On First-Order Fragments for Words and Mazurkiewicz Traces

A Survey

Volker Diekert[1] and Manfred Kufleitner[2]

[1] Universität Stuttgart, FMI, Germany
`diekert@fmi.uni-stuttgart.de`
[2] Université Bordeaux 1, LaBRI, France
`manfred.kufleitner@labri.fr`

Abstract. We summarize several characterizations, inclusions, and separations on fragments of first-order logic over words and Mazurkiewicz traces. The results concerning Mazurkiewicz traces can be seen as generalizations of those for words. It turns out that over traces it is crucial, how easy concurrency can be expressed. Since there is no concurrency in words, this distinction does not occur there. In general, the possibility of expressing concurrency also increases the complexity of the satisfiability problem.
In the last section we prove an algebraic and a language theoretic characterization of the fragment $\Sigma_2[E]$ over traces. Over words the relation E is simply the order of the positions. The algebraic characterization yields decidability of the membership problem for this fragment. For words this result is well-known, but although our proof works in a more general setting it is quite simple and direct. An essential step in the proof consists of showing that every homomorphism from a free monoid to a finite aperiodic monoid M admits a factorization forest of finite height. We include a simple proof that the height is bounded by $3\,|M|$.

1 Introduction

The concept of partially commutative free monoids has first been considered by Cartier and Foata [1]. Later Keller and Mazurkiewicz used them as a model for concurrent systems and Mazurkiewicz established the notion of *trace* monoids for these structures [16, 19, 20]. Since then the elements of partially commutative monoids are called *Mazurkiewicz traces*. Many aspects of traces and trace languages have been researched, see *The Book of Traces* [7] for an overview.

Over words it has turned out that finite monoids are a powerful technique to refine the class of recognizable languages [9]. For fragments of first-order logic, in many cases it is a characterization in terms of algebra which leads to decidability of the membership problem. For example, on the algebraic side first-order logics as well as temporal logics corresponds to aperiodic monoids, see e.g. [12]. The probably most interesting fragment of them is given by the variety **DA**. It admits many different characterizations, which led to the title *Diamonds are Forever* in

[30]. One of the purposes of this paper is to survey the situation over words and Mazurkiewicz traces.

Words can be seen as a special case of Mazurkiewicz traces and the corresponding results for words have been known before their generalizations to traces. Since over words we do not have any concurrency the situation is more complex for traces, and therefore not all word results remain valid for traces. It turns out that for traces the distinction between so-called dependence graphs and partial orders is rather crucial. Over words, both notions coincide.

The paper is organized as follows. In Section 2 we introduce Mazurkiewicz traces using a graph theoretic approach since this directly translates into the logic setting. After that we present further notions used in this paper which include the definition of fragments of first-order logic and temporal logic, some language operations, and the connections to finite monoids. In Section 3 we give several characterizations of languages whose syntactic monoid is aperiodic or in the variety **DA**. In a second part of this section we describe the alternation hierarchy of first-order logic using language operations. Section 4 contains some ideas and approaches revealing how concurrency increases the expressive power of logical fragments and in Section 5 we present some results showing that in general, concurrency also increases the complexity of the satisfiability problem.

Finally, in Section 6 we give a self-contained proof of a language theoretic and an algebraic characterization of the fragment Σ_2 over traces. The algebraic characterization yields decidability of the membership problem for this fragment. For words this result is well-known, but although our proof works in a more general setting it is quite simple and direct. A main tool in this proof are factorization forests. We give a simple and essentially self-contained proof for Simon's theorem on factorization forests in the special case of finite aperiodic monoids M. Our proof can be generalized to arbitrary monoids and still yields that the height of the factorization forests is bounded by $3\,|M|$. The previously published bound was $7\,|M|$, see [2]. After having completed our paper we learned that the bound $3\,|M|$ has been stated in the Technical Report [3], too.

2 Preliminaries

Words and Mazurkiewicz traces

A *dependence alphabet* is a pair (Γ, D) where the alphabet Γ is a finite set (of actions) and the *dependence relation* $D \subseteq \Gamma \times \Gamma$ is reflexive and symmetric. The *independence relation* I is the complement of D. A *Mazurkiewicz trace* is an isomorphism class of a node-labeled directed acyclic graph $t = [V, E, \lambda]$, where V is a finite set of vertices labeled by $\lambda : V \to \Gamma$ and $E \subseteq (V \times V) \setminus \mathrm{id}_V$ is the edge relation such that for any two different vertices $x, y \in V$ we have either $(x, y) \in E$ or $(y, x) \in E$.

We call $[V, E, \lambda]$ a *dependence graph*. By $<$ we mean the transitive closure of E. We write $x \parallel y$ if $x \neq y$ and the vertices x and y are incomparable with respect to $<$. In this case we say that x and y are *independent* or *concurrent*.

Node labeled graphs (V, E, λ) and (V', E', λ') are isomorphic if and only if the corresponding labeled *partial orders* $(V, <, \lambda)$ and $(V', <', \lambda')$ are isomorphic. The transitive reduction of a trace is called the *Hasse diagram*.

For $D = \Gamma \times \Gamma$ we obtain *words*. The vertices in words are linearly ordered and the relations E and $<$ are identical. Let $t_1 = [V_1, E_1, \lambda_1]$ and $t_2 = [V_2, E_2, \lambda_2]$ be traces. Then we define the concatenation of t_1 and t_2 to be $t_1 \cdot t_2 = [V, \leq, \lambda]$ where $V = V_1 \cup V_2$ is a disjoint union, $\lambda = \lambda_1 \cup \lambda_2$, and $E = E_1 \cup E_2 \cup \{ (x, y) \in V_1 \times V_2 \mid (\lambda(x), \lambda(y)) \in D \}$. The set $\mathbb{M}$ of traces becomes a monoid with the empty trace $1 = (\emptyset, \emptyset, \emptyset)$ as unit. It is generated by Γ, where a letter a is viewed as a graph with a single vertex labeled by a. Thus, we obtain a canonical surjective homomorphism $\pi : \Gamma^* \to \mathbb{M}$. The effect of the mapping π can be made explicit as follows. We start with a word $w = a_1 \cdots a_n$ where all a_x are letters in Γ. Each x is viewed as an element in $\{ 1, \ldots, n \}$ with label $\lambda(x) = a_x$. We draw an arc from x to y if and only if both, $x < y$ and $(a_x, a_y) \in D$. This dependence graph is $\pi(w)$. Note that $\mathbb{M}$ is also canonically isomorphic to the quotient monoid $\Gamma^* / \{ ab = ba \mid (a, b) \in I \}$. By abuse of notation we often identify a trace t and its word representatives $w \in \pi^{-1}(t)$.

Example 1. Let $(\Gamma, D) = a — b — c — d$ where self-loops are omitted. Consider the trace $t = acdbca$. We have $acdbca = cabadc$ in $\mathbb{M}$. The trace t has the following graphical presentations:

Hasse diagram:

$$\begin{tikzcd} a \arrow[r] & b \arrow[r] \arrow[dr] & a \\ c \arrow[r] \arrow[ur] & d \arrow[r] & c \end{tikzcd}$$

Dependence graph E:

$$\begin{tikzcd} a \arrow[r] \arrow[rr, bend left] & b \arrow[r] \arrow[dr] & a \\ c \arrow[r] \arrow[ur] \arrow[rr, bend right] & d \arrow[r] & c \end{tikzcd}$$

Partial order $<$:

$$\begin{tikzcd} a \arrow[r] \arrow[rr, bend left] \arrow[drr] & b \arrow[r] \arrow[dr] & a \\ c \arrow[r] \arrow[ur] \arrow[urr] \arrow[rr, bend right] & d \arrow[r] & c \end{tikzcd}$$

In t, the node labeled with d is concurrent to all nodes labeled with a or b. $\square$

There is a basic observation which holds for all $t \in \mathbb{M}$ and all vertices x, y of t:

$$(x, y) \in E \;\Leftrightarrow\; (x, y) \in E^+ \wedge (\lambda(x), \lambda(y)) \in D \tag{1}$$

$$(x, y) \in E^+ \;\Leftrightarrow\; \exists x_1 \cdots \exists x_{|\Gamma|} \colon \left\{ \begin{array}{c} x_{|\Gamma|} = y \,\wedge\, (x, x_1) \in E \,\wedge \\ \displaystyle\bigwedge_{1 \leq i < |\Gamma|} (x_i, x_{i+1}) \in E \cup \mathrm{id}_V \end{array} \right\} \tag{2}$$

This shows that traces can be either represented by their dependence graphs or as a partial order without losing any information. There are some standard notations we adopt here. By $\mathrm{alph}(t)$ we denote the *alphabet* of a trace t, i.e., the set of letters occurring as labels of some position. By $|t|$ we denote the *length* of a trace, i.e., the number of vertices of t. A *trace language* L is a subset of $\mathbb{M}$.

