

How many times do you need to go back to the future in Unary Temporal Logic?*

Thomas Place^{1,2} and Marc Zeitoun¹[0000-0003-4101-8437]

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

² Institut Universitaire de France

`firstname.name@labri.fr`

Abstract. Unary temporal logic (UTL) can express properties on finite words with the temporal modalities “sometimes in the future/past”. The languages definable in UTL are well-understood. In particular, they correspond to *unambiguous languages*, which are built by applying successively three standard operators to the trivial class of languages (consisting of the empty language and the universal one): polynomial, Boolean, and finally unambiguous polynomial closures. Moreover, it is known that one can decide whether a given regular language is expressible in UTL. We extend these results in two ways. First, we use *generalized* temporal modalities “sometimes in the future/past”, which depend on a class “ \mathcal{C} ” of languages. Second, we investigate a hierarchy inside such a variant of UTL: its *future/past hierarchy*. Each level in this hierarchy consists of all languages definable with a *bounded* number of *alternations* between the “sometimes in the future” and “sometimes in the past” modalities. We show that if \mathcal{C} is a class of *group languages* with mild properties, there is a correspondence between levels of such a \mathcal{C} -specified hierarchy and classes of languages obtained from \mathcal{C} by applying standard operators: the polynomial, the Boolean, and the left/right deterministic closures. We also show that if \mathcal{C} has decidable “separation problem”, then one can decide membership of a regular language within any level of the corresponding future/past hierarchy. Finally, these results extend to the case where we allow “tomorrow” and “yesterday” temporal modalities.

1 Introduction

The goal of this paper is to understand the fine-grained structure of fragments of a standard temporal logic, and in particular, their expressive power. Temporal logics are common formalisms in computer science, whose purpose is to specify properties of finite or infinite structures, such as words or trees. Their success stems from the good balance between their ease of use and their expressiveness. For instance, it is well known [6] that on words, linear time temporal logic is exactly as powerful as first-order logic. Another standard logic is *unary temporal logic* (UTL, or TL for short), which shares ties with navigational logics for trees. Here, we focus on fragments of TL on finite words over some alphabet A .

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

The logic TL has two temporal operators: “sometimes in the future” and “sometimes in the past”. Each TL formula defines a regular language on A^* . Therefore, TL defines a class of regular languages, which turns out to be one of the most robust ones [26]: first, Etessami, Vardi and Wilke [3,4] proved that TL has the same expressiveness as FO^2 , the restriction of first-order logic to two variables. Second, Thérien and Wilke [27] designed a decidable characterization of FO^2 , *i.e.*, a *membership* algorithm that decides whether a given regular language is definable in this logic. Obtaining such an algorithm is important, as this requires a solid understanding on the investigated languages. Third, Schützenberger [24], Pin, Straubing and Thérien [13,15] described this class in terms of languages built from \emptyset and A^* by applying standard operators: the Boolean, polynomial and unambiguous polynomial closures. Here, the Boolean closure $\text{Bool}(\mathcal{C})$ of a class of languages \mathcal{C} is the smallest Boolean algebra containing \mathcal{C} . The polynomial closure $\text{Pol}(\mathcal{C})$ of \mathcal{C} is the smallest class containing \mathcal{C} closed under union and (marked) language concatenation. Finally, the unambiguous polynomial closure $\text{UPol}(\mathcal{C})$ of \mathcal{C} is a subclass of $\text{Pol}(\mathcal{C})$, defined by semantic restrictions on the allowed unions and marked products. The results of [15] show that languages definable in TL are exactly those of $\text{UPol}(\text{Bool}(\text{Pol}(\{\emptyset, A^*\})))$.

This smoothly generalizes to $\text{TL}(\mathcal{C})$, a version of TL parameterized by a “base class” \mathcal{C} of languages [19,23] (TL corresponds to $\text{TL}(\{\emptyset, A^*\})$). On the logical side, this boils down to enriching TL with temporal operators (or FO^2 with predicates) built from \mathcal{C} in a natural way. Moreover, if \mathcal{C} is a class of *group languages* satisfying mild properties, $\text{TL}(\mathcal{C})$ is exactly $\text{UPol}(\text{Bool}(\text{Pol}(\mathcal{C})))$. In this case, [19,23] provide yet another definition of the languages in $\text{TL}(\mathcal{C})$. They are built from $\text{Bool}(\text{Pol}(\mathcal{C}))$ by applying two other closure operators in alternation: the left (resp. right) deterministic closure LPol (resp. RPol). Finally, membership remains decidable for $\text{TL}(\mathcal{C})$, provided that \mathcal{C} satisfies some properties.

Contributions. These multiple equivalent definitions of $\text{TL}(\mathcal{C})$ lead to natural hierarchies: the quantifier alternation hierarchy of FO^2 , the LPol/RPol alternation hierarchy, called the *deterministic hierarchy*, and the future-past hierarchy, which counts the number of alternations between future and past operators. While the first has been already investigated [5,8,11] for a particular base class, this is not the case for the future-past hierarchy. Our first contribution connects the future-past hierarchy with the deterministic hierarchy: we show that the future-past hierarchy inside $\text{TL}(\mathcal{C})$ coincides with the deterministic hierarchy of base class $\text{Bool}(\text{Pol}(\mathcal{C}))$. This holds when \mathcal{C} is a class of group languages satisfying mild properties, and for extensions of such classes capturing the variants of TL allowing the “tomorrow” and “yesterday” operators. In practice, this makes it possible to cope with temporal operators controlling the words that may be used along future/past jumps (typical properties are modulo tests on the length of such words, or on the number of occurrences of a specific letter). The second result is that membership is decidable for all levels of this hierarchy. For some of them, we use the language-theoretic characterization together with generic results of [19,23,18]. For others (the so-called *join levels*), this requires specific work. Altogether, these results generalize work by Kufleitner and Lauser [9,8].

Other hierarchies. Two alternative hierarchies inside TL are known, albeit specific to $\text{TL}(\{\emptyset, A^*\})$. First, a hierarchy based on the notion of “ranker” is considered in [10]. Another hierarchy is investigated in [8]. It is based on unambiguous interval temporal logic (this is another logic equivalent to TL introduced in [12]). These hierarchies are independent from our work.

Organization. We present terminology in Section 2 and future-past hierarchies in Section 3. Their characterization by deterministic hierarchies is proved in Section 4. Finally, Section 5 is devoted to membership for the *join levels*. Due to space limitations, several proofs are postponed to the full version of the paper.

2 Preliminaries

We fix a finite alphabet A for the whole paper. As usual, A^* is 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 . Given $w \in A^*$, we write $|w| \in \mathbb{N}$ for the length of w . We also consider *positions*. A word $w = a_1 \cdots a_{|w|} \in A^*$ is viewed as an *ordered set* $\text{Pos}(w) = \{0, 1, \dots, |w|, |w| + 1\}$ of $|w| + 2$ positions. Each position i such that $1 \leq i \leq |w|$ carries the label $a_i \in A$. Positions 0 and $|w| + 1$ are *artificial* leftmost and rightmost positions, which carry *no label*. For every $i \in \text{Pos}(w)$, we define an element $w[i] \in A \cup \{\text{min}, \text{max}\}$ (where “*min*” and “*max*” do not belong to A). We let $w[0] = \text{min}$, $w[i] = a_i$ if $1 \leq i \leq |w|$ and $w[|w| + 1] = \text{max}$. Finally, given $i, j \in \text{Pos}(w)$ such that $i < j$, we write $w(i, j) = a_{i+1} \cdots a_{j-1} \in A^*$ (*i.e.*, we keep the letters carried by all positions that are *strictly* between i and j). Note that $w(0, |w| + 1) = w$.

A language is a subset of A^* . We look at *regular* languages, *i.e.*, 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 multiplication $s, t \mapsto st$, which is associative and has an identity element written “ 1_M ”. An element $e \in M$ is *idempotent* if it satisfies $ee = e$. 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 $s \in M$. Clearly, A^* equipped with concatenation is a monoid (ε is the identity). Hence, we may consider morphisms $\alpha : A^* \rightarrow M$ into a monoid M . We say that $L \subseteq A^*$ is *recognized* by such a morphism α when there exists $F \subseteq M$ such that $L = \alpha^{-1}(F)$. 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 on, we implicitly assume that every other monoid M, N, \dots in this paper is finite.

Classes of languages. A *class of languages* \mathcal{C} is a set of languages. Such a class forms a *lattice* if it is closed under both union and intersection, and contains the languages \emptyset and A^* . It is a *Boolean algebra* if it is additionally 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 $\{w \in A^* \mid uwv \in L\}$ belongs to \mathcal{C} as well. A class \mathcal{C} is a *prevariety* when it is a quotient-closed Boolean algebra and contains *only regular languages*.

