LTL is expressively complete for Mazurkiewicz traces

Volker Diekert[1] and Paul Gastin[2]

[1] *Inst. für Informatik, Universität Stuttgart, Breitwiesenstr. 20-22, D-70565 Stuttgart*
`diekert@informatik.uni-stuttgart.de`
and
[2] *LIAFA, Université Paris 7, 2, place Jussieu, F-75251 Paris Cedex 05*
`Paul.Gastin@liafa.jussieu.fr`

A long standing open problem in the theory of (Mazurkiewicz) traces has been the question whether LTL (Linear Temporal Logic) is expressively complete with respect to the first order theory. We solve this problem positively for finite and infinite traces and for the simplest temporal logic, which is based only on next and until modalities. Similar results were established previously, but they were all weaker, since they used additional past or future modalities. Another feature of our work is that our proof is direct and does not use any reduction to the word case.

Key Words: Temporal Logic, Mazurkiewicz Traces, Concurrency

1. INTRODUCTION

Nowadays, it is widely accepted that we need to develop methods to verify critical systems. For this, we need formal specifications for the expected behaviors of systems. Conveniently, these formal specifications are given by temporal logic formulae. When dealing with concurrent systems, a possible approach is to reduce them to sequential ones by considering all linearizations. Then one can use techniques and tools developed for sequential systems, but usually, one faces a combinatorial explosion. In order to avoid this problem, one could try to work directly on concurrent systems and this explains why a lot of research has been devoted recently to the study of temporal logics for concurrency. A major aim is to find a temporal logic which is expressive enough to ensure that all desired specifications can be formalized.

Trace theory, initiated by Mazurkiewicz, is one of the most popular settings for studying concurrency. See [6] for the general background of trace theory, in particular, [18] for traces and logic and [12] for infinite traces. It is no surprise that various temporal logics for traces have been extensively studied [1, 15, 16, 17, 19, 20, 21]. A long standing problem is to find natural temporal logics that are expressively complete with respect to the first order theory of finite and infinite traces.

There are two main classes of temporal logics for traces, the global ones and the local ones. With *local logics*, formulae are evaluated at local events, i.e., local states of the processes. With the techniques presented here, we have obtained a positive result in the special case of traces associated with cograph dependence alphabets [4]: various natural local temporal logics are expressively complete with respect to first order logic. But the general problem is still open for arbitrary trace alphabets.

Global logics formulae are evaluated at global configurations of the system. For sequential systems there is no difference between a global or local viewpoint since a cut of a sequence is defined by a single event. The expressive completeness of the linear temporal logic is however highly non trivial. It is a celebrated result of Kamp [13] which states that linear temporal logic has the same expressive power as the first order theory of words. Originally Kamp used future and past modalities, but it was established later that past modalities can be avoided, see [10] and [9] for more details.

In this paper, we only deal with global temporal logics and we will omit the word *global* from now on. The first completeness result for traces is by Ebinger [7]. He proved that a linear temporal logic with both past and future modalities is expressively complete for finite traces but his approach did not cope with infinite traces. Then, Thiagarajan and Walukiewicz [22] proved the completeness both for finite and infinite traces of LTrL, a linear temporal logic with future modalities and past modalities in the restricted form of past constants. These two results were obtained using a reduction to the word case. In [14] it was claimed that LTL (the basic linear temporal logic based on the usual next and until modalities) is expressively complete for finite traces, but the proof contained a flaw. The expressive completeness for a pure future temporal logic LTL_f was established in [2]. The result holds for finite and infinite traces, but LTL_f contains new filter modalities in addition to the usual next and until modalities. Without independency these filter modalities are simple macros, but in general there seems to be no direct way to remove them; and the problem of the expressive completeness of the basic linear temporal logic over traces remained open.

In this paper, we solve this problem positively. Previous expressive completeness results for words and for traces are now formal corollaries of our main theorem. It should be noted that, contrary to most previous works, our proof does not use any reduction to the word case. Instead, we extend the new proof introduced by Wilke for finite words [24] which is based on the well-known fact that first order languages are aperiodic. Basically, we follow here the same approach as in [2]. In the former paper, filter modalities were used to express some special products of trace languages. Our new proof is a substantial revision of the previous one. We are now able to express the products mentioned above directly with basic formulae without using the filter modalities. [1]

2. PRELIMINARIES

By (Σ, I) we mean a finite *independence alphabet* where Σ denotes a finite alphabet and $I \subseteq \Sigma \times \Sigma$ is an irreflexive and symmetric relation called the *independence relation*. The complementary relation $D = (\Sigma \times \Sigma) \setminus I$ is called the *dependence*

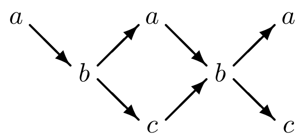
[1] The present paper is the journal version of an extended abstract presented in [3].

relation. The monoid of *finite traces* $\mathbb{M}(\Sigma, I)$ is defined as a quotient monoid with respect to the congruence relation induced by I, i.e., $\mathbb{M}(\Sigma, I) = \Sigma^*/\{\, ab = ba \mid (a, b) \in I \,\}$. We also write $\mathbb{M}$ instead of $\mathbb{M}(\Sigma, I)$.

A trace $x \in \mathbb{M}$ is given by a congruence class of a word $a_1 \cdots a_n \in \Sigma^*$ where $a_i \in \Sigma$, $1 \leq n$. By abuse of language, but for simplicity we denote a trace x by one of its representing words $a_1 \cdots a_n$. The number n is called the length of x, denoted by $|x|$. For $n = 0$ we obtain the *empty* trace, denoted by 1. The *alphabet* $\operatorname{alph}(x)$ of a trace x is the set of letters occurring in x. A *trace language* is a subset $L \subseteq \mathbb{M}$. The product of trace languages is defined as usual:

$$KL = \{xy \in \mathbb{M} \mid x \in K, y \in L\}.$$

Every trace $a_1 \cdots a_n \in \mathbb{M}$ can be identified with its *dependence graph*. This is (an isomorphism class of) a node–labeled, acyclic, directed graph $[V, E, \lambda]$, where $V = \{1, \ldots, n\}$ is a set of vertices, each $i \in V$ is labeled by $\lambda(i) = a_i$, and there is an edge $(i, j) \in E$ if and only if both $i < j$ and $(\lambda(i), \lambda(j)) \in D$. In pictures it is common to draw the Hasse diagram only. Thus, all redundant edges are omitted. For instance, let $(\Sigma, D) = a — b — c$, i.e., $I = \{(a, c), (c, a)\}$. Then the trace $x = abcabca$ is given by



An *infinite trace* is an infinite dependence graph $[V, E, \lambda]$ such that for all $j \in V$ the set $\downarrow j = \{i \in V \mid i \leq j\}$ is finite. A *real trace* is a finite or infinite trace. The set of real traces is denoted by $\mathbb{R}(\Sigma, I)$ or simply by $\mathbb{R}$. For a real trace $x = [V, E, \lambda]$ the *alphabet* is $\operatorname{alph}(x) = \lambda^{-1}(V)$ and the *alphabet at infinity* is the set of letters occurring infinitely many times in x, i.e., $\operatorname{alphinf}(x) = \{a \in \Sigma \mid |\lambda^{-1}(a)| = \infty\}$. Usually we consider real trace languages, i.e., we consider subsets of $\mathbb{R}$. If we speak about *finitary* languages, then we refer to subsets of $\mathbb{M}$. For $A \subseteq \Sigma$ we denote by $\mathbb{M}_A$ the submonoid of $\mathbb{M}(\Sigma, I)$ generated by A:

$$\mathbb{M}_A = \mathbb{M}(A, I \cap A \times A) = \{x \in \mathbb{M}(\Sigma, I) \mid \operatorname{alph}(x) \subseteq A\}.$$

Accordingly, we let $\mathbb{R}_A = \{x \in \mathbb{R} \mid \operatorname{alph}(x) \subseteq A\}$.

By $\min(x)$ and $\max(x)$ we refer to the *minimal* and *maximal letters* in the dependence graph. (We shall use the notation $\max(x)$ only for finite traces $x \in \mathbb{M}$.) In the example above $\min(x) = \{a\}$ and $\max(x) = \{a, c\}$. Formally:

$$\begin{aligned} \min(x) &= \{a \in \Sigma \mid x \in a\mathbb{R}\}, \\ \max(x) &= \{a \in \Sigma \mid x \in \mathbb{M}a\}. \end{aligned}$$

For $B \subseteq \Sigma$ and $\# \in \{\subseteq, =, \supseteq, \neq\}$ we define:

$$\begin{aligned} I(B) &= \{a \in \Sigma \mid \forall b \in B : (a, b) \in I\}, \\ D(B) &= \{a \in \Sigma \mid \exists b \in B : (a, b) \in D\}, \\ (\mathrm{Min} \mathbin{\#} B) &= \{x \in \mathbb{R} \mid \min(x) \mathbin{\#} B\}, \\ (\mathrm{Max} \mathbin{\#} B) &= \{x \in \mathbb{M} \mid \max(x) \mathbin{\#} B\}. \end{aligned}$$

Note that (Max $\#$ B) denotes a finitary language, whereas (Min $\#$ B) is a real trace language. The alphabet Σ is the disjoint union of the sets $I(B)$ and $D(B)$. If B happens to be a singleton, then we usually omit braces, e.g. we write $I(b)$ or $(\mathrm{Max} = b)$.

3. TEMPORAL LOGIC FOR TRACES

The syntax of the temporal logic LTL(Σ) is defined as follows. There are: a constant symbol $\perp$ representing *false*, the logical connectives $\neg$ (not) and $\vee$ (or), for each $a \in \Sigma$ a unary operator $\langle a \rangle$, called *next-a*, and a binary operator $\mathbf{U}$, called *until*. Formally, the syntax is given by:

$$\varphi ::= \perp \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle a \rangle \varphi \mid \varphi \,\mathbf{U}\, \varphi,$$

where $a \in \Sigma$.

The semantics is usually defined by saying when some formula φ is satisfied by some real trace z at some configuration (i.e., finite prefix) x; hence by defining $(z, x) \models \varphi$. Since our temporal logic uses future modalities only, we have $(z, x) \models \varphi$ if and only if $(y, 1) \models \varphi$, where y is the unique trace satisfying $z = xy$. Therefore, we do not need to deal with configurations and it is enough to say when a trace satisfies a formula at the empty configuration, denoted simply by $z \models \varphi$. This is done inductively on the formula as follows:

$$\begin{array}{ll}
z \not\models \perp, & \\
z \models \neg\varphi & \text{if } z \not\models \varphi, \\
z \models \varphi \vee \psi & \text{if } z \models \varphi \text{ or } z \models \psi, \\
z \models \langle a \rangle \varphi & \text{if } z = ay \text{ and } y \models \varphi, \\
z \models \varphi \,\mathbf{U}\, \psi & \text{if } z = xy,\ x \in \mathbb{M},\ y \models \psi, \text{ and } x = x'x'',\ x'' \neq 1 \text{ implies } x''y \models \varphi.
\end{array}$$

As usual, we define $\mathrm{L}_{\mathbb{R}}(\varphi) = \{x \in \mathbb{R} \mid x \models \varphi\}$. We say that a trace language $L \subseteq \mathbb{R}$ is *expressible in* LTL(Σ), if there exists a formula $\varphi \in$ LTL(Σ) such that $L = \mathrm{L}_{\mathbb{R}}(\varphi)$.