First-order logic and temporal logic

The syntax of first-order logic formulas $\mathrm{FO}[E]$ is built upon atomic formulas of type

$$\top, \;\; \lambda(x) = a, \;\; \text{and} \;\; (x, y) \in E,$$

where $\top$ means *true*, x, y are variables and $a \in \Gamma$ is a letter. If φ, ψ are first-order formulas, then $\neg\varphi$, $\varphi \vee \psi$, $\exists x\, \varphi$ are first-order formulas, too. We use the usual shortcuts as $\bot = \neg\top$ meaning *false*, $\varphi \wedge \psi = \neg(\neg\varphi \vee \neg\psi)$, and $\forall x\, \varphi = \neg\exists x\, \neg\varphi$. Note that $x = y$ can be expressed by

$$\bigvee_{a \in \Gamma} \big(\lambda(x) = a \ \wedge\ \lambda(y) = a\big) \ \wedge\ (x, y) \notin E \ \wedge\ (y, x) \notin E$$

We let $\mathrm{FO}^m[E]$ be the set of all formulas with at most m different names for variables. There are completely analogous definitions for the first-order logic $\mathrm{FO}[<]$. The only difference is that instead of $(x, y) \in E$ we have an atomic predicate $x < y$.

Given $\varphi \in \mathrm{FO}[E] \cup \mathrm{FO}[<]$ the semantics is defined as usual [32]. In particular, if all free variables in φ belong to a set $\{ x_1, \ldots, x_m \}$, then for all $t \in \mathbb{M}$ and all $x_1, \ldots, x_m \in t$ we write $t, x_1, \ldots, x_m \models \varphi$ if t satisfies $\varphi(x_1, \ldots, x_m)$. We identify formulas by semantic equivalence (over finite traces). Hence, if φ and ψ are formulas with m free variables, then we write $\varphi = \psi$ as soon as $t, x_1, \ldots, x_m \models (\varphi \leftrightarrow \psi)$ for all $t \in \mathbb{M}$ and all $x_1, \ldots, x_m \in t$. Due to (1) we have that $\mathrm{FO}^m[E]$ is a fragment of $\mathrm{FO}^m[<]$. A *first-order sentence* is a formula in $\mathrm{FO}[E]$ or $\mathrm{FO}[<]$ without free variables. For a first-order sentence φ we define $L(\varphi) = \{ t \in \mathbb{M} \mid t \models \varphi \}$. A trace language $L \subseteq \mathbb{M}$ is called *first-order definable* if $L = L(\varphi)$ for some first-order sentence φ and we let $\mathrm{FO}(\mathbb{M}) = \{ L(\varphi) \mid \varphi \in \mathrm{FO}[E] \}$. We do not write $\mathrm{FO}[E](\mathbb{M})$, because $\mathrm{FO}(\mathbb{M}) = \{ L(\varphi) \mid \varphi \in \mathrm{FO}[<] \}$ as well, due to (2). So, in first-order it is not necessary to distinguish between E and $<$. However, for subclasses of FO we need this distinction. We define the following classes for $E' = E$ and $E' = <$, respectively.

The fragment $\Sigma_n[E']$ contains all formulas in prenex normal form with n blocks of alternating quantifiers starting with a block of existential quantifiers whereas in $\Pi_n[E']$ formulas start with a block of universal quantifiers. According to our convention to identify equivalent formulas, it makes sense to write e.g. $\varphi \in \Sigma_n[E'] \Leftrightarrow \neg\varphi \in \Pi_n[E']$. Although in general the transitive closure of binary relations is not expressible in first-order logic, we have $\bigcup_{0 \leq n} \Sigma_n[E] = \mathrm{FO}[<]$ due to the following observation obtained from (1) and (2):

$$\Sigma_n[E] \subseteq \Sigma_n[<] \subseteq \Sigma_{n+1}[E]$$

For $E' = E$ and $E' = <$ we define the following language classes:

- $\mathrm{FO}^m[E'](\mathbb{M}) = \{ L(\varphi) \mid \varphi \in \mathrm{FO}^m[E'] \}$.
- $\Sigma_n[E'](\mathbb{M}) = \{ L(\varphi) \mid \varphi \in \Sigma_n[E'] \}$.
- $\Pi_n[E'](\mathbb{M}) = \{ L(\varphi) \mid \varphi \in \Pi_n[E'] \}$.
- $\Delta_n[E'](\mathbb{M}) = \Sigma_n[E'](\mathbb{M}) \cap \Pi_n[E'](\mathbb{M})$.

Now, $\mathrm{FO}^m[E'](\mathbb{M})$ and $\Delta_n[E'](\mathbb{M})$ are Boolean algebras and $\Sigma_n[E'](\mathbb{M})$ and $\Pi_n[E'](\mathbb{M})$ are closed under union and intersection.

Local temporal logic formulas are defined by first-order formulas having at most one free variable. In this paper we focus on unary operators and local semantics. In temporal logic we write $a(x)$ for the atomic formula $\lambda(x) =$

a. Inductively, we define $\mathsf{SF}\varphi(x)$ (S*trict* F*uture*), $\mathsf{SP}\varphi(x)$ (S*trict* P*ast*), $\mathsf{M}\varphi(x)$ (*so*M*ewhere*), $\mathsf{Eco}\,\varphi(x)$ (E*xists* co*ncurrently*) as follows.

$$\begin{aligned}
\mathsf{SF}\varphi(x) &= \exists y : x < y \wedge \varphi(y)\\
\mathsf{SP}\varphi(x) &= \exists y : y < x \wedge \varphi(y)\\
\mathsf{M}\varphi(x) &= \exists y : \varphi(y)\\
\mathsf{Eco}\,\varphi(x) &= \exists y : x \parallel y \wedge \varphi(y)
\end{aligned}$$

It is common to write φ instead of $\varphi(x)$. Let $\mathcal{C}$ be a subset of temporal operators from the set above, then $\mathrm{TL}[\mathcal{C}]$ means the formulas where all operators are from $\mathcal{C}$. In order to pass to languages we would like to define $L(\varphi) \subseteq \mathbb{M}$, even if φ has a free variable. There is however no canonical choice, so we use an existential variant; and we define here:

$$L_{\exists}(\varphi) = \{\, t \in \mathbb{M} \mid \exists x \in t\colon t, x \models \varphi \,\} = L(\mathsf{M}\varphi).$$

Define $\mathrm{TL}[\mathcal{C}](\mathbb{M})$ as the Boolean closure of languages defined by $L_{\exists}(\varphi)$ with $\varphi \in \mathrm{TL}[\mathcal{C}]$.

Languages and language operations

We now define some operations on classes of languages that are used to describe the expressive power of logical fragments. Let $\mathcal{V}$ be a class of trace languages. By $\mathbb{B}(\mathcal{V})$ we denote the Boolean closure of $\mathcal{V}$. A language L is a *monomial* over $\mathcal{V}$ of *degree* m if there exist $n \leq m$, $a_i \in \Gamma$ and $L_i \in \mathcal{V}$ with

$$L = L_0 a_1 L_1 \cdots a_n L_n$$

Note that the degree of a monomial is not unique. A finite union of monomials over $\mathcal{V}$ is called a *polynomial* over $\mathcal{V}$. A polynomial has *degree* m if it can be written as a union of monomials of degree m. The class of all polynomials over $\mathcal{V}$ is denoted by $\mathrm{Pol}(\mathcal{V})$. The class $\mathrm{Pol}(\mathcal{V})$ is often called the *polynomial closure* or the closure under product and union of the class $\mathcal{V}$. By co-$\mathrm{Pol}(\mathcal{V})$ we denote the class of languages L such that $\mathbb{M} \setminus L \in \mathrm{Pol}(\mathcal{V})$. If we speak of monomials and polynomials without referring to some class $\mathcal{V}$ then we mean monomials and polynomials over $\mathcal{A} = \{\, A^* \mid A \subseteq \Gamma \,\}$, respectively. In particular, $\mathrm{Pol} = \mathrm{Pol}(\mathcal{A})$ and co-Pol $=$ co-$\mathrm{Pol}(\mathcal{A})$. For example, if $A, B \subseteq \Gamma$ then $A^*B^* \in \mathrm{Pol}$ since

$$A^*B^* = A^* \cup \bigcup_{b \in B} A^* b B^*$$

The class of *star-free* languages SF is the closure of the empty set under Boolean operations and polynomials. If $\mathcal{V}$ is a class of word languages then $\mathrm{UPol}(\mathcal{V})$ consists of the word languages that are disjoint finite unions of unambiguous monomials. A monomial $L_0 a_1 L_1 \cdots a_n L_n$ is *unambiguous* if every $w \in L_0 a_1 L_1 \cdots a_n L_n$

has a unique factorization $w = w_0 a_0 w_1 \cdots a_n w_n$ with $w_i \in L_i$. A similar language operation is $\mathbb{B}$-UPol. By $\mathbb{B}\text{-UPol}(\mathcal{V})$ we denote the closure of $\mathcal{V}$ under Boolean operations and unambiguous products. An *unambiguous product* is an unambiguous monomial of the form $L_0 a_1 L_1$. We set $\mathrm{UPol} = \mathrm{UPol}(\mathcal{A})$ and $\mathbb{B}\text{-UPol} = \mathbb{B}\text{-UPol}(\mathcal{A})$. For example, the word language $\{a,b\}^* \, ab \, \{a,b\}^*$ is in UPol since

$$\{a,b\}^* \, ab \, \{a,b\}^* = \{b\}^* \, a \, \{a\}^* \, b \, \{a,b\}^*$$

whereas the polynomials $\{a,b\}^* \, aa \, \{a,b\}^*$ and $\{a,b,c\}^* \, ab \, \{a,b,c\}^*$ are not in UPol. See [23] for more information on the language operations $\mathrm{UPol}(\mathcal{V})$ and $\mathbb{B}\text{-UPol}(\mathcal{V})$. The operation $\mathbb{B}\text{-UPol}(\mathcal{V})$ has been extended to classes of trace languages [18].

