:

$$\bigwedge_{c \in \Gamma_B} B(c)$$

Each of the considered counters is bounded iff there is a common bound k for all of them. Therefore L can be defined as:

$$L = \{w : \exists\rho. \bigwedge_{c\in\Gamma_B} val_\rho(c) \text{ is bounded but infinite}\}$$
$$= \bigcup_k \underbrace{\{w : \exists\rho. \bigwedge_{c\in\Gamma_B} val_\rho(c) \text{ is bounded by } k \text{ and infinite}\}}_{L_k},$$

where the quantification is over the set of all runs of $\mathcal{A}$ on w.

It is easy to see that for a fixed k, L_k can be recognized by a nondeterministic Büchi automaton. We simply store the counter values in the state and do not allow them to be incremented above k. The acceptance condition requires each of the counters $c \in \Gamma_B$ to be reset infinitely often. Hence L_k is ω-regular. Since all ω-regular languages are in $\mathcal{BC}_2$, $L \in \mathbf{\Sigma}^0_3$ as a countable union of $\mathcal{BC}_2$-sets.

In the general form, the acceptance condition of an ωB-automaton is a positive boolean combination of B-conditions. We can write such a condition in disjunctive normal form (DNF). The language accepted by this automaton is a union of languages corresponding to each disjunct. Hence it is in $\mathbf{\Sigma}^0_3$. □

Thanks to the complementation result of [BC06], we have:

Corollary 2. *Each ωS-regular language is in $\mathbf{\Pi}^0_3$.*

The complexity bounds given by Theorem 2 and Corollary 2 are tight.

Theorem 3. *There is a $\mathbf{\Sigma}^0_3$-complete set that is ωB-regular and a $\mathbf{\Pi}^0_3$-complete set that is ωS-regular.*

Proof. Because ωB-regular languages are complements of ωS-regular languages, it suffices to show only one of the claims.

We recall that the language L_S is in $\mathbf{\Pi}^0_3$ and ωS-regular. $\mathbf{\Pi}^0_3$-completeness of L_S follows from [Kec95, Exercise 23.2] via an obvious reduction. □

3.2 Complexity of ωBS-Regular Languages

Now we switch to languages recognized by automata that can use both S- and B-conditions. We prove the following.

Theorem 4. *Each ωBS-regular language is in $\mathbf{\Sigma}^0_4$.*

Proof. The proof, on the one hand, will use the result of Theorem 2 and, on the other hand, will repeat a similar reasoning.

Let us fix an ωBS-regular language L and an automaton $\mathcal{A}$ recognizing it. First consider an acceptance condition of the form:

$$\bigwedge_{c\in\Gamma_B} B(c) \quad \wedge \quad \bigwedge_{c\in\Gamma_S} S(c)$$

The language L can then be defined by:

$$L = \bigcup_k \underbrace{\left\{ w : \exists \rho. \begin{array}{l} \bigwedge_{c \in \Gamma_B} \; val_\rho(c) \text{ is bounded by } k \text{ and infinite} \\ \bigwedge_{c \in \Gamma_S} \; val_\rho(c) \text{ converges to } \infty \end{array} \right\}}_{L_k}$$

Note that each L_k language is ωS-regular, hence (by Theorem 2), it is in $\mathbf{\Pi}^0_3$. So L, as a countable union of such languages, is in $\mathbf{\Sigma}^0_4$.

A general acceptance condition can be written in disjunctive normal form (DNF). Again, the language accepted by such an automaton is a union of languages corresponding to each disjunct, so it is in $\mathbf{\Sigma}^0_4$. □

Now we show that the bound is tight. For that we consider the language, that was used in [BC06, Corollary 2.8] to show that the class of ωBS-regular languages is not closed under complements. Let

$$G = \left\{ a^{n_1} b a^{n_2} b \ldots : \begin{array}{l} \text{the sequence } n_1, n_2, \ldots \text{ can be partitioned into} \\ \text{a (possibly empty) bounded subsequence and} \\ \text{a (possibly empty) subsequence tending to } \infty \end{array} \right\}$$

The following fact is presented as an example in [TL93, page 595].

Lemma 2. *The language G is $\mathbf{\Sigma}^0_4$-complete.*

Now it suffices to note that the language G is ωBS-regular. It is proven in [BC06] (by showing an appropriate ωBS-regular expression), but it is straightforward to construct an automaton recognizing it.

4 Alternating ωBS-Automata

On the way towards finding a model of automata for the logic MSO+U we consider alternating ωBS-automata.

Alternating ωBS-automata are defined similarly as nondeterministic ωBS-automata. The difference is that the state space Q is split into $Q_\forall$ (universal states) and $Q_\exists$ (existential states).

We use standard game semantics for such automata. For a given alternating automaton $\mathcal{A}$ and word $w \in A^\omega$ we define a two-player game. A play in this game starts in the initial state of the automaton and in the first position of the word and proceeds by applying transitions of the automaton on the word w consistent with current state and a letter in current position in the word. Player $\forall$ chooses transitions when the automaton is in a state from $Q_\forall$, Player $\exists$ — from $Q_\exists$. Finally the play produces an infinite sequence of transitions consistent with consecutive letters of the word. The word w is accepted by the automaton iff Player $\exists$ has a winning strategy in the game with the winning condition of exactly the same form as an acceptance condition of nondeterministic ωBS-automata, i.e. a boolean combination of B- and S-conditions.

4.1 Languages Complete for the Classes $\mathbf{\Pi}^0_{2n}$

We will now present examples of languages of infinite words complete for the Borel classes $\mathbf{\Pi}^0_{2n}$, which are recognized by alternating ωBS-automata.

To make proofs easier, we will work with the spaces of sequences of vectors of numbers $\mathcal{N}_n = (\mathbb{N}^n)^\omega$. An easy embedding will transfer the results into the space of infinite words. For $n = 0$, the above definition gives a space of sequences of empty tuples, i.e. $\mathcal{N}_0 = \{\omega\}$.

Let us fix an alphabet $A = \{a, b, c\}$. We use encoding of a sequence of vectors into the space A^ω, where each vector $(z_n, z_{n-1}, \ldots, z_1)$ is mapped to the word $a^{z_n}ba^{z_{n-1}}b \ldots a^{z_1}c$. We will call the embedding defined this way $W_n \colon \mathcal{N}_n \to A^\omega$.

We will use the following notations to easily operate on sequences of vectors.

- For $\eta \in \mathcal{N}_n$ and $m \in \mathbb{N}$, let $\eta{\restriction}_m$ be a subsequence of η consisting of those vectors that have value m at the first coordinate.
- Let $\pi_n \colon \mathcal{N}_n \to \mathcal{N}_{n-1}$ be the projection which cuts off the first coordinate from each vector in a given sequence.

Definition 1. *Let $L_n \subseteq \mathcal{N}_n$ for $n > 0$ be the set of all $\eta \in \mathcal{N}_n$ such that*

$$\exists^\infty_{m_n} \exists^\infty_{m_{n-1}} \ldots \exists^\infty_{m_1} \exists^\infty_{x \in \omega} \eta(x) = (m_n, m_{n-1}, \ldots, m_1),$$

where $\exists^\infty$ stands for "exists infinitely many". Additionally, let $L_0 = \{\omega\} = \mathcal{N}_0$.

The following lemma describes the languages L_n in an inductive fashion.

Lemma 3. *For $n > 0$, $\eta \in L_n$ iff there exist infinitely many $m \in \mathbb{N}$ such that $\eta{\restriction}_m$ is an infinite sequence and $\pi_n(\eta{\restriction}_m) \in L_{n-1}$.*

The topological complexity of the languages $W_n(L_n)$ is presented as an example in [TL93, pages 595–596], here we only recall it.

Theorem 5. *For every $n>0$, the language $W_n(L_n)$ is $\mathbf{\Pi}^0_{2n+2}$-complete.*

Logic. Now we present MSO+U formulas describing the languages L_n. We do not formally prove that the formulas define exactly the desired sets, but they will serve as a guideline for us in the construction of alternating automata recognizing the languages.

First define a formula over $\mathcal{N}_n$, expressing boundedness of the first coordinates of vectors marked by X:

$$\mathrm{Bnd}_n(X) \equiv \exists_{k \in \mathbb{N}} \forall_{x \in X}\, \eta(x)_1 \leq k.$$

Now we build the formula for the language L_n inductively:

$$\varphi_n \equiv \forall X.\, \mathrm{Bnd}_n(X) \implies \exists Y.\, \mathrm{Bnd}_n(Y) \wedge (X \cap Y = \emptyset) \wedge (\varphi_{n-1}|_Y), \qquad (2)$$

where $\varphi_{n-1}|_Y$ is φ_{n-1} with all quantifiers restricted to Y and operating on $\mathcal{N}_n$ by ignoring the first coordinates of vectors, and φ_0 simply states that a sequence is infinite.

The formula (2) deals with sequences of vectors, but it is easy to rewrite it in such a way that it works on ω-words over A and defines $W_n(L_n)$. It is possible because properties like "being a maximal block of consecutive a's that correspond to the k-th coordinate of one of the vectors in a sequence" are expressible in MSO. Expressing Bnd_n in this context requires U.

4.2 Automata Construction

Theorem 6. *For each $n \in \mathbb{N}$ there is an alternating ωBS-automaton recognizing a $\mathbf{\Pi}^0_{2n+2}$-hard language.*

Proof. For a fixed n, it is possible to construct an ωBS-automaton recognizing exactly the language $W_n(L_n)$. However, to avoid some technical inconveniences, we construct an automaton $\mathcal{A}_n$ for which we only require that it accepts a word $W_n(\eta)$ iff $\eta \in L_n$.

The automaton will mimic the formula φ_n. The problem that we face is that alternation in automata and quantifier alternation in logic have different semantics. In logic, using the second order quantifier refers to choosing a set all at once, while in automata, players make decisions step by step (position by position). We will be able to overcome this problem using properties of the B-condition.

Automaton. The automaton $\mathcal{A}_n$ will be defined in the following way.

While reading the code of a sequence of vectors, before reading each vector Player $\forall$ decides if he selects the first component of the vector. If $\forall$ has not chosen the component, $\exists$ can choose it. If the component was chosen by $\forall$, counter a_n counts its length and then resets. If the component was chosen by $\exists$, counter e_n counts its length and then resets.

If the first component was chosen by $\exists$ then the procedure is repeated for the second component and for the counters a_{n-1} and e_{n-1}. We continue with the next components until Player $\exists$ does not select a component or all components of the vector are selected by $\exists$.

The whole process is repeated for all the vectors in a word.

Player $\forall$ can additionally reset any of a_i counters at any time (except the moment when it is actually incremented). This is to allow Player $\forall$ to select a finite (even empty) set.

The acceptance condition (winning condition for $\exists$ in the game) requires that among the counters $a_n, e_n, a_{n-1}, e_{n-1}, \ldots, a_1, e_1$, the left-most which is unbounded (or reset finitely many times) is an a-counter, or all counters are reset infinitely many times and are bounded.

Soundness. For a given word $w = W_n(\eta)$ such that $\eta \in L_n$, we have to prove that the existential player has a winning strategy in $\mathcal{A}_n$ on w. We proceed by induction. As stated above, $\eta \in L_n$ if and only if there exist infinitely many $m \in \mathbb{N}$ such that

$$\eta\restriction_m \text{ is infinite} \quad \text{and} \quad \pi_n\left(\eta\restriction_m\right) \in L_{n-1} \tag{3}$$

Player $\exists$ uses the following strategy. Let k be the greatest value of the first component among vectors selected by $\forall$ so far. Let m_k be the least m greater than k, for which condition (3) holds. Player $\exists$ selects a vector if its first component is equal to m_k.

Note that we may assume that k is increased only finitely many times during the run (otherwise Player $\forall$ loses). Hence, there exists a value m_{k0} that occurs at the first component of almost all vectors selected by Player $\exists$. By the assumption, $\pi_n(\eta\restriction_{m_{k0}}) \in L_{n-1}$. Since the set L_n is prefix-independent (i.e. for all $\nu \in (\mathbb{N}^n)^*$,

$\eta \in L_n$ iff $\nu\eta \in L_n$), also η restricted to the vectors selected by $\exists$, with the first component erased, belongs to L_{n-1}. It follows by inductive assumption that $\exists$ has a strategy on further components of vectors of this restricted sequence.

Induction basis: The automaton recognizing the set L_0 simply accepts all infinite sequences.

Correctness. Now let us take $w = W_n(\eta)$ such that $\eta \notin L_n$. We prove that the universal player has a winning strategy in $\mathcal{A}_n$ on w. We, again, proceed by induction. Note that if $\eta \notin L_n$ there exists m_0 such that for all $m \geq m_0$

$$\eta\restriction_m \text{ is finite} \quad \text{or} \quad \pi_n\left(\eta\restriction_m\right) \notin L_{n-1} \tag{4}$$

Player $\forall$ marks all the vectors whose first coordinate is less than m_0. If there are only finitely many such vectors, $\forall$ uses additional resets. During the game, Player $\forall$ remembers the largest first coordinate M of a vector selected by Player $\exists$.

For every $i \in \{m_0, \ldots, M\}$ we have $\eta\restriction_i$ is finite or $\pi_n\left(\eta\restriction_i\right) \notin L_{n-1}$, so

$$\overline{\eta_M} := \eta\restriction_{\{m_0, m_0+1, \ldots, M\}} \quad \text{is finite} \quad \text{or} \quad \eta_M := \pi_n\left(\overline{\eta_M}\right) \notin L_{n-1}.$$

If $\overline{\eta_M}$ is finite $\exists$ will lose the game (if he doesn't increase M). Otherwise, at every moment, Player $\forall$ assumes that M will not increase and uses the winning strategy from the inductive assumption for η_M at the next coordinates.

The value M can increase only finitely many times during the game (otherwise $\exists$ loses). Using prefix independence, we obtain that $\forall$ wins the game.

The inductive basis is trivial here, since there is no $\eta \in \mathcal{N}_0 \setminus L_0$. □

5 Conclusion

Our results seem to indicate that deciding MSO+U (if possible at all) might require considering the emptiness problem of some rather complicated model of automata, such as alternating ωBS-automata.

Acknowledgements. We would like to thank the anonymous referees for their constructive comments.

References

[BC06] Bojańczyk, M., Colcombet, T.: Bounds in ω-regularity. In: LICS, pp. 285–296 (2006)

[Boj09] Bojańczyk, M.: Weak MSO with the unbounding quantifier. In: STACS, pp. 159–170 (2009)

[Boj10] Bojańczyk, M.: Beyond ω-regular languages. In: STACS, pp. 11–16 (2010)

[BT09] Bojańczyk, M., Toruńczyk, S.: Deterministic automata and extensions of Weak MSO. In: FSTTCS, pp. 73–84 (2009)

[Kec95] Kechris, A.S.: Classical Descriptive Set Theory. Graduate Texts in Mathematics, vol. 156. Springer, Heidelberg (1995)

[Sri98] Srivastava, S.M.: A Course on Borel Sets. Graduate Texts in Mathematics, vol. 180. Springer, Heidelberg (1998)

[TL93] Thomas, W., Lescow, H.: Logical specifications of infinite computations. In: REX School/Symposium, pp. 583–621 (1993)