Equivalently, we can define inductively the language $\mathrm{L}_{\mathbb{R}}(\varphi)$ as follows:

$$\begin{array}{rcl}
\mathrm{L}_{\mathbb{R}}(\perp) & = & \emptyset \\
\mathrm{L}_{\mathbb{R}}(\neg\varphi) & = & \mathbb{R} \setminus \mathrm{L}_{\mathbb{R}}(\varphi) \\
\mathrm{L}_{\mathbb{R}}(\varphi \vee \psi) & = & \mathrm{L}_{\mathbb{R}}(\varphi) \cup \mathrm{L}_{\mathbb{R}}(\psi) \\
\mathrm{L}_{\mathbb{R}}(\langle a \rangle \varphi) & = & a\mathrm{L}_{\mathbb{R}}(\varphi) \\
\mathrm{L}_{\mathbb{R}}(\varphi \,\mathbf{U}\, \psi) & = & \mathrm{L}_{\mathbb{R}}(\varphi) \,\mathbf{U}\, \mathrm{L}_{\mathbb{R}}(\psi)
\end{array}$$

where the *until* operator $\mathbf{U}$ is defined on real trace languages by

$$L \,\mathbf{U}\, K = \{xy \mid x \in \mathbb{M},\ y \in K, \text{ and } x = x'x'',\ x'' \neq 1 \text{ implies } x''y \in L\}.$$

Remark. For comparison let us mention that the syntax and semantics of the logics LTrL defined in [22] and LTL_f defined in [2] are very similar. For LTrL, the difference is only that there is in addition for each letter $a \in \Sigma$ a constant $\langle a^{-1} \rangle \top$. Since the constant $\langle a^{-1} \rangle \top$ refers to the past, we need to use configurations to define its semantics: We have $(z, x) \models \langle a^{-1} \rangle \top$ if and only if $a \in \max(x)$. For LTL_f, the difference is that for each subalphabet $B \subseteq \Sigma$ there is in addition a modality $\langle B^* \rangle \varphi$

whose semantics is given by $z \models \langle B^* \rangle \varphi$ if $z = xy$, $x \in \mathbb{M}_B$, and $y \models \varphi$. A consequence or our main result (Theorem 7.1), these three logics have the same expressive power. However, it is not clear whether there is a direct (e.g. inductive) translation of LTrL or LTL_f to LTL; or between LTrL and LTL_f.

The following operators are standard abbreviations.

$$
\begin{array}{rcll}
\top & := & \neg\bot & \textit{true}, \\
a & := & \langle a \rangle \top & \text{for } a \in \Sigma, \\
A & := & \bigvee_{a \in A} \langle a \rangle \top & \text{for } A \subseteq \Sigma, \\
\mathbf{X}\,\varphi & := & \bigvee_{a \in \Sigma} \langle a \rangle \varphi & \textit{neXt } \varphi, \\
\mathrm{stop} & := & \neg\,\mathbf{X}\,\top & \textit{termination}, \\
\mathbf{F}\,\varphi & := & \top\,\mathbf{U}\,\varphi & \textit{future} \text{ or } \textit{eventually } \varphi, \\
\mathbf{G}\,\varphi & := & \neg\,\mathbf{F}\,\neg\varphi & \textit{globally} \text{ or } \textit{always } \varphi.
\end{array}
$$

EXAMPLE 3.1. $\mathrm{L}_{\mathbb{R}}(\mathrm{stop}) = \{1\}$, $\mathrm{L}_{\mathbb{R}}(\mathbf{F}\,\mathrm{stop}) = \mathbb{M}$ and for $A \subseteq \Sigma$ we have:

$$
\begin{array}{rcl}
\mathbb{R}_A & = & \mathrm{L}_{\mathbb{R}}(\neg\,\mathbf{F}(\Sigma \setminus A)), \\
\mathbb{M}_A & = & \mathrm{L}_{\mathbb{R}}(\mathbf{F}\,\mathrm{stop} \wedge \neg\,\mathbf{F}(\Sigma \setminus A)), \\
(\mathrm{Min} = A) & = & \mathrm{L}_{\mathbb{R}}(\neg(\Sigma \setminus A) \wedge \bigwedge_{a \in A} a), \\
(\mathrm{Max} \supseteq A) & = & \mathrm{L}_{\mathbb{R}}(\bigwedge_{a \in A} \mathbf{F}\langle a \rangle \mathrm{stop}), \\
(\mathrm{alphinf} = A) & = & \mathrm{L}_{\mathbb{R}}(\mathbf{F}\,\mathbf{G}\,\neg(\Sigma \setminus A) \wedge \bigwedge_{a \in A} \mathbf{G}\,\mathbf{F}\,a).
\end{array}
$$

Later we shall perform an induction on the size of Σ leading to formulae $\varphi \in \mathrm{LTL}(A)$ for $A \subseteq \Sigma$. Such a formula may be interpreted over $\mathbb{R}_A$ or over $\mathbb{R}$. Note that the main difference between the two interpretations is with negation since the complement is taken with respect to $\mathbb{R}_A$ or $\mathbb{R}$. The following lemma gives the relationship between the two interpretations.

LEMMA 3.1.

1.*Let* $\varphi \in \mathrm{LTL}(A)$. *Then,* $\mathrm{L}_{\mathbb{R}_A}(\varphi) = \mathrm{L}_{\mathbb{R}}(\varphi \wedge \neg\,\mathbf{F}(\Sigma \setminus A))$.

2.*Let* $\varphi \in \mathrm{LTL}$. *We can find a formula* $\varphi_A \in \mathrm{LTL}(A)$ *such that* $\mathrm{L}_{\mathbb{R}_A}(\varphi_A) = \mathrm{L}_{\mathbb{R}}(\varphi) \cap \mathbb{R}_A$.

Proof. 1. We have: $\mathbb{R}_A = \mathrm{L}_{\mathbb{R}}(\neg\,\mathbf{F}(\Sigma \setminus A))$ and $\mathrm{L}_{\mathbb{R}_A}(\varphi) = \mathrm{L}_{\mathbb{R}}(\varphi) \cap \mathbb{R}_A$.
2. The formula φ_A is constructed by induction:

- $\bot_A = \bot$,
- $(\neg\varphi)_A = \neg(\Phi_A)$,
- $(\varphi \vee \psi)_A = \varphi_A \vee \psi_A$,
- $(\langle a \rangle \varphi)_A = \begin{cases} \langle a \rangle (\varphi_A) & \text{if } a \in A, \\ \bot & \text{otherwise,} \end{cases}$
- $(\varphi\,\mathbf{U}\,\psi)_A = \varphi_A\,\mathbf{U}\,\psi_A$.

■

We will also use an induction on $|\Sigma|$ when the graph (Σ, D) is not connected. Then we find a partition of the alphabet $\Sigma = \Sigma_1 \cup \Sigma_2$ with $\Sigma_1 \times \Sigma_2 \subseteq I$ and each trace $x \in \mathbb{R}$ can be split into two independent traces x_1 and x_2 over Σ_1 and Σ_2 with $x = x_1x_2 = x_2x_1$.

Lemma 3.2. *Assume that* $\Sigma = \Sigma_1 \cup \Sigma_2$ *with* $\Sigma_1 \times \Sigma_2 \subseteq I$. *Let* $\mathbb{R}_i = \mathbb{R}_{\Sigma_i}$ *and* $\varphi_i \in \mathrm{LTL}(\Sigma_i)$ *for* $i = 1, 2$. *Then* $\mathrm{L}_{\mathbb{R}}(\varphi_1 \wedge \varphi_2) = \mathrm{L}_{\mathbb{R}_1}(\varphi_1) \cdot \mathrm{L}_{\mathbb{R}_2}(\varphi_2)$.

Proof. We show by induction that $\mathrm{L}_{\mathbb{R}}(\varphi_1) = \mathrm{L}_{\mathbb{R}_1}(\varphi_1) \cdot \mathbb{R}_2$. The result follows since then $\mathrm{L}_{\mathbb{R}}(\varphi_1 \wedge \varphi_2) = \mathrm{L}_{\mathbb{R}}(\varphi_1) \cap \mathrm{L}_{\mathbb{R}}(\varphi_2) = (\mathrm{L}_{\mathbb{R}_1}(\varphi_1) \cdot \mathbb{R}_2) \cap (\mathbb{R}_1 \cdot \mathrm{L}_{\mathbb{R}_2}(\varphi_2)) = \mathrm{L}_{\mathbb{R}_1}(\varphi_1) \cdot \mathrm{L}_{\mathbb{R}_2}(\varphi_2)$.
• $\perp$ and $\varphi_1 \vee \psi_1$: trivial.
• $\neg\varphi_1$: We have, $\mathrm{L}_{\mathbb{R}}(\neg\varphi_1) = \mathbb{R} \setminus \mathrm{L}_{\mathbb{R}}(\varphi_1) = (\mathbb{R}_1 \cdot \mathbb{R}_2) \setminus (\mathrm{L}_{\mathbb{R}_1}(\varphi_1) \cdot \mathbb{R}_2) = (\mathbb{R}_1 \setminus \mathrm{L}_{\mathbb{R}_1}(\varphi_1)) \cdot \mathbb{R}_2) = \mathrm{L}_{\mathbb{R}_1}(\neg\varphi_1) \cdot \mathbb{R}_2$.
• $\langle a \rangle \varphi_1$ with $a \in \Sigma_1$: Easy since $\mathrm{L}_{\mathbb{R}}(\langle a \rangle \varphi_1) = a\mathrm{L}_{\mathbb{R}}(\varphi_1) = a(\mathrm{L}_{\mathbb{R}_1}(\varphi_1) \cdot \mathbb{R}_2) = (a\mathrm{L}_{\mathbb{R}_1}(\varphi_1)) \cdot \mathbb{R}_2 = \mathrm{L}_{\mathbb{R}_1}(\langle a \rangle \varphi_1) \cdot \mathbb{R}_2$.
• $\varphi_1 \,\mathbf{U}\, \psi_1$: Let $z = xy$ with $y \in \mathrm{L}_{\mathbb{R}}(\psi_1)$ and for all $x = x'x''$ with $x'' \neq 1$ we have $x''y \in \mathrm{L}_{\mathbb{R}}(\varphi_1)$. Write $x = x_1x_2$ and $y = y_1y_2$ with $x_i, y_i \in \mathbb{R}_i$ for $i = 1, 2$. By induction hypothesis, $y_1 \in \mathrm{L}_{\mathbb{R}_1}(\psi_1)$ and for all factorizations $x_1 = x'x''$ with $x'' \neq 1$ we have $x''y \in \mathrm{L}_{\mathbb{R}_1}(\varphi_1) \cdot \mathbb{R}_2$, that is, $x''y_1 \in \mathrm{L}_{\mathbb{R}_1}(\varphi_1)$. Hence $x_1y_1 \in \mathrm{L}_{\mathbb{R}_1}(\varphi_1 \,\mathbf{U}\, \psi_1)$ and we obtain the first inclusion since $z = (x_1x_2)(y_1y_2) = (x_1y_1)(x_2y_2)$.