Algebraic descriptions

Finite monoids are an elementary tool in the description and classification of recognizable languages. Remember that a *monoid* M is a set equipped with an associative binary operation and a neutral element 1. An *ordered monoid* is a monoid M equipped with a partial order relation $\leq$ such that $a \leq b$ implies $ca \leq cb$ and $ac \leq bc$ for all $a, b, c \in M$. Every monoid M forms an ordered monoid $(M, =)$. For homomorphisms $h : (M, \leq) \to (N, \preceq)$ between ordered monoids we additionally require that $a \leq b$ implies $h(a) \preceq h(b)$ for all $a, b \in M$. If a is an element of an ordered monoid $(M, \leq)$ then we define $\lfloor a \rfloor = \{ b \in M \mid b \leq a \}$. More details on ordered monoids can be found in [22]. An element e of a monoid is called *idempotent* if $e^2 = e$. For every finite monoid M there exists a number $\omega \in \mathbb{N}$ such that a^ω is idempotent for every $a \in M$. The element a^ω is the unique idempotent generated by a. Therefore we use the ω-notation also if the finite monoid M is not fixed to denote the idempotent generated by some element. A language L is called *recognizable* if $L = h^{-1}h(L)$ for some homomorphism $h : \mathbb{M} \to M$, where M is a finite monoid. In this case we say that M recognizes L. The minimal monoid recognizing L is its syntactic monoid. For a language $L \subseteq \mathbb{M}$ we define its *syntactic pre-order* $\leq_L$ by

$$s \leq_L t \;\Leftrightarrow\; (\forall p, q \in \mathbb{M} \colon ptq \in L \;\Rightarrow\; psq \in L)$$

and its *syntactic congruence* $\sim_L$ by $s \sim_L t$ if and only if $s \leq_L t$ and $t \leq_L s$. The natural homomorphism $\mu_L : \mathbb{M} \to \mathbb{M}/{\sim_L} : t \mapsto [t]_{\sim_L}$ is called the *syntactic homomorphism* of L and the monoid $M(L) = \mathbb{M}/{\sim_L}$ is called the *syntactic monoid* of L. A language L is recognizable if and only if $M(L)$ is finite. The *syntactic pre-order* $\leq_L$ of L induces a partial order on $M(L)$ such that $(M(L), \leq_L)$ forms an ordered monoid. It is called the *syntactic ordered monoid* of L. For $\mu_L : (\mathbb{M}, =) \to (M(L), \leq_L)$ we have

$$L = \bigcup_{a \in \mu_L(L)} \mu_L^{-1}(\lfloor a \rfloor)$$

A class of recognizable languages $\mathcal{V}$ is a *language variety* if it is closed under Boolean operations, left and right quotients, and inverse homomorphic images.

A class of finite monoids $\mathbf{V}$ is called a *variety* if it is closed under taking finite products, submonoids and homomorphic images [21]. Eilenberg has shown that language varieties of word languages and varieties of finite monoids are in a one-to-one correspondence [9]. Ordered monoids are designed to serve as a similar tool for classes of languages which are not closed under complementation. Syntactic (ordered) monoids play a crucial role in these correspondences. This yields to the observation that properties of classes of languages can be expressed in terms of properties of syntactic monoids. In a lot of cases, a description of the variety generated by the syntactic monoids $M(L)$ for $L \in \mathcal{V}$ yields decidability of the membership problem for this language variety $\mathcal{V}$. An important tool to describe the structure of monoids are *Green's relations*. For $a, b \in M$ we define

$$\begin{array}{rclcrcl} a \mathrel{\mathcal{J}} b & \Leftrightarrow & MaM = MbM & \qquad & a \leq_{\mathcal{J}} b & \Leftrightarrow & MaM \subseteq MbM \\ a \mathrel{\mathcal{R}} b & \Leftrightarrow & aM = bM & & a \leq_{\mathcal{R}} b & \Leftrightarrow & aM \subseteq bM \\ a \mathrel{\mathcal{L}} b & \Leftrightarrow & Ma = Mb & & a \leq_{\mathcal{L}} b & \Leftrightarrow & Ma \subseteq Mb \\ a \mathrel{\mathcal{H}} b & \Leftrightarrow & a \mathrel{\mathcal{R}} b \text{ and } a \mathrel{\mathcal{L}} b & & & & \end{array}$$

Note that $\mathcal{J}$, $\mathcal{R}$, $\mathcal{L}$, and $\mathcal{H}$ are equivalence relations, whereas $\leq_{\mathcal{J}}$, $\leq_{\mathcal{R}}$, and $\leq_{\mathcal{L}}$ are pre-orders. Equations are another tool to describe properties of finite monoids. Let Ω be a finite set and let $v, w \in \Omega^*$. A monoid M *satisfies* the equation $v = w$, if for all homomorphisms $h : \Omega^* \to M$ we have $h(v) = h(w)$. For example, commutative monoids satisfy $xy = yx$. We also allow the ω-operator in equations and define $h(v^\omega) = h(v)^\omega$. By $\llbracket\, v = w \,\rrbracket$ we denote the class of finite monoids satisfying $v = w$. The class of all monoids satisfying an equation forms a variety. We define the variety of *aperiodic monoids* $\mathbf{A}$ by $\mathbf{A} = \llbracket\, x^\omega = x^{\omega+1} \,\rrbracket$. Another important variety is $\mathbf{DA} = \llbracket\, (xy)^\omega x (xy)^\omega = (xy)^\omega \,\rrbracket$. By mapping y to 1 we see that $\mathbf{DA} \subseteq \mathbf{A}$. In the following we summarize some basic properties of these varieties.

Proposition 1 ([21]). *For every finite monoid M the following are equivalent:*

1. $M \in \mathbf{A}$.
2. *M is $\mathcal{H}$-trivial, i.e., every $\mathcal{H}$-class contains exactly one element.*
3. *All groups in M are trivial, i.e., if a subsemigroup of M is a group then it contains only one element.*

Proposition 2 ([17]). *For every finite monoid M the following are equivalent:*

1. $M \in \mathbf{DA}$.
2. $M \in \llbracket\, (xy)^\omega y (xy)^\omega = (xy)^\omega \,\rrbracket$.
3. $M \in \llbracket\, (xyz)^\omega y (xyz)^\omega = (xyz)^\omega \,\rrbracket$.
4. $M \in \mathbf{A}$ *and* $\forall a, b, e \in M \colon e = e^2$ *and* $a \mathrel{\mathcal{J}} b \mathrel{\mathcal{J}} e$ *implies* $ab \mathrel{\mathcal{J}} e$.
5. $\forall e, f \in M \colon e = e^2$ *and* $e \mathrel{\mathcal{J}} f$ *implies* $f = f^2$.

3 Expressivity results

In the following two theorems we summarize characterizations of trace languages whose syntactic monoid is aperiodic or in $\mathbf{DA}$. Note that this includes the special case of word languages. The results are using some temporal operators which we did not introduce yet. The operator X is an existential next-operator, i.e., $\mathsf{X}\,\varphi$ is true at a position x if at some minimal position in the future of x the formula φ holds. Over words, this position is unique. The until-operator U is a binary operator. The formula $\varphi \,\mathsf{U}\, \psi$ is true at a position x if there exists a position $y \geq x$ at which ψ holds and all positions between x and y (i.e., all positions from the current position x *"until"* y) satisfy φ. The formula $\mathsf{X}_a\,\varphi$ for $a \in \Gamma$ is true at a position x if there exists a position $y > x$ labeled by a and if at the first of these a-labeled positions in the future of x the formula φ holds. The operator Y_a is left-right symmetric to X_a. With $\mathrm{TL}[\mathsf{X}_a, \mathsf{Y}_a]$ we mean that we have X_a and Y_a operators for every $a \in \Gamma$. The definition of the languages generated by formulas in $\mathrm{TL}[\mathsf{X}, \mathsf{U}]$ and $\mathrm{TL}[\mathsf{X}_a, \mathsf{Y}_a]$ is slightly different from the one that we propose above for unary temporal logic.

Theorem 1 ([5, 8, 14, 15]). *Let $L \subseteq \mathbb{M}$. Then the following are equivalent:*

1. $M(L) \in \mathbf{A}$.
2. $L \in \mathrm{SF}$.
3. *L is expressible in* $\mathrm{FO}^3[<]$.
4. *L is expressible in* $\mathrm{FO}[<]$.
5. *L is expressible in* $\mathrm{FO}[E]$.
6. *L is expressible in* $\mathrm{TL}[\mathsf{X}, \mathsf{U}]$.

Theorem 2 ([6, 18]). *Let $L \subseteq \mathbb{M}$. Then the following are equivalent:*

1. $M(L) \in \mathbf{DA}$.
2. $L \in \mathrm{Pol} \cap \text{co-Pol}$.
3. $L \in \mathbb{B}\text{-UPol}$.
4. *L is expressible in* $\mathrm{FO}^2[E]$.
5. *L is expressible in* $\Delta_2[E]$.
6. *L is expressible in* $\mathrm{TL}[\mathsf{X}_a, \mathsf{Y}_a]$.
7. *L is expressible in* $\mathrm{TL}[\mathsf{SF}, \mathsf{SP}]$.
8. *L is expressible in* $\mathrm{TL}[\mathsf{SF}, \mathsf{SP}, \mathsf{M}]$.

For word languages $L \subseteq \Gamma^*$ we additionally have $M(L) \in \mathbf{DA}$ if and only if $L \in \mathrm{UPol}$, see [25]. In particular, UPol is closed under complementation. Since membership in both varieties $\mathbf{A}$ and $\mathbf{DA}$ is decidable, membership for all characterizations in Theorem 1 and Theorem 2 is decidable.

Theorem 3 ([6, 11]). *Let $L \subseteq \mathbb{M}$. Then the following are equivalent:*

1. *L is expressible in* $\mathrm{FO}^2[<]$.
2. *L is expressible in* $\mathrm{TL}[\mathsf{SF}, \mathsf{SP}, \mathsf{E}\infty]$.

The following theorem gives a language theoretic characterization of the alternation hierarchy for first-order logic over words. It is the connection to the *Straubing-Thérien hierarchy* in which one describes classes of word languages by alternating Boolean closure and polynomial closure starting with the empty set. By definition, the limit of this process is the class of star-free languages. In the following we use $\mathbb{B}\Sigma_n$ as a shortcut for $\mathbb{B}\big(\Sigma_n[<](\Gamma^*)\big)$. Note that $\mathbb{B}\Sigma_n = \mathbb{B}\Pi_n$.

Theorem 4 ([24]). *Over words we have the following*

1. $\Sigma_0[<](\Gamma^*) \;=\; \mathbb{B}(\Sigma_0) \;=\; \{\emptyset, \Gamma^*\}$.
2. $\Sigma_{n+1}[<](\Gamma^*) \;=\; \mathrm{Pol}(\mathbb{B}\Sigma_n)$.
3. $\Pi_{n+1}[<](\Gamma^*) \;=\; \text{co-Pol}(\mathbb{B}\Sigma_n)$.
4. $\Delta_{n+1}[<](\Gamma^*) \;=\; \mathrm{UPol}(\mathbb{B}\Sigma_n)$.

A basis for the last part of this theorem is the more general fact that $\mathrm{UPol}(\mathcal{V}) = \mathrm{Pol}(\mathcal{V}) \cap \text{co-Pol}(\mathcal{V})$ if $\mathcal{V}$ is a variety of word languages. This follows from an algebraic description in terms of Mal'cev products [24]. Another language theoretic characterization of Σ_2 is $\Sigma_2[<](\Gamma^*) = \mathrm{Pol}$. We give a detailed proof of this characterization in the more general setting of traces over dependence graphs in Section 6. It is well-known that the alternation hierarchy for first-order logic is strict [29], i.e.:

– For $n \geq 1$ the classes $\Sigma_n[<](\Gamma^*)$ and $\Pi_n[<](\Gamma^*)$ are incomparable.
– For $n \geq 1$ the class $\Sigma_n[<](\Gamma^*)$ is strictly contained in $\Delta_{n+1}[<](\Gamma^*)$.
– For $n \geq 1$ the class $\Delta_n[<](\Gamma^*)$ is strictly contained in the class $\Sigma_n[<](\Gamma^*)$.

Recently, Weis and Immerman have shown that the alternation hierarchy for FO^2 on words is strict [33]. In the next section we consider the alternation hierarchy for first-order logic over traces. The distinction between partial orders $<$ and dependence graphs E turns out to be crucial. Using (2) we can express $<$ in terms of E, but this requires variables and it requires quantifiers, but in FO^2 the number of variables is restricted whereas in Σ_n the number of quantifier alternations is bounded.

4 Separation results

We start this section with a simple observation. Let $(\Gamma, D) = a - b - c$ and consider the traces $x = abc$ and $y = b$. Then for all $n \in \mathbb{N}$ the trace $(xy)^n$ is a sequence in which all positions are totally ordered whereas in the trace $(xy)^n x (xy)^n$ we have a factor xx whose Hasse diagram is

$$\begin{tikzcd}
 & & a \arrow[r] & b \arrow[r] & c \\
a \arrow[r] & b \arrow[r] \arrow[ur] & c \arrow[ur] & &
\end{tikzcd}$$

In particular, in xx there exist two concurrent actions. Consider the formula $\varphi = \exists z_1 \exists z_2 \colon z_1 \| z_2 \in \mathrm{FO}^2[<] \cap \Sigma_1[<]$ where $z_1 \| z_2$ is a macro for $\neg(z_1 = z_2 \;\vee\; z_1 < z_2 \;\vee\; z_2 < z_1)$. Then for all $n \geq 1$ we have

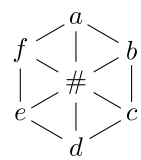
$$(xy)^n x (xy)^n \models \varphi \qquad \text{and} \qquad (xy)^n \not\models \varphi$$

This shows that the syntactic monoid of the trace language $L(\varphi)$ is not in $\mathbf{DA} = [\![\, (xy)^\omega x (xy)^\omega = (xy)^\omega \,]\!]$. Now, whenever the dependence relation is not transitive we find some letters a, b and c with the dependencies $a - b - c$. On the other hand, if the dependence relation is transitive then the partial order $<$ and the edge relation E of the dependence graph are identical. Together with $\Sigma_1[<](\mathbb{M}) \subseteq \Delta_2[<](\mathbb{M})$ we obtain the following theorem.

Theorem 5. *Let $\mathbb{M}$ be the trace monoid generated by the dependence alphabet (Γ, D). The following are equivalent:*

1. *The dependence relation D is transitive.*
2. *For every trace, the relations $<$ and E are identical.*
3. $\mathrm{FO}^2[E](\mathbb{M}) = \mathrm{FO}^2[<](\mathbb{M})$.
4. $\Delta_2[E](\mathbb{M}) = \Delta_2[<](\mathbb{M})$.

The main technique in the proofs of the following theorems are Ehrenfeucht-Fraïssé games, see e.g. [29, 32]. Let $\mathbb{M}$ be a trace monoid over the following dependence alphabet:

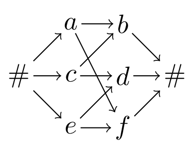
$(\Gamma, D) \ =$ 

Theorem 6 ([6]). *For the above trace monoid $\mathbb{M}$ the trace language*

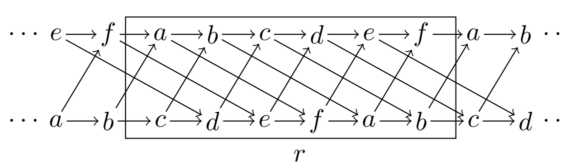
$$L = \{\, t \in \mathbb{M} \mid \exists x, y, z \in t\colon\ (x \parallel y \wedge y \parallel z \wedge z \parallel x) \,\}$$

consisting of all traces with three pairwise concurrent actions is expressible in $\Sigma_1[<]$ but not in $\mathrm{FO}^2[<]$.

The main idea in the proof of this theorem is to consider the traces $\#q\#$ and $p = \#r^n\#$ in which every action has the same set of concurrent actions, but in p there are at most two pairwise independent actions. The Hasse diagram of $\#q\#$ is:

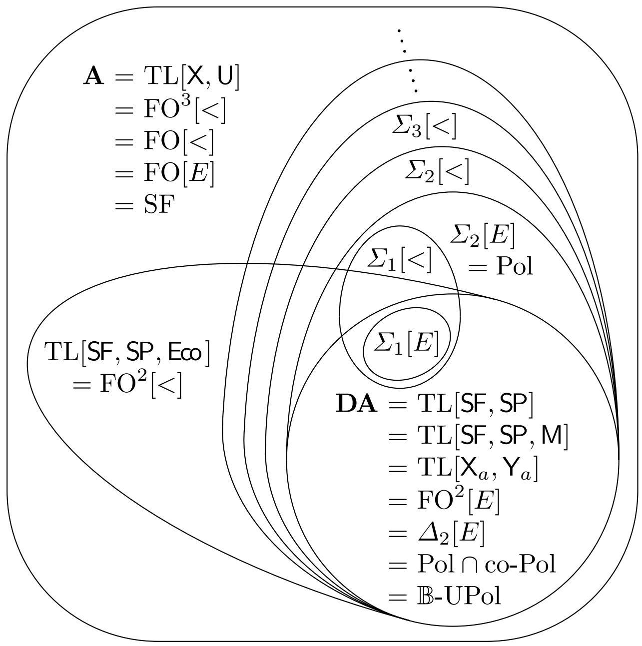


and the Hasse diagram of the trace $p = \#r^n\#$ is sketched below:



For every formula $\varphi \in \mathrm{FO}^2[<]$ we can find a sufficiently large number n such that the two traces $p^n q p^n \in L$ and $p^{2n} \notin L$ either both are models of φ or none of them is a model. Therefore, $L \notin \mathrm{FO}^2[<](\mathbb{M})$. The previous two results can be summarized as follows: "two concurrent actions" is in $\mathrm{FO}^2[<]$ and $\Delta_2[<]$ but not in $\mathrm{FO}^2[E] = \Delta_2[E]$ and "three concurrent actions" is in $\Delta_2[<]$ but not in $\mathrm{FO}^2[<]$. The next theorem implies that in general $\mathrm{FO}^2[<]$ and $\Delta_2[<]$ are incomparable. It is open whether membership is decidable for $\mathrm{FO}^2[<](\mathbb{M})$ or $\Delta_2[<](\mathbb{M})$. Also note that the following result is rather unexpected since $\mathrm{FO}^2[<] \subseteq \mathrm{FO}[<] = \bigcup_n \Sigma_n[<]$.

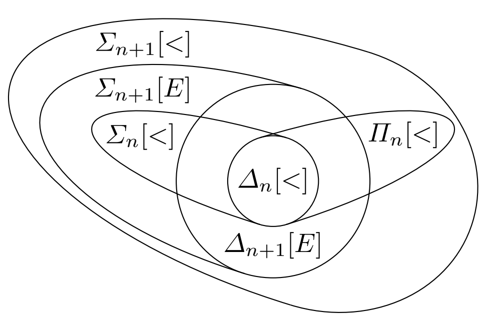
Theorem 7 **([6]).** *For every $n \geq 0$ there exists a trace monoid $\mathbb{M}$ and a trace language $L \subseteq \mathbb{M}$ such that $L \in \mathrm{FO}^2[<](\mathbb{M})$ but $L \notin \Sigma_n\,[<]\,(\mathbb{M})$.*



Remember $\Sigma_n[<] \subseteq \Sigma_{n+1}[E] \subseteq \Sigma_{n+1}[<]$. We already know from the word case that the inclusion $\Sigma_n[<] \subseteq \Sigma_{n+1}[E]$ is strict. The following theorem says that in general the second inclusion is also strict and that the fragments $\Pi_{n-1}[<]$ and $\Sigma_n[E]$ are incomparable.

Theorem 8 **([6]).** *Let $\mathbb{M}$ be the trace monoid generated by the dependence alphabet (Γ, D). The following are equivalent:*

1. *The dependence relation D is transitive.*
2. *$\exists n \geq 1 : \Sigma_n[E](\mathbb{M}) = \Sigma_n[<](\mathbb{M})$.*
3. *$\exists n \geq 2 : \Pi_{n-1}[<](\mathbb{M}) \subseteq \Sigma_n[E](\mathbb{M})$.*



5 Complexity of satisfiability

The possibility of being able to speak about concurrency increases the expressivity of most first-order fragments. In this section we will see how it also increases the complexity of the satisfiability problem. The (uniform) *satisfiability problem* (SAT) for some class of logical formulas $\mathcal{C}$ is the following:

Input: A dependence alphabet (Γ, D) and a formula $\varphi \in \mathcal{C}$.
Question: Does there exists $t \in \mathbb{M} = \mathbb{M}(\Gamma, D)$ such that $t \in L(\varphi)$?

and the *non-uniform satisfiability problem* for $\mathcal{C}$ over a dependence alphabet (Γ, D) is the satisfiability problem where the dependence alphabet (Γ, D) is fixed and not part of the input:

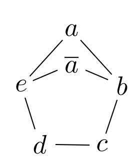
Input: A formula $\varphi \in \mathcal{C}$.
Question: Does there exists $t \in \mathbb{M}$ such that $t \in L(\varphi)$?

We summarize some complexity results in the following theorem.

Theorem 9 ([6, 10, 11, 13, 27, 28]).

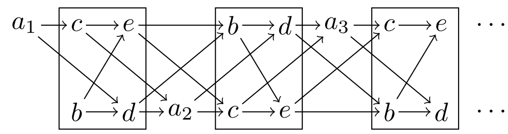
1. *SAT for temporal logics is* PSPACE*-complete.*
2. *SAT for* $\mathrm{FO}[<]$ *is not elementary.*
3. *The non-uniform satisfiability problem for* $\mathrm{TL}[\mathsf{X}, \mathsf{F}]$ *over* $\{a, \overline{a}\}^*$ *is* PSPACE*-hard.*
4. *SAT for* $\mathrm{TL}[\mathsf{SF}, \mathsf{SP}, \mathsf{M}]$ *is* NP*-complete.*
5. *The non-uniform satisfiability problem for* $\mathrm{TL}[\mathsf{SF}, \mathsf{SP}, \mathsf{Eco}]$ *over some dependence alphabet is* PSPACE*-hard. In fact, non-uniform satisfiability for the stutter-invariant fragment* $\mathrm{TL}[\mathsf{F}, \mathsf{Eco}]$ *is already* PSPACE*-hard.*
6. *SAT for* $\mathrm{FO}^2[E]$ *is in* NEXPTIME*.*
7. *The satisfiability problem for* $\mathrm{FO}^2[<]$ *is in* EXPSPACE *and* NEXPTIME*-hard. In fact, satisfiability for* $\mathrm{FO}^2[\|]$ *in which* $\|$ *is the only binary relation is already* NEXPTIME*-hard.*

The parts *"4."* and *"5."* in Theorem 9 show that allowing the Eco operator increases the complexity of the satisfiability problem (unless NP = PSPACE). Part *"4."* is proved by giving a *small model property* for $\mathrm{TL}[\mathsf{SF}, \mathsf{SP}, \mathsf{M}]$, i.e., if there exists a model then there also exists a model whose size is polynomially bounded. For part *"5."* a reduction of *"3."* is used. In the following we sketch the idea of how to simulate the X-operator using the Eco-operator over the following *independence* alphabet:

$(\Gamma, I) =$ 

For a word $w = a_1 \cdots a_n \in \{a, \overline{a}\}^+$ we define a trace $\widetilde{w} = a_1(bcde) \cdots a_n(bcde) \in \mathbb{M} = \mathbb{M}(\Gamma, I)$.

$\widetilde{w} =$



For the trace $\widetilde{w}$ we can use Eco to simulate X on the positions with label $a, \overline{a}$. The transformation of $\mathsf{X}\,\psi$ is given by

$$\widetilde{\mathsf{X}\,\psi} = \mathsf{Eco}\,(b \wedge \mathsf{Eco}\,(c \wedge \mathsf{Eco}\,(d \wedge \mathsf{Eco}\,(e \wedge \mathsf{Eco}\,((a \vee \overline{a}) \wedge \widetilde{\psi})))))$$

It is easy to verify that $\widetilde{\mathsf{X}\,\psi}$ indeed reaches the next a or $\overline{a}$ position.

6 The fragment $\boldsymbol{\Sigma_2[E]}$

In this section we give a self-contained proof of the following theorem. An important tool in the proof are factorization forests.

Theorem 10. *Let $L \subseteq \mathbb{M} = \mathbb{M}(\Gamma, I)$ be a recognizable trace language and let $\mu : \mathbb{M} \to (M(L), \leq) : t \mapsto [t]$ be the syntactic homomorphism onto its syntactic ordered monoid. The following are equivalent:*

1. *For all $e, s \in \mathbb{M}$: $[e] = [e^2]$ and $\mathrm{alph}(s) \subseteq \mathrm{alph}(e)$ implies $[ese] \leq [e]$.*
2. *L is a polynomial.*
3. *L is expressible in $\Sigma_2[E]$.*

The syntactic ordered monoid of a recognizable trace language (given in any reasonable presentation) is effectively computable. Since property "1." in Theorem 10 can be effectively verified we obtain the following corollary.

Corollary 1. *It is decidable if $L \subseteq \mathbb{M}$ is definable in $\Sigma_2[E]$.*

6.1 Factorization forests

Let M be a finite monoid. A *factorization forest* of a homomorphism $\varphi : \Gamma^* \to M$ is a function d which maps every word w with length $|w| \geq 2$ to a factorization $d(w) = (w_1, \ldots, w_n)$ of $w = w_1 \cdots w_n$ such that $n \geq 2$ and w_i is not empty for all $i \in \{1, \ldots, n\}$ and such that $n \geq 3$ implies that $\varphi(w_1) = \ldots = \varphi(w_n)$ is idempotent in M. The *height* h of a word w is defined as

$$h(w) = \begin{cases} 0 & \text{if } |w| \leq 1 \\ 1 + \max\{\, h(w_1), \ldots, h(w_n) \,\} & \text{if } d(w) = (w_1, \ldots, w_n) \end{cases}$$

We call the tree defined by the "branching" d for the word w the *factorization tree* of w. The height $h(w)$ is the height of this tree. The *height* of d is defined as $\sup\{\, h(w) \mid w \in \Gamma^* \,\}$. A famous theorem of Simon says that every homomorphism $\varphi : \Gamma^* \to M$ has a factorization forest of height $\leq 9\,|M|$, see [26]. By generalizing techniques of [2] we can improve this bound to $3\,|M|$. Using another approach, this bound has been shown independently in [3]. Below we present a simple proof of this fact in the special case of aperiodic monoids. The proof requires only basic facts from the theory of finite semigroups such as:

– The intersection of an $\mathcal{R}$-class and an $\mathcal{L}$-class within the same $\mathcal{J}$-class yields a unique $\mathcal{H}$-class within that $\mathcal{J}$-class.
– $x \leq_{\mathcal{L}} y$ and $x \mathrel{\mathcal{J}} y$ implies $x \mathrel{\mathcal{L}} y$; $x \leq_{\mathcal{R}} y$ and $x \mathrel{\mathcal{J}} y$ implies $x \mathrel{\mathcal{R}} y$.
– In aperiodic monoids every $\mathcal{H}$-class consists of only one element.

Theorem 11. *Let M be a finite aperiodic monoid. Every homomorphism $\varphi : \Gamma^* \to M : w \mapsto [w]$ has a factorization forest of height $< 3\,|M|$.*

Proof. We show that for every $w \in \Gamma^*$ there exists a factorization tree of height $h(w) < 3\,|\{\, x \in M \mid [w] \leq_{\mathcal{J}} x \,\}|$. The $\mathcal{J}$-class of 1 in aperiodic monoids is trivial. Let $w \in \Gamma^*$ with $|w| \geq 2$. If $[w] = 1$ then for all $b \in \mathrm{alph}(w)$ we have $[b] = 1$. Hence $d(w) = (b_1, \ldots, b_n)$ yields a factorization tree of height 1 for $w = b_1 \ldots b_n$. Now let $[w] <_{\mathcal{J}} 1$. Then w has a unique factorization

$$w = w_0 a_1 w_1 \cdots a_m w_m$$

with $a_i \in \Gamma$ and $w_i \in \Gamma^*$ satisfying the following two conditions:

$$\forall\, 1 \leq i \leq m\colon [a_i w_i] \mathrel{\mathcal{J}} [w] \quad \text{and} \quad \forall\, 0 \leq i \leq m\colon [w] <_{\mathcal{J}} [w_i]$$

Let $w'_i = a_i w_i$ for $1 \leq i \leq m$. For each $1 \leq i < m$ define a pair (L_i, R_i) where L_i is the $\mathcal{L}$-class of $[w'_i]$ and R_i is the $\mathcal{R}$-class of $[w'_{i+1}]$. Every such pair represents an $\mathcal{H}$-class within the $\mathcal{J}$-class of $[w]$. Therefore, the number of different such pairs does not exceed $|\{\, x \mid [w] \mathrel{\mathcal{J}} x \,\}|$. For the above factorization of w we perform an induction on the cardinality of the set $\{\, (L_i, R_i) \mid 1 \leq i < m \,\}$ to show that w has a factorization tree of height

$$h(w) \;<\; 3\,|\{\, (L_i, R_i) \mid 1 \leq i < m \,\}| \;+\; 3\,|\{\, x \mid [w] <_{\mathcal{J}} x \,\}|$$

Note that the number on the right-hand side of this inequality does not exceed $3\,|\{\, x \in M \mid [w] \leq_{\mathcal{J}} x \,\}|$. If every pair (L, R) occurs at most twice then we have $m - 1 \leq 2\,|\{\, (L_i, R_i) \mid 1 \leq i < m \,\}|$. We define a factorization tree for w by $d(w) = (w_0 w'_1,\, w'_2 \cdots w'_m)$, $d(w_0 w'_1) = (w_0,\, w'_1)$, $d(w'_i \cdots w'_m) = (w'_i, w'_{i+1} \cdots w'_m)$ for $2 \leq i < m$ and $d(w'_i) = (a_i, w_i)$ for $1 \leq i \leq m$. Since $[w] <_{\mathcal{J}} [w_i]$, by induction every w_i has a factorization tree of height $h(w_i) \;<\; 3\,|\{\, x \mid [w_i] \leq_{\mathcal{J}} x \,\}| \;\leq\; 3\,|\{\, x \mid [w] <_{\mathcal{J}} x \,\}|$. This yields:

$$\begin{aligned} h(w) \;&<\; m \;+\; 3\,|\{\, x \mid [w] <_{\mathcal{J}} x \,\}| \\ &\leq\; 3\,|\{\, (L_i, R_i) \mid 1 \leq i < m \,\}| \;+\; 3\,|\{\, x \mid [w] <_{\mathcal{J}} x \,\}| \end{aligned}$$

Note that the height might decrease if some of the w_i are empty. Now suppose that there exists a pair $(L, R) \in \{\, (L_i, R_i) \mid 1 \leq i < m \,\}$ occurring (at least) three times. Let $i_0 < \cdots < i_k$ be the sequence of all positions with $(L, R) = (L_{i_j}, R_{i_j})$. Let $\widehat{w_j} = w'_{i_{j-1}+1} \cdots w'_{i_j}$ for $1 \leq j \leq k$. For all $1 \leq j \leq \ell \leq k$ we have

– $[\widehat{w_j} \cdots \widehat{w_\ell}] \leq_{\mathcal{L}} [w'_{i_\ell}] \mathrel{\mathcal{L}} [w'_{i_0}]$.
– $[\widehat{w_j} \cdots \widehat{w_\ell}] \leq_{\mathcal{R}} [w'_{i_{j-1}+1}] \mathrel{\mathcal{R}} [w'_{i_0+1}]$.
– $[w'_{i_\ell}] \leq_{\mathcal{J}} [\widehat{w_j} \cdots \widehat{w_\ell}] \leq_{\mathcal{J}} [w] \mathrel{\mathcal{J}} [w'_{i_\ell}] \mathrel{\mathcal{J}} [w'_{i_0}] \mathrel{\mathcal{J}} [w'_{i_0+1}]$ by assumption on the factorization.

Thus for all $1 \le j \le \ell \le k$ and $1 \le j' \le \ell' \le k$ we get

– $[\widehat{w_j} \cdots \widehat{w_\ell}] \;\mathcal{L}\; [w'_{i_1}] \;\mathcal{L}\; [\widehat{w_{j'}} \cdots \widehat{w_{\ell'}}]$ and
– $[\widehat{w_j} \cdots \widehat{w_\ell}] \;\mathcal{R}\; [w'_{i_1+1}] \;\mathcal{R}\; [\widehat{w_{j'}} \cdots \widehat{w_{\ell'}}]$ and therefore
– $[\widehat{w_j} \cdots \widehat{w_\ell}] \;\mathcal{H}\; [\widehat{w_{j'}} \cdots \widehat{w_{\ell'}}]$ and since M is aperiodic we find
– $[\widehat{w_j} \cdots \widehat{w_\ell}] = [\widehat{w_{j'}} \cdots \widehat{w_{\ell'}}]$.

Therefore, all $[\widehat{w_j} \cdots \widehat{w_\ell}]$ denote the same element in M and since $k \ge 2$ this element is idempotent. In particular, we have $[\widehat{w_j}]^2 = [\widehat{w_j}] = [\widehat{w_\ell}]$ for all $1 \le j, \ell \le k$. We construct a factorization tree of w by

$$\begin{aligned} d(w) &= (w_0 w'_1 \cdots w'_{i_0},\ w'_{i_0+1} \cdots w'_m) \\ d(w'_{i_0+1} \cdots w'_m) &= (\widehat{w_1} \cdots \widehat{w_k},\ w'_{i_k+1} w'_m) \\ d(\widehat{w_1} \cdots \widehat{w_k}) &= (\widehat{w_1}, \ldots, \widehat{w_k}) \end{aligned}$$

Now, the pair (L, R) does not occur in any of the words $w_0 w'_1 \cdots w'_{i_0}$, $w'_{i_k+1} w'_m$ and $\widehat{w_j}$. By induction on the number of pairs (L_i, R_i) there exist factorization trees for them whose height is bounded by

$$3\,|\{\, (L_i, R_i) \mid 1 \le i < m \,\} \setminus \{(L, R)\}| \;+\; 3\,|\{\, x \mid [w] <_{\mathcal{J}} x \,\}|$$

Hence the height of the factorization tree of w satisfies the desired bound. ☐

6.2 Proof of Theorem 10

Lemma 1. *Let $\mu : \mathbb{M} \to (M, \le) : t \mapsto [t]$ be a homomorphism into an ordered monoid. If M is finite and satisfies the following property for all $e, s \in \mathbb{M}$:*

$$[e] = [e^2] \;\text{ and }\; \mathrm{alph}(s) \subseteq \mathrm{alph}(e) \;\text{ implies }\; [ese] \le [e] \tag{3}$$

then for every $p \in M$ the language $\mu^{-1}(\lfloor p \rfloor)$ is a polynomial.

Proof. By considering the case $s^\omega = e$ the property (3) implies $[s^\omega s s^\omega] = [s^\omega s] \le [s^\omega]$ and furthermore

$$[s^\omega] = [s^\omega s^\omega] \le [s^\omega s^{\omega-1}] \le [s^\omega s^{\omega-2}] \le \cdots \le [s^\omega s]$$

Hence $[s^\omega s] = [s^\omega]$ for all $s \in \mathbb{M}$ and therefore M is aperiodic. By Theorem 11 there exists a factorization forest d of height $< 3\,|M|$ for the homomorphism $\Gamma^* \to M : w \mapsto [\pi(w)]$ where $\pi : \Gamma^* \to \mathbb{M}$ is the natural projection. We define the height $h(t)$ of a trace t with respect to this factorization forest as the minimal height of one of its word representatives $w \in \pi^{-1}(t)$ and set $d(t) = \big(\pi(w_1), \ldots, \pi(w_n)\big)$ where $d(w) = (w_1, \ldots, w_n)$. We show that for every $t \in \mathbb{M}$ there exists a monomial L_t of the form

$$a_1 A_1^* a_2 \cdots A_n^* a_{n+1}$$

whose (minimal) degree is bounded by (a sufficiently large function in) the height $h(t)$ of the factorization tree of t and that has the property $t \in L_t \subseteq \mu^{-1}(\lfloor [t] \rfloor)$.

Since $h(t) < 3\,|M|$ there exist only finitely many such languages and therefore the following union

$$\bigcup_{t\in\mu^{-1}(\lfloor p \rfloor)} L_t$$

is finite and gives a polynomial representation for $\mu^{-1}(\lfloor p \rfloor)$.

If $|t| \leq 1$ then $L_t = \{\, t\,\}$ is a monomial with constant degree. Now let $|t| > 1$. The first case is $d(t) = (t_1, t_2)$. Then by induction on the height there exist monomials for t_1 and t_2 with $t_i \in L_{t_i} \subseteq \mu^{-1}(\lfloor [t_i] \rfloor)$ for $i = 1, 2$ whose degree is bounded by a function in $h(t) - 1$. We define the monomial $L_t = L_{t_1}\emptyset^* L_{t_2}$. Clearly, we have $t \in L_t$. It remains to verify $L_t \subseteq \mu^{-1}(\lfloor [t] \rfloor)$. Let $t_1' t_2' \in L_t$ with $t_1' \in L_{t_1}$ and $t_2' \in L_{t_2}$. Then

$$[t_1' t_2'] = [t_1'][t_2'] \leq [t_1][t_2] = [t_1 t_2] = [t]$$

The second case is $d(t) = (t_1, \ldots, t_n)$ with $[t_1]^2 = [t_1] = [t_2] = \ldots = [t_n] = [t]$. By induction there exist languages L_i with $t_i \in L_{t_i} \subseteq \mu^{-1}(\lfloor [t_i] \rfloor)$ for $i = 1, n$ whose degree is bounded by a function in $h(t) - 1$. We define the monomial $L_t = L_{t_1}\big(\mathrm{alph}(t)\big)^* L_{t_n}$. Again, $t \in L_t$ is clear. It remains to verify $L_t \subseteq \mu^{-1}(\lfloor [t] \rfloor)$. Let $t_1' s t_n' \in L_t$ with $t_1' \in L_{t_1}$, $t_n' \in L_{t_n}$ and $\mathrm{alph}(s) \subseteq \mathrm{alph}(t)$. Then

$$[t_1' s t_n'] = [t_1'][s][t_n'] \leq [t_1][s][t_n] = [t][s][t] \leq [t]$$

where the last inequality follows by (3). $\square$

Lemma 2. *Every monomial $A_0^* a_1 A_1^* \cdots a_m A_m^*$ is expressible in $\Sigma_2[E]$.*

Proof. We show that for every trace $t = t_0 a_1 t_1 \cdots a_m t_m$ with $\mathrm{alph}(t_i) \subseteq A_i$ there exists a $\Sigma_2[E]$-sentence φ_t whose size is bounded by a function in m and the size of the alphabet Γ (and not by $|t|$) such that

$$t \in L(\varphi_t) \subseteq A_0^* a_1 A_1^* \cdots a_m A_m^*$$

Since there are only finitely many such sentences the following disjunction is finite

$$\bigvee_{t\in A_0^* a_1 A_1^* \cdots a_m A_m^*} \varphi_t$$

and it describes exactly the monomial $A_0^* a_1 A_1^* \cdots a_m A_m^*$. The lemma then follows since $\Sigma_2[E]$ is closed under finite disjunctions.

Using the convention that a_0 is the empty trace we define $B_i = \mathrm{alph}(a_i t_i)$ for $0 \leq i \leq m$. For each i and each letter $b \in B_i$ fix a first position $x_{f,i,b}$ with label b in the factor $a_i t_i$ and a last position $x_{\ell,i,b}$ with label b in the factor $a_i t_i$. There is a $\Sigma_2[E]$-formula $\psi_t(\overline{x})$ with free variables $\overline{x} = (x_{f,i,b}, x_{\ell,i,b})_{0\leq i\leq m,\, b\in B_i}$ which reflects exactly the labeling and the partial ordering (i.e., not only the edge relation in the dependence graph) of the chosen positions in t. Furthermore the size of $\psi_t(\overline{x})$ does only depend on m and Γ. The formula φ_t we are looking for can be specified as follows:

$$\varphi_t \;=\; \exists \overline{x}\colon \psi_t(\overline{x}) \;\wedge\; \forall y\colon \bigvee_{b\in B_i,\, 0\leq i\leq m} \lambda(y) = b \;\wedge\; x_{f,i,b} \leq y \leq x_{\ell,i,b}$$

Note that it is allowed to write $x_{f,i,b} \le y \le x_{\ell,i,b}$ also over dependence graphs because $\psi_t(\overline{x})$ specifies the labels such that $\lambda(x_{f,i,b}) = \lambda(x_{\ell,i,b}) = b$. $\square$

Lemma 3. *Let $L \subseteq \mathbb{M}$ be a trace language and let $\mu : \mathbb{M} \to (M(L), \le)$ be its syntactic ordered homomorphism. If L is definable in $\Sigma_2[E]$ then $M(L)$ has the property that $[e] = [e^2]$ and $\mathrm{alph}(s) \subseteq \mathrm{alph}(e)$ implies $[ese] \le [e]$ for all $e, s \in \mathbb{M}$.*

Proof. Let $\varphi = \exists \overline{x}\, \forall \overline{y} \colon \psi(\overline{x}, \overline{y}) \in \Sigma_2[E]$ where $\overline{x} = (x_1, \dots, x_n)$, $\overline{y} = (y_1, \dots, y_n)$, and ψ is a propositional formula. Let $p, q, s, t \in \mathbb{M}$ and assume $\mathrm{alph}(s) \subseteq \mathrm{alph}(t)$. We show that for all $k \ge (n+1)^2$ we have

$$pt^k q \models \varphi \;\Rightarrow\; pt^k s t^k q \models \varphi \tag{4}$$

If $u = pt^k q$ models φ then there exist positions $X_1, \dots, X_n$ in the trace u such that

$$u, \overline{X} \models \forall \overline{y} \colon \psi(\overline{X}, \overline{y}) \tag{5}$$

where $X = (X_1, \dots, X_n)$. We refer to the k copies of the factor t in u as *blocks* numbered by 1 to k from left to right. By choice of k there exist n consecutive blocks such that no X_i is a position within these blocks, i.e.,

$$u = pt^{k_1} \cdot t^n \cdot t^{k_2} q$$

and all X_i are positions either in the prefix pt^{k_1} or in the suffix $t^{k_2} q$ of u. Consider the following factorization of $v = pt^k s t^k q$:

$$v = pt^{k_1} \cdot t^{k_1'} s t^{k_2'} \cdot t^{k_2} q$$

Since the prefix and suffix in this factorization are equal to that in the factorization of u and since all X_i correspond to positions in these parts of u we can choose the corresponding positions $X_1', \dots, X_n'$ in the identical parts of v. We claim that for $\overline{X'} = (X_1', \dots, X_n')$ we have

$$v, \overline{X'} \models \forall \overline{y} \colon \psi(\overline{X'}, \overline{y})$$

By contradiction, suppose there exist positions $Y_1', \dots, Y_n'$ in v such that for $\overline{Y'} = (Y_1', \dots, Y_n')$ we have

$$v, \overline{X'}, \overline{Y'} \models \neg\psi(\overline{X'}, \overline{Y'})$$

We show that this contradicts (5). If Y_i' is a position in the prefix pt^{k_1} or in the suffix $t^{k_2} q$ of v we can choose an analogous position Y_i in u. W.l.o.g. we assume that all Y_i are positions in the middle factor $t^{k_1'} s t^{k_2'}$ and that $i < j$ implies $(Y_j', Y_i') \notin E$, i.e., $Y_1', \dots, Y_n'$ is a linearization of the positions in $\overline{Y'}$. We now let Y_i be any position in the block $k_1 + i$ of u with the same label as Y_i'. This is possible since $\mathrm{alph}(s) \subseteq \mathrm{alph}(t)$. Now, all Y_i are positions in the middle factor t^n of u. By construction, we have

$$\begin{aligned}
(X_i, X_j) \in E \;&\Leftrightarrow\; (X_i', X_j') \in E \\
(Y_i, Y_j) \in E \;&\Leftrightarrow\; (Y_i', Y_j') \in E \\
(X_i, Y_j) \in E \;&\Leftrightarrow\; (X_i', Y_j') \in E \\
(Y_i, X_j) \in E \;&\Leftrightarrow\; (Y_i', X_j') \in E
\end{aligned}$$

Note that this would not be true for partial orders instead of dependence graphs. From $v, \overline{X'}, \overline{Y'} \models \neg\psi(\overline{X'}, \overline{Y'})$ it now follows

$$u, \overline{X}, \overline{Y} \models \neg\psi(\overline{X}, \overline{Y})$$

in contradiction to (5). This proves (4). For $L = L(\varphi)$ it follows that $[t^k s t^k] \leq [t^k]$ holds in the syntactic ordered monoid $(M(L), \leq)$ of L. The lemma now follows since $[t^k] = [t]$ if $[t] = [e]$ is idempotent. $\square$

Proof (Theorem 10). The implication "*1.* $\Rightarrow$ *2.*" follows by Lemma 1 since L is the union of languages of the form $\mu^{-1}(\lfloor p \rfloor)$ with $p \in M(L)$. "*2.* $\Rightarrow$ *3.*" follows from Lemma 2 since $\Sigma_2[E]$ is closed under finite disjunctions. Finally, the implication "*3.* $\Rightarrow$ *1.*" is Lemma 3. $\square$

References

1. P. Cartier and D. Foata. *Problèmes combinatoires de commutation et réarrangements.* Number 85 in Lecture Notes in Mathematics. Springer, 1969.
2. J. Chalopin and H. Leung. On factorization forests of finite height. *Theoretical Computer Science*, 310(1-3):489–499, 2004.
3. T. Colcombet. On Factorization Forests. Technical report, number hal-00125047, Irisa, Rennes, 2007.
4. V. Diekert and P. Gastin. LTL is expressively complete for Mazurkiewicz traces. *Journal of Computer and System Sciences*, 64:396–418, 2002.
5. V. Diekert and P. Gastin. Pure future local temporal logics are expressively complete for Mazurkiewicz traces. *Information and Computation*, 204:1597–1619, 2006.
6. V. Diekert, M. Horsch, and M. Kufleitner. On first-order fragments for Mazurkiewicz traces, to appear in Fundamenta Informaticae.
7. V. Diekert and G. Rozenberg, editors. *The Book of Traces.* World Scientific, Singapore, 1995.
8. W. Ebinger and A. Muscholl. Logical definability on infinite traces. *Theoretical Computer Science*, 154:67–84, 1996.
9. S. Eilenberg. *Automata, Languages, and Machines*, volume B. Academic Press, New York and London, 1976.
10. E. A. Emerson. Temporal and modal logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, chapter 16, pages 995–1072. Elsevier Science Publisher B. V., 1990.
11. K. Etessami, M. Y. Vardi, and T. Wilke. First-order logic with two variables and unary temporal logic. *Information and Computation*, 179(2):279–295, 2002.
12. D. Gabbay, I. Hodkinson, and M. Reynolds. *Temporal Logic: Mathematical Foundations and Computational Aspects.* Clarendon Press, Oxford, 1994.
13. P. Gastin and D. Kuske. Satisfiability and model checking for MSO-definable temporal logics are in PSPACE. In R. M. Amadio and D. Lugiez, editors, *Proc. of CONCUR'03*, volume 2761 of *LNCS*, pages 222–236. Springer, 2003.
14. P. Gastin and M. Mukund. An elementary expressively complete temporal logic for Mazurkiewicz traces. In *Proc. of ICALP'02*, number 2380 in LNCS, pages 938–949. Springer, 2002.
15. G. Guaiana, A. Restivo, and S. Salemi. Star-free trace languages. *Theoretical Computer Science*, 97:301–311, 1992.

16. R. M. Keller. Parallel program schemata and maximal parallelism I. Fundamental results. *Journal of the Association for Computing Machinery*, 20(3):514–537, 1973.
17. M. Kufleitner. *Logical Fragments for Mazurkiewicz Traces: Expressive Power and Algebraic Characterizations*. Dissertation, Universität Stuttgart, 2006.
18. M. Kufleitner. Polynomials, fragments of temporal logic and the variety DA over traces. In O. H. Ibarra and Z. Dang, editors, *Proc. of DLT'06*, volume 4036 of *LNCS*, pages 37–48. Springer, 2006.
19. A. Mazurkiewicz. Concurrent program schemes and their interpretations. DAIMI Rep. PB 78, Aarhus University, Aarhus, 1977.
20. A. Mazurkiewicz. Trace theory. In W. Brauer et al., editors, *Petri Nets, Applications and Relationship to other Models of Concurrency*, number 255 in LNCS, pages 279–324. Springer, 1987.
21. J.-É. Pin. *Varieties of Formal Languages*. North Oxford Academic, London, 1986.
22. J.-É. Pin. A variety theorem without complementation. In *Russian Mathematics (Izvestija vuzov.Matematika)*, volume 39, pages 80–90, 1995.
23. J.-É. Pin, H. Straubing, and D. Thérien. Locally trivial categories and unambiguous concatenation. *Journal of Pure and Applied Algebra*, 52:297–311, 1988.
24. J.-É. Pin and P. Weil. Polynominal closure and unambiguous product. *Theory Comput. Syst*, 30(4):383–422, 1997.
25. M. P. Schützenberger. Sur le produit de concatenation non ambigu. *Semigroup Forum*, 13:47–75, 1976.
26. I. Simon. Factorization forests of finite height. *Theoretical Computer Science*, 72(1):65–94, 1990.
27. A. P. Sistla and E. Clarke. The complexity of propositional linear time logic. *Journal of the Association for Computing Machinery*, 32:733–749, 1985.
28. L. Stockmeyer. The complexity of decision problems in automata theory and logic. PhD thesis, TR 133, M.I.T., Cambridge, 1974.
29. H. Straubing. *Finite Automata, Formal Logic, and Circuit Complexity*. Birkhäuser, Boston, Basel and Berlin, 1994.
30. P. Tesson and D. Thérien. Diamonds are Forever: The Variety DA. In G. M. dos Gomes Moreira da Cunha, P. V. A. da Silva, and J.-É. Pin, editors, *Semigroups, Algorithms, Automata and Languages, Coimbra (Portugal) 2001*, pages 475–500. World Scientific, 2002.
31. P. S. Thiagarajan and I. Walukiewicz. An expressively complete linear time temporal logic for Mazurkiewicz traces. In *Proc. of LICS'97*, pages 183–194, 1997.
32. W. Thomas. Languages, automata and logic. In A. Salomaa and G. Rozenberg, editors, *Handbook of Formal Languages*, volume 3, Beyond Words. Springer, Berlin, 1997.
33. P. Weis and N. Immerman. Structure theorem and strict alternation hierarchy for FO^2 on words. Technical report, Department of Computer Science University of Massachusetts, Amherst, 2006.