We use a decision problem called *membership* as a means to investigate classes. For a class \mathcal{C} , the \mathcal{C} -membership problem takes as input a regular language L and asks if $L \in \mathcal{C}$. Intuitively, obtaining a procedure for \mathcal{C} -membership requires a solid understanding of \mathcal{C} . We also look at a more involved problem, called separation. For a class \mathcal{C} and two languages L_0 and L_1 , we say that L_0 is \mathcal{C} -separable from L_1 when 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 . We do *not* present separation algorithms in this paper: we only use them as an intermediary to investigate *membership*.

We turn to a key tool. Let \mathcal{C} be a prevariety. A \mathcal{C} -morphism is a surjective morphism $\eta : A^* \rightarrow N$ such that $\eta^{-1}(F) \in \mathcal{C}$ for all $F \subseteq N$. We use this notion to handle membership. It is well known that for every regular language L , there exists a canonical morphism recognizing L . We briefly recall its definition. We associate with L an equivalence \equiv_L on A^* : given $u, v \in A^*$, we let $u \equiv_L v$ when $xuy \in L \Leftrightarrow xvy \in L$ for all $x, y \in A^*$. One may verify that \equiv_L is a congruence and that, since L is regular, it has finite index. Thus, the quotient set $M_L = A^*/\equiv_L$ is a finite monoid. The *syntactic morphism* of L is the morphism $\alpha_L : A^* \rightarrow M_L$ which maps a word to its \equiv_L -class. It can be computed from any representation of L . We have the following standard property (see, e.g., [17, Proposition 3])

Proposition 1. *Let \mathcal{C} be a prevariety. A regular language L belongs to \mathcal{C} if and only if its syntactic morphism $\alpha_L : A^* \rightarrow M_L$ is a \mathcal{C} -morphism.*

In view of Proposition 1, getting an algorithm for \mathcal{C} -membership boils down to finding a procedure that decides whether an input morphism $\alpha : A^* \rightarrow M$ 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 1.

Proposition 2. *Let \mathcal{C} be a prevariety, $k \geq 1$ and $L_1, \dots, L_k \in \mathcal{C}$. There exists a \mathcal{C} -morphism $\eta : A^* \rightarrow N$ such that L_1, \dots, L_k are recognized by η .*

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 shall 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 example is the class AMT of *alphabet modulo testable languages*. For all $w \in A^*$ and $a \in A$, we let $\#_a(w) \in \mathbb{N}$ be 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 $k, m \in \mathbb{N}$ are such that $k < m$ (these are exactly the languages recognized by commutative groups). We also look at 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.

By definition, $\{\varepsilon\}$ and A^+ are *not* group languages. This motivates the next definition: the *well-suited extension* of a class \mathcal{C} , written \mathcal{C}^+ , consists of all languages $L \cap A^+$ and $L \cup \{\varepsilon\}$ for $L \in \mathcal{C}$ (hence, $\mathcal{C} \subseteq \mathcal{C}^+$). The following lemma is easy.

Lemma 1. *We have $\{\varepsilon\}, A^+ \in \mathcal{C}^+$. In addition, if \mathcal{C} is a prevariety, so is \mathcal{C}^+ .*

We conclude with a lemma concerning \mathcal{G} -morphisms and \mathcal{G}^+ -morphisms.

Lemma 2. *Let \mathcal{G} be a group prevariety and $\eta : A^* \rightarrow M$ be a morphism. If η is a \mathcal{G} -morphism, then M is a group. If η is a \mathcal{G}^+ -morphism, then $\eta(A^+)$ is a group.*

3 Future/past hierarchies of unary temporal logic

We define unary temporal logic and its future/past hierarchies. We work with a generalized definition of unary temporal logic introduced in [23]: with every class \mathcal{C} , we associate a logic $\text{TL}(\mathcal{C})$ and its future/past hierarchy.

3.1 Definition

Syntax. We first recall the definition of the full logic used in [23]. A TL formula is built from atomic formulas using Boolean connectives and temporal operators. The atomic formulas are \top , \perp , \min , \max and a for every letter $a \in A$. All Boolean connectives are allowed: if ψ_1 and ψ_2 are TL formulas, then so are $(\psi_1 \vee \psi_2)$, $(\psi_1 \wedge \psi_2)$ and $(\neg\psi_1)$. We also associate *two temporal operators* with every language $L \subseteq A^*$ which we write F_L and P_L : if ψ is a TL formula, then so are $(F_L \psi)$ and $(P_L \psi)$. We omit parentheses when there is no ambiguity.

We now classify the TL formulas by counting the alternations between future and past operators in their parse tree. With all $n \in \mathbb{N}$, we associate three sets of TL formulas: FL^n , PL^n and BL^n (where F, P and B stand for future, past and Boolean combinations respectively). The first two are defined by induction on $n \in \mathbb{N}$. The FL^0 and PL^0 formulas are the Boolean combinations of atomic formulas (*i.e.*, they do not contain temporal operators). Assume now that $n \geq 1$.

- FL^n is the least set containing the PL^{n-1} formulas and closed under Boolean connectives and *future* operators: if $\psi \in \text{FL}^n$ and $L \subseteq A^*$, then $F_L \psi \in \text{FL}^n$.
- PL^n is the least set containing the FL^{n-1} formulas and closed under Boolean connectives and *past* operators: if $\psi \in \text{PL}^n$ and $L \subseteq A^*$, then $P_L \psi \in \text{PL}^n$.

Finally, for each $n \in \mathbb{N}$, we define the BL^n formulas as the Boolean combinations of FL^n and PL^n formulas.

Semantics. Evaluating a TL formula φ requires a word $w \in A^*$ and a position $i \in \text{Pos}(w)$. We use structural induction on φ to define what it means for (w, i) to *satisfy* φ . We denote this property by $w, i \models \varphi$:

- $w, i \models \top$ always holds and $w, i \models \perp$ never holds.
- for $\ell \in A \cup \{\min, \max\}$, $w, i \models \ell$ holds if $w[i] = \ell$.
- $w, i \models \psi_1 \vee \psi_2$ if $w, i \models \psi_1$ or $w, i \models \psi_2$.
- $w, i \models \psi_1 \wedge \psi_2$ if $w, i \models \psi_1$ and $w, i \models \psi_2$.
- $w, i \models \neg\psi$ if $w, i \models \psi$ does not hold.
- $w, i \models F_L \psi$ if there is $j \in \text{Pos}(w)$ such that $i < j$, $w(i, j) \in L$ and $w, j \models \psi$.
- $w, i \models P_L \psi$ if there is $j \in \text{Pos}(w)$ such that $j < i$, $w(j, i) \in L$ and $w, j \models \psi$.

When no distinguished position is specified, we evaluate formulas at *the two unlabeled positions* 0 and $|w| + 1$ *simultaneously*. Given a TL formula φ and $w \in A^*$, we write $w \models \varphi$ and say that w *satisfies* φ when $w, 0 \models \varphi$ and $w, |w| + 1 \models \varphi$. The *language defined by* φ is $L(\varphi) = \{w \in A^* \mid w \models \varphi\}$.

Each language L is defined by “ $(\min \wedge F_L \max) \vee \max$ ”. Thus, we restrict the available formulas using a class \mathcal{C} . Let $n \in \mathbb{N}$ and $Z \in \{\text{TL}, \text{FL}^n, \text{PL}^n, \text{BL}^n\}$. We write $Z[\mathcal{C}]$ for the set of all Z formulas φ such that every operator F_L or P_L occurring in φ satisfies $L \in \mathcal{C}$. Finally, we write $Z(\mathcal{C})$ for the class of all languages that can be defined by a $Z[\mathcal{C}]$ formula. We are mainly interested in the case when \mathcal{C} is a group prevariety \mathcal{G} or its well-suited extension \mathcal{G}^+ .

Example 1. Let $\text{ST} = \{\emptyset, A^*\}$. Then, $\text{TL}(\text{ST})$ corresponds to a classic variant of unary temporal logic: F_{A^*} or P_{A^*} are the standard operators “sometimes in the future” (F) and “sometimes in the past” (P). Let $A = \{a, b, c\}$. The language $L = a^*bA^*ca^*$ is defined by $(\min \wedge F(c \wedge \neg F b)) \vee (\max \wedge P(b \wedge \neg P c))$, which is a $\text{BL}^1[\text{ST}]$ formula. Note that it is important here that formulas be evaluated at *both* unlabeled positions simultaneously: the formula states that $F(c \wedge \neg F b)$ holds at “0” and $P(b \wedge \neg P c)$ holds at “ $|w| + 1$ ”. We get $L \in \text{BL}^1(\text{ST})$.