Conversely, let $z = z_1z_2$ with $z_1 \in \mathrm{L}_{\mathbb{R}_1}(\varphi_1 \,\mathbf{U}\, \psi_1)$ and $z_2 \in \mathbb{R}_2$. Write $z_1 = x_1y_1$ with $y_1 \in \mathrm{L}_{\mathbb{R}_1}(\psi_1)$ and for all $x_1 = x'x''$ with $x'' \neq 1$ we have $x''y_1 \in \mathrm{L}_{\mathbb{R}_1}(\varphi_1)$. Then $y_1z_2 \in \mathrm{L}_{\mathbb{R}}(\psi_1)$ and $x''y_1z_2 \in \mathrm{L}_{\mathbb{R}}(\varphi_1)$ which proves the converse inclusion. ■

4. FIRST-ORDER LOGIC

The first order theory of traces is given by the syntax of $\mathrm{FO}(\Sigma, <)$:

$$\varphi ::= P_a(x) \mid x < y \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x \varphi,$$

where $a \in \Sigma$ and $x, y \in \mathrm{Var}$ are first order variables. Given a trace $t = [V, E, \lambda]$ and a valuation of the free variables into the vertices $\sigma : \mathrm{Var} \to V$, the semantics is obtained by interpreting the relation $<$ as the transitive closure of E and the predicate $P_a(x)$ by $\lambda(\sigma(x)) = a$. Then we can say when $(t, \sigma) \models \varphi$. If φ is a closed formula (a sentence), then the valuation σ has an empty domain and we define the language $\mathrm{L}_{\mathbb{R}}(\varphi) = \{t \in \mathbb{R} \mid t \models \varphi\}$. We say that a trace language $L \subseteq \mathbb{R}$ is expressible in $\mathrm{FO}(\Sigma, <)$ if there exists some sentence $\varphi \in \mathrm{FO}(\Sigma, <)$ such that $L = \mathrm{L}_{\mathbb{R}}(\varphi)$.

Passing from a temporal logic formula to a first order one is well-known and belongs to folklore. The transformation relies on the fact that a prefix (configuration) p of a trace t can be defined by its maximal vertices. Such a set of maximal vertices is bounded by the maximal number of pairwise independent letters in Σ. Therefore, a prefix inside a trace can be defined using a bounded number of first order variables.

Proposition 4.1. *If a trace language is expressible in* $\mathrm{LTL}(\Sigma)$*, then it is expressible in* $\mathrm{FO}(\Sigma, <)$.

Proof. Though this result is not new, we give a sketch of the proof for the sake of completeness. We first define the relativization ψ_X of a formula $\psi \in$ FO with respect to some set $X = \{x_1, \ldots, x_k\}$ of first order variables not occurring in ψ. The idea is that a trace $t = (V, E, \lambda)$ satisfies ψ_X if and only if $t' \models \psi$ where t' is the suffix trace defined by the vertices which are not in the past of the meanings of the first order variables in X. Formally, ψ_X is defined inductively as follows. $(P_a(x))_X = P_a(x)$, $(x < y)_X = (x < y)$, $(\neg\psi)_X = \neg(\psi_X)$, $(\psi' \vee \psi'')_X = \psi'_X \vee \psi''_X$, and $(\exists x \psi)_X = \exists x((\neg \bigvee_{1\leq i\leq k} x \leq x_i) \wedge \psi_X)$.

Now, we associate by induction with each formula $\varphi \in$ LTL a closed formula $\widetilde{\varphi} \in$ FO defining the same language.

$$\begin{aligned}
\widetilde{\bot} &= \exists x(x < x), \\
\widetilde{\neg\varphi} &= \neg\widetilde{\varphi}, \\
\widetilde{\varphi \vee \psi} &= \widetilde{\varphi} \vee \widetilde{\psi}, \\
\widetilde{\langle a\rangle\varphi} &= \exists x\Big((\forall y \neg(y < x)) \wedge P_a(x) \wedge \widetilde{\varphi}_{\{x\}}\Big), \\
\widetilde{\varphi \,\mathbf{U}\, \psi} &= \widetilde{\psi} \vee (\widetilde{\varphi} \wedge (\exists x_1 \cdots x_k, (\widetilde{\psi}_X \wedge \forall y_1 \cdots y_k, \downarrow Y \subset \downarrow X \longrightarrow \widetilde{\varphi}_Y))),
\end{aligned}$$

where k is the size of the maximal clique of (Σ, I), $X = \{x_1, \ldots, x_k\}$, $Y = \{y_1, \ldots, y_k\}$ and

$$\downarrow Y \subset \downarrow X = \left(\bigwedge_{1\leq i\leq k} \bigvee_{1\leq j\leq k} y_i \leq x_j\right) \wedge \left(\bigvee_{1\leq i\leq k} \bigwedge_{1\leq j\leq k} \neg(x_i \leq y_j)\right).$$

■

As in the case of LTrL, this translation yields a non-elementary decision procedure for the uniform satisfiability problem of LTL. (See also [11] for a modular decision procedure based on automata constructions.) For the lower bound, we can use [23], since the lower bound is given there for the fragment of LTrL without the previous constants $\langle a^{-1}\rangle\top$. Putting this together the result of Walukiewicz becomes:

Proposition 4.2 ([23]). *The satisfiability problem for both logics* LTrL *and* LTL *is non-elementary over Mazurkiewicz traces.*

5. APERIODIC LANGUAGES

Recall that a finite monoid S is *aperiodic*, if there is some $n \geq 0$ such that $s^n = s^{n+1}$ for all $s \in S$. A finitary trace language $L \subseteq \mathbb{M}$ is *aperiodic*, if there exists a morphism to some finite aperiodic monoid $h : \mathbb{M} \to S$ such that $L = h^{-1}(h(L))$. Since our considerations include infinite traces, we have to extend the notion of an aperiodic language to real trace languages such that it becomes equivalent for finitary languages.

Let $h : \mathbb{M} \to S$ be a morphism to some finite monoid S. For $x, y \in \mathbb{R}$, we say that x and y are h-similar, denoted by $x \sim_h y$ if either $x, y \in \mathbb{M}$ and $h(x) = h(y)$ or x and y have infinite factorizations in non-empty finite traces $x = x_1x_2\cdots$,

$y = y_1 y_2 \cdots$ with $x_i, y_i \in \mathbb{M} \setminus \{1\}$ and $h(x_i) = h(y_i)$ for all i. According to the definition of h-similarity, we never have $x \sim_h y$ when x is finite and y is infinite. We denote by $\approx_h$ the transitive closure of $\sim_h$ which is therefore an equivalence relation. An equivalence class is denoted by $[x]_{\approx_h} = \{y \in \mathbb{R} \mid y \approx_h x\}$. For a finite trace $x \in \mathbb{M}$ we have $[x]_{\approx_h} = h^{-1}(h(x))$ and the monoid $\mathbb{M}$ is covered by at most $|S|$ classes. Using a Ramsey-type argument we can show that $\mathbb{R} \setminus \mathbb{M}$ is covered by at most $|S|^2$ classes. Indeed, let $x \in \mathbb{R} \setminus \mathbb{M}$ and consider a factorization $x = x_1 x_2 \cdots$ in finite traces. For all $0 \le i < j$ we define $s_{i,j} = h(x_{i+1} \cdots x_j) \in S$. Since S is finite, Ramsey's theorem implies the existence of $e \in S$ and an increasing sequence of integers $0 = i_0 < i_1 < i_2 < \cdots$ such that $s_{i_p,i_q} = e$ for all $0 < p < q$. If we let $s = s_{i_0,i_1}$ then we obtain $x \in h^{-1}(s)h^{-1}(e)^{\omega} \subseteq [x]_{\approx_h}$, which proves that the number of $\approx_h$-classes in $\mathbb{R} \setminus \mathbb{M}$ is at most $|S|^2$. Therefore, $\{[x]_{\approx_h} \mid x \in \mathbb{R}\}$ defines a finite partition of $\mathbb{R}$ of cardinality at most $|S|^2 + |S|$. A real trace language $L \subseteq \mathbb{R}$ is *recognized* by h if it is saturated by $\sim_h$, i.e., $x \in L$ implies $[x]_{\approx_h} \subseteq L$ for all $x \in \mathbb{R}$. A real trace language $L \subseteq \mathbb{R}$ is *aperiodic* if it is recognized by some morphism to some finite and aperiodic monoid.

The following results will be useful later. The first proposition will allow us to use an induction on the size of the alphabet Σ.

Proposition 5.1. *Let $L \subseteq \mathbb{R}$ be a language recognized by the morphism $h : \mathbb{M} \to S$ into a finite monoid S and let $A \subseteq \Sigma$. Then, $L \cap \mathbb{R}_A$ and $L \cap \mathbb{M}_A$ are recognized by the restriction $h\restriction_{\mathbb{M}_A}$ of h to $\mathbb{M}_A$.*

Proof. Let $x \in L \cap \mathbb{R}_A$ and let $y \in \mathbb{R}$. Assume that x, y are $h\restriction_{\mathbb{M}_A}$-similar. Then they are also h-similar and we deduce that $y \in L \cap \mathbb{R}_A$. The proof is the same for $L \cap \mathbb{M}_A$. ∎

Then, we consider the case where the dependence alphabet (Σ, D) is non-connected. The following analogue of Mezei's Theorem holds for real trace languages:

Proposition 5.2. *Let $L \subseteq \mathbb{R}$ be a language recognized by the morphism $h : \mathbb{M} \to S$ into a finite monoid S. Assume that $\Sigma = \Sigma_1 \cup \Sigma_2$ with $\Sigma_1 \times \Sigma_2 \subseteq I$ and let $\mathbb{M}_i = \mathbb{M}_{\Sigma_i}$, $\mathbb{R}_i = \mathbb{R}_{\Sigma_i}$ and $h_i = h\restriction_{M_i}$ for $i = 1, 2$. Then, L is a finite union of products $L_1 \cdot L_2$ where $L_i \subseteq \mathbb{R}_i$ is recognized by h_i for $i = 1, 2$.*

Proof. Let $(x, y) \in \mathbb{R}_1 \times \mathbb{R}_2$ be such that $x \cdot y \in L$. We claim that $[x]_{\approx_1} \cdot [y]_{\approx_2} \subseteq L$. Indeed, let $(x', y') \in \mathbb{R}_1 \times \mathbb{R}_2$ be such that $x \sim_1 x'$ and $y \sim_2 y'$. We have $xy \sim_h x'y'$. We prove this when x, y are both infinite. The other cases are easier. We have $x = x_1 x_2 \cdots$, $x' = x'_1 x'_2 \cdots$ with $h_1(x_i) = h_1(x'_i)$ for all $i > 0$. Similarly, $y = y_1 y_2 \cdots$, $y' = y'_1 y'_2 \cdots$ with $h_2(y_i) = h_2(y'_i)$ for all $i > 0$. Since $\Sigma_1 \times \Sigma_2 \subseteq I$, we obtain $xy = (x_1 y_1)(x_2 y_2) \cdots$, $x'y' = (x'_1 y'_1)(x'_2 y'_2) \cdots$ and $h(x_i y_i) = h_1(x_i) h_2(y_i) = h_1(x'_i) h_2(y'_i) = h(x'_i y'_i)$. Therefore, $x'y' \sim_h xy$ and we obtain $x'y' \in L$ which proves the claim. We deduce immediately that

$$L = \bigcup \quad [x]_{\approx_1} \cdot [y]_{\approx_2}$$

where the union ranges over all $(x, y) \in \mathbb{R}_1 \times \mathbb{R}_2$ be such that $x \cdot y \in L$ Note that this union is finite since there are only finitely many equivalence classes for $\approx_1$ and $\approx_2$. Also, the languages $[x]_{\approx_1}$ and $[y]_{\approx_2}$ are clearly recognized by h_1 and h_2. ■

Let $h : \mathbb{M} \to S$ be a morphism to some finite monoid S and let $L \subseteq \mathbb{R}$. For $s \in S$ we define $L(s) = \{x \in \mathbb{R} \mid h^{-1}(s)x \cap L \neq \emptyset\}$.

Proposition 5.3. *If L is recognized by h then $L = \bigcup_{s \in S} h^{-1}(s) \cdot L(s)$ and for each $s \in S$, the language $L(s)$ is recognized by h.*

Proof. Clearly, $L \subseteq h^{-1}(1) \cdot L(1)$. Conversely, let $s \in S$, $x \in h^{-1}(s)$ and $y \in L(s)$. By definition of $L(s)$ there exists some $x' \in h^{-1}(s)$ such that $x'y \in L$. Now clearly $xy \sim_h x'y$ and since L is recognized by h we deduce that $xy \in L$. Therefore, $h^{-1}(s) \cdot L(s) \subseteq L$.

Now, let $y \in L(s)$ and choose $x \in h^{-1}(s)$ such that $xy \in L$. Let $z \in \mathbb{R}$ such that $z \sim_h y$. We have $xz \sim_h xy$ and we deduce that $xz \in L$. Therefore, $z \in L(s)$ which proves that $L(s)$ is recognized by h. ■

In order to prove the main theorem we shall use the equivalence between $\mathrm{FO}(\Sigma, <)$–definability and aperiodic languages.

Theorem 5.1 ([7, 8]). *A real trace language is expressible in $\mathrm{FO}(\Sigma, <)$ if and only if it is aperiodic.*

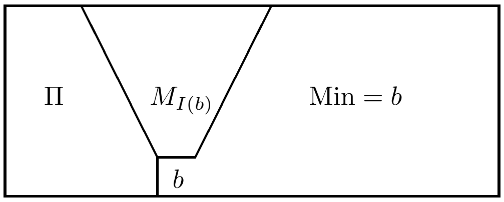
Remark. Due to the theorem above the work of the present paper consists of showing that an aperiodic language is expressible in $\mathrm{LTL}(\Sigma)$. This implies the desired equivalence between LTL-definability, first-order definability, and aperiodic languages. Recently, a direct translation from LTL-definability to aperiodic languages using an inductive construction has been given [5]. Together with Theorem 7.1 we obtain the equivalence between LTL-definability and aperiodic languages without using the above theorem and passing through first order logic.

6. COMPOSITION OF LTL LANGUAGES

We show that some restricted products of expressible languages are also expressible in $\mathrm{LTL}(\Sigma)$. This will be used in Section 7 where we prove by induction that aperiodic languages are expressible in $\mathrm{LTL}(\Sigma)$. For this, we will write an aperiodic language as a finite union of products of simpler languages. Since union corresponds with disjunction, the only problem is indeed with taking products. It turns out that the induction can be based on a restricted use of products, which is considered in this section. Note that an arbitrary product of languages expressible in $\mathrm{LTL}(\Sigma)$ is also expressible in $\mathrm{LTL}(\Sigma)$, but we do not have a direct proof for this general statement. Actually, this becomes a consequence of our main theorem since aperiodic trace languages are closed under product.

We give now the two crucial composition lemmas. Their proofs use some technical lemmas whose statements and proofs are postponed to Section 6.1.

The first composition is visualized by the following picture.



LEMMA 6.1. *Let $b \in \Sigma$, $B = \Sigma \setminus \{b\}$ and $\Pi = \mathbb{M}_B \cap (\mathrm{Max} \subseteq D(b))$. Let $L_1, L_2 \subseteq \mathbb{R}$ and $L_3 \subseteq (\mathrm{Min} = b)$ be trace languages expressible in* LTL(Σ). *Then the language $(L_1 \cap \Pi)(L_2 \cap \mathbb{M}_{I(b)})L_3$ is expressible in* LTL(Σ).

Proof. The product $\Pi(b\mathbb{R})$ is unambiguous hence we have

$$(L_1 \cap \Pi)(L_2 \cap \mathbb{M}_{I(b)})L_3 = ((L_1 \cap \Pi)b\mathbb{R}) \cap (\Pi(L_2 \cap \mathbb{M}_{I(b)})L_3).$$

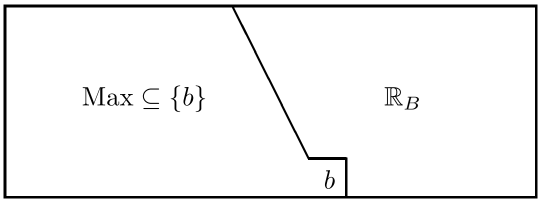
Using Lemma 6.8, the language $(L_1 \cap \Pi)b\mathbb{R}$ is expressible in LTL(Σ).

Now, the product $\mathbb{M}_{I(b)}(\mathrm{Min} = b)$ is also unambiguous hence we have

$$(L_2 \cap \mathbb{M}_{I(b)})L_3 = ((L_2 \cap \mathbb{M}_{I(b)})(\mathrm{Min} = b)) \cap (\mathbb{M}_{I(b)}L_3).$$

By Lemma 6.5 the language $\mathbb{M}_{I(b)}L_3$ is expressible in LTL(Σ) and using Lemma 6.9 the language $(L_2 \cap \mathbb{M}_{I(b)})(\mathrm{Min} = b)$ is also expressible in LTL(Σ). Therefore, $(L_2 \cap \mathbb{M}_{I(b)})L_3$ is expressible in LTL(Σ) and since $(L_2 \cap \mathbb{M}_{I(b)})L_3 \subseteq b\mathbb{R}$ we can apply Lemma 6.4 in order to show that $\Pi(L_2 \cap \mathbb{M}_{I(b)})L_3$ is expressible in LTL(Σ). ■

The second composition is visualized by the following picture.



LEMMA 6.2. *Let $b \in \Sigma$ and $B = \Sigma \setminus \{b\}$. Let $L_1 \subseteq \mathbb{R}$ and $L_2 \subseteq \mathbb{R}_B$ be expressible in* LTL(Σ). *Then the language $(L_1 \cap (\mathrm{Max} \subseteq \{b\}))L_2$ is expressible in* LTL(Σ).

Proof. The product $(\mathrm{Max} \subseteq \{b\})\mathbb{R}_B$ is unambiguous and we have

$$(L_1 \cap (\mathrm{Max} \subseteq \{b\}))L_2 = (L_1 \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B \cap (\mathrm{Max} \subseteq \{b\})L_2.$$

The languages $(L_1 \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B$ and $(\mathrm{Max} \subseteq \{b\})L_2$ are expressible in LTL(Σ) by Lemmas 6.10 and 6.3. ■

6.1. Technical lemmas

This subsection is devoted to the technical lemmas that are used in the two proofs above. Since these technical lemmas are not used afterwards, this subsection may be skipped in a first reading.

LEMMA 6.3. *Let $b \in \Sigma$, $B = \Sigma \setminus \{b\}$ and let $\varphi \in \mathrm{LTL}(\Sigma)$ be a formula such that $\mathrm{L}_{\mathbb{R}}(\varphi) \subseteq \mathbb{R}_B$. Then,*

$$(\mathrm{Max} \subseteq \{b\}) \cdot \mathrm{L}_{\mathbb{R}}(\varphi) = \mathrm{L}_{\mathbb{R}}((\mathbf{F}\, b)\ \mathbf{U}\ \varphi).$$

Proof. Let $z \in (\mathrm{Max} \subseteq \{b\}) \cdot \mathrm{L}_{\mathbb{R}}(\varphi)$ and write $z = xy$ with $x \in (\mathrm{Max} \subseteq \{b\})$ and $y \in \mathrm{L}_{\mathbb{R}}(\varphi)$. Then, for all factorizations $x = x'x''$ with $x'' \neq 1$ we have $\max(x'') = b$, hence $x''y \models \mathbf{F}\, b$. Therefore, $z \in \mathrm{L}_{\mathbb{R}}((\mathbf{F}\, b)\ \mathbf{U}\ \varphi)$.

Conversely, assume that $z \models (\mathbf{F}\, b)\ \mathbf{U}\ \varphi$ and consider a factorization $z = xy$ such that $x \in \mathbb{M}$, $y \models \varphi$ and for all $x = x'x''$ with $x'' \neq 1$ we have $x''y \models \mathbf{F}\, b$. Since $\mathrm{alph}(y) \subseteq B$ and $b \notin B$, this implies $b \in \mathrm{alph}(x'')$. We deduce that $\max(x) \subseteq \{b\}$. ∎

LEMMA 6.4. *Let $b \in \Sigma$, $B = \Sigma \setminus \{b\}$ and $\Pi = \mathbb{M}_B \cap (\mathrm{Max} \subseteq D(b))$. Let $\varphi \in \mathrm{LTL}(\Sigma)$ be a formula such that $\mathrm{L}_{\mathbb{R}}(\varphi) \subseteq b\mathbb{R}$. Then,*

$$\Pi \cdot \mathrm{L}_{\mathbb{R}}(\varphi) = \mathrm{L}_{\mathbb{R}}((\neg b)\ \mathbf{U}\ \varphi).$$

Proof. Let $z \in \Pi \cdot \mathrm{L}_{\mathbb{R}}(\varphi)$ and write $z = xy$ with $x \in \Pi$ and $y \in \mathrm{L}_{\mathbb{R}}(\varphi)$. Then, for all factorizations $x = x'x''$ with $x'' \neq 1$ we have $\min(x''y) = \min(x'') \cup (\min(y) \setminus D(x''))$. Since $x \in \mathbb{M}_B$ and $b \in D(x'')$ we deduce that $x''y \models \neg b$. Therefore, $z \in \mathrm{L}_{\mathbb{R}}((\neg b)\ \mathbf{U}\ \varphi)$.

Conversely, assume that $z \models (\neg b)\ \mathbf{U}\ \varphi$ and consider a factorization $z = xy$ such that $x \in \mathbb{M}$, $y \models \varphi$ and for all $x = x'x''$ with $x'' \neq 1$, we have $x''y \models \neg b$ Let $a \in \mathrm{alph}(x)$ and write $x = x'x''$ with $\min(x'') = \{a\}$. We have $a \in \min(x''y)$ and therefore $a \neq b$, which implies that $x \in \mathbb{M}_B$. Now, let $a \in \max(x)$ and write $x = x'a$. Since $ay \models \neg b$ and $b \in \min(y)$ we must have $a \in D(b)$ and we deduce that $x \in \Pi$. ∎

LEMMA 6.5. *Let $C \subseteq \Sigma$ and let $\varphi \in \mathrm{LTL}(\Sigma)$ be a formula such that $\mathrm{L}_{\mathbb{R}}(\varphi) \subseteq (\mathrm{Min} \subseteq \Sigma \setminus C)$. Then,*

$$\mathbb{M}_C \cdot \mathrm{L}_{\mathbb{R}}(\varphi) = \mathrm{L}_{\mathbb{R}}(C\ \mathbf{U}\ \varphi).$$

Proof. Let $z \in \mathbb{M}_C \cdot \mathrm{L}_{\mathbb{R}}(\varphi)$ and write $z = xy$ with $x \in \mathbb{M}_C$ and $y \in \mathrm{L}_{\mathbb{R}}(\varphi)$. Then, for all factorizations $x = x'x''$ with $x'' \neq 1$ we have $\min(x''y) \cap C \neq \emptyset$, hence $x''y \models C$. Therefore, $z \in \mathrm{L}_{\mathbb{R}}(C\ \mathbf{U}\ \varphi)$.

Conversely, assume that $z \models C\,\mathbf{U}\,\varphi$ and consider a factorization $z = xy$ such that $x \in \mathbb{M}$, $y \models \varphi$ and for all $x = x'x''$ with $x'' \neq 1$, we have $x''y \models C$. Let $c \in \mathrm{alph}(x)$ and write $x = x'x''$ with $\min(x'') = \{c\}$. We have $\emptyset \neq \min(x''y) \cap C = \min(x'') \cap C$ where the last equality holds since $\min(y) \cap C = \emptyset$. Therefore, $c \in C$ which proves that $x \in \mathbb{M}_C$. ∎

For the next two lemmas, we need a new notation. Let $L \subseteq \mathbb{R}$ and $x \in \mathbb{M}$. Then, the left quotient of L by x is the language $x^{-1}L = \{y \in \mathbb{R} \mid xy \in L\}$.

LEMMA 6.6. *Let $a, b \in \Sigma$, $B = \Sigma \setminus \{b\}$ and $\Pi = \mathbb{M}_B \cap (\mathrm{Max} \subseteq D(b))$. Then, the language $(a^{-1}\Pi)b\mathbb{R}$ is expressible in* $\mathrm{LTL}(\Sigma)$.

Proof. First, if $a = b$ then $a^{-1}\Pi = \emptyset$ and therefore $(a^{-1}\Pi)b\mathbb{R} = \mathrm{L}_{\mathbb{R}}(\bot)$. Now, assume that $a \neq b$. We claim that $(a^{-1}\Pi)b\mathbb{R} = \mathrm{L}_{\mathbb{R}}(\psi)$ with

$$\psi = \bigvee_{a - b_1 - \cdots - b_k - b} \neg b \,\mathbf{U}\, (\neg b \wedge \langle b_1 \rangle (\cdots \neg b \,\mathbf{U}\, (\neg b \wedge \langle b_k \rangle (\neg b \,\mathbf{U}\, b)) \cdots))$$

where the disjunction ranges over all simple paths from a to b in the graph of the dependence alphabet (Σ, D). Note that if $a - b$ then we may have $k = 0$ and in this case, $\neg b \,\mathbf{U}\, b$ is part of the disjunction.

Let $z = xby$ with $ax \in \Pi$. Since $\max(ax) \subseteq D(b)$ there is a path in ax from a to some maximal letter of ax which depends on b. More precisely, we find a simple path $a - b_1 - \cdots - b_k - b$ in the graph of the dependence alphabet (Σ, D) and a factorization $x = x_0 b_1 x_1 \cdots b_k x_k$. Note that we may choose $k = 0$ and $x = x_0$ if a is independent of x. Now, $x \in \Pi$ and thus, for all $x = x'x''$ with $x'' \neq 1$, we have $\mathrm{alph}(x'') \subseteq B$ and $b \in D(x'')$. Therefore, $x''by \models \neg b$ and we deduce that $z \models \psi$.

Conversely, assume that $z \models \psi$. Consider a simple path $a - b_1 - \cdots - b_k - b$ in the graph of the dependence alphabet (Σ, D) and a factorization $z = x_0 b_1 x_1 \cdots b_k x_k b y$ such that for all $0 \leq i \leq k$ and $x_i = x'x''$ we have $x'' b_{i+1} \cdots b_k x_k b y \models \neg b$. Then, $x = x_0 b_1 x_1 \cdots b_k x_k \in \mathbb{M}_B$ and if we write $x = x'x''$ with $x' \in \Pi$ and $x'' \in \mathbb{M}_{I(b)}$ then necessarily $\{b_1, \ldots, b_k\} \subseteq \mathrm{alph}(x')$. Now, using that $a \in D(b_1)$ (or $a \in D(b)$ if $k = 0$) we deduce that $ax' \in \Pi$. Therefore, $z = x'bx''y \in (a^{-1}\Pi)b\mathbb{R}$. ∎

LEMMA 6.7. *Let $a, b \in \Sigma$ and $B = \Sigma \setminus \{b\}$. Then, the language $(a^{-1}(\mathrm{Max} = b)) \cdot \mathbb{R}_B$ is expressible in* $\mathrm{LTL}(\Sigma)$.

Proof. We claim that $(a^{-1}(\mathrm{Max} = b)) \cdot \mathbb{R}_B = \mathrm{L}_{\mathbb{R}}(\psi)$ with

$$\psi = \bigvee_{a = b_0 - b_1 - \cdots - b_k = b} \mathbf{F}\langle b_1 \rangle \cdots \mathbf{F}\langle b_k \rangle (\neg \mathbf{F}\, b)$$

where the disjunction ranges over all simple paths from a to b in the graph of the dependence alphabet (Σ, D).

Let $z = xy$ with $x \in \mathbb{M}$, $\max(ax) = \{b\}$ and $y \in \mathbb{R}_B$. As in the proof above, we find a simple path $a = b_0 - b_1 - \cdots - b_k = b$ in the graph of the dependence alphabet (Σ, D) and a factorization $x = x_1 b_1 \cdots x_k b_k$. Note that if $x = 1$ is the empty trace then we have $a = b$ and we choose $k = 0$. Since $\mathrm{L}_{\mathbb{R}}(\neg \mathbf{F}\, b) = \mathbb{R}_B$ we deduce that $z \models \psi$.

Conversely, assume that $z \models \psi$. Consider a path $a = b_0 - b_1 - \cdots - b_k = b$ in the graph of the dependence alphabet (Σ, D) and a factorization $z = x_1 b_1 \cdots x_k b_k y$ with $y \in \mathbb{R}_B$. Let $x = x_1 b_1 \cdots x_k b_k$. We have $b = b_k \in \max(ax)$ hence we can write $ax = ax'x''$ with $\max(ax') = \{b\}$ and $x'' \in \mathbb{M}_{I(b)}$. Therefore, $z = x'(x''y) \in (a^{-1}(\mathrm{Max} = b)) \cdot \mathbb{R}_B$. ∎

LEMMA 6.8. *Let $b \in \Sigma$, $B = \Sigma \setminus \{b\}$ and $\Pi = \mathbb{M}_B \cap (\mathrm{Max} \subseteq D(b))$. Let $\varphi \in \mathrm{LTL}(\Sigma)$. Then the language $(\mathrm{L}_{\mathbb{R}}(\varphi) \cap \Pi)b\mathbb{R}$ is expressible in* $\mathrm{LTL}(\Sigma)$.

Proof. We proceed by structural induction on φ. The cases $\bot$ and $\varphi \vee \psi$ are trivial.
• $\neg\varphi$: Since the product $\Pi b\mathbb{R}$ is unambiguous, we have $(\mathrm{L}_{\mathbb{R}}(\neg\varphi) \cap \Pi)b\mathbb{R} = \Pi b\mathbb{R} \setminus (\mathrm{L}_{\mathbb{R}}(\varphi) \cap \Pi)b\mathbb{R}$. We can conclude by induction since $\Pi b\mathbb{R} = \mathbb{M}b\mathbb{R} = \mathrm{L}_{\mathbb{R}}(\mathbf{F}\, b)$.
• $\langle a\rangle\varphi$: We first claim that

$$(\mathrm{L}_{\mathbb{R}}(\langle a\rangle\varphi) \cap \Pi)b\mathbb{R} = a \cdot \Big((\mathrm{L}_{\mathbb{R}}(\varphi) \cap \Pi)b\mathbb{R} \cap (a^{-1}\Pi)b\mathbb{R}\Big).$$

Indeed, let $x \in (\mathrm{L}_{\mathbb{R}}(\langle a\rangle\varphi) \cap \Pi)b\mathbb{R}$, we can write $x = aybz$ with $y \in \mathrm{L}_{\mathbb{R}}(\varphi)$, $ay \in \Pi$ and $z \in \mathbb{R}$. Then, $y \in a^{-1}\Pi \subseteq \Pi$ and we deduce that $ybz \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap \Pi)b\mathbb{R} \cap (a^{-1}\Pi)b\mathbb{R}$.

Conversely, let $x \in a \cdot ((\mathrm{L}_{\mathbb{R}}(\varphi) \cap \Pi)b\mathbb{R} \cap (a^{-1}\Pi)b\mathbb{R})$. We can write $x = aybz$ with $y \in a^{-1}\Pi$, $z \in \mathbb{R}$ and $ybz \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap \Pi)b\mathbb{R}$. Then, $y \in \Pi$ and since the product $\Pi b\mathbb{R}$ is unambiguous, we deduce that $y \in \mathrm{L}_{\mathbb{R}}(\varphi)$. Therefore, $ay \in (\mathrm{L}_{\mathbb{R}}(\langle a\rangle\varphi) \cap \Pi)$ which proves the claim.

By Lemma 6.6, the language $(a^{-1}\Pi)b\mathbb{R}$ is expressible in $\mathrm{LTL}(\Sigma)$, hence we can conclude by induction.
• $\varphi\,\mathbf{U}\,\psi$: Here, we claim that

$$(\mathrm{L}_{\mathbb{R}}(\varphi\,\mathbf{U}\,\psi) \cap \Pi)b\mathbb{R} = ((\mathrm{L}_{\mathbb{R}}(\varphi) \cap \Pi)b\mathbb{R} \setminus b\mathbb{R})\;\mathbf{U}\;(\mathrm{L}_{\mathbb{R}}(\psi) \cap \Pi)b\mathbb{R}.$$

First, let $z \in (\mathrm{L}_{\mathbb{R}}(\varphi\,\mathbf{U}\,\psi) \cap \Pi)b\mathbb{R}$. We write $z = xby$ with $x \in \mathrm{L}_{\mathbb{R}}(\varphi\,\mathbf{U}\,\psi) \cap \Pi$ and $y \in \mathbb{R}$. Then, $x = x_1x_2$ with $x_2 \models \psi$ and for all factorizations $x_1 = x'x''$ with $x'' \neq 1$, we have $x''x_2 \models \varphi$. Since $x'' \neq 1$ and $x''x_2 \in \Pi$ we have $x''x_2by \notin b\mathbb{R}$ and we obtain $x''x_2by \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap \Pi)b\mathbb{R} \setminus b\mathbb{R}$. Also, $x_2by \in (\mathrm{L}_{\mathbb{R}}(\psi) \cap \Pi)b\mathbb{R}$ which proves the first inclusion.

Conversely, let z be in the right hand side and write $z = z_1z_2$ with $z_2 \in (\mathrm{L}_{\mathbb{R}}(\psi) \cap \Pi)b\mathbb{R}$ and for all $z_1 = z'z''$ with $z'' \neq 1$ we have $z''z_2 \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap \Pi)b\mathbb{R} \setminus b\mathbb{R}$. Then clearly, $b \notin \mathrm{alph}(z_1)$. We write $z_2 = x_2by_2$ with $x_2 \in \mathrm{L}_{\mathbb{R}}(\psi) \cap \Pi$. We can also write $z_1 = x_1y_1$ with y_1 independent of x_2b and $x = x_1x_2 \in \Pi$. Therefore, we obtain $z = x_1y_1x_2by_2 = (x_1x_2)b(y_1y_2)$. Now, for all factorizations $x_1 = x'x''$ with $x'' \neq 1$, we have $z_1 = x'(x''y_1)$ and we obtain $(x''x_2)b(y_1y_2) = x''y_1z_2 \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap \Pi)b\mathbb{R}$. Since the product $\Pi b\mathbb{R}$ is unambiguous and $x''x_2 \in \Pi$, we deduce that $x''x_2 \in \mathrm{L}_{\mathbb{R}}(\varphi)$. Therefore, $x_1x_2 \in \mathrm{L}_{\mathbb{R}}(\varphi\,\mathbf{U}\,\psi) \cap \Pi$ which proves the claim.

We can conclude by induction since we have $b\mathbb{R} = \mathrm{L}_{\mathbb{R}}(b)$. ■

Lemma 6.9. *Let $b \in \Sigma$ and $C \subseteq \Sigma$ such that $C \times \{b\} \subseteq I$. Let $\varphi \in \mathrm{LTL}(\Sigma)$. Then the language $(\mathrm{L}_{\mathbb{R}}(\varphi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b)$ is expressible in $\mathrm{LTL}(\Sigma)$.*

Proof. Again, we proceed by structural induction on φ. The cases $\bot$ and $\varphi \vee \psi$ are trivial.
• $\neg\varphi$: Since the product $\mathbb{M}_C \cdot (\mathrm{Min} = b)$ is unambiguous, we have

$$\begin{aligned}(\mathrm{L}_{\mathbb{R}}(\neg\varphi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b) &= (\mathbb{M}_C \setminus \mathrm{L}_{\mathbb{R}}(\varphi)) \cdot (\mathrm{Min} = b)\\ &= \Big(\mathbb{M}_C \cdot (\mathrm{Min} = b)\Big) \setminus \Big((\mathrm{L}_{\mathbb{R}}(\varphi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b)\Big),\end{aligned}$$

and we can conclude by induction using Lemma 6.5 since the language $(\mathrm{Min} = b)$ is expressible in $\mathrm{LTL}(\Sigma)$.

• $\langle a\rangle\varphi$: Easy, since if $a \notin C$ we have $(\mathrm{L}_{\mathbb{R}}(\langle a\rangle\varphi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b) = \emptyset$ and if $a \in C$ we have $(\mathrm{L}_{\mathbb{R}}(\langle a\rangle\varphi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b) = a \cdot (\mathrm{L}_{\mathbb{R}}(\varphi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b)$.
• $\varphi \mathbin{\mathbf{U}} \psi$: This case follows directly by induction from the equality

$$(\mathrm{L}_{\mathbb{R}}(\varphi \mathbin{\mathbf{U}} \psi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b) = \\ \Big((\mathrm{L}_{\mathbb{R}}(\varphi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b)\Big) \mathbin{\mathbf{U}} \Big((\mathrm{L}_{\mathbb{R}}(\psi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b)\Big).$$

First, let $z = xy$ with $x \in \mathrm{L}_{\mathbb{R}}(\varphi \mathbin{\mathbf{U}} \psi) \cap \mathbb{M}_C$ and $\min(y) = \{b\}$. Write $x = x_1x_2$ with $x_2 \models \psi$ and for all factorizations $x_1 = x'x''$ with $x'' \neq 1$, we have $x''x_2 \models \varphi$. Then, $x_2y \in (\mathrm{L}_{\mathbb{R}}(\psi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b)$ and $x''x_2y \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b)$.

Conversely, let z be in the right hand side and write $z = z_1z_2$ with $z_2 \in (\mathrm{L}_{\mathbb{R}}(\psi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b)$ and for all $z_1 = z'z''$ with $z'' \neq 1$ we have $z''z_2 \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b)$. Write $z_2 = x_2y_2$ with $x_2 \in \mathrm{L}_{\mathbb{R}}(\psi) \cap \mathbb{M}_C$ and $y_2 \in (\mathrm{Min} = b)$. Let $a \in \mathrm{alph}(z_1)$. Then we can write $z_1 = z'z''$ with $\min(z'') = \{a\}$. From $z''z_2 \in \mathbb{M}_C \cdot (\mathrm{Min} = b)$ we deduce that $a \in \min(z''z_2) \subseteq C \cup \{b\}$. Hence, $\mathrm{alph}(z_1) \subseteq C \cup \{b\}$ and we can write $z = x_1y_1$ with $x_1 \in \mathbb{M}_C$ and $\mathrm{alph}(y_1) \subseteq \{b\}$. Now, $y = y_1y_2 \in (\mathrm{Min} = b)$ and for all factorizations $x_1 = x'x''$ with $x'' \neq 1$, we have $(x''x_2)(y_1y_2) = x''y_1z_2 \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap \mathbb{M}_C) \cdot (\mathrm{Min} = b)$. Since the product $\mathbb{M}_C \cdot (\mathrm{Min} = b)$ is unambiguous, we deduce that $x''x_2 \models \varphi$. Therefore, $x = x_1x_2 \models \varphi \mathbin{\mathbf{U}} \psi$ and we are done. ■

LEMMA 6.10. *Let $b \in \Sigma$, $B = \Sigma \setminus \{b\}$ and let $\varphi \in \mathrm{LTL}(\Sigma)$. Then the language $(\mathrm{L}_{\mathbb{R}}(\varphi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B$ is expressible in* $\mathrm{LTL}(\Sigma)$.

Proof. We use a structural induction on φ. The cases $\bot$ and $\varphi \vee \psi$ are trivial.
• $\neg\varphi$: Since the product $(\mathrm{Max} \subseteq \{b\})\mathbb{R}_B$ is unambiguous, we have

$$(\mathrm{L}_{\mathbb{R}}(\neg\varphi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B = (\mathrm{Max} \subseteq \{b\})\mathbb{R}_B \setminus (\mathrm{L}_{\mathbb{R}}(\varphi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B.$$

Moreover $(\mathrm{Max} \subseteq \{b\})\mathbb{R}_B$ is the set $(\mathrm{Alphinf} \subseteq B)$, which is expressible in $\mathrm{LTL}(\Sigma)$ by the formula $\mathbf{F}\,\mathbf{G}\,\neg b$.
• $\langle a\rangle\varphi$: We first claim that

$$(\mathrm{L}_{\mathbb{R}}(\langle a\rangle\varphi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B = \\ a\big((\mathrm{L}_{\mathbb{R}}(\varphi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B \cap (a^{-1}(\mathrm{Max} = b))\mathbb{R}_B\big)$$

Indeed, let $z = axy$ with $ax \in \mathrm{L}_{\mathbb{R}}(\langle a\rangle\varphi) \cap (\mathrm{Max} \subseteq \{b\})$ and $y \in \mathbb{R}_B$. Then, $x \in a^{-1}(\mathrm{Max} = b) \subseteq (\mathrm{Max} \subseteq \{b\})$ and also $x \in \mathrm{L}_{\mathbb{R}}(\varphi)$. Hence, z belongs to the right hand side.

Conversely, let z be in the right hand side and write $z = axy$ with $x \in a^{-1}(\mathrm{Max} = b)$, $y \in \mathbb{R}_B$ and $xy \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B$. Since the product $(\mathrm{Max} \subseteq \{b\})\mathbb{R}_B$ is unambiguous, we deduce that $x \in \mathrm{L}_{\mathbb{R}}(\varphi)$. Hence, $ax \in \mathrm{L}_{\mathbb{R}}(\langle a\rangle\varphi) \cap (\mathrm{Max} \subseteq \{b\})$ and we are done.

By Lemma 6.7, the language $(a^{-1}(\mathrm{Max} = b))\mathbb{R}_B$ is expressible in $\mathrm{LTL}(\Sigma)$, hence we can conclude by induction.
• $\varphi \mathbin{\mathbf{U}} \psi$: This case follows by induction from the formula

$$(\mathrm{L}_{\mathbb{R}}(\varphi \mathbin{\mathbf{U}} \psi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B = \\ \big(\mathrm{L}_{\mathbb{R}}(\varphi) \cap (\mathrm{Max} \subseteq \{b\})\big)\mathbb{R}_B \mathbin{\mathbf{U}} \big(\mathrm{L}_{\mathbb{R}}(\psi) \cap (\mathrm{Max} \subseteq \{b\})\big)\mathbb{R}_B.$$

First, let $z = xy$ with $x \in \mathrm{L}_{\mathbb{R}}(\varphi\, \mathbf{U}\, \psi) \cap (\mathrm{Max} \subseteq \{b\})$ and $y \in \mathbb{R}_B$. Then, $x = x_1 x_2$ with $x_2 \models \psi$ and for all factorizations $x_1 = x'x''$ with $x'' \neq 1$, we have $x''x_2 \models \varphi$. Since each suffix of x is in $(\mathrm{Max} \subseteq \{b\})$, we deduce $x_2 y \in (\mathrm{L}_{\mathbb{R}}(\psi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B$ and $x''x_2 y \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B$ which proves the first inclusion.

Conversely, let z be in the right hand side and consider a factorization $z = z_1 z_2$ such that $z_2 \in (\mathrm{L}_{\mathbb{R}}(\psi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B$ and for all $z_1 = z'z''$ with $z'' \neq 1$ we have $z''z_2 \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B$. We write $z_2 = x_2 y_2$ with $y_2 \in \mathbb{R}_B$ and $x_2 \in \mathrm{L}_{\mathbb{R}}(\psi) \cap (\mathrm{Max} \subseteq \{b\})$. We can also write $z_1 = x_1 y_1$ with y_1 independent of $x_2 b$ and $\max(x_1 x_2) \subseteq \{b\}$. Note that $y_1 \in \mathbb{R}_B$. We obtain $z = x_1 y_1 x_2 y_2 = (x_1 x_2)(y_1 y_2)$. Now, for all factorizations $x_1 = x'x''$ with $x'' \neq 1$, we have $z_1 = x'(x''y_1)$ and we obtain $(x''x_2)(y_1 y_2) = x''y_1 z_2 \in (\mathrm{L}_{\mathbb{R}}(\varphi) \cap (\mathrm{Max} \subseteq \{b\}))\mathbb{R}_B$. Since the product $(\mathrm{Max} \subseteq \{b\})\mathbb{R}_B$ is unambiguous, we deduce that $x''x_2 \in \mathrm{L}_{\mathbb{R}}(\varphi)$. Therefore, $x_1 x_2 \in \mathrm{L}_{\mathbb{R}}(\varphi\, \mathbf{U}\, \psi) \cap (\mathrm{Max} \subseteq \{b\})$. ∎

7. KAMP'S THEOREM FOR REAL TRACES

THEOREM 7.1. *A real trace language is expressible in* $\mathrm{FO}(\Sigma, <)$ *if and only if it is expressible in* $\mathrm{LTL}(\Sigma)$.

By Proposition 4.1 and Theorem 5.1 it is enough to show that all aperiodic languages in $\mathbb{R}(\Sigma, I)$ are expressible in $\mathrm{LTL}(\Sigma)$.

Let Q be a finite set of states. We denote by $\mathrm{Trans}(Q)$ the monoid of mappings from Q to Q. The multiplication is the composition (in reverse order) of mappings: $(fg)(x) = g(f(x))$; and the unit element is the identity id_Q. We will use the fact that every finite monoid S can be realized as a submonoid of some $\mathrm{Trans}(Q)$ where $|Q| \leq |S|$. Indeed, it suffices to consider the right action of S over itself. More precisely, if we define $\chi(s) \in \mathrm{Trans}(S)$ by $\chi(s)(t) = ts$ then it is easy to see that $\chi : S \to \mathrm{Trans}(S)$ is an injective morphism.

We deduce that every aperiodic trace language can be recognized by some morphism $h : \mathbb{M}(\Sigma, I) \to S \subseteq \mathrm{Trans}(Q)$ where S is aperiodic. We show by induction on $(|Q|, |\Sigma|)$ that all languages recognized by h are expressible in $\mathrm{LTL}(\Sigma)$. For this induction, we use the following well-founded lexicographic order on $\mathbb{N}^2$: $(m, n) < (m', n')$ if and only if $m < m'$ or $m = m'$ and $n < n'$.

First, assume that $h(a) = \mathrm{id}_Q$ for all $a \in \Sigma$, which is in particular the case when $|Q| = 1$. If $L \subseteq \mathbb{R}$ is recognized by h then L is one of the sets $\emptyset$, $\mathbb{R}$, $\mathbb{M}$ or $\mathbb{R} \setminus \mathbb{M}$ which are respectively defined by the formulas $\bot$, $\top$, $\mathbf{F}\,\mathrm{stop}$, and $\mathbf{G}\,\mathbf{X}\,\top$. This shows the basis case of the induction.

Second, assume that $h(b) \neq \mathrm{id}_Q$ for some $b \in \Sigma$. The crucial observation here is that $h(b)$ is not a permutation of Q. Indeed, since S is aperiodic, there exists some n such that $h(b)^n = h(b)^{n+1}$. If $h(b)$ were a permutation then it would be invertible and the last equality would imply $h(b) = \mathrm{id}_Q$, a contradiction. Hence, $h(b)(Q) = Q'$ for some $Q' \subset Q$ with $|Q'| < |Q|$.

We let $B = \Sigma \setminus \{b\}$ and we define two subsets of $\mathbb{M}$:

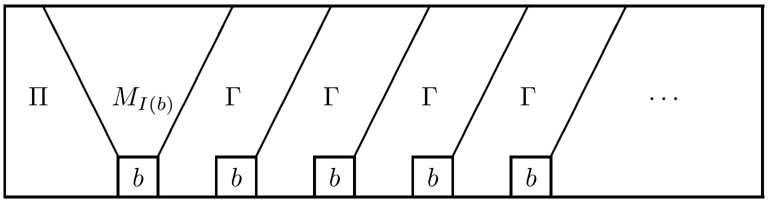
$$\begin{aligned} \Pi &= \{x \in \mathbb{M}_B \mid \max(x) \subseteq D(b)\}, \\ \Gamma &= \{x \in \mathbb{M}_B \mid \min(x) \subseteq D(b)\}. \end{aligned}$$

The notation Π is chosen since Πb are exactly the *pyramids* of $\mathbb{M}$ where the unique maximal element is b. It should be noted that $(\Pi b)^*$ and $(\Gamma b)^*$ are free submonoids of $\mathbb{M}$, both being infinitely generated if $D(b) \neq \{b\}$.

By Δ we denote the subset of real traces x which are either in $\mathbb{M}b$ or which can be factorized into an infinite product of *finite* traces such that all factors except the first belong to Γb, that is, $\Delta = \mathbb{M}b(\Gamma b)^\infty$. The set Δ, which plays a key-role, admits the following unambiguous decomposition:

$$\Delta = \Pi \mathbb{M}_{I(b)} b (\Gamma b)^\infty.$$

This decomposition is best visualized by the following picture; it is in some sense the guide for the modular construction of a formula defining the language $L \cap \Delta$.



The core of the proof is now the following proposition.

Proposition 7.1. *Let $L \subseteq \mathbb{R}$ be recognized by h. Then, $L \cap \Delta$ is expressible in* LTL(Σ).

The proof of this proposition will be given later in Section 7.1. Here we show how it is used in the proof of Theorem 7.1. We start with two corollaries.

Corollary 7.1. *Assume that (Σ, D) is connected and let $L \subseteq \mathbb{R}$ be recognized by h. Then the language $L \cap (\mathrm{Alphinf} = \Sigma)$ is expressible in* LTL(Σ).

Proof. Since (Σ, D) is connected, we have $(\mathrm{Alphinf} = \Sigma) \subseteq \Delta$. Therefore,

$$L \cap (\mathrm{Alphinf} = \Sigma) = (L \cap \Delta) \cap (\mathrm{Alphinf} = \Sigma).$$

By Proposition 7.1 we know that $L \cap \Delta$ is expressible in LTL(Σ) and we conclude easily since $(\mathrm{Alphinf} = \Sigma) = \mathrm{L}_{\mathbb{R}}(\wedge_{a \in \Sigma} \mathbf{G}\,\mathbf{F}\,a)$. ■

Corollary 7.2. *Let $c \in \Sigma$, $C = \Sigma \setminus \{c\}$ and let $L \subseteq \mathbb{R}$ be recognized by h. Then the language $L \cap (\mathrm{Alphinf} \subseteq C)$ is expressible in* LTL(Σ).

Proof. We claim that $L \cap (\mathrm{Alphinf} \subseteq C)$ is the finite union

$$\bigcup_{u,v \in S} \Big(\big((h^{-1}(u) \cap (\mathrm{Max} \subseteq \{b\})) \cdot (h^{-1}(v) \cap \mathbb{M}_B) \big) \cap (\mathrm{Max} \subseteq \{c\}) \Big) \cdot (L(uv) \cap \mathbb{R}_C)$$

First, let $t = xyz$ with $x \in h^{-1}(u)$, $y \in h^{-1}(v)$ and $z \in L(uv)$ for some $u, v \in S$. Then $h(xy) = uv$ and we deduce from Proposition 5.3 that $t = xyz \in L$.

Conversely, we have the unambiguous decompositions

$$(\text{Alphinf} \subseteq C) = (\text{Max} \subseteq \{c\}) \cdot \mathbb{R}_C \qquad \text{and} \qquad \mathbb{M} = (\text{Max} \subseteq \{b\}) \cdot \mathbb{M}_B.$$

Therefore,

$$(\text{Alphinf} \subseteq C) = \Big(\big((\text{Max} \subseteq \{b\}) \cdot \mathbb{M}_B\big) \cap (\text{Max} \subseteq \{c\})\Big) \cdot \mathbb{R}_C.$$

Hence, if $t \in L \cap (\text{Alphinf} \subseteq C)$, we can write $t = xyz$ with $x \in (\text{Max} \subseteq \{b\})$, $y \in \mathbb{M}_B$, $xy \in (\text{Max} \subseteq \{c\})$ and $z \in \mathbb{R}_C$. If we let $u = h(x)$ and $v = h(y)$ then we get $z \in L(uv)$ which concludes the proof of the claim.

We will now apply Lemma 6.2 twice in order to conclude the proof of Corollary 7.2. We fix some $u, v \in S$. First, by Propositions 5.1 and 5.3 we know that $K_2 = L(uv) \cap \mathbb{R}_C$ is recognized by the morphism $h{\restriction_{\mathbb{M}_C}}$ and $L_2 = h^{-1}(v) \cap \mathbb{M}_B$ is recognized by the morphism $h{\restriction_{\mathbb{M}_B}}$. By induction on the size of the alphabet, we deduce that L_2 and K_2 are expressible in LTL(B) and LTL(C) respectively. By Lemma 3.1 they are also expressible in LTL(Σ).

Second, note that $(\text{Max} \subseteq \{b\}) \subseteq \Delta \cup \{1\}$. Hence,

$$h^{-1}(u) \cap (\text{Max} \subseteq \{b\}) = (h^{-1}(u) \cap (\Delta \cup \{1\})) \cap (\text{Max} \subseteq \{b\}).$$

By Proposition 7.1, the language $L_1 = h^{-1}(u) \cap (\Delta \cup \{1\})$ is expressible in LTL(Σ). Applying once Lemma 6.2 we deduce first that

$$K_1 = (L_1 \cap (\text{Max} \subseteq \{b\})) \cdot L_2 = (h^{-1}(u) \cap (\text{Max} \subseteq \{b\})) \cdot (h^{-1}(v) \cap \mathbb{M}_B)$$

is expressible in LTL(Σ). Next, applying a second time Lemma 6.2 we deduce that $(K_1 \cap (\text{Max} \subseteq \{c\})) \cdot K_2$, that is,

$$\Big(\Big((h^{-1}(u) \cap (\text{Max} \subseteq \{b\})) \cdot (h^{-1}(v) \cap \mathbb{M}_B)\Big) \cap (\text{Max} \subseteq \{c\})\Big) \cdot (L(uv) \cap \mathbb{R}_C)$$

is expressible in LTL(Σ). Therefore, $L \cap (\text{Alphinf} \subseteq C)$ is expressible in LTL(Σ). ∎

In order to conclude the proof of Theorem 7.1, we use the following proposition.

PROPOSITION 7.2. *Assume that* (Σ, D) *is non-connected and let* $L \subseteq \mathbb{R}$ *be recognized by* h*. Then* L *is expressible in* LTL(Σ).

Proof. Assume that $\Sigma = \Sigma_1 \cup \Sigma_2$ with $\Sigma_1 \times \Sigma_2 \subseteq I$ and let $\mathbb{M}_i = \mathbb{M}_{\Sigma_i}$, $\mathbb{R}_i = \mathbb{R}_{\Sigma_i}$ and $h_i = h{\restriction_{M_i}}$ for $i = 1, 2$. By Proposition 5.2, L is a finite union of products $L_1 \cdot L_2$ where $L_i \subseteq \mathbb{R}_i$ is recognized by h_i for $i = 1, 2$. By induction on the size of the alphabet, we deduce that L_1 and L_2 are expressible in LTL(Σ_1) and LTL(Σ_2) respectively. Using Lemma 3.2, we deduce that $L_1 \cdot L_2$ is expressible in LTL(Σ). ∎

7.1. Proof of Proposition 7.1

Let us first show that the set $(\Gamma b)^\infty$ is expressible in LTL(Σ).

Lemma 7.1. *The real trace language* $(\Gamma b)^\infty$ *is expressed by the formula*

$$(\mathrm{Min} \subseteq D(b)) \wedge \Big(\mathrm{stop} \vee \mathbf{F}\langle b\rangle \mathrm{stop} \vee \mathbf{G}\,\mathbf{F}(\mathrm{Min} = b)\Big)$$

where we write simply $(\mathrm{Min} \subseteq D(b))$ *and* $(\mathrm{Min} = b)$ *instead of the corresponding* $\mathrm{LTL}(\Sigma)$ *formula.*

Proof. We have $(\Gamma b)^+ = (\mathrm{Min} \subseteq D(b)) \cap \mathbb{M}b$, therefore $(\Gamma b)^*$ is expressed by the formula $(\mathrm{Min} \subseteq D(b)) \wedge (\mathrm{stop} \vee \mathbf{F}\langle b\rangle\mathrm{stop})$.

Now, if $x \in (\Gamma b)^\omega$ then clearly $\min(x) \subseteq D(b)$ and $x \models \mathbf{G}\,\mathbf{F}(\mathrm{Min} = b)$. Conversely, assume that $\min(x) \subseteq D(b)$ and $x \models \mathbf{G}\,\mathbf{F}(\mathrm{Min} = b)$. Consider any infinite factorization $x = z_0 z_1 \cdots$ of x into *finite* traces. Since $x \models \mathbf{G}\,\mathbf{F}(\mathrm{Min} = b)$ and z_0 is a finite prefix of x we can write $x = y_0 b x_1$ with $z_0 \le y_0 \in \mathbb{M}$ and $\min(x_1) \subseteq D(b)$. From $\min(x) \subseteq D(b)$, we deduce that $y_0 b \in (\mathrm{Min} \subseteq D(b)) \cap \mathbb{M}b = (\Gamma b)^+$.

Now, $z_0 z_1$ and $y_0 b$ are finite prefixes of x, hence we can write $x = y_0 b y_1 b x_2$ with $z_0 z_1 \le y_0 b y_1 \in \mathbb{M}$ and $\min(x_2) \subseteq D(b)$. From $\min(x_1) \subseteq D(b)$, we deduce that $y_1 b \in (\mathrm{Min} \subseteq D(b)) \cap \mathbb{M}b = (\Gamma b)^+$.

In this way, we construct an infinite sequence of finite traces y_i such that for all i we have $y_i b \in (\Gamma b)^+$ and $z_0 \cdots z_i \le y_0 b \cdots b y_i \le x$. We deduce that $x = y_0 b y_1 b \cdots \in (\Gamma b)^\omega$. ∎

Now, recall that $h(b)(Q) = Q'$. Each $s \in h((\Gamma b)^*)$ maps the subset Q' to Q'. Hence we may define two subsets $T, T' \subseteq \mathrm{Trans}(Q')$ by $T = \{s\restriction_{Q'} \mid s \in h(\Gamma b)\}$ and $T' = \{s\restriction_{Q'} \mid s \in h((\Gamma b)^*)\}$. Since $h((\Gamma b)^*)$ is a submonoid of S, the set T' is a monoid. Moreover, the monoid T' is generated by T and is aperiodic since S is aperiodic.

By T^* we denote the free monoid generated by the finite set T (here T is viewed as an alphabet). Accordingly, T^∞ means the set of finite or infinite words over the alphabet T. The inclusion $T \subseteq T'$ induces a canonical morphism $e : T^* \to T'$ which is called the *evaluation*.

The mapping $\sigma : \Gamma b \to T$, defined by the restriction $\sigma(x) = h(x)\restriction_{Q'}$, induces a morphism $\sigma : (\Gamma b)^* \to T^*$ between free monoids. The mapping σ is also extended to infinite sequences $\sigma : (\Gamma b)^\omega \to T^\omega$, so finally we have $\sigma : (\Gamma b)^\infty \to T^\infty$.

Since T' is a submonoid of $\mathrm{Trans}(Q')$ and $|Q'| < |Q|$, we may use induction (Although we might have $|T| > |\Sigma|$). More precisely, we may assume that every language $K \subseteq T^\infty$ which is recognized by the morphism e is expressible in $\mathrm{LTL}(T)$. The following lemma allows us to make use of this induction step.

Lemma 7.2. *Let* $K \subseteq T^\infty$ *be a word language expressible in* $\mathrm{LTL}(T)$*. Then the real trace language* $\sigma^{-1}(K) \subseteq (\Gamma b)^\infty$ *is expressible in* $\mathrm{LTL}(\Sigma)$*.*

Proof. We prove this by induction on the formula $\varphi \in \mathrm{LTL}(T)$ which defines the language K. Let us denote in this proof by $\sigma^{-1}(\varphi)$ the language $\sigma^{-1}(\mathrm{L}_{T^\infty}(\varphi))$. The cases $\perp$ and $\varphi \vee \psi$ are trivial.

• $\neg\varphi$: We have $\sigma^{-1}(\neg\varphi) = (\Gamma b)^\infty \setminus \sigma^{-1}(\varphi)$. We conclude by induction using Lemma 7.1.

• $\langle t \rangle \varphi$: We first claim that

$$\sigma^{-1}(\langle t \rangle \varphi) = (\mathrm{Min} \subseteq D(b)) \cap \left(\bigcup_{t=(uvh(b))\restriction_{Q'}} (h^{-1}(u) \cap \Pi)(h^{-1}(v) \cap \mathbb{M}_{I(b)}) b \sigma^{-1}(\varphi) \right).$$

This is based on the unambiguous decomposition

$$(\Gamma b)(\Gamma b)^\infty = (\mathrm{Min} \subseteq D(b)) \cap \Pi \mathbb{M}_{I(b)} b (\Gamma b)^\infty.$$

Let $x \in \sigma^{-1}(\langle t \rangle \varphi)$. Write $x = yz$ with $y \in \sigma^{-1}(t) \subseteq (\Gamma b)$ and $z \in \sigma^{-1}(\varphi) \subseteq (\Gamma b)^\infty$. Using the unambiguous decomposition above we deduce that $y = y' y'' b$ with $y' \in \Pi$ and $y'' \in \mathbb{M}_{I(b)}$. Let $u = h(y')$ and $v = h(y'')$. We obtain $t = \sigma(y) = (uvh(b))\restriction_{Q'}$ and $x = yz \in (\mathrm{Min} \subseteq D(b)) \cap ((h^{-1}(u) \cap \Pi)(h^{-1}(v) \cap \mathbb{M}_{I(b)}) b \sigma^{-1}(\varphi))$.

Conversely, let $x = y' y'' b z \in (\mathrm{Min} \subseteq D(b))$ with $y' \in h^{-1}(u) \cap \Pi$, $y'' \in h^{-1}(v) \cap \mathbb{M}_{I(b)}$ and $z \in \sigma^{-1}(\varphi)$ for some $u, v \in S$ such that $t = (uvh(b))\restriction_{Q'}$. We deduce that $y = y' y'' b \in \Gamma b$ and $\sigma(y) = t$. Therefore, $x = yz \in \sigma^{-1}(\langle t \rangle \varphi)$.

By induction, the language $\sigma^{-1}(\varphi)$ is expressible in LTL(Σ). Hence, so is $L_3 = b\sigma^{-1}(\varphi) \subseteq (\mathrm{Min} = b)$. By Proposition 5.1, the languages $L_1 = h^{-1}(u) \cap \mathbb{M}_B$ and $L_2 = h^{-1}(v) \cap \mathbb{M}_B$ are recognized by the restriction of h to $\mathbb{M}_B$. Hence, by induction on the size of the alphabet, they are expressible in LTL(B), and also in LTL(Σ) by Lemma 3.1. Since $\Pi \cup \mathbb{M}_{I(b)} \subseteq \mathbb{M}_B$, we have $h^{-1}(u) \cap \Pi = L_1 \cap \Pi$ and $h^{-1}(v) \cap \mathbb{M}_{I(b)} = L_2 \cap \mathbb{M}_{I(b)}$. Using Lemma 6.1, we deduce that

$$(L_1 \cap \Pi)(L_2 \cap \mathbb{M}_{I(b)}) L_3 = (h^{-1}(u) \cap \Pi)(h^{-1}(v) \cap \mathbb{M}_{I(b)}) b \sigma^{-1}(\varphi)$$

is expressible in LTL(Σ).

• $\varphi \, \mathbf{U} \, \psi$: An until–formula $\varphi \, \mathbf{U} \, \psi$ over words is equivalent with $\psi \vee (\varphi \wedge \mathbf{X}(\varphi \, \mathbf{U} \, \psi))$. Thus it is enough to consider $\mathbf{X}(\varphi \, \mathbf{U} \, \psi)$. We claim that

$$\sigma^{-1}(\mathbf{X}(\varphi \, \mathbf{U} \, \psi)) = (\Gamma b)^\infty \cap \Big(\big(b\sigma^{-1}(\varphi) \cup (\mathrm{Min} \neq b)\big) \, \mathbf{U} \, \big(b\sigma^{-1}(\psi)\big) \Big).$$

Let $x \in \sigma^{-1}(t)$ for some $t = t_0 t_1 t_2 \cdots \in T^\infty$ with $t \models \mathbf{X}(\varphi \, \mathbf{U} \, \psi)$. We write $x = x_0 b x_1 b x_2 b \cdots$ with $x_i \in \Gamma$ and $\sigma(x_i b) = t_i$. Let $i > 0$ be such that $t_i t_{i+1} \cdots \models \psi$ and for all $0 < j < i$, $t_j t_{j+1} \cdots \models \varphi$. We deduce that $z = b x_i b x_{i+1} \cdots \in b\sigma^{-1}(\psi)$ and if $y = x_0 b x_1 \cdots b x_{i-1} = uv$ with $v \neq 1$ then either $\min(v) \neq \{b\}$ and also $\min(vz) \neq \{b\}$, or $\min(v) = \{b\}$ and we have $v = b x_j \cdots b x_{i-1}$ for some $0 < j < i$. In this case, we have $vz = b x_j b x_{j+1} \cdots \in b\sigma^{-1}(\varphi)$. Therefore, x belongs to the right hand side.

Conversely, assume that x belongs to the right hand side, then we can write $x = x_0 b x_1 b x_2 b \cdots$ with $x_i \in \Gamma$. Let $t = \sigma(x) = t_0 t_1 t_2 \cdots$ with $t_i = \sigma(x_i b)$. Let $x = yz$ be a factorization such that $z \in b\sigma^{-1}(\psi)$ and for all $y = uv$ with $v \neq 1$ we have either $\min(vz) \neq \{b\}$ or $vz \in b\sigma^{-1}(\varphi)$. We deduce that $z = b x_i b x_{i+1} \cdots$ for some $i > 0$ and $t_i t_{i+1} \cdots \models \psi$. Now, for all $0 < j < i$ we have $\min(b x_j b x_{j+1} \cdots) = \{b\}$ hence $b x_j b x_{j+1} \cdots \in b\sigma^{-1}(\varphi)$ and $t_j t_{j+1} \cdots \models \psi$. Therefore, $t \models \mathbf{X}(\varphi \, \mathbf{U} