

Characterizing level one in group-based concatenation hierarchies^{*}

Thomas Place^{1,2} and Marc Zeitoun¹

¹ Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR 5800, 33400 Talence, France

² Institut Universitaire de France

`firstname.name@labri.fr`

Abstract. We investigate two operators on classes of regular languages: polynomial closure (*Pol*) and Boolean closure (*Bool*). We apply these operators to classes of *group languages* \mathcal{G} and to their well-suited extensions \mathcal{G}^+ , which is the least Boolean algebra containing \mathcal{G} and $\{\varepsilon\}$. This yields the classes $Bool(Pol(\mathcal{G}))$ and $Bool(Pol(\mathcal{G}^+))$. These classes form the first level in important classifications of classes of regular languages, called *concatenation hierarchies*, which admit natural *logical characterizations*. We present *generic* algebraic characterizations of these classes. They imply that one may decide whether a regular language belongs to such a class, provided that a more general problem called separation is decidable for the input class \mathcal{G} . The proofs are constructive and rely exclusively on notions from language and automata theory.

Keywords: Regular languages · Group languages · Concatenation hierarchies · Membership.

1 Introduction

An active line of research in automata theory is to investigate natural subclasses of regular languages. We are particularly interested in classes associated to fragments of standard pieces of syntax used to define the regular languages (*e.g.*, regular expressions or monadic second-order logic). Given a fragment, we consider the class of all languages that can be defined by an expression of this fragment. For each such class \mathcal{C} , a standard approach for its investigation is to look for a *\mathcal{C} -membership algorithm*: given a regular language L as input, decide whether $L \in \mathcal{C}$. Getting such an algorithm requires a solid understanding of \mathcal{C} . We are not only interested in a yes/no answer on the decidability of \mathcal{C} -membership but also in the techniques and proof arguments involved for obtaining this answer.

We look at classifications called *concatenation hierarchies*. A concatenation hierarchy is built from an input class of languages, called its *basis*, using two operators. The *polynomial closure* of a class \mathcal{C} , written $Pol(\mathcal{C})$, consists in all finite unions of languages $L_0 a_1 L_1 \cdots a_n L_n$ where a_1, \dots, a_n are letters and L_0, \dots, L_n are languages in \mathcal{C} . The *Boolean closure* of \mathcal{C} , denoted by $Bool(\mathcal{C})$, is the least

^{*} Funded by the DeLTA project (ANR-16-CE40-0007).

class containing \mathcal{C} and closed under Boolean operations. We investigate level *one* of concatenation hierarchies: the classes $Bool(Pol(\mathcal{C}))$ (abbreviated $BPol(\mathcal{C})$). Moreover, we consider special bases \mathcal{C} . The *group languages* are those recognized by a finite group, or equivalently by a permutation automaton (*i.e.*, a complete, deterministic *and* co-deterministic automaton). We only consider bases that are either a class \mathcal{G} containing only group languages, or its well-suited extension \mathcal{G}^+ (roughly, \mathcal{G}^+ is the least Boolean algebra containing \mathcal{G} and the singleton $\{\varepsilon\}$). The motivation for using such bases stems from the logical characterizations of concatenation hierarchies [22,12]. A word can be viewed as a logical structure consisting of a sequence of *labeled positions*. Therefore, we may use first-order sentences to define languages. It turns out that $BPol(\mathcal{G})$ and $BPol(\mathcal{G}^+)$ correspond to the logical classes $\mathcal{B}\Sigma_1(<, \mathbb{P}_{\mathcal{G}})$ and $\mathcal{B}\Sigma_1(<, +1, \mathbb{P}_{\mathcal{G}})$ where $\mathcal{B}\Sigma_1$ is the fragment of first-order logic containing only the Boolean combinations of purely existential formulas. Here, the predicates “ $<$ ” and “ $+1$ ” are interpreted as the linear order and the successor relation. Moreover, $\mathbb{P}_{\mathcal{G}}$ is a set of predicates built from \mathcal{G} : for each language $L \in \mathcal{G}$, it contains a unary predicate that checks whether the prefix preceding a given position belongs to L .

In the paper, we present generic algebraic characterizations of $BPol(\mathcal{G})$ and $BPol(\mathcal{G}^+)$. They apply to all classes of group languages \mathcal{G} satisfying mild hypotheses (namely, \mathcal{G} must be closed under Boolean operations and quotients). Moreover, they imply that *membership* is decidable for $BPol(\mathcal{G})$ and $BPol(\mathcal{G}^+)$ provided that a more general problem, *separation*, is decidable for \mathcal{G} . Separation takes two input regular languages L_0, L_1 and asks whether there exists $K \in \mathcal{G}$ such that $L_0 \subseteq K$ and $L_1 \cap K = \emptyset$. From the decidability point of view, the results are not entirely new. In particular, for $BPol(\mathcal{G})$, it is even known [14] that *separation* is decidable for $BPol(\mathcal{G})$ when it is already decidable for \mathcal{G} (on the other hand, this is open for $BPol(\mathcal{G}^+)$). Hence, our main contribution consists in the characterizations themselves and the techniques that we use to prove them. In particular, the proof arguments are constructive. For example, given a language L satisfying the characterization of $BPol(\mathcal{G})$, we prove directly that L belongs to $BPol(\mathcal{G})$ by explicitly building a description of L as a Boolean combination of products $L_0 a_1 L_1 \cdots a_n L_n$ where $L_0, \dots, L_n \in \mathcal{G}$.

With these characterizations, we generalize a number of known results for particular classes of group languages \mathcal{G} . Let us first consider the case when \mathcal{G} is the trivial Boolean algebra, which we denote by ST : we have $ST = \{\emptyset, A^*\}$ and $ST^+ = \{\emptyset, \{\varepsilon\}, A^+, A^*\}$ (where A is the alphabet). In this case, we obtain two well-known classes: $BPol(ST) = \mathcal{B}\Sigma_1(<)$ defines the *piecewise testable languages* and $BPol(ST^+) = \mathcal{B}\Sigma_1(<, +1)$ the *languages of dot-depth one*. The famous algebraic characterizations of these classes by Simon [17] and Knast [6] are simple corollaries of our generic results. Another key example is the class MOD of *modulo languages*: membership of a word in such a language depends only on its length modulo some fixed integer. In this case, the logical counterparts of $BPol(MOD)$ and $BPol(MOD^+)$ are the classes $\mathcal{B}\Sigma_1(<, MOD)$ and $\mathcal{B}\Sigma_1(<, +1, MOD)$ where “ MOD ” denotes the set of *modular predicates*. It is again possible to use our results to reprove the known characterizations of these

classes by Chaubard, Pin and Straubing [4] and Maciel, Péladeau and Thérien [7]. Our result also applies to the important case when \mathcal{G} is the class GR of *all* group languages [8]. In particular, there exists a specialized characterization of $BPol(\text{GR})$ by Henckell, Margolis, Pin and Rhodes [5], independent from GR-separation. While it is also possible to reprove this result as a corollary of our characterization, this requires a bit of technical work as well as knowledge of the GR-separation algorithm [2] which is a difficult result. Finally, another generic characterization of the classes $BPol(\mathcal{G})$ follows from an algebraic theorem of Steinberg [18] (though it only applies under more restrictive hypotheses on \mathcal{G}).

Our techniques differ from those used in the aforementioned specialized papers. Historically, classes of the form $BPol(\mathcal{G})$ or $BPol(\mathcal{G}^+)$ are often approached via alternate definitions based on an algebraic construction called “*wreath product*”. Indeed, it turns out that all classes of this kind can be built from the piecewise testable languages (*i.e.*, the class $BPol(\text{ST})$) using this product [20,9]. The arguments developed in [4,7,8,5,18] build exclusively on this construction. The paper is completely independent from these techniques: we work *directly* with the language theoretic definition of our classes based on the operator $BPol$. This matches our original motivation: investigating classes of regular languages.

We introduce the needed terminology in Section 2, look at classes of the form $BPol(\mathcal{G})$ in Section 3, and devote Section 4 to classes of the form $BPol(\mathcal{G}^+)$. Due to space limitations, several proofs are postponed to the full version of the paper [16].

2 Preliminaries

We present the objects that we investigate and the terminology that we require to manipulate them. The proofs are available in the full version of the paper.

2.1 Words, regular languages and classes

We fix a finite alphabet A for the whole paper. As usual, A^* denotes the set of all finite words over A , including the empty word ε . We let $A^+ = A^* \setminus \{\varepsilon\}$. For $u, v \in A^*$, we let uv be the word obtained by concatenating u and v . Additionally, given $w \in A^*$, we write $|w| \in \mathbb{N}$ for the length of w . A language is a subset of A^* . We denote the singleton language $\{u\}$ by u . We lift concatenation to languages: for $K, L \subseteq A^*$, we let $KL = \{uv \mid u \in K \text{ and } v \in L\}$. We shall consider *marked products*: given languages $L_0, \dots, L_n \subseteq A^*$, a marked product of L_0, \dots, L_n is a product of the form $L_0 a_1 L_1 \cdots a_n L_n$ where $a_1, \dots, a_n \in A$ (note that “ L_0 ” is a marked product: this is the case $n = 0$).

Regular languages. All languages considered in the paper are *regular*. These are the languages that can be equivalently defined by a regular expression, an automaton or a morphism into a finite monoid. We work with the latter definition. A *monoid* is a set M equipped with a binary operation $s, t \mapsto st$ (called multiplication) which is associative and has a neutral element denoted by “ 1_M ”. Recall that an idempotent of a monoid M is an element $e \in M$ such that $ee = e$.

For all $S \subseteq M$, we write $E(S)$ for the set of all idempotents in S . It is standard that when M is *finite*, there exists $\omega(M) \in \mathbb{N}$ (written ω when M is understood) such that s^ω is idempotent for every $s \in M$.