The logic $\text{TL}(\text{MOD})$ is also interesting. Let $K = A^*b(aa)^*$ which is defined by the $\text{FL}^1[\text{MOD}]$ formula $(\min \wedge F(b \wedge (\neg F(b \vee c)) \wedge F_{(AA)^*} \max)) \vee \max$ (clearly, $A^*, (AA)^* \in \text{MOD}$). We get $K \in \text{FL}^1(\text{MOD})$. Finally, it is also natural to consider $\text{TL}(\text{AMT})$: when using a temporal operator, one may then count the number of occurrences of each letter between two positions, modulo some integer.

Remark 2. Well-suited extensions are natural inputs as well. It is shown in [23] that for all prevarieties \mathcal{C} , we have $\text{TL}(\mathcal{C}^+) = \text{TLX}(\mathcal{C})$ where TLX is a stronger variant which allows additional operators “tomorrow” (X) and “yesterday” (Y). Roughly, the idea is that since $\{\varepsilon\} \in \mathcal{C}^+$ by Lemma 1, one may use the operators $F_{\{\varepsilon\}}$ and $P_{\{\varepsilon\}}$ in $\text{TL}[\mathcal{C}^+]$ formulas. They are clearly equivalent to X and Y. Hence, $\text{TL}(\text{ST}^+) = \text{TLX}(\text{ST})$ is the standard variant of unary temporal logic (which allows F, P, X and Y). While we do not detail this point due to space limitations, the result of [23] extends to future/past hierarchies (the proof is identical). For example, $\text{BL}^n(\mathcal{C}^+) = \text{BLX}^n(\mathcal{C})$ for all $n \in \mathbb{N}$.

3.2 Tools

We define equivalence relations that we shall use as tools. For each TL formula φ , we define the *rank* of φ as the length of the longest sequence of *nested temporal operators* within its parse tree: the rank of an atomic formula is 0, the rank of $(\psi_1 \vee \psi_2)$ and $(\psi_1 \wedge \psi_2)$ is the maximum between the ranks of ψ_1 and ψ_2 , the rank of $(\neg\psi)$ is the rank of ψ and the rank of $(F_L \psi)$ and $(P_L \psi)$ is the rank of ψ plus one. Moreover, given $n \in \mathbb{N}$ and a morphism $\eta : A^* \rightarrow N$, an $\text{FL}^n[\eta]$ (resp. $\text{PL}^n[\eta]$) formula is an FL^n (resp. PL^n) formula φ such that for all operators F_L or P_L occurring in φ , the language $L \subseteq A^*$ is recognized by η .

We are ready to define our equivalences. Let $\eta : A^* \rightarrow N$ be a morphism and let $k, n \in \mathbb{N}$. We define two relations on pairs (w, i) , where $w \in A^*$ and $i \in$

$\text{Pos}(w)$. Let $w, w' \in A^*$, $i \in \text{Pos}(w)$ and $i' \in \text{Pos}(w')$. We write $w, i \blacktriangleright_{n,\eta,k} w', i'$ (resp. $w, i \blacktriangleleft_{n,\eta,k} w', i'$) to mean that for every $\text{FL}^n[\eta]$ (resp. $\text{PL}^n[\eta]$) formula φ of rank at most k , we have $w, i \models \varphi \Leftrightarrow w', i' \models \varphi$. Note that despite the notation, $w, i \blacktriangleright_{n,\eta,k} w', i'$ does not entail $w', i' \blacktriangleleft_{n,\eta,k} w, i$ in general. By definition, $\blacktriangleright_{n,\eta,k}$ and $\blacktriangleleft_{n,\eta,k}$ are *equivalences* of finite index (one may verify that there are finitely many non-equivalent $\text{FL}^n[\eta]$ (resp. $\text{PL}^n[\eta]$) formulas of rank at most k).

We adapt these relations to words. Let $\cong \in \{\blacktriangleright_{n,\eta,k}, \blacktriangleleft_{n,\eta,k}\}$ and $w, w' \in A^*$. We write $w \cong w'$ when $w, 0 \cong w', 0$ and $w, |w| + 1 \cong w', |w'| + 1$. This defines equivalences of finite index on A^* . We use them to characterize the two classes $\text{FL}^n(\mathcal{C})$ and $\text{PL}^n(\mathcal{C})$ for a given prevariety \mathcal{C} (the proof is presented in the full version of the paper).

Lemma 3. *Let \mathcal{C} be a prevariety, $n \geq 1$ and $L \subseteq A^*$. Then, $L \in \text{FL}^n(\mathcal{C})$ (resp. $L \in \text{PL}^n(\mathcal{C})$) if and only if there exists a \mathcal{C} -morphism $\eta : A^* \rightarrow N$ and $k \in \mathbb{N}$ such that L is a union of $\blacktriangleright_{n,\eta,k}$ -classes (resp. of $\blacktriangleleft_{n,\eta,k}$ -classes).*

We complete the definition with properties. The proofs rely on arguments similar to Ehrenfeucht-Fraïssé games (strictly speaking, we use alternative inductive definitions of $\blacktriangleright_{n,\eta,k}$ and $\blacktriangleleft_{n,\eta,k}$ rather than a “game”). They are presented in the full version of the paper. First, these equivalences are congruences.

Lemma 4. *Let $\eta : A^* \rightarrow N$ be a morphism, and let $n \geq 1$, $k \in \mathbb{N}$ and $\cong \in \{\blacktriangleright_{n,\eta,k}, \blacktriangleleft_{n,\eta,k}\}$. For every $u, v, u', v' \in A^*$, if $u \cong v$ and $u' \cong v'$, then $uu' \cong vv'$.*

We now consider the special case of morphisms $\eta : A^* \rightarrow N$ such that $\eta(A^+)$ is a *group* (this is mandatory in the next two results). In view of Lemma 2, this corresponds to the case $\mathcal{C} \in \{\mathcal{G}, \mathcal{G}^+\}$ for a group prevariety \mathcal{G} . We present two properties for the cases $n = 1$ and $n > 1$. We shall use them in Section 4 to establish the language theoretic characterization of future/past hierarchies.

We start with the case $n = 1$. Let $\eta : A^* \rightarrow N$ be a morphism and $k \in \mathbb{N}$. We define an equivalence $\sim_{\eta,k}$ on A^* , in *two* steps. First, given $w, w' \in A^*$, $i \in \text{Pos}(w)$ and $i' \in \text{Pos}(w')$, we write $w, i \equiv_{\eta,k} w', i'$ if the following conditions hold:

1. We have $w[i] = w'[i']$.
2. If $k \geq 1$, then $\eta(w(0, i)) = \eta(w(0, i'))$ and $\eta(w(i, |w|+1)) = \eta(w'(i', |w'|+1))$.
3. If $\eta^{-1}(1_N) = \{\varepsilon\}$, then for every $h \in \mathbb{Z}$ such that $-k \leq h \leq k$, we have $i + h \in \text{Pos}(w) \Leftrightarrow i' + h \in \text{Pos}(w')$ and in that case, $w[i + h] = w'[i' + h]$.

When $\eta^{-1}(1_N) \cap A^+ \neq \emptyset$, Condition 3 is trivial: it can be discarded (roughly, this hypothesis distinguishes the case $\mathcal{C} = \mathcal{G}$ from $\mathcal{C} = \mathcal{G}^+$). Finally, given $w, w' \in A^*$, we let $w \sim_{\eta,k} w'$ if for all $i \in \text{Pos}(w)$ (resp. $i' \in \text{Pos}(w')$) there exists $i' \in \text{Pos}(w')$ (resp. $i \in \text{Pos}(w)$) such that $w, i \equiv_{\eta,k} w', i'$. We may now state our first property.

Lemma 5. *Let $\eta : A^* \rightarrow N$ be a morphism such that $\eta(A^+)$ is a group and let $k \in \mathbb{N}$. Let $p = \omega(N)$. For all $u, v \in A^*$, if $u \sim_{\eta,k} v$, then $uw^{2kp} \blacktriangleright_{1,\eta,k} v^{2kp+1}$ and $v^{2kp}u \blacktriangleleft_{1,\eta,k} v^{2kp+1}$.*

We now present a second property, which we shall use in order to handle the case when $n > 1$.

Proposition 3. *Let $\eta : A^* \rightarrow N$ be a morphism such that $\eta(A^+)$ is a group and $k \in \mathbb{N}$. Let $p = \omega(N)$. The following properties hold for all $n \geq 1$ and $u, v \in A^*$,*

- *If n is even and $u \blacktriangleright_{n,\eta,k^2} v$, then $uv^{2kp} \blacktriangleright_{n+1,\eta,k} v^{2kp+1}$.*
- *If n is odd and $u \blacktriangleleft_{n,\eta,k^2} v$, then $uv^{2kp} \blacktriangleleft_{n+1,\eta,k} v^{2kp+1}$.*
- *If n is odd and $u \blacktriangleright_{n,\eta,k^2} v$, then $v^{2kp}u \blacktriangleright_{n+1,\eta,k} v^{2kp+1}$.*
- *If n is even and $u \blacktriangleleft_{n,\eta,k^2} v$, then $v^{2kp}u \blacktriangleleft_{n+1,\eta,k} v^{2kp+1}$.*

Remark 3. It might be surprising that there are *four* cases in Proposition 3. This is because the property does not only depend on the outermost kind of temporal operator in formulas (*i.e.*, future for $\text{FL}^n[\eta]$ and past for $\text{PL}^n[\eta]$) but also on the *innermost* kind (which depends on whether n is odd or even).

4 Characterization by deterministic hierarchies

We present a generic language theoretic characterization of future/past hierarchies associated with *group* prevarieties. It generalizes the characterization of the full logic presented in [23]. It is based on variants of *polynomial closure*.

4.1 Polynomial closure

Given finitely many 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 $a_1, \dots, a_n \in A$. In particular, a single language L_0 is a marked product (this is the case $n = 0$).

The *polynomial closure* of a class \mathcal{C} , denoted by $\text{Pol}(\mathcal{C})$, consists of all *finite unions* of marked products $L_0 a_1 L_1 \cdots a_n L_n$ such that $L_0, \dots, L_n \in \mathcal{C}$. If \mathcal{C} is a prevariety, then $\text{Pol}(\mathcal{C})$ is a quotient-closed lattice (this is due to Arfi [2], see also [14,20] for recent proofs). Yet, $\text{Pol}(\mathcal{C})$ need not be closed under complement. Hence, it is often combined with another operator. The Boolean closure of a class \mathcal{D} , denoted by $\text{Bool}(\mathcal{D})$, is the smallest Boolean algebra containing \mathcal{D} . Finally, we write $\text{BPol}(\mathcal{C})$ for $\text{Bool}(\text{Pol}(\mathcal{C}))$. The following result is standard (see [20], for example).

Proposition 4. *If \mathcal{C} is a prevariety, then so is $\text{BPol}(\mathcal{C})$.*

Remark 4. The classes $\text{Pol}(\mathcal{C})$ and $\text{BPol}(\mathcal{C})$ are quite prominent. For example, $\text{Pol}(\text{ST})$ contains exactly the finite unions of languages $A^* a_1 A^* \cdots a_n A^*$ where $n \in \mathbb{N}$ and $a_1, \dots, a_n \in A$ are letters. Moreover, $\text{BPol}(\text{ST})$ consists of all finite Boolean combinations of such languages: this is the famous class of *piecewise testable languages* [25]. In the literature, such classes are more often associated with classical logic rather than with temporal logic. Indeed, it is well known [28,20] that BPol corresponds to the quantifier alternation free fragment of first-order logic (\mathcal{BS}_1). For each prevariety \mathcal{C} , there exists a set of first-order predicates $\mathbb{I}_{\mathcal{C}}$ such that $\text{BPol}(\mathcal{C})$ contains exactly the languages that can be defined by a \mathcal{BS}_1 sentence using only predicates in $\mathbb{I}_{\mathcal{C}}$. On the other hand, no characterization of BPol based on temporal logic is known. In order to establish a connection with unary temporal logic, we have to apply additional operators on top of BPol .

4.2 Deterministic variants

We consider four restrictions of *Pol*: *UPol*, *LPol*, *RPol* and *MPol*. The first three are standard (see for example [24,13,15]). On the other hand, *MPol* was introduced recently in [18]. We restrict the marked products to those satisfying specific semantic conditions and the unions to *disjoint* ones. Consider a marked product $L_0a_1L_1 \cdots a_nL_n$. For $1 \leq i \leq n$, we let $L'_i = L_0a_1L_1 \cdots a_{i-1}L_{i-1}$ and $L''_i = L_ia_{i+1} \cdots L_{n-1}a_nL_n$. In particular, $L'_1 = L_0$ and $L''_n = L_n$. We say that,

- $L_0a_1L_1 \cdots a_nL_n$ is *left deterministic* when $L'_i \cap L'_ia_iA^* = \emptyset$ for $1 \leq i \leq n$.
- $L_0a_1L_1 \cdots a_nL_n$ is *right deterministic* when $L''_i \cap A^*a_iL''_i = \emptyset$ for $1 \leq i \leq n$.
- $L_0a_1L_1 \cdots a_nL_n$ is *mixed deterministic* when either $L'_i \cap L'_ia_iA^* = \emptyset$, or $L''_i \cap A^*a_iL''_i = \emptyset$ for $1 \leq i \leq n$.
- $L_0a_1L_1 \cdots a_nL_n$ is *unambiguous* when for all $w \in L_0a_1L_1 \cdots a_nL_n$, there is a *unique* decomposition $w = w_0a_1w_1 \cdots a_nw_n$ where $w_i \in L_i$ for $1 \leq i \leq n$.

By definition, a left or right deterministic marked product is also mixed deterministic. It is also simple to verify that mixed deterministic marked products are unambiguous. Note that these four notions depend on the product itself and not only on the resulting language. For example, A^*aA^* (which is not unambiguous) and $(A \setminus \{a\})^*aA^*$ (which is left deterministic) evaluate to the same language.

Remark 5. A mixed deterministic product need not be left or right deterministic. For example, let $L_1 = (ab)^+$, $L_2 = c^+$ and $L_3 = (ba)^+$. The product $L_1cL_2cL_3$ is mixed deterministic since $L_1 \cap L_1cA^* = \emptyset$ and $L_3 \cap A^*cL_3 = \emptyset$. However, it is neither left deterministic nor right deterministic. Similarly, a unambiguous product need not be mixed deterministic. If $L_4 = (ca)^+$, the product L_1aL_4 is unambiguous but it neither left nor right deterministic.

The *left polynomial closure* of a class \mathcal{C} , written $LPol(\mathcal{C})$, consists of all *finite disjoint unions* of *left deterministic marked products* $L_0a_1L_1 \cdots a_nL_n$ such that $L_0, \dots, L_n \in \mathcal{C}$ (by “disjoint” we mean that the languages in the union must be pairwise disjoint). The *right polynomial closure* of \mathcal{C} ($RPol(\mathcal{C})$), the *mixed polynomial closure* of \mathcal{C} ($MPol(\mathcal{C})$) and the *unambiguous polynomial closure* of \mathcal{C} ($UPol(\mathcal{C})$) are defined analogously by replacing the requirement to be “left deterministic” for marked products by the appropriate one.

We introduce a key property of these operators, which is not apparent on the definition: when applied to a prevariety, they also yield a prevariety. Moreover, in that case, the four operators preserve the decidability of *membership*. This is proved in [19,23] for *UPol* and in [18] for *LPol*, *RPol* and *MPol*. From this, we shall obtain decidability of membership for classes built with FL^n , PL^n and BL^n .

Theorem 1 ([23,18]). *Let $X \in \{UPol, LPol, RPol, MPol\}$. For every prevariety \mathcal{C} , the class $X(\mathcal{C})$ is a prevariety as well. Moreover, if \mathcal{C} has decidable membership, then so does $X(\mathcal{C})$.*

For each operator, Theorem 1 is based on a generic algebraic characterization of the classes that it builds. In Theorem 2 below, we recall the symmetric

characterizations of $LPol$ and $RPol$, as we shall need them in order to establish the correspondence with future/past hierarchies. This requires two notions.

Let \mathcal{C} be a prevariety and let $\alpha : A^* \rightarrow M$ be a morphism. We define *two* relations on M (both depending on α). Given $s, t \in M$, we say that (s, t) is a \mathcal{C} -pair if $\alpha^{-1}(s)$ is *not* \mathcal{C} -separable from $\alpha^{-1}(t)$. This relation is not very robust: it is reflexive (if α is surjective) and symmetric (this is tied to \mathcal{C} being closed under complement). The second relation is an equivalence “ $\sim_{\mathcal{C}}$ ” on M . For $s, t \in M$, we write $s \sim_{\mathcal{C}} t$ when $s \in F \Leftrightarrow t \in F$ for every $F \subseteq M$ such that $\alpha^{-1}(F) \in \mathcal{C}$. By definition, $\sim_{\mathcal{C}}$ is an equivalence relation. In fact, it is shown in [23,20] that it is the reflexive transitive closure of the “ \mathcal{C} -pair” relation (we do not use this property). We now present the characterizations of $LPol$ and $RPol$ taken from [18]. They are crucial for proving Theorem 3 below, which expresses $FL^n(\mathcal{C}), PL^n(\mathcal{C}), BL^n(\mathcal{C})$ in terms of the operators $LPol, RPol$ and $BPol$.

Theorem 2 ([18]). *Let \mathcal{C} be a prevariety and let $\alpha : A^* \rightarrow M$ be a surjective morphism. The three following properties are equivalent:*

1. *The morphism α is an $LPol(\mathcal{C})$ -morphism (resp. an $RPol(\mathcal{C})$ -morphism).*
2. *For all \mathcal{C} -pairs $(s, t) \in M^2$, we have $s^{\omega+1} = s^{\omega}t$ (resp. $s^{\omega+1} = ts^{\omega}$).*
3. *For all $s, t \in M$ such that $s \sim_{\mathcal{C}} t$, we have $s^{\omega+1} = s^{\omega}t$ (resp. $s^{\omega+1} = ts^{\omega}$).*

This implies the statement on membership in Theorem 1 for $LPol$ and $RPol$. Let us explain how on $LPol$, for instance. By Proposition 1, deciding $LPol(\mathcal{C})$ -membership boils down to deciding if an input morphism $\alpha : A^* \rightarrow M$ is an $LPol(\mathcal{C})$ -morphism. By the third assertion in Theorem 2, this is possible if one can compute the equivalence $\sim_{\mathcal{C}}$ on M . By definition, this boils down to \mathcal{C} -membership (it suffices to compute all subsets $F \subseteq M$ such that $\alpha^{-1}(F) \in \mathcal{C}$).

4.3 Characterization of future/past hierarchies

It is well known that there is a correspondence between full unary temporal logic and $UPol$. This was first proved for the standard variants in [27,4,16], and was then generalized to our extended definition in [23]. More precisely, for each *group* prevariety \mathcal{G} , we have $TL(\mathcal{G}) = UPol(BPol(\mathcal{G}))$ and $TL(\mathcal{G}^+) = UPol(BPol(\mathcal{G}^+))$. Here, we generalize these results to future/past hierarchies.

We use $LPol$ and $RPol$ to define hierarchies (the definition is taken from [18]). It is shown in [19,23] that for all prevarieties \mathcal{C} , the class $UPol(\mathcal{C})$ is the least one containing \mathcal{C} and closed under left and right deterministic marked products as well as disjoint union. Thus, applying $LPol$ and $RPol$ in alternation builds a classification of $UPol(\mathcal{C})$: the *deterministic hierarchy of basis \mathcal{C}* . For all $n \in \mathbb{N}$, we define two levels $LPol_n(\mathcal{C})$ and $RPol_n(\mathcal{C})$. We let $LPol_0(\mathcal{C}) = RPol_0(\mathcal{C}) = \mathcal{C}$. For $n \geq 1$, $LPol_n(\mathcal{C}) = LPol(RPol_{n-1}(\mathcal{C}))$ and $RPol_n(\mathcal{C}) = RPol(LPol_{n-1}(\mathcal{C}))$. The union of all levels is exactly $UPol(\mathcal{C})$. These are strict hierarchies and the levels $LPol_n(\mathcal{C})$ and $RPol_n(\mathcal{C})$ are incomparable for all $n \geq 1$, in general. This motivates additional intermediary levels “combining” $LPol_n(\mathcal{C})$ and $RPol_n(\mathcal{C})$: for all $n \geq 1$, we let $LPol_n(\mathcal{C}) \vee RPol_n(\mathcal{C})$ be the least Boolean algebra containing both $LPol_n(\mathcal{C})$ and $RPol_n(\mathcal{C})$.

For every prevariety \mathcal{C} , we connect the future/past hierarchy of $\text{TL}(\mathcal{C})$ with the deterministic hierarchy of basis $\text{BPol}(\mathcal{C})$. In the general case, we only prove that the latter is included in the former. This inclusion is *strict* in general: an example of prevariety \mathcal{C} such that $\text{UPol}(\text{BPol}(\mathcal{C}))$ is *strictly* included in $\text{TL}(\mathcal{C})$ is provided in [23] (strictness follows from results of [7]).

Proposition 5. *Let \mathcal{C} be prevariety. The following properties hold for all $n \geq 1$:*

1. *If n is odd, $\text{RPol}_n(\text{BPol}(\mathcal{C})) \subseteq \text{FL}^n(\mathcal{C})$ and $\text{LPol}_n(\text{BPol}(\mathcal{C})) \subseteq \text{PL}^n(\mathcal{C})$.*
2. *If n is even, $\text{LPol}_n(\text{BPol}(\mathcal{C})) \subseteq \text{FL}^n(\mathcal{C})$ and $\text{RPol}_n(\text{BPol}(\mathcal{C})) \subseteq \text{PL}^n(\mathcal{C})$.*

Remark 6. There are four cases in Proposition 5. This is because for every level $\text{RPol}_n(\text{BPol}(\mathcal{C}))$ or $\text{LPol}_n(\text{BPol}(\mathcal{C}))$, the notation highlights the last operator used in its construction from $\text{BPol}(\mathcal{C})$. However, the logic corresponding to this level is determined by the *first* operator in the construction. For example, we have $\text{RPol}(\text{BPol}(\mathcal{C})) \subseteq \text{FL}^1(\mathcal{C})$ and all classes which are built from $\text{RPol}(\text{BPol}(\mathcal{C}))$ by applying LPol and RPol in alternation are included in a level $\text{FL}^n(\mathcal{C})$.

Proof (of Proposition 5). We use induction on n . There are four cases. We prove that if n is odd, then $\text{RPol}_n(\text{BPol}(\mathcal{C})) \subseteq \text{FL}^n(\mathcal{C})$ (the other cases are symmetric). Let $\mathcal{D} = \text{LPol}_{n-1}(\text{BPol}(\mathcal{C}))$ and fix $L \in \text{RPol}_n(\text{BPol}(\mathcal{C})) = \text{RPol}(\mathcal{D})$. We prove that $L \in \text{FL}^n(\mathcal{C})$. We need the next easy lemma.

Lemma 6. *Let $K \in \text{BPol}(\mathcal{C})$. There exists an $\text{FL}^1[\mathcal{C}]$ formula ξ_K such that for all $w \in A^*$ and all $i \in \text{Pos}(w)$, we have*

$$w, i \models \xi_K \iff i \leq |w| \text{ and } w(i, |w| + 1) \in K.$$

Proof. By definition, $K \in \text{BPol}(\mathcal{C})$ is a Boolean combination of languages of the form $K_0 a_1 K_1 \cdots a_n K_n$ with $a_1, \dots, a_n \in A$ and $K_0, \dots, K_n \in \mathcal{C}$. Since we may use Boolean connectives freely in $\text{FL}^1[\mathcal{C}]$, we may assume without loss of generality that K itself is of the form $K_0 a_1 K_1 \cdots a_n K_n$. It now suffices to verify that the following formula ξ_K satisfies the desired property:

$$\xi_K = \text{F}_{K_0} (a_1 \wedge \text{F}_{K_1} (a_2 \wedge \text{F}_{K_2} (\cdots a_n \wedge \text{F}_{K_n} \text{max}))). \quad \square$$

Since $L \in \text{RPol}(\mathcal{D})$ with $\mathcal{D} = \text{LPol}_{n-1}(\text{BPol}(\mathcal{C}))$, it is shown in [18, Proposition 5.3] that L is a finite union of products $L_0 a_1 L_1 \cdots a_m L_m$ satisfying the two following conditions: (1) $L_h \in \mathcal{D}$ for every $h \leq m$, and (2) there exists a right deterministic marked product $K_0 a_1 K_1 \cdots a_m K_m$ such that $K_h \in \text{BPol}(\mathcal{C})$ and $L_h \subseteq K_h$ for every $h \leq m$. Hence, by closure under union, it suffices to prove that every such product $L_0 a_1 L_1 \cdots a_m L_m$ belongs to $\text{FL}^n(\mathcal{C})$. We use a subinduction on h to prove that $L_0 a_1 L_1 \cdots a_h L_h \in \text{FL}^n(\mathcal{C})$ for $0 \leq h \leq m$.

When $h = 0$, there are two cases. If $n = 1$, then $\mathcal{D} = \text{BPol}(\mathcal{C})$ which means that $L_0 \in \text{BPol}(\mathcal{C})$ and it is defined by the $\text{FL}^1[\mathcal{C}]$ formula $(\text{min} \wedge \xi_{L_0}) \vee \text{max}$, where ξ_{L_0} is given by Lemma 6. Otherwise, $n > 1$ and $\mathcal{D} = \text{LPol}_{n-1}(\text{BPol}(\mathcal{C}))$ with $n - 1 \geq 1$. Consequently, since $(n - 1)$ is even, the main induction on n yields $L_0 \in \mathcal{D} \subseteq \text{FL}^{n-1}(\mathcal{C}) \subseteq \text{FL}^n(\mathcal{C})$, which completes this case.

We now assume that $h \geq 1$. Let $R = L_0 a_1 L_1 \cdots a_{h-1} L_{h-1}$. We prove that $R a_h L_h \in \text{FL}^n(\mathcal{C})$. Induction on h yields $\text{FL}^n[\mathcal{C}]$ formulas φ_R and φ_{L_h} defining R and L_h . We also use the $\text{FL}^1[\mathcal{C}]$ formula ξ_{K_h} associated with $K_h \in \text{BPol}(\mathcal{C})$ by Lemma 6. We write ψ for the $\text{FL}^1[\mathcal{C}]$ formula $a_h \wedge \xi_{K_h}$. Since $K_0 a_1 K_1 \cdots a_m K_m$ is right deterministic, one may verify that $A^* a_h K_h$ is unambiguous. Hence, by definition of ψ , for every $w \in A^*$, there exists *at most one* position $i \in \text{Pos}(w)$ such that $w, i \models \psi$. Therefore, since $L_h \subseteq K_h$, it follows that for every $w \in A^*$, we have $w \in R a_h L_h$ if and only if w satisfies the three following conditions:

1. There exists $i \in \text{Pos}(w)$ (which must be unique) such that $w, i \models \psi$.
2. The prefix $w(0, i)$ belongs to R (i.e., we have $w(0, i) \models \varphi_R$).
3. The suffix $w(i, |w| + 1)$ belongs to L_h (i.e., we have $w(i, |w| + 1) \models \varphi_{L_h}$).

It remains to prove that these properties can be expressed in $\text{FL}^n[\mathcal{C}]$. Condition 1 is expressed by the $\text{FL}^1[\mathcal{C}]$ formula $(\min \wedge \text{F } \psi) \vee \max$. We turn to Condition 2. We modify φ_R into a new formula φ'_R expressing the desired property. For every word $w \in A^*$, we restrict the evaluation of φ_R to the positions $j \in \text{Pos}(w)$ such that either $w, j \models \text{F } \psi$ or $j = |w| + 1$. We build φ'_R by applying the two following modifications to φ_R . First, we recursively replace every subformula $\text{P}_U \zeta$ by $(\max \wedge \text{P } (\psi \wedge \text{P}_U \zeta)) \vee (\text{F } \psi \wedge \text{P}_U \zeta)$. Second, we replace every subformula $\text{F}_U \zeta$ by $(\text{F}_U (\zeta \wedge \text{F } \psi)) \vee (\text{F}_U (\psi \wedge \text{F } (\max \wedge \zeta)))$. Since φ_R is an $\text{FL}^n[\mathcal{C}]$ formula and n is *odd*, one may verify that φ'_R is also an $\text{FL}^n[\mathcal{C}]$ formula. The key point is that since n is odd, the $\text{FL}^n[\mathcal{C}]$ formulas are defined inductively from the $\text{FL}^1[\mathcal{C}]$ formulas and inserting the $\text{FL}^1[\mathcal{C}]$ formula ψ in an $\text{FL}^n[\mathcal{C}]$ formula yields a new $\text{FL}^n[\mathcal{C}]$ formula. It can be verified that φ'_R expresses Condition 2.

We turn to Condition 3. We modify φ_{L_h} into another formula φ'_{L_h} expressing the desired property. For all $w \in A^*$, we restrict the evaluation of φ_{L_h} to the positions $j \in \text{Pos}(w)$ such that $j = 0$ or $w, j \models \text{P } \psi$. We build φ'_{L_h} by applying the two following modifications to φ_{L_h} . We recursively replace every subformula $\text{P}_U \zeta$ by $(\text{P}_U (\psi \wedge \text{P } (\min \wedge \zeta))) \vee (\text{P}_U (\zeta \wedge \text{P } \psi))$. Moreover, we replace every subformula $\text{F}_U \zeta$ by $(\min \wedge \text{F } (\psi \wedge \text{F}_U \zeta)) \vee (\text{P } \psi \wedge \text{F}_U \zeta)$. As in the previous case, since n is odd, one may verify that φ'_{L_h} remains an $\text{FL}^n[\mathcal{C}]$ formula and that it expresses Condition 3. Finally, the language $R a_h L_h$ is now defined by the following $\text{FL}^n[\mathcal{C}]$ formula: $((\min \wedge \text{F } \psi) \vee \max) \wedge \varphi'_R \wedge \varphi'_{L_h}$. \square

With Proposition 5 in hand, we may now consider the case when $\mathcal{C} \in \{\mathcal{G}, \mathcal{G}^+\}$ for a group prevariety \mathcal{G} . In this case, there is an exact correspondence.

Theorem 3. *Let \mathcal{G} be a group prevariety and let $\mathcal{C} \in \{\mathcal{G}, \mathcal{G}^+\}$. The three following properties hold for every $n \geq 1$:*

1. *If n is odd, $\text{FL}^n(\mathcal{C}) = \text{RPol}_n(\text{BPol}(\mathcal{C}))$ and $\text{PL}^n(\mathcal{C}) = \text{LPol}_n(\text{BPol}(\mathcal{C}))$.*
2. *If n is even, $\text{FL}^n(\mathcal{C}) = \text{LPol}_n(\text{BPol}(\mathcal{C}))$ and $\text{PL}^n(\mathcal{C}) = \text{RPol}_n(\text{BPol}(\mathcal{C}))$.*
3. *We have $\text{BL}^n(\mathcal{C}) = \text{LPol}_n(\text{BPol}(\mathcal{C})) \vee \text{RPol}_n(\text{BPol}(\mathcal{C}))$.*

Before we prove Theorem 3, let us discuss its consequences. An important application is membership for future/past hierarchies. Let \mathcal{G} be a group prevariety and $\mathcal{C} \in \{\mathcal{G}, \mathcal{G}^+\}$. In view of Theorem 3 and Theorem 1, it is immediate

that membership is decidable for all levels $\text{FL}^n(\mathcal{C})$ and $\text{PL}^n(\mathcal{C})$ as soon as this problem is decidable for $\text{BPol}(\mathcal{C})$. Since $\mathcal{C} \in \{\mathcal{G}, \mathcal{G}^+\}$, it follows from results of [22] that $\text{BPol}(\mathcal{C})$ -membership boils down to \mathcal{G} -separation (this is based on independent techniques). Hence, we obtain the following corollary.

Corollary 1. *Consider a group prevariety \mathcal{G} with decidable separation and let $\mathcal{C} \in \{\mathcal{G}, \mathcal{G}^+\}$. For every $n \geq 1$, membership is decidable for $\text{FL}^n(\mathcal{C})$ and $\text{PL}^n(\mathcal{C})$.*

Remark 7. We do not mention the levels $\text{BL}^n(\mathcal{C})$ yet as this requires more work. This is the topic of Section 5: we prove that Corollary 1 also holds for them.

Proof (of Theorem 3). By definition, the third assertion is an immediate consequence of the others. Hence, we focus on the first two. We use induction on n . By symmetry, we only treat the case when n is odd and show that $\text{FL}^n(\mathcal{C}) = \text{RPol}_n(\text{BPol}(\mathcal{C}))$. Proposition 5 yields the right to left inclusion. We fix $L \in \text{FL}^n(\mathcal{C})$ and prove that $L \in \text{RPol}_n(\text{BPol}(\mathcal{C}))$. Let $\mathcal{D} = \text{LPol}_{n-1}(\text{BPol}(\mathcal{C}))$: we prove that $L \in \text{RPol}(\mathcal{D})$. Let $\alpha_L : A^* \rightarrow M_L$ be the syntactic morphism of L . By Proposition 1 and Theorem 2, it suffices to prove that for every \mathcal{D} -pair $(s, t) \in M_L^2$, we have $s^{\omega+1} = ts^\omega$. Since $L \in \text{FL}^n(\mathcal{C})$, Lemma 3 yields a \mathcal{C} -morphism $\eta : A^* \rightarrow N$ and $k \in \mathbb{N}$ such that L is a union of $\blacktriangleright_{n,\eta,k}$ -classes. Note that since $\mathcal{C} \in \{\mathcal{G}, \mathcal{G}^+\}$, we know that $\eta(A^+)$ is a group by Lemma 2. Let $p = \omega(N)$. We claim that since (s, t) is \mathcal{D} -pair, there exist $u, v \in A^*$ such that $\alpha_L(u) = t$, $\alpha_L(v) = s$ and $v^{2kp+1} \blacktriangleright_{n,\eta,k} uv^{2kp}$. Let us first explain why this completes the proof. Since $\blacktriangleright_{n,\eta,k}$ is a congruence by Lemma 4, we get $xv^{2kp+1}y \blacktriangleright_{n,\eta,k} xuv^{2kp}y$ for every $x, y \in A^*$. Since L is a union of $\blacktriangleright_{n,\eta,k}$ -classes, this yields $xv^{2kp+1}y \in L \Leftrightarrow xuv^{2kp}y \in L$ for all $x, y \in A^*$, i.e., $v^{2kp+1} \equiv_L uv^{2kp}$. Hence, $\alpha_L(v^{2kp+1}) = \alpha_L(uv^{2kp})$, i.e., $s^{2kp+1} = ts^{2kp}$. We now multiply by enough copies of s on the right to get $s^{\omega+1} = ts^\omega$, as desired.

We now build $u, v \in A^*$. There are two cases depending on n . If $n = 1$, then $\mathcal{D} = \text{BPol}(\mathcal{C})$. We use the equivalence $\sim_{\eta,k}$ defined in Section 3. Observe that the $\sim_{\eta,k}$ -classes belong to $\text{BPol}(\mathcal{C})$. There are two subcases depending on η .

- If $\eta^{-1}(1_N) \cap A^+ \neq \emptyset$, then Condition 3 in the definition of $\sim_{\eta,k}$ is trivial. Hence, one may verify that the $\sim_{\eta,k}$ -classes are Boolean combinations of languages of the form $\eta^{-1}(g)$ and $\eta^{-1}(g_1)a\eta^{-1}(g_2)$ for $a \in A$ and $g, g_1, g_2 \in N$, i.e., languages in $\text{BPol}(\mathcal{C})$, since η is a \mathcal{C} -morphism.
- Otherwise, $\eta^{-1}(1_N) = \{\varepsilon\}$ and one may verify that the $\sim_{\eta,k}$ -classes are Boolean combinations of languages $w\eta^{-1}(g)$, $\eta^{-1}(g)w$ and $\eta^{-1}(g_1)w\eta^{-1}(g_2)$ for $w \in A^*$ and $g, g_1, g_2 \in N$. These are languages in $\text{BPol}(\mathcal{C})$ since η is a \mathcal{C} -morphism (which implies that $\{\varepsilon\} \in \mathcal{C}$, since it is recognized by η).

Since (s, t) is a $\text{BPol}(\mathcal{C})$ -pair, it follows that there exist $u, v \in A^*$ such that $\alpha_L(u) = t$, $\alpha_L(v) = s$ and $u \sim_{\eta,k} v$ (otherwise, $\alpha_L^{-1}(s)$ can be separated from $\alpha_L^{-1}(t)$ by a union of $\sim_{\eta,k}$ -classes, i.e., by a language in $\text{BPol}(\mathcal{C})$). Thus, since $\alpha(A^+)$ is a group, Lemma 5 yields $v^{2kp+1} \blacktriangleright_{1,\eta,k} uv^{2kp}$, completing the case $n = 1$.

We now assume that $n > 1$. Lemma 3 implies that the $\blacktriangleright_{n-1,\eta,k^2}$ -classes belong to $\text{FL}^{n-1}(\mathcal{C})$. Hence, since $n-1$ is even (n is odd by hypothesis) induction

yields that the $\blacktriangleright_{n-1,\eta,k^2}$ -classes belong to $\mathcal{D} = LPol_{n-1}(BPol(\mathcal{C}))$. Since (s, t) is a \mathcal{D} -pair, this yields $u, v \in A^*$ such that $\alpha(u) = t$, $\alpha(v) = s$ and $u \blacktriangleright_{n-1,\eta,k^2} v$ (otherwise, $\alpha^{-1}(s)$ can be separated from $\alpha^{-1}(t)$ by a union of $\blacktriangleright_{n-1,\eta,k^2}$ -classes, *i.e.*, by a language in \mathcal{D}). Since $n-1$ is even and $\alpha(A^+)$ is a group, it then follows from Proposition 3 that $uv^{2kp} \blacktriangleright_{n,\eta,k} v^{2kp+1}$, completing the proof. \square

5 Intermediary levels

We now consider the levels $BL^n(\mathcal{C})$ in future/past hierarchies. We prove that when $\mathcal{C} \in \{\mathcal{G}, \mathcal{G}^+\}$ where \mathcal{G} is a group prevariety with decidable separation, membership is decidable for all levels $BL^n(\mathcal{C})$. By Theorem 3, we know that $BL^n(\mathcal{C}) = LPol_n(BPol(\mathcal{C})) \vee RPol_n(BPol(\mathcal{C}))$ for every $n \geq 1$. A key ingredient in our approach is a result of [18] based on mixed polynomial closure ($MPol$).

Theorem 4 ([18]). *Let \mathcal{D} be a prevariety. For every number $n \geq 1$, we have $LPol_{n+1}(\mathcal{D}) \vee RPol_{n+1}(\mathcal{D}) = MPol(LPol_n(\mathcal{D}) \vee RPol_n(\mathcal{D}))$.*

Combining Theorem 4 with Theorem 3 yields $BL^{n+1}(\mathcal{C}) = MPol(BL^n(\mathcal{C}))$. Now, recall that by Theorem 1, $MPol$ preserves the decidability of membership (this is shown in [18]). Therefore, an immediate induction reduces membership for $BL^n(\mathcal{C})$ to membership for $BL^1(\mathcal{C})$. Thus, we concentrate on this case: we prove that for every group prevariety \mathcal{G} with decidable separation, if $\mathcal{C} \in \{\mathcal{G}, \mathcal{G}^+\}$, then membership is decidable for $BL^1(\mathcal{C}) = LPol(BPol(\mathcal{C})) \vee RPol(BPol(\mathcal{C}))$.

We present algebraic characterizations of $LPol(BPol(\mathcal{C})) \vee RPol(BPol(\mathcal{C}))$. There are two statements depending on whether $\mathcal{C} = \mathcal{G}$ or $\mathcal{C} = \mathcal{G}^+$. We use the \mathcal{G} -pair relation defined in Section 4 (this is how the statement depends on \mathcal{G} -separation). We first characterize the classes $LPol(BPol(\mathcal{G})) \vee RPol(BPol(\mathcal{G}))$.

Theorem 5. *Let \mathcal{G} be a group prevariety and let $\alpha : A^* \rightarrow M$ be a surjective morphism. Then, α is an $LPol(BPol(\mathcal{G})) \vee RPol(BPol(\mathcal{G}))$ -morphism if and only if it satisfies the following property:*

$$(sq(tq')^\omega)^\omega s((r't)^\omega rs)^\omega = (sq(tq')^\omega)^\omega t((r't)^\omega rs)^\omega \quad (1)$$

for all $q, q', r, r' \in M$ and all \mathcal{G} -pairs $(s, t) \in M^2$.

We complete Theorem 5 with a second statement, which applies to the classes $LPol(BPol(\mathcal{G}^+)) \vee RPol(BPol(\mathcal{G}^+))$.

Theorem 6. *Let \mathcal{G} be a group prevariety and let $\alpha : A^* \rightarrow M$ be a surjective morphism. Then, α is an $LPol(BPol(\mathcal{G}^+)) \vee RPol(BPol(\mathcal{G}^+))$ -morphism if and only if it satisfies the following property:*

$$(esfq(etfq')^\omega)^\omega esf((r'etf)^\omega resf)^\omega = (esfq(etfq')^\omega)^\omega etf((r'etf)^\omega resf)^\omega \quad (2)$$

for all $q, q', r, r' \in M$, all $e, f \in E(\alpha(A^+))$ and all \mathcal{G} -pairs $(s, t) \in M^2$.

Recall that the \mathcal{G} -pairs associated with a morphism can be computed provided that \mathcal{G} -separation is decidable (by definition, (s, t) is a \mathcal{G} -pair if and only

if $\alpha^{-1}(s)$ is not \mathcal{G} -separable from $\alpha^{-1}(t)$. Hence, Theorem 5 and Theorem 6 imply that if \mathcal{G} is a group prevariety *with decidable separation* and $\mathcal{C} \in \{\mathcal{G}, \mathcal{G}^+\}$, then membership is decidable for the class $LPol(BPol(\mathcal{C})) \vee RPol(BPol(\mathcal{C}))$. Using Theorem 1 and Theorem 4, one may then lift decidability to *all* levels $LPol_n(BPol(\mathcal{C})) \vee RPol_n(BPol(\mathcal{C}))$ for $n \geq 1$. Finally, Theorem 3 yields the following corollary.

Corollary 2. *Let \mathcal{G} be a group prevariety with decidable separation. For every $n \geq 1$, membership is decidable for $BL^n(\mathcal{G})$ and $BL^n(\mathcal{G}^+)$.*

Remark 8. Theorem 5 generalizes a known result in the special case when \mathcal{G} is the trivial class $ST = \{\emptyset, A^*\}$. In this case, it is known [1,9] that a surjective morphism $\alpha : A^* \rightarrow M$ is an $LPol(BPol(ST)) \vee RPol(BPol(ST))$ -morphism if and only if M satisfies the equation $(sq)^\omega s(rs)^\omega = (sq)^\omega (rs)^\omega$ for all $q, r, s \in M$. This equation is equivalent to (1) in this case. Indeed, since ST is trivial, *every* pair in M^2 is an ST -pair. In particular, if $q, r, s \in M$, then $(s, 1_M)$ is an ST -pair. Hence, the above equation is the special case of (1) when $t = q' = r' = 1_M$. Conversely, if the above equation holds, then given elements $s, t, q, q', r, r' \in M$, we may apply the equation twice to get $(sq(tq')^\omega)^\omega s((r't)^\omega rs)^\omega = (sq(tq')^\omega)^\omega ((r't)^\omega rs)^\omega$ and $(tq')^\omega t(r't)^\omega = (tq')^\omega (r't)^\omega$. When combined, the two imply that (1) holds.

The proofs of Theorem 5 and Theorem 6 are presented in the full version of the paper. The two arguments are similar (though the one of Theorem 6 is technically more involved). Let us point out that these proofs are nontrivial. As for most algebraic characterizations of this kind, the challenging direction consists in proving that if some morphism $\alpha : A^* \rightarrow M$ satisfies (1) (resp. (2)), then it must be an $LPol(BPol(\mathcal{G})) \vee RPol(BPol(\mathcal{G}))$ -morphism (resp. an $LPol(BPol(\mathcal{G}^+)) \vee RPol(BPol(\mathcal{G}^+))$ -morphism). In particular, this part of the proof relies heavily on properties of the operators $LPol$, $RPol$ and $UPol$ established in [18] and [19,23].

6 Conclusion

For all group prevarieties \mathcal{G} , we characterized the future/past hierarchies within the variants $TL(\mathcal{G})$ and $TL(\mathcal{G}^+)$ of unary temporal logic with the deterministic hierarchies of bases $BPol(\mathcal{G})$ and $BPol(\mathcal{G}^+)$. We used these results to prove that if \mathcal{G} -separation is decidable, then membership is also decidable for *all* levels $FL^n(\mathcal{G})$, $PL^n(\mathcal{G})$, $BL^n(\mathcal{G})$, $FL^n(\mathcal{G}^+)$, $PL^n(\mathcal{G}^+)$ and $BL^n(\mathcal{G}^+)$ in such hierarchies.

A natural question is whether decidability can be pushed to more general problems than membership, *e.g.*, separation. When \mathcal{G} is the trivial class $ST = \{\emptyset, A^*\}$, it is known that separation is decidable for all levels $FL^n(ST)$ and $PL^n(ST)$ (this is shown for their counterparts in deterministic hierarchies [18]). Moreover, it is also known that if \mathcal{G} is a group prevariety with decidable separation, then $BPol(\mathcal{G})$ - and $BPol(\mathcal{G}^+)$ -separation are also decidable [21]. In view of our characterizations, this suggests that similar results may hold for the whole future/past hierarchies of $TL(\mathcal{G})$ and $TL(\mathcal{G}^+)$.

References

1. Almeida, J., Azevedo, A.: The join of the pseudovarieties of \mathcal{R} -trivial and \mathcal{L} -trivial monoids. *Journal of Pure and Applied Algebra* **60**(2), 129–137 (1989)
2. Arfi, M.: Polynomial operations on rational languages. In: *Proceedings of the 4th Annual Symposium on Theoretical Aspects of Computer Science*. pp. 198–206. STACS’87, Springer-Verlag, Berlin, Heidelberg (1987)
3. Etessami, K., Vardi, M.Y., Wilke, T.: First-order logic with two variables and unary temporal logic. In: *Proceedings of the 12th Annual IEEE Symposium on Logic in Computer Science*. pp. 228–235. LICS’97 (1997)
4. Etessami, K., Vardi, M.Y., Wilke, T.: First-order logic with two variables and unary temporal logic. *Information and Computation* **179**(2), 279–295 (2002)
5. Fleischer, L., Kufleitner, M., Lauser, A.: The half-levels of the FO^2 alternation hierarchy. *Theory of Computing Systems* **61**(2), 352–370 (2017)
6. Kamp, H.W.: *Tense Logic and the Theory of Linear Order*. Phd thesis, Computer Science Department, University of California at Los Angeles, USA (1968)
7. Krebs, A., Lodaya, K., Pandya, P.K., Straubing, H.: Two-variable logics with some betweenness relations: Expressiveness, satisfiability and membership. *Logical Methods in Computer Science* **Volume 16, Issue 3** (2020)
8. Kufleitner, M., Lauser, A.: The join levels of the Trotter-Weil hierarchy are decidable. In: *Proceedings of the 37th International Symposium on Mathematical Foundations of Computer Science*. MFCS’12, vol. 7464, pp. 603–614 (2012)
9. Kufleitner, M., Lauser, A.: The join of \mathcal{R} -trivial and \mathcal{L} -trivial monoids via combinatorics on words. *Discrete Mathematics & Theoretical Computer Science* **14**(1), 141–146 (2012)
10. Kufleitner, M., Weil, P.: On logical hierarchies within FO^2 -definable languages. *Logical Methods in Computer Science* **8**(3:11), 1–30 (2012)
11. Kufleitner, M., Weil, P.: The FO^2 alternation hierarchy is decidable. In: *Proceedings of the 21st International Conference on Computer Science Logic*. pp. 426–439. CSL’12 (2012)
12. Lodaya, K., Pandya, P.K., Shah, S.S.: Marking the chops: an unambiguous temporal logic. In: Ausiello, G., Karhumäki, J., Mauri, G., Ong, C.L. (eds.) *Fifth IFIP International Conference On Theoretical Computer Science - TCS 2008, IFIP 20th World Computer Congress, TC 1, Foundations of Computer Science, September 7-10, 2008, Milano, Italy*. IFIP, vol. 273, pp. 461–476. Springer (2008)
13. Pin, J.E.: Propriétés syntactiques du produit non ambigu. In: *Proceedings of the 7th International Colloquium on Automata, Languages and Programming*. pp. 483–499. ICALP’80 (1980)
14. 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)
15. Pin, J.E., Straubing, H., Thérien, D.: Locally trivial categories and unambiguous concatenation. *Journal of Pure and Applied Algebra* **52**(3), 297 – 311 (1988)
16. Pin, J.E., Weil, P.: Polynomial closure and unambiguous product. *Theory of Computing Systems* **30**(4), 383–422 (1997)
17. Place, T.: Deciding classes of regular languages: The covering approach. In: *Language and Automata Theory and Applications - 14th International Conference, LATA 2020, Milan, Italy, March 4-6, 2020, Proceedings*. *Lecture Notes in Computer Science*, vol. 12038, pp. 89–112. Springer (2020)

18. Place, T.: The amazing mixed polynomial closure and its applications to two-variable first-order logic. In: Proceedings of the 37th Annual ACM/IEEE Symposium on Logic in Computer Science. LICS'22 (2022)
19. Place, T., Zeitoun, M.: Separating without any ambiguity. In: Proceedings of the 45th International Colloquium on Automata, Languages, and Programming. pp. 137:1–137:14. ICALP'18 (2018)
20. Place, T., Zeitoun, M.: Generic results for concatenation hierarchies. *Theory of Computing Systems (ToCS)* **63**(4), 849–901 (2019), selected papers from CSR'17
21. 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)
22. Place, T., Zeitoun, M.: Characterizing level one in group-based concatenation hierarchies. In: Proceeding of the 17th International Computer Science Symposium in Russia. CSR'22 (2022)
23. Place, T., Zeitoun, M.: Unambiguous polynomial closure explained (2022). <https://doi.org/10.48550/ARXIV.2205.12703>, <https://arxiv.org/abs/2205.12703>
24. Schützenberger, M.P.: Sur le produit de concaténation non ambigu. *Semigroup Forum* **13**, 47–75 (1976)
25. 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)
26. Tesson, P., Therien, D.: Diamonds are forever: The variety DA. In: *Semigroups, Algorithms, Automata and Languages*, Coimbra (Portugal) 2001. pp. 475–500. World Scientific (2002)
27. Thérien, D., Wilke, T.: Over words, two variables are as powerful as one quantifier alternation. In: Proceedings of the 30th Annual ACM Symposium on Theory of Computing. pp. 234–240. STOC'98, ACM, New York, NY, USA (1998)
28. Thomas, W.: Classifying regular events in symbolic logic. *Journal of Computer and System Sciences* **25**(3), 360–376 (1982)