An *ordered monoid* is a pair (M, \leq) where M is a monoid and \leq is a partial order on M which is compatible with multiplication: for every $s, t, s', t' \in M$, if $s \leq t$ and $s' \leq t'$, then $ss' \leq tt'$. An upper set of M (for \leq) is a set $S \subseteq M$ which is upward closed for \leq : for every $s, t \in M$ such that $s \leq t$, we have $s \in S \Rightarrow t \in S$. For every $s \in M$, we write $\uparrow s$ for the least upper set of M containing s (*i.e.*, $\uparrow s$ consists of all $t \in M$ such that $s \leq t$). We may view arbitrary monoids as being ordered, as follows: we view any monoid M with no ordering specified as the ordered monoid $(M, =)$: we use equality as the ordering. In this special case, *all* subsets of M are upper sets.

Clearly, A^* is a monoid for concatenation as the multiplication (ε is neutral). Given an ordered monoid (M, \leq) , we may consider morphisms $\alpha : A^* \rightarrow (M, \leq)$. We say that a language $L \subseteq A^*$ is *recognized* by such a morphism α when there exists an *upper set* $F \subseteq M$ such that $L = \alpha^{-1}(F)$ (the definition depends on the ordering \leq , since F must be an upper set). Note that this also defines the languages recognized by a morphism $\eta : A^* \rightarrow N$ into an *unordered* monoid N since we view N as the ordered monoid $(N, =)$. It is well-known that a language is regular if and only if it can be recognized by a morphism into a *finite* monoid.

Remark 1. The only infinite monoid that we consider is A^* . From now, we implicitly assume that every other monoid M, N, \dots that we consider is finite.

Classes of languages. A class of languages \mathcal{C} is a set of languages. A *lattice* is a class closed under both union and intersection, and containing the languages \emptyset and A^* . Moreover, a *Boolean algebra* is a lattice closed under complement. Finally, a class \mathcal{C} is *quotient-closed* when for all $L \in \mathcal{C}$ and $u, v \in A^*$, the language $\{w \in A^* \mid uvw \in L\}$ belongs to \mathcal{C} as well. We say that a class \mathcal{C} is a *positive prevariety* (resp. *prevariety*) to indicate that it is a quotient-closed lattice (resp. Boolean algebra) containing *only regular languages*.

We rely on a decision problem called *membership* as a means to investigate classes of languages. Given a class \mathcal{C} , the \mathcal{C} -membership problem takes as input a regular language L and asks whether $L \in \mathcal{C}$. Intuitively, obtaining a procedure for \mathcal{C} -membership requires a solid understanding of \mathcal{C} . We also look at more involved problem called *separation*. Given a class \mathcal{C} , and two languages L_0 and L_1 , we say that L_0 is \mathcal{C} -separable from L_1 if and only if there exists $K \in \mathcal{C}$ such that $L_0 \subseteq K$ and $L_1 \cap K = \emptyset$. The \mathcal{C} -separation problem takes two regular languages L_0 and L_1 as input and asks whether L_0 is \mathcal{C} -separable from L_1 . Let us point out that we do *not* present separation algorithms in this paper. We shall need this problem as an intermediary in our investigation of membership.

Group languages. A group is a monoid G such that every element $g \in G$ has an inverse $g^{-1} \in G$, *i.e.*, such that $gg^{-1} = g^{-1}g = 1_G$. We call “*group language*” a language recognized by a morphism into a *finite group*. We consider classes \mathcal{G} that are group prevarieties (*i.e.*, containing group languages only).

We let GR be the class of *all* group languages. Another important example is the class AMT of *alphabet modulo testable languages*. For every $w \in A^*$ and

every $a \in A$, we write $\#_a(w) \in \mathbb{N}$ for the number of occurrences of “ a ” in w . The class AMT consists of all finite Boolean combinations of languages $\{w \in A^* \mid \#_a(w) \equiv k \pmod{m}\}$ where $a \in A$ and $k, m \in \mathbb{N}$ such that $k < m$. One may verify that these are exactly the languages recognized by commutative groups. We also consider the class MOD, which consists of all finite Boolean combinations of languages $\{w \in A^* \mid |w| \equiv k \pmod{m}\}$ with $k, m \in \mathbb{N}$ such that $k < m$. Finally, we write ST for the trivial class $ST = \{\emptyset, A^*\}$. One may verify that GR, AMT, MOD and ST are all group prevarieties.

It follows from the definition that $\{\varepsilon\}$ and A^+ are *not* group languages. This motivates the next definition: for a class \mathcal{C} , the *well-suited extension of \mathcal{C}* , denoted by \mathcal{C}^+ , consists of all languages of the form $L \cap A^+$ or $L \cup \{\varepsilon\}$ where $L \in \mathcal{C}$. The next lemma follows from the definition.

Lemma 2. *Let \mathcal{C} be a prevariety. Then, \mathcal{C}^+ is a prevariety containing the languages $\{\varepsilon\}$ and A^+ .*

2.2 Polynomial and Boolean closure

In the paper, we look at classes built using two standard operators. Consider a class \mathcal{C} . The *Boolean closure* of \mathcal{C} , denoted by $Bool(\mathcal{C})$ is the least Boolean algebra that contains \mathcal{C} . Moreover, the *polynomial closure* of \mathcal{C} , denoted by $Pol(\mathcal{C})$, contains all finite unions of marked products $L_0 a_1 L_1 \cdots a_n L_n$ where $L_0, \dots, L_n \in \mathcal{C}$. Finally, we write $BPol(\mathcal{C})$ for $Bool(Pol(\mathcal{C}))$. It is known that when \mathcal{C} is a prevariety, $Pol(\mathcal{C})$ is a positive prevariety and $BPol(\mathcal{C})$ is a prevariety. This is not immediate (proving that $Pol(\mathcal{C})$ is closed under intersection is difficult). This was first shown by Arfi [1], see also [10,12] for more recent proofs.

Theorem 3. *Let \mathcal{C} be a prevariety. Then $Pol(\mathcal{C})$ is a positive prevariety and $BPol(\mathcal{C})$ is a prevariety.*

In the literature, these operators are used to define classifications called concatenation hierarchies. Given a prevariety \mathcal{C} , the concatenation hierarchy of basis \mathcal{C} is built from \mathcal{C} by iteratively applying Pol and $Bool$ to \mathcal{C} . In the paper, we only look at the classes $Pol(\mathcal{C})$ and $BPol(\mathcal{C})$. These are the levels 1/2 and one in the concatenation hierarchy of basis \mathcal{C} . Moreover, we look at bases that are either a group prevariety \mathcal{G} or its well-suited extension \mathcal{G}^+ . Most of the prominent concatenation hierarchies in the literature use bases of this kind.

The hierarchy of basis $ST = \{\emptyset, A^*\}$ is called the Straubing-Thérien hierarchy [19,21]. In particular, $BPol(ST)$ is the class of piecewise testable languages [17]. Another prominent example is the basis $ST^+ = \{\emptyset, \{\varepsilon\}, A^+, A^*\}$ which yields the *dot-depth hierarchy* [3]. Non-trivial group prevarieties also yield important hierarchies. For example, the group hierarchy, whose basis is GR was first investigated in [8]. The hierarchies of bases MOD and MOD^+ are also prominent (see for example [4,7]). These hierarchies are also interesting for their logical counterparts, which were first discovered by Thomas [22]. Let us briefly recall them (see [12,14] for more details).

Consider a word $w = a_1 \cdots a_{|w|} \in A^*$. We view w as a linearly ordered set of $|w|+2$ positions $\{0, 1, \dots, |w|, |w|+1\}$ such that each position $1 \leq i \leq |w|$ carries the label $a_i \in A$ (on the other hand, 0 and $|w|+1$ are artificial unlabeled leftmost and rightmost positions). We use first-order logic to describe properties of words: a sentence can quantify over the positions of a word and use a predetermined set of predicates to test properties of these positions. We also allow two constants “*min*” and “*max*”, which we interpret as the artificial unlabeled positions 0 and $|w|+1$ in a given word w . Each first-order sentence φ defines the language of all words satisfying the property stated by φ . Let us present the predicates that we use. For each $a \in A$, we associate a unary predicate (also denoted by a), which selects the positions labeled by “ a ”. We also consider two binary predicates: the (strict) linear order “ $<$ ” and the successor relation “ $+1$ ”.

Example 4. The sentence “ $\exists x \exists y (x < y) \wedge a(x) \wedge b(y)$ ” defines the language $A^*aA^*bA^*$. The sentence “ $\exists x \exists y a(x) \wedge c(y) \wedge (y + 1 = \text{max})$ ” defines A^*aA^*c .

We associate a (possibly infinite) set of predicates $\mathbb{P}_{\mathcal{G}}$ to every group prevariety \mathcal{G} . For every language $L \in \mathcal{G}$, $\mathbb{P}_{\mathcal{G}}$ contains a unary predicate P_L which is interpreted as follows. Let $w = a_1 \cdots a_{|w|} \in A^*$. The unary predicate P_L selects all positions $i \in \{0, \dots, |w|+1\}$ such that $i \neq 0$ and $a_1 \cdots a_{i-1} \in L$. It is standard to write “ $\mathcal{B}\Sigma_1$ ” for the fragment of first-order logic, containing exactly the Boolean combinations of existential first-order sentences. We let $\mathcal{B}\Sigma_1(<, \mathbb{P}_{\mathcal{G}})$ be the class of all languages defined by a sentence of $\mathcal{B}\Sigma_1$ using only the label predicates, the linear order “ $<$ ” and those in $\mathbb{P}_{\mathcal{G}}$. Moreover, we let $\mathcal{B}\Sigma_1(<, +1, \mathbb{P}_{\mathcal{G}})$ be the class of all languages defined by a sentence of $\mathcal{B}\Sigma_1$, which additionally allows the successor predicate “ $+1$ ”. The following proposition follows from the generic logical characterization of concatenation hierarchies presented in [12] and the properties of group languages.

Proposition 5. *For every group prevariety \mathcal{G} , we have $BPol(\mathcal{G}) = \mathcal{B}\Sigma_1(<, \mathbb{P}_{\mathcal{G}})$ and $BPol(\mathcal{G}^+) = \mathcal{B}\Sigma_1(<, +1, \mathbb{P}_{\mathcal{G}})$.*

Remark 6. When $\mathcal{G} = \text{ST}$, all predicates in \mathbb{P}_{ST} are trivial. Hence, we get the classes $\mathcal{B}\Sigma_1(<)$ and $\mathcal{B}\Sigma_1(<, +1)$. When $\mathcal{G} = \text{MOD}$, one may verify that we obtain the classes $\mathcal{B}\Sigma_1(<, \text{MOD})$ and $\mathcal{B}\Sigma_1(<, +1, \text{MOD})$ where “*MOD*” is the set of *modular predicates* (for all $r, q \in \mathbb{N}$ such that $r < q$, it contains a unary predicate $M_{r,q}$ selecting the positions i such that $i \equiv r \pmod{q}$). When $\mathcal{G} = \text{AMT}$, one may verify that we obtain the classes $\mathcal{B}\Sigma_1(<, \text{AMOD})$ and $\mathcal{B}\Sigma_1(<, +1, \text{AMOD})$ where “*AMOD*” is the set of *alphabetic modular predicates* (for all $a \in A$ and $r, q \in \mathbb{N}$ such that $r < q$, it contains a unary predicate $M_{r,q}^a$ selecting the positions i such that the number of positions $j < i$ with label a is congruent to r modulo q).

2.3 \mathcal{C} -morphisms

We now introduce a key tool, which we shall use to formulate our results. Let \mathcal{C} be a positive prevariety. A *\mathcal{C} -morphism* is a *surjective* morphism $\eta : A^* \rightarrow (N, \leq)$ into a finite ordered monoid such that every language recognized by η belongs

to \mathcal{C} . Let us make a key remark: when \mathcal{C} is a prevariety, it suffices to consider *unordered* monoids (we view them as monoids ordered by equality).

Lemma 7. *Let \mathcal{C} be a prevariety and $\eta : A^* \rightarrow (N, \leq)$ a morphism. Then, η is a \mathcal{C} -morphism if and only if $\eta : A^* \rightarrow (N, =)$ is a \mathcal{C} -morphism.*

While simple, this notion is a key tool in the paper. First, it is involved in the membership problem. It is well-known that for every regular language L , there exists a canonical morphism $\alpha_L : A^* \rightarrow (M_L, \leq_L)$ into a finite ordered monoid recognizing L and called the *syntactic morphism* of L (we do not recall the definition as we shall not use it, see [11] for example). It can be computed from any representation of L and we have the following standard property.

Proposition 8. *Let \mathcal{C} be a positive prevariety. A regular language L belongs to \mathcal{C} if and only if its syntactic morphism $\alpha_L : A^* \rightarrow (M_L, \leq_L)$ is a \mathcal{C} -morphism.*

In view of Proposition 8, getting an algorithm for \mathcal{C} -membership boils down to finding a procedure to decide whether an input morphism $\alpha : A^* \rightarrow (M, \leq)$ is a \mathcal{C} -morphism. This is how we approach the question in the paper. We shall also use \mathcal{C} -morphisms as mathematical tools in proof arguments. In this context, we shall need the following simple corollary of Proposition 8.

Proposition 9. *Let \mathcal{C} be a positive prevariety and consider finitely many languages $L_1, \dots, L_k \in \mathcal{C}$. There exists a \mathcal{C} -morphism $\eta : A^* \rightarrow (N, \leq)$ such that L_1, \dots, L_k are recognized by η .*

Finally, we state the following simple lemma, which considers group languages.

Lemma 10. *Let \mathcal{G} be a group prevariety and let $\eta : A^* \rightarrow G$ be a \mathcal{G} -morphism. Then, G is a group.*

2.4 \mathcal{C} -pairs

Given a positive prevariety \mathcal{C} and a morphism $\alpha : A^* \rightarrow M$, we associate a relation on M . The definition is taken from [12], where it is used to characterize all classes of the form $Pol(\mathcal{C})$ for an arbitrary positive prevariety \mathcal{C} (we recall this characterization below). We say that $(s, t) \in M^2$ is a \mathcal{C} -pair (for α) if and only if $\alpha^{-1}(s)$ is *not* \mathcal{C} -separable from $\alpha^{-1}(t)$. The \mathcal{C} -pair relation is not robust. One may verify that it is reflexive when α is surjective and symmetric when \mathcal{C} is closed under complement. However, it is *not* transitive in general. We shall use the following lemma, which connects this notion to \mathcal{C} -morphisms.

Lemma 11. *Let \mathcal{C} be a positive prevariety and let $\alpha : A^* \rightarrow M$ be a morphism into a finite monoid. The two following properties hold:*

- for every \mathcal{C} -morphism $\eta : A^* \rightarrow (N, \leq)$ and every \mathcal{C} -pair $(s, t) \in M^2$ for α , there exist $u, v \in A^*$ such that $\eta(u) \leq \eta(v)$, $\alpha(u) = s$ and $\alpha(v) = t$.
- there exists a \mathcal{C} -morphism $\eta : A^* \rightarrow (N, \leq)$ such that for all $u, v \in A^*$, if $\eta(u) \leq \eta(v)$, then $(\alpha(u), \alpha(v))$ is a \mathcal{C} -pair for α .

Application to polynomial closure. We now recall the characterization of $Pol(\mathcal{C})$ from [12].

Theorem 12. *Let \mathcal{C} be a positive prevariety and $\alpha : A^* \rightarrow (M, \leq)$ a surjective morphism. Then, α is a $Pol(\mathcal{C})$ -morphism if and only if the following condition holds:*

$$s^{\omega+1} \leq s^{\omega} t s^{\omega} \quad \text{for every } \mathcal{C}\text{-pair } (s, t) \in M^2. \quad (1)$$

By definition, one can compute all \mathcal{C} -pairs associated to a morphism provided that \mathcal{C} -separation is decidable. Hence, in view of Proposition 8, it follows from Theorem 12 that when \mathcal{C} is a positive prevariety with decidable separation, membership is decidable for $Pol(\mathcal{C})$.

An interesting point is that Theorem 12 can be simplified in the special case when \mathcal{C} is a group prevariety \mathcal{G} or its well-suited extension \mathcal{G}^+ . This will be useful later when dealing with $BPol(\mathcal{G})$ and $BPol(\mathcal{G}^+)$. We first present a specialized characterization of the $Pol(\mathcal{G})$ -morphisms.

Theorem 13. *Let \mathcal{G} be a group prevariety and $\alpha : A^* \rightarrow (M, \leq)$ a surjective morphism. Then, α is a $Pol(\mathcal{G})$ -morphism if and only if the following condition holds:*

$$1_M \leq s \quad \text{for every } s \in M \text{ such that } (1_M, s) \text{ is a } \mathcal{G}\text{-pair}. \quad (2)$$

Finally, we present a similar statement for classes of the form $Pol(\mathcal{G}^+)$.

Theorem 14. *Let \mathcal{G} be a group prevariety, $\alpha : A^* \rightarrow (M, \leq)$ a surjective morphism and $S = \alpha(A^+)$. Then, α is a $Pol(\mathcal{G}^+)$ -morphism if and only if the following condition holds:*

$$e \leq e s e \quad \text{for every } e \in E(S) \text{ and } s \in M \text{ such that } (1_M, s) \text{ is a } \mathcal{G}\text{-pair}. \quad (3)$$

3 Group languages

In this section, we look at classes of the form $BPol(\mathcal{G})$ when \mathcal{G} is a group prevariety. We present a generic algebraic characterization of such classes, which implies that $BPol(\mathcal{G})$ -membership is decidable when this is already the case for \mathcal{G} -separation.

3.1 Preliminaries

We present two key results that we use to build languages of $BPol(\mathcal{G})$ in the proof. The first one is a concatenation principle for the classes $BPol(\mathcal{C})$ (where \mathcal{C} is an arbitrary prevariety) which is proved in [15, Lemma 3.6]. It is based on the notion of ‘‘cover’’. Given a language L , a cover of L is a *finite* set \mathbf{K} of languages satisfying $L \subseteq \bigcup_{K \in \mathbf{K}} K$. If \mathcal{D} is a class, we say that \mathbf{K} is a \mathcal{D} -cover of L , if \mathbf{K} is a cover of L such that $K \in \mathcal{D}$ for every $K \in \mathbf{K}$.

Proposition 15. *Let \mathcal{C} be a prevariety, and let $n \in \mathbb{N}$, $L_0, \dots, L_n \in \text{Pol}(\mathcal{C})$ and $a_1, \dots, a_n \in A$. For every $i \leq n$, let \mathbf{H}_i be a $B\text{Pol}(\mathcal{C})$ -cover of L_i . There exists a $B\text{Pol}(\mathcal{C})$ -cover \mathbf{K} of $L_0 a_1 L_1 \cdots a_n L_n$ such that for every $K \in \mathbf{K}$, there exists $H_i \in \mathbf{H}_i$ for each $i \leq n$ satisfying $K \subseteq H_0 a_1 H_1 \cdots a_n H_n$.*

Using Proposition 15 requires building a language $L_0 a_1 L_1 \cdots a_n L_n$ where $L_0, \dots, L_n \in \text{Pol}(\mathcal{C})$. We do this with an independent result which is tailored to the special case that we investigate in the section: \mathcal{C} is a group prevariety \mathcal{G} . Let $L \subseteq A^*$ be a language. For every word $w \in A^*$, we associate a language $\uparrow_L w \subseteq A^*$. Let $a_1, \dots, a_n \in A$ be the letters such that $w = a_1 \cdots a_n$. We define $\uparrow_L w = L a_1 L \cdots a_n L \subseteq A^*$ (in particular, $\uparrow_L \varepsilon = L$). The next proposition is proved in the full version of the paper (the proof is based on Higman's lemma).

Proposition 16. *Let $H \subseteq A^*$ be an arbitrary language and let $L \subseteq A^*$ be a group language such that $\varepsilon \in L$. There exists a cover \mathbf{K} of H such that every $K \in \mathbf{K}$ is of the form $K = \uparrow_L w$ for some word $w \in H$.*

3.2 Characterization of $B\text{Pol}(\mathcal{G})$

We are ready to present the characterization. As announced, we actually characterize the $B\text{Pol}(\mathcal{G})$ -morphisms. Recall that since $B\text{Pol}(\mathcal{G})$ is a prevariety, it suffices to consider unordered monoids by Lemma 7.

Theorem 17. *Let \mathcal{G} be a group prevariety and $\alpha : A^* \rightarrow M$ a surjective morphism. Then, α is a $B\text{Pol}(\mathcal{G})$ -morphism if and only if the following property holds:*

$$(qr)^\omega (st)^{\omega+1} = (qr)^\omega qt(st)^\omega \quad (4)$$

for every $q, r, s, t \in M$ such that (q, s) is a \mathcal{G} -pair.

Computing the \mathcal{G} -pairs associated to a morphism boils down to \mathcal{G} -separation. Hence, in view of Proposition 8, Theorem 17 implies that if *separation* is decidable for a group prevariety \mathcal{G} , then *membership* is decidable for $B\text{Pol}(\mathcal{G})$.

Remark 18. The decidability result itself is not new. In fact, it is even known [14] that *separation* is decidable for $B\text{Pol}(\mathcal{G})$ when this is already the case for \mathcal{G} . Our main contribution is the algebraic characterization and its proof, which relies on self-contained language theoretic arguments.

We may use Theorem 17 to reprove well-known results for specific classes \mathcal{G} . For example, since $\text{ST} = \{\emptyset, A^*\}$, every pair $(s, t) \in M^2$ is an ST-pair. Hence, using Theorem 17, one may verify that a surjective morphism $\alpha : A^* \rightarrow M$ is a $B\text{Pol}(\text{ST})$ -morphism if and only if the equation $(st)^\omega s = (st)^\omega = t(st)^\omega$ holds for all $s, t \in M$. This is exactly the characterization of the class $B\text{Pol}(\text{ST}) = \mathcal{B}\Sigma_1(<)$ of piecewise testable languages by Simon [17]. We also get a characterization of the class $B\text{Pol}(\text{MOD}) = \mathcal{B}\Sigma_1(<, \text{MOD})$. Though the statement does not really simplify in this case, it is easily shown to be equivalent to the one presented in [4]. Finally, there exists a simple characterization of $B\text{Pol}(\text{GR})$ presented in [5]: a surjective morphism $\alpha : A^* \rightarrow M$ is a $B\text{Pol}(\text{GR})$ -morphism if and only

if $(ef)^\omega = (fe)^\omega$ for all idempotents $e, f \in E(M)$. This is also a corollary of Theorem 17. Yet, this requires a bit of technical work as well as a knowledge of the GR-separation algorithm [2] (needed to describe the GR-pairs).

Proof (of Theorem 17). We first assume that α is a $BPol(\mathcal{G})$ -morphism and prove that it satisfies (4). There exists a finite set \mathbf{H} of languages in $Pol(\mathcal{G})$ such that for every $s \in M$, the language $\alpha^{-1}(s)$ is a Boolean combination of languages in \mathbf{H} . Since $Pol(\mathcal{G})$ is a positive prevariety, Proposition 9 yields a $Pol(\mathcal{G})$ -morphism $\eta : A^* \rightarrow (N, \leq)$ recognizing every $H \in \mathbf{H}$. Moreover, Lemma 11 yields a \mathcal{G} -morphism $\beta : A^* \rightarrow G$ such that for every $u, v \in A^*$, if $\beta(u) = \beta(v)$, then $(\eta(u), \eta(v)) \in N^2$ is a \mathcal{G} -pair for η . We know that G is a group by Lemma 10. We let $n = \omega(M) \times \omega(N) \times \omega(G)$.

We may now prove that (4) holds. Let $q, r, s, t \in M$ such that (q, s) is a \mathcal{G} -pair. We prove that $(qr)^\omega(st)^{\omega+1} = (qr)^\omega qt(st)^\omega$. Since $\beta : A^* \rightarrow G$ is a \mathcal{G} -morphism and (q, s) is a \mathcal{G} -pair, Lemma 11 yields two words $u, x \in A^*$ and $g \in G$ such that $\beta(u) = \beta(x) = g$, $\alpha(u) = q$ and $\alpha(x) = s$. Since α is surjective, we get $v, y \in A^*$ such that $\alpha(v) = r$ and $\alpha(y) = t$. Moreover, since G is a group, we have $\beta((uv)^n) = \beta((xy)^n) = 1_G$ by definition of n . Let $v' = v(uv)^{n-1}$ and $y' = y(xy)^{n-1}$. Since $\beta(u) = \beta(x) = g$, we also get $\beta(v') = \beta(y') = g^{-1}$, $\beta(uy') = 1_G$ and $\beta(v'x) = 1_G$. Hence, by definition of β , $(1_N, \eta(uy'))$ and $(1_N, \eta(v'x))$ are \mathcal{G} -pairs. Since η is a $Pol(\mathcal{G})$ -morphism by definition, it follows from Theorem 13 that $1_N \leq \eta(uy')$ and $1_N \leq \eta(v'x)$. We may now multiply to obtain that $\eta((uv)^n(xy)^{n+1}) \leq \eta((uv)^n uy'(xy)^{n+1})$ and $\eta((uv)^n uy(xy)^n) \leq \eta((uv)^n uv'xy(xy)^n)$. By definition of n , y' and v' , one may verify that this yields the inequalities $\eta((uv)^n(xy)^{n+1}) \leq \eta((uv)^n uy(xy)^n)$ and $\eta((uv)^n uy(xy)^n) \leq \eta((uv)^n(xy)^{n+1})$. Since η recognizes all $H \in \mathbf{H}$ by definition, it follows that $(uv)^n(xy)^{n+1} \in H \Leftrightarrow (uv)^n uy(xy)^n \in H$ for every $H \in \mathbf{H}$. Since all languages recognized by α are Boolean combination of languages in \mathbf{H} , we get $\alpha((uv)^n(xy)^{n+1}) = \alpha((uv)^n uy(xy)^n)$. By definition, this exactly says that $(qr)^\omega(st)^{\omega+1} = (qr)^\omega qt(st)^\omega$ as desired.

We turn to the converse implication. Assume that α satisfies (4). We prove that α is a $BPol(\mathcal{G})$ -morphism. Lemma 11 yields a \mathcal{G} -morphism $\beta : A^* \rightarrow G$ such that for every $u, v \in A^*$, if $\beta(u) = \beta(v)$, then $(\alpha(u), \alpha(v))$ is a \mathcal{G} -pair. We write $L = \beta^{-1}(1_G) \in \mathcal{G}$. By hypothesis on \mathcal{G} , L is a group language. Moreover, we have $\varepsilon \in L$ by definition. Given a finite set of languages \mathbf{K} , and $s, t \in M$, we say that \mathbf{K} is (s, t) -safe if for every $K \in \mathbf{K}$ and $w, w' \in K$, we have $s\alpha(w)t = s\alpha(w')t$. The argument is based on the following lemma.

Lemma 19. *Let $s, t \in M$. There exists a $BPol(\mathcal{G})$ -cover of L which is (s, t) -safe.*

Before proving Lemma 19 we first use it to prove that every language recognized by α belongs to $BPol(\mathcal{G})$, thus concluding the argument. We apply Lemma 19 with $s = t = 1_M$. This yields a $BPol(\mathcal{G})$ -cover \mathbf{K}_L of L which is $(1_M, 1_M)$ -safe. We use it to build a $BPol(\mathcal{G})$ -cover \mathbf{K} of A^* which is $(1_M, 1_M)$ -safe. Since $L \in \mathcal{G}$ and $\varepsilon \in L$, Proposition 16 yields a cover \mathbf{P} of A^* such that

every $P \in \mathbf{P}$, there exist $n \in \mathbb{N}$ and $a_1, \dots, a_n \in A$ such that $P = La_1L \cdots a_nL$. We cover each $P \in \mathbf{P}$ independently. Consider a language $P \in \mathbf{P}$. By definition, $P = La_1L \cdots a_nL$ for $a_1, \dots, a_n \in A$. Since $L \in \mathcal{G}$ and \mathbf{K}_L is a $BPol(\mathcal{G})$ -cover of L , Proposition 15 yields a $BPol(\mathcal{G})$ -cover \mathbf{K}_P of $P = La_1L \cdots a_nL$ such that for every $K \in \mathbf{K}_P$, there exist $K_0, \dots, K_n \in \mathbf{K}_L$ satisfying $K \subseteq K_0a_1K_1 \cdots a_nK_n$. Since \mathbf{K}_L is $(1_M, 1_M)$ -safe, it is immediate that \mathbf{K}_P is $(1_M, 1_M)$ -safe as well. Finally, since \mathbf{P} is a cover of A^* , it is now immediate that $\mathbf{K} = \bigcup_{P \in \mathbf{P}} \mathbf{K}_P$ is a $(1_M, 1_M)$ -safe $BPol(\mathcal{G})$ -cover of A^* . Since \mathbf{K} is $(1_M, 1_M)$ -safe, we know that for every $K \in \mathbf{K}$, there exists $s \in M$ such that $K \subseteq \alpha^{-1}(s)$. Hence, since \mathbf{K} is a cover of A^* , it is immediate that for every $F \subseteq M$, the language $\alpha^{-1}(F)$ is a union of languages in \mathbf{K} . By closure under union, it follows that $\alpha^{-1}(F) \in BPol(\mathcal{G})$. This exactly says that all languages recognized by α belong to $BPol(\mathcal{G})$.

It remains to prove Lemma 19. We define a preorder on M^2 that we shall use as an induction parameter. Consider $(s, t), (s', t') \in M^2$. We write $(s, t) \leq_L (s', t')$ if there exist $x, y \in A^*$ such that $xy \in L$, $s' = s\alpha(x)$ and $t' = \alpha(y)t$. It is immediate that \leq_L is reflexive since we have $\varepsilon = \varepsilon\varepsilon \in L$. Let us verify that \leq_L is transitive. Let $(s, t), (s', t'), (s'', t'') \in M^2$ such that $(s, t) \leq_L (s', t')$ and $(s', t') \leq_L (s'', t'')$. We show that $(s, t) \leq_L (s'', t'')$. By definition, we have $xy, x'y' \in L$ such that $s' = s\alpha(x)$, $t' = \alpha(y)t$, $s'' = s'\alpha(x')$ and $t'' = \alpha(y')t'$. Hence, $s'' = s\alpha(xx')$ and $t'' = \alpha(y'y)t$. Moreover, since $L = \beta^{-1}(1_G)$, we have $\beta(xx'y'y) = \beta(xy) = 1_G$, which yields $xx'y'y \in L$. We conclude that $(s, t) \leq_L (s'', t'')$, as desired.

We may now start the proof. Let $s, t \in M$. We construct a $BPol(\mathcal{G})$ -cover \mathbf{K} of L which is (s, t) -safe. We proceed by descending induction on the number of pairs $(s', t') \in M^2$ such that $(s, t) \leq_L (s', t')$. We handle the base case and the inductive step simultaneously. Consider a word $w \in L$. We say w stabilizes (s, t) if there exist $u, v \in A^*$ such that $uv \in \uparrow_L w$, $s\alpha(u) = s$ and $\alpha(v)t = t$. Observe that by definition, ε stabilizes (s, t) since we have $\varepsilon\varepsilon = \varepsilon \in L = \uparrow_L \varepsilon$. We let $H \subseteq L$ be the language of all words $w \in L$ that do *not* stabilize (s, t) . Note that by definition $\varepsilon \notin H$. We first use induction to build a $BPol(\mathcal{G})$ -cover \mathbf{K}_H of H and then complete it to build \mathbf{K} . Let us point out that it may happen that H is empty. This is the base case, it suffices to define $\mathbf{K}_H = \emptyset$.

Let $P \subseteq M^2$ be the set of all pairs $(s', t') \in M^2$ such that $(s, t) \leq_L (s', t')$ and $(s', t') \not\leq_L (s, t)$. We define $\ell = |P|$ and write $P = \{(s'_1, t'_1), \dots, (s'_\ell, t'_\ell)\}$. For every $i \leq \ell$, we may apply induction in the proof of Lemma 19 to obtain a $BPol(\mathcal{G})$ -cover \mathbf{K}_i of L which is (s'_i, t'_i) -safe. We define \mathbf{K}_L as the finite set of all languages $L \cap K_1 \cap \cdots \cap K_\ell$ where $K_i \in \mathbf{K}_i$ for every $i \leq \ell$. Since $L \in \mathcal{G}$, it is immediate that \mathbf{K}_L is a $BPol(\mathcal{G})$ -cover of L which is (s', t') -safe for every $(s', t') \in P$. We use it to construct \mathbf{K}_H .

Lemma 20. *There exists an (s, t) -safe $BPol(\mathcal{G})$ -cover \mathbf{K}_H of H .*

Proof. Since L is a group language such that $\varepsilon \in L$, Proposition 16 yields a cover \mathbf{U} of H such that for every $U \in \mathbf{U}$, there exist $n \geq 1$ and $a_1, \dots, a_n \in A$ such that $a_1 \cdots a_n \in H$ and $U = La_1L \cdots a_nL$ (note that $n \geq 1$ as $\varepsilon \notin H$). For each

$U \in \mathbf{U}$, we build an (s, t) -safe $BPol(\mathcal{G})$ -cover \mathbf{K}_U of U . Since \mathbf{U} is a cover of H , it will then suffice to define \mathbf{K}_H as the union of all covers \mathbf{K}_U . We fix $U \in \mathbf{U}$.

By definition, $U = La_1L \cdots a_nL$ where $a_1 \cdots a_n \in H$. Since $L \in \mathcal{G}$, $\varepsilon \in L$ and \mathbf{K}_L is a $BPol(\mathcal{G})$ -cover of L , Proposition 15 yields a $BPol(\mathcal{G})$ -cover \mathbf{K}_U of U such that for each $K \in \mathbf{K}_U$, we have $K \subseteq K_0a_1K_1 \cdots a_nK_n$ for $K_0, \dots, K_n \in \mathbf{K}_L$. It remains to show that \mathbf{K}_U is (s, t) -safe. We fix $K \in \mathbf{K}_U$ as described above and $w, w' \in K$. We show that $s\alpha(w)t = s\alpha(w')t$. By definition, we have $w_i, w'_i \in K_i$ for all $i \leq n$ such that $w = w_0a_1w_1 \cdots a_nw_n$ and $w' = w'_0a_1w'_1 \cdots a_nw'_n$. We let $u_i = w_0a_1 \cdots w_{i-1}a_i$ and $u'_i = w'_0a_1 \cdots w'_{i-1}a_i$ for $0 \leq i \leq n$ ($u_0 = u'_0 = \varepsilon$). We also let $v_i = a_{i+1}w_{i+1} \cdots a_nw_n$ and $v'_i = a_{i+1}w'_{i+1} \cdots a_nw'_n$ ($v_n = v'_n = \varepsilon$). Note that $u_iw'_iv'_i = u_{i-1}w_{i-1}v'_{i-1}$ for $1 \leq i \leq n$. Hence, it suffices to prove that $s\alpha(u_iw_iv'_i)t = s\alpha(u_iw'_iv'_i)t$ for $0 \leq i \leq n$. By transitivity, it will then follow that $s\alpha(u_nw_nv'_n)t = s\alpha(u_0w'_0v'_0)t$, *i.e.*, $s\alpha(w)t = s\alpha(w')t$ as desired.

We fix $i \leq n$ and show that $s\alpha(u_iw_iv'_i)t = s\alpha(u_iw'_iv'_i)t$. By hypothesis, $w_i, w'_i \in K_i$. Since $K_i \in \mathbf{K}_L$ is (s', t') -safe for all $(s', t') \in P$, it suffices to prove that $(s\alpha(u_i), \alpha(v'_i)t) \in P$. There are two conditions to verify. First, we show that $(s, t) \leq_L (s\alpha(u_i), \alpha(v'_i)t)$. By definition of \leq_L , this boils down to proving that $u_iv'_i \in L$. By definition, $w_j, w'_j \in K_j$ for every $j \leq n$. Moreover, since $K_j \in \mathbf{K}_L$, it follows that $w_j, w'_j \in L$ for every $j \leq n$ by definition of \mathbf{K}_L . It follows that $\beta(w_j) = \beta(w'_j) = 1_G$ since $L = \beta^{-1}(1_G)$. Therefore, by definition of u_i and v'_i , we obtain $\beta(u_i) = \beta(a_1 \cdots a_i)$ and $\beta(v'_i) = \beta(a_{i+1} \cdots a_n)$. This yields $\beta(u_iv'_i) = \beta(a_1 \cdots a_n)$. Finally, since $a_1 \cdots a_n \in H \subseteq L$ and L is recognized by β , we get $u_iv'_i \in L$, as desired. It remains to prove that $(s\alpha(u_i), \alpha(v'_i)t) \not\leq_L (s, t)$. By contradiction, assume that $(s\alpha(u_i), \alpha(v'_i)t) \leq_L (s, t)$. This yields $x, y \in A^*$ such that $xy \in L$ and $s = s\alpha(u_ix)$ and $t = \alpha(yv'_i)t$. Since $xy \in L$ and $w_j, w'_j \in L$, it is immediate by definition of u_i and v'_i that $u_ixyv_i \in \uparrow_L(a_1 \cdots a_n)$. Hence, $a_1 \cdots a_n$ stabilizes (s, t) . This is a contradiction since $a_1 \cdots a_n \in H$. \square

We are ready to construct the desired (s, t) -safe $BPol(\mathcal{G})$ -cover \mathbf{K} of L . Let \mathbf{K}_H be the $BPol(\mathcal{G})$ -cover of H given by Lemma 20. We let $K_\perp = L \setminus (\bigcup_{K \in \mathbf{K}_H} K)$. Finally, we define $\mathbf{K} = \{K_\perp\} \cup \mathbf{K}_H$. It is immediate that \mathbf{K} is a $BPol(\mathcal{G})$ -cover of L since $BPol(\mathcal{G})$ is a Boolean algebra (recall that $L \in \mathcal{G}$). It remains to verify that \mathbf{K} is (s, t) -safe. Since we already know that \mathbf{K}_H is (s, t) -safe, it suffices to prove that for every $w, w' \in K_\perp$, we have $s\alpha(w)t = s\alpha(w')t$. We actually show that $s\alpha(w)t = st$ for every $w \in K_\perp$. Since this is immediate when $w = \varepsilon$, we assume that $w \in A^+$ and let $a_1, \dots, a_n \in A$ be the letters such that $w = a_1 \cdots a_n$.

By definition of K_\perp , we know that $w \notin K'$ for every $K' \in \mathbf{K}_H$. Since \mathbf{K}_H is a cover of H , it follows that $w \notin H$, which means that w stabilizes (s, t) by definition of H . We get $u', v' \in A^*$ such that $u'v' \in \uparrow_L w$, $s\alpha(u') = s$ and $\alpha(v')t = t$. Since $u'v' \in \uparrow_L w$, there exist $0 \leq i \leq n$ and $x_0, \dots, x_i, y_i, \dots, y_n \in A^*$ which satisfy $x_0, \dots, x_{i-1}, x_iy_i, y_{i+1}, \dots, y_n \in L$, $u' = x_0a_1x_1 \cdots a_ix_i$ and $v' = y_ia_{i+1}x_{i+1} \cdots a_nx_n$. We write $u = a_1 \cdots a_i$ and $v = a_{i+1} \cdots a_n$. By definition $w = uv$. We show that $s = s\alpha(ux_i)$ and $t = \alpha(y_iv)t$. Let us first assume that this holds and explain why this implies $st = s\alpha(w)t$.

Since $uv = w$ and $w \in K_\perp \subseteq L = \beta^{-1}(1_G)$, we have $\beta(u)\beta(v) = 1_G$. Let $p = \omega(G)$. We have $1_G = \beta((y_i v)^p)$. Thus, since G is a group, it follows that $\beta(u) = \beta((y_i v)^{p-1} y_i)$. By definition of β , it follows that $(\alpha(u), \alpha((y_i v)^{p-1} y_i))$ is a \mathcal{G} -pair. Consequently, we obtain from (4) that,

$$(\alpha(ux_i))^\omega (\alpha((y_i v)^{p-1} y_i v))^{\omega+1} = (\alpha(ux_i))^\omega \alpha(uv) (\alpha((y_i v)^{p-1} y_i v))^\omega.$$

We may now multiply by s on the left and t on the right. Since $s = s\alpha(ux_i)$ and $t = \alpha(y_i v)t$, this yields $st = s\alpha(uv)t$. This concludes the proof since $uv = w$.

It remains to show that $s = s\alpha(ux_i)$ and $t = \alpha(y_i v)t$. We prove the former (the latter is symmetrical and left to the reader). For every j such that $0 \leq j \leq i$, we write $z_j = x_j a_{j+1} \cdots x_{i-1} a_i x_i$ (when $i = j$, we let $z_i = x_i$). We use induction on i to prove that $s = s\alpha(a_1 \cdots a_j z_j)$ for $0 \leq j \leq i$. Clearly, the case $j = i$ yields $s = s\alpha(a_1 \cdots a_i x_i)$ which exactly says that $s = s\alpha(ux_i)$. When $j = 0$, we have $z_0 = x_0 a_1 x_1 \cdots a_i x_i = u'$ and $s\alpha(u') = s$ by hypothesis. Assume now that $1 \leq j \leq i$. Since $x_{j-1} \in L$ and $L = \beta^{-1}(1_G)$, we have $\beta(x_{j-1}) = \beta(\varepsilon) = 1_G$. Hence, $(\alpha(x_{j-1}), 1_M)$ is a \mathcal{G} -pair by definition of β . Applying (4) with the values $\alpha(x_{j-1}), \alpha(a_j z_j a_1 \cdots a_{j-1}), 1_M, 1_M$ yields that,

$$(\alpha(x_{j-1} a_j z_j a_1 \cdots a_{j-1}))^\omega = (\alpha(x_{j-1} a_j z_j a_1 \cdots a_{j-1}))^\omega \alpha(x_{j-1}). \quad (5)$$

By induction hypothesis, we know that $s = s\alpha(a_1 \cdots a_{j-1} z_{j-1})$. Since it is immediate by definition that $a_1 \cdots a_{j-1} z_{j-1} = a_1 \cdots a_{j-1} x_{j-1} a_j z_j$, we get,

$$\begin{aligned} s &= s\alpha(a_1 \cdots a_{j-1} x_{j-1} a_j z_j) \\ &= s(\alpha(a_1 \cdots a_{j-1} x_{j-1} a_j z_j))^{\omega+1} \\ &= s\alpha(a_1 \cdots a_{j-1}) (\alpha(x_{j-1} a_j z_j a_1 \cdots a_{j-1}))^\omega \alpha(x_{j-1}) \alpha(a_j z_j) \\ &= s\alpha(a_1 \cdots a_{j-1}) (\alpha(x_{j-1} a_j z_j a_1 \cdots a_{j-1}))^\omega \alpha(a_j z_j) \quad \text{by (5)} \\ &= s(\alpha(a_1 \cdots a_{j-1} x_{j-1} a_j z_j))^\omega \alpha(a_1 \cdots a_{j-1} a_j z_j) \\ &= s\alpha(a_1 \cdots a_j z_j). \end{aligned}$$

This concludes the proof. \square

4 Well-suited extensions

We turn to the classes $BPol(\mathcal{G}^+)$ where \mathcal{G} is an arbitrary group prevariety. Again, we present a generic algebraic characterization, which implies that $BPol(\mathcal{G}^+)$ -membership is decidable when this is already the case for \mathcal{G} -separation.

4.1 Preliminaries

In this case as well, we start with preliminary results that we use to build languages of $BPol(\mathcal{G}^+)$. The first one is a simple corollary of Proposition 15 (the concatenation principle for $BPol(\mathcal{C})$) which is more convenient to manipulate when considering $BPol(\mathcal{G}^+)$ (see the full version of the paper).

Corollary 21. *Let \mathcal{C} be a prevariety, $L \in \text{Pol}(\mathcal{C}^+)$, \mathbf{H} a $\text{BPol}(\mathcal{C}^+)$ -cover of L , $n \in \mathbb{N}$ and $n+1$ nonempty words $w_1, \dots, w_{n+1} \in A^+$. There exists a $\text{BPol}(\mathcal{C}^+)$ -cover \mathbf{K} of $w_1 L \cdots w_n L w_{n+1}$ such that for every language $K \in \mathbf{K}$, we have $K \subseteq w_1 H_1 \cdots w_n H_n w_{n+1}$ for $H_1, \dots, H_n \in \mathbf{H}$.*

We complete Corollary 21 with a result that we use to build languages of the form $w_1 L \cdots w_n L w_{n+1}$ with $L \in \text{Pol}(\mathcal{C}^+)$. It is tailored to the case considered in the section: \mathcal{C} is a group prevariety \mathcal{G} . Consider a morphism $\alpha : A^* \rightarrow M$ and a nonempty word $w \in A^+$. An α -guarded decomposition of w is a tuple (w_1, \dots, w_{n+1}) where $n \in \mathbb{N}$ and $w_1, \dots, w_{n+1} \in A^+$ are nonempty words such that $w = w_1 \cdots w_{n+1}$ and, if $n \geq 1$, then for $1 \leq i \leq n$, there exists an idempotent $e_i \in \alpha(A^+)$ such that $\alpha(w_i) e_i = \alpha(w_i)$ and $e_i \alpha(w_{i+1}) = \alpha(w_{i+1})$. The next result is a corollary of Proposition 16. It is proved in the full version of the paper.

Proposition 22. *Let $H \subseteq A^+$ be a language, $\alpha : A^* \rightarrow M$ be a morphism and $L \subseteq A^*$ be a group language such that $\varepsilon \in L$. There is a cover \mathbf{K} of H such that for all $K \in \mathbf{K}$, there are $w \in H$ and an α -guarded decomposition (w_1, \dots, w_{n+1}) of w for some $n \in \mathbb{N}$ such that $K = w_1 L \cdots w_n L w_{n+1}$ (if $n = 0$, then $K = \{w_1\}$).*

4.2 Characterization

We may now present the characterization. As we explained, we actually characterize the $\text{BPol}(\mathcal{G}^+)$ -morphisms. Recall that since $\text{BPol}(\mathcal{G}^+)$ is a prevariety, it suffices to consider unordered monoids by Lemma 7.

Theorem 23. *Let \mathcal{G} be a group prevariety, $\alpha : A^* \rightarrow M$ a surjective morphism and $S = \alpha(A^+)$. Then, α is a $\text{BPol}(\mathcal{G}^+)$ -morphism if and only if the following property holds:*

$$(eqfre)^\omega (esfte)^{\omega+1} = (eqfre)^\omega qft(esfte)^\omega \quad (6)$$

for all $q, r, s, t \in M$ and $e, f \in E(S)$ such that (q, s) is a \mathcal{G} -pair.

Again, by Proposition 8, Theorem 23 implies that if *separation* is decidable for a group prevariety \mathcal{G} , then *membership* is decidable for $\text{BPol}(\mathcal{G}^+)$.

Theorem 23 can also be used to reprove famous results for specific classes \mathcal{G} . As seen in Section 3, since $\text{ST} = \{\emptyset, A^*\}$, every pair $(s, t) \in M^2$ is an ST-pair. Thus, one may verify from Theorem 23 that a surjective morphism $\alpha : A^* \rightarrow M$ is a $\text{BPol}(\text{ST}^+)$ -morphism if and only if $(eqfre)^\omega (esfte)^\omega = (eqfre)^\omega qft(esfte)^\omega$ for every $q, r, s, t \in S$ and $e, f \in E(S)$ (where $S = \alpha(A^+)$). This is exactly the well-known characterization of the languages of dot-depth one by Knast [6] (i.e., the class $\text{BPol}(\text{ST}^+) = \mathcal{B}\Sigma_1(<, +1)$). Additionally, there exists a specialized characterization of $\text{BPol}(\text{MOD}^+) = \mathcal{B}\Sigma_1(<, +1, \text{MOD})$ in the literature [7]. It can also be reproved as a corollary of Theorem 23. Yet, this requires some technical work involving the MOD-pairs.

Proof (of Theorem 23). Assuming that α satisfies (6), we prove that it is a $\text{BPol}(\mathcal{G}^+)$ -morphism. The converse implication is proved in the full version of the paper (the argument is based on Theorem 23).

Lemma 11 yields a \mathcal{G} -morphism $\beta : A^* \rightarrow G$ such that for $u, v \in A^*$, if $\beta(u) = \beta(v)$, then $(\alpha(u), \alpha(v))$ is a \mathcal{G} -pair. Let $L = \beta^{-1}(1_G) \in \mathcal{G}$. By hypothesis on \mathcal{G} , L is a group language. Moreover, $\varepsilon \in L$ by definition. Given a finite set of languages \mathbf{K} , and $s, t \in M$, we say that \mathbf{K} is (s, t) -safe if for every $K \in \mathbf{K}$ and $w, w' \in K$, we have $s\alpha(w)t = s\alpha(w')t$. The proof is based on the next lemma.

Lemma 24. *Let $s, t \in M$. There exists an (s, t) -safe $BPol(\mathcal{G}^+)$ -cover of L .*

We first apply Lemma 24 to prove that α is a $BPol(\mathcal{G}^+)$ -morphism. We use for $s = t = 1_M$. This yields a $BPol(\mathcal{G}^+)$ -cover \mathbf{K}_L of L which is $(1_M, 1_M)$ -safe. One may now apply Proposition 15 and Proposition 16 to build a $BPol(\mathcal{G})$ -cover \mathbf{K} of A^* which is $(1_M, 1_M)$ -safe from \mathbf{K}_L (see the proof of Theorem 17 for details). Hence, for every $F \subseteq M$, the language $\alpha^{-1}(F)$ is a union of languages in \mathbf{K} . By closure under union, it follows that $\alpha^{-1}(F) \in BPol(\mathcal{G}^+)$. This exactly says that all languages recognized by α belong to $BPol(\mathcal{G}^+)$.

It remains to prove Lemma 24. We define a preorder on M^2 that we shall use as an induction parameter. Let $(s, t), (s', t') \in M^2$. We write $(s, t) \leq_L^+ (s', t')$ if either $(s, t) = (s', t')$ or there exist $x, y \in A^*$ and $e \in E(S)$ such that $xy \in L$, $\alpha(x)e = \alpha(x)$, $e\alpha(y) = \alpha(y)$, $s' = s\alpha(x)$ and $t' = \alpha(y)t$. One may verify that \leq_L^+ is a preorder. We may now start the proof. Let $s, t \in M$. We construct a $BPol(\mathcal{G}^+)$ -cover \mathbf{K} of L which is (s, t) -safe. We proceed by induction on the number of pairs $(s', t') \in M^2$ such that $(s, t) \leq_L^+ (s', t')$. The base case and the inductive step are handled simultaneously. First, we define a language $H \subseteq L$. Let $w \in L$. We say w stabilizes (s, t) if $w = \varepsilon$ or $w \in A^+$ and there exists $n \geq 1$, an α -guarded decomposition (w_1, \dots, w_{n+1}) of w , an index $1 \leq i \leq n$, $x_1, \dots, x_i, y_i, \dots, y_n \in A^*$ and $e \in E(S)$ which satisfy the following conditions:

- $x_1, \dots, x_{i-1}, x_i y_i, y_{i+1}, \dots, y_n \in L$, and,
- $s\alpha(w_1 x_1 \cdots w_i x_i) e = s$, and,
- $e\alpha(y_i w_{i+1} \cdots y_n w_{n+1}) t = t$.

We let $H \subseteq L$ be the language of all words $w \in L$ which do *not* stabilize (s, t) . Observe that by definition, we have $\varepsilon \notin H$. We first use induction to build an (s, t) -safe $BPol(\mathcal{G}^+)$ -cover of H . Then, we complete it to obtain the desired $BPol(\mathcal{G}^+)$ -cover of L . It may happen that H is empty. In this case, we do not need induction: it suffices to use \emptyset as this $BPol(\mathcal{G}^+)$ -cover.

We let $P \subseteq M^2$ be the set of all $(s', t') \in M^2$ such that $(s, t) \leq_L^+ (s', t')$ and $(s', t') \not\leq_L^+ (s, t)$. We define $\ell = |P|$ and write $P = \{(s'_1, t'_1), \dots, (s'_\ell, t'_\ell)\}$. For every $i \leq \ell$, we may apply induction in the proof of Lemma 24 to obtain a $BPol(\mathcal{G}^+)$ -cover \mathbf{K}_i of L which is (s'_i, t'_i) -safe. We met \mathbf{K}_L as the finite set of all languages $L \cap K_1 \cap \cdots \cap K_\ell$ where $K_i \in \mathbf{K}_i$ for every $i \leq \ell$. Since $L \in \mathcal{G}$, it is immediate that \mathbf{K}_L is a $BPol(\mathcal{G}^+)$ -cover of L which is (s', t') -safe for all $(s', t') \in P$. We use it to build \mathbf{K}_H .

Lemma 25. *There exists an (s, t) -safe $BPol(\mathcal{G}^+)$ -cover \mathbf{K}_H of H .*

Proof. Since L is a group language such that $\varepsilon \in L$ and $\varepsilon \notin H$, Proposition 22 yields a cover \mathbf{U} of H such that each $U \in \mathbf{U}$ is of the form $U = w_1 L \cdots w_n L w_{n+1}$

where (w_1, \dots, w_{n+1}) is an α -guarded decomposition of a word $w \in H$. For each $U \in \mathbf{U}$, we build an (s, t) -safe $BPol(\mathcal{G}^+)$ -cover \mathbf{K}_U of U . As \mathbf{U} is a cover of H , it will then suffice to define \mathbf{K}_H as the union of all covers \mathbf{K}_U . We fix $U \in \mathbf{U}$.

By definition of \mathbf{U} , $U = w_1 L \cdots w_n L w_{n+1}$ where (w_1, \dots, w_{n+1}) is an α -guarded decomposition of a word $w \in H$. Since $L \in \mathcal{G}$, $\varepsilon \in L$ and \mathbf{K}_L is a $BPol(\mathcal{G}^+)$ -cover of L by hypothesis, Corollary 21 yields a $BPol(\mathcal{G}^+)$ -cover \mathbf{K}_U of U such that for each $K \in \mathbf{K}_U$, we have $K \subseteq w_1 K_1 \cdots w_n K_n w_{n+1}$ for $K_1, \dots, K_n \in \mathbf{K}_L$. Let us prove that \mathbf{K}_U is (s, t) -safe. We fix $K \in \mathbf{K}_U$ as described above. For $u, u' \in K$, we show that $s\alpha(u)t = s\alpha(u')t$. If $n = 0$, then $K \subseteq \{w_1\}$. Hence $u = u' = w_1$ and the result is immediate. Assume now that $n \geq 1$. We get $u_i, u'_i \in K_i$ for $1 \leq i \leq n$ such that $u = w_1 u_1 \cdots w_n u_n w_{n+1}$ and $u' = w_1 u'_1 \cdots w_n u'_n w_{n+1}$. For $1 \leq i \leq n$, we write $x_i = w_1 u_1 w_2 \cdots u_{i-1} w_i$ and $x'_i = w_1 u'_1 w_2 \cdots u'_{i-1} w_i$. Moreover, we let $y_i = w_{i+1} u_{i+1} \cdots w_n u_n w_{n+1}$ and $y'_i = w_{i+1} u'_{i+1} \cdots w_n u'_n w_{n+1}$. For $1 \leq i \leq n$, we have $x_i u'_i y'_i = x_{i-1} u_{i-1} y'_{i-1}$. Moreover, one may use the hypotheses that $w \in H$ and (w_1, \dots, w_{n+1}) is an α -guarded decomposition of w to verify that $(s\alpha(x_i), \alpha(y'_i)t) \in P$. Hence, since $K_i \in \mathbf{K}_L$ which is (s', t') -safe for every $(s', t') \in P$, we have $s\alpha(x_i u_i y'_i)t = s\alpha(x_i u'_i y'_i)t$ for $1 \leq i \leq n$. It is now immediate by transitivity that $s\alpha(x_n u_n y'_n)t = s\alpha(x_1 u'_1 y'_1)t$, *i.e.* $s\alpha(u)t = s\alpha(u')t$ as desired. \square

We now define the desired (s, t) -safe $BPol(\mathcal{G}^+)$ -cover \mathbf{K} of L . Lemma 25 yields a $BPol(\mathcal{G}^+)$ -cover \mathbf{K}_H of H . We let $K_\perp = L \setminus (\bigcup_{K \in \mathbf{K}_H} K)$. Finally, we let $\mathbf{K} = \{K_\perp\} \cup \mathbf{K}_H$. Clearly, \mathbf{K} is a $BPol(\mathcal{G}^+)$ -cover of L since $BPol(\mathcal{G}^+)$ is a Boolean algebra (recall that $L \in \mathcal{G}$). It remains to verify that \mathbf{K} is (s, t) -safe. Since we already know that \mathbf{K}_H is (s, t) -safe, it suffices to prove that for every $w, w' \in K_\perp$, we have $s\alpha(w)t = s\alpha(w')t$. We actually show that $s\alpha(w)t = st$ for every $w \in K_\perp$. Since this is immediate when $w = \varepsilon$, we assume that $w \in A^+$.

By definition of K_\perp , we have $w \notin K'$ for all $K' \in \mathbf{K}_H$. Since \mathbf{K}_H is a cover of H , this yields $w \notin H$, *i.e.* w stabilizes (s, t) . Since $w \neq \varepsilon$, we get an α -guarded decomposition (w_1, \dots, w_{n+1}) of w , an index $i \leq n$, $x_1, \dots, x_i, y_i, \dots, y_1 \in A^*$ and $e \in E(S)$ such that $x_1, \dots, x_{i-1}, x_i y_i, y_{i-1}, \dots, y_n \in L$, $s\alpha(w_1 x_1 \cdots w_i x_i)e = s$ and $e\alpha(y_i w_{i+1} \cdots w_n w_n)t = t$. Let $u = w_1 \cdots w_i$ and $v = w_{i+1} \cdots w_{n+1}$. We show that $s = s\alpha(ux_i)e$ and $t = e\alpha(y_i v)t$ (note that since e is an idempotent, this also implies $s = se$ and $t = et$) Let us first assume that this holds and explain why this implies $st = s\alpha(w)t$.

Since (w_1, \dots, w_{n+1}) is an α -guarded decomposition, there exist an idempotent $f \in E(S)$ such that $\alpha(w_i)f = \alpha(w_i)$ and $f\alpha(w_{i+1}) = \alpha(w_{i+1})$. By definition of u and v , we have $\alpha(u)f = \alpha(u)$ and $f\alpha(v) = \alpha(v)$. Clearly, we have $wv = w$. Thus, since $w \in L = \beta^{-1}(1_G)$, we have $\beta(u)\beta(v) = 1_G$. Let $p = \omega(G)$. We have $1_G = \beta((y_i v)^p)$. Thus, since G is a group, it follows that $\beta(u) = \beta((y_i v)^{p-1} y_i)$. By definition of β , it follows that $(\alpha(u), \alpha((y_n v)^{p-1} y_i))$ is a \mathcal{G} -pair. Let $q = \alpha(u)$, $r = \alpha(x_i)$, $q' = \alpha((y_n v)^{p-1} y_i)$ and $r' = \alpha(v)$. Since we just proved that (q, q') is a \mathcal{G} -pair, we obtain from (6) that,

$$(eqfre)^\omega (eq'fr'e)^{\omega+1} = (eqfre)^\omega qfr'(eq'fr'e)^\omega. \quad (7)$$

Since $\alpha(u)f = \alpha(u)$, we have $eqfre = e\alpha(ux_i)e$ and $qfr' = \alpha(uv) = \alpha(w)$. Moreover, since $f\alpha(v) = \alpha(v)$, we have $eq'fr'e = e\alpha((y_iv)^p)e$. Hence, since we have $s = s\alpha(ux_i)e = se$ and $t = e\alpha(y_iv)t = et$, it is immediate that $seqfre = s$ and $eq'fr'et = t$. We may now multiply by s on the left and t on the right in (7) to obtain $st = sqfr't = s\alpha(w)t$ as desired.

It remains to prove that $s = s\alpha(ux_i)e$ and $t = e\alpha(y_iv)t$. We concentrate on $s = s\alpha(ux_i)e$ (the other equality is symmetrical and left to the reader). For every j such that $1 \leq j \leq i$, we write $r_j = \alpha(w_jx_j \cdots w_ix_i)e$ and $u_j = w_1 \cdots w_{j-1}$ (we let $u_1 = \varepsilon$). We use induction on j to prove that $s = s\alpha(u_j)r_j$ for $1 \leq j \leq i$. This concludes the argument: when $j = i$, we get $s = s\alpha(w_1 \cdots w_{i-1}w_ix_i)e$. Since $u = w_1 \cdots w_i$, this exactly says that $s = s\alpha(ux_i)e$ as desired. The case $j = 1$ is immediate by definition: we have $s\alpha(w_1x_1 \cdots w_ix_i)e = s$. Thus, we now assume that $2 \leq j \leq i$. Since (w_1, \dots, w_{n+1}) is an α -guarded decomposition, there exist an idempotent $f \in E(S)$ such that $\alpha(w_{j-1})f = \alpha(w_{j-1})$ and $f\alpha(w_j) = \alpha(w_j)$. By definition of u_j and r_j , we have $\alpha(u_j)f = \alpha(u_j)$ and $fr_j = r_j$. Moreover, since $x_{j-1} \in L$ and $L = \beta^{-1}(1_G)$, we have $\beta(x_{j-1}) = \beta(\varepsilon) = 1_G$. By definition of β , it follows that $(\alpha(x_{j-1}), 1_M)$ is a \mathcal{G} -pair. Hence, we may apply (6) for $q = \alpha(x_{j-1})$, $r = r_j\alpha(u_j)$ and $s = t = 1_M$ to obtain,

$$(f\alpha(x_{j-1})fr_j\alpha(u_j)f)^\omega = (f\alpha(x_{j-1})fr_j\alpha(u_j)f)^\omega\alpha(x_{j-1})f. \quad (8)$$

Induction yields that $s = s\alpha(u_{j-1})r_{j-1}$. Moreover, it is immediate from the definitions that $\alpha(u_{j-1})r_{j-1} = \alpha(u_j)\alpha(x_{j-1})r_j = \alpha(u_j)f\alpha(x_{j-1})fr_j$ which yields,

$$\begin{aligned} s &= s\alpha(u_j)f\alpha(x_{j-1})fr_j \\ &= s(\alpha(u_j)f\alpha(x_{j-1})fr_j)^{\omega+1} \\ &= s\alpha(u_j)(f\alpha(x_{j-1})fr_j\alpha(u_j)f)^\omega\alpha(x_{j-1})fr_j \\ &= s\alpha(u_j)(f\alpha(x_{j-1})fr_j\alpha(u_j)f)^\omega r_j && \text{by (8)} \\ &= s(\alpha(u_j)f\alpha(x_{j-1})fr_j)^\omega\alpha(u_j)fr_j \\ &= s\alpha(u_j)fr_j. \end{aligned}$$

This exactly says that $q = s\alpha(u_j)r_j$ which completes the proof. \square

5 Conclusion

We presented generic algebraic characterizations for classes of the form $BPol(\mathcal{G})$ and $BPol(\mathcal{G}^+)$ when \mathcal{G} is a group prevariety. They imply that membership is decidable for these two classes as soon as separation is decidable for the input class \mathcal{G} . The most natural follow-up question is whether these characterizations can be generalized to encompass all classes $BPol(\mathcal{C})$ where \mathcal{C} is an *arbitrary* prevariety and obtain a characterization similar to the one provided by Theorem 12 for $Pol(\mathcal{C})$. This is a difficult question. In particular, it seems unlikely that $BPol(\mathcal{C})$ -membership boils down to \mathcal{C} -separation in the general case. Indeed, a specialized characterization for the class $BPol(BPol(ST))$ is known [13]. Yet, deciding it involves looking at a more general question than $BPol(ST)$ -separation.

References

1. Arfi, M.: Polynomial operations on rational languages. In: STACS'87. pp. 198–206. Springer-Verlag, Berlin, Heidelberg (1987)
2. Ash, C.J.: Inevitable graphs: a proof of the type II conjecture and some related decision procedures. *Int. J. Algebra Comput.* **1**(1), 127–146 (1991)
3. Brzozowski, J.A., Cohen, R.S.: Dot-depth of star-free events. *Journal of Computer and System Sciences* **5**(1), 1–16 (1971)
4. Chaubard, L., Éric Pin, J., Straubing, H.: First order formulas with modular predicates. In: Proceedings of the 21th IEEE Symposium on Logic in Computer Science (LICS'06). pp. 211–220 (2006)
5. Henckell, K., Margolis, S., Pin, J.E., Rhodes, J.: Ash's type II theorem, profinite topology and Malcev products. *Int. J. Algebra Comput.* **1**, 411–436 (1991)
6. Knast, R.: A semigroup characterization of dot-depth one languages. *RAIRO - Theoretical Informatics and Applications* **17**(4), 321–330 (1983)
7. Maciel, A., Péladeau, P., Thérien, D.: Programs over semigroups of dot-depth one. *Theoretical Computer Science* **245**(1), 135–148 (2000)
8. Margolis, S., Pin, J.E.: Product of Group Languages. In: FCT Conference. vol. *Lecture Notes in Computer Science*, pp. 285–299. Springer-Verlag (1985)
9. Pin, J.E.: Algebraic tools for the concatenation product. *Theoretical Computer Science* **292**, 317–342 (2003)
10. Pin, J.E.: An explicit formula for the intersection of two polynomials of regular languages. In: DLT 2013. *Lect. Notes Comp. Sci.*, vol. 7907, pp. 31–45. Springer (2013)
11. Pin, J.E.: Mathematical foundations of automata theory (2020), <http://www.irif.fr/~jep/PDF/MPRI/MPRI.pdf>. In preparation
12. Place, T., Zeitoun, M.: Generic results for concatenation hierarchies. *Theory of Computing Systems (ToCS)* **63**(4), 849–901 (2019), selected papers from CSR'17
13. Place, T., Zeitoun, M.: Going higher in first-order quantifier alternation hierarchies on words. *Journal of the ACM* **66**(2), 12:1–12:65 (2019)
14. Place, T., Zeitoun, M.: Separation and covering for group based concatenation hierarchies. In: Proceedings of the 34th Annual ACM/IEEE Symposium on Logic in Computer Science. pp. 1–13. LICS'19 (2019)
15. Place, T., Zeitoun, M.: Separation for dot-depth two. *Logical Methods in Computer Science* **Volume 17, Issue 3** (2021)
16. Place, T., Zeitoun, M.: Characterizing level one in group-based concatenation hierarchies. *CoRR* **abs/2201.06826** (2022), <https://arxiv.org/abs/2201.06826>
17. Simon, I.: Piecewise testable events. In: Proceedings of the 2nd GI Conference on Automata Theory and Formal Languages. pp. 214–222. Springer-Verlag, Berlin, Heidelberg (1975)
18. Steinberg, B.: Inevitable graphs and profinite topologies: Some solutions to algorithmic problems in monoid and automata theory, stemming from group theory. *Int. J. Algebra Comput.* **11**(1), 25–72 (2001)
19. Straubing, H.: A generalization of the schützenberger product of finite monoids. *Theoretical Computer Science* **13**(2), 137–150 (1981)
20. Straubing, H.: Finite semigroup varieties of the form $V * D$. *Journal of Pure and Applied Algebra* **36**, 53–94 (1985)
21. Thérien, D.: Classification of finite monoids: The language approach. *Theoretical Computer Science* **14**(2), 195–208 (1981)
22. Thomas, W.: Classifying regular events in symbolic logic. *Journal of Computer and System Sciences* **25**(3), 360–376 (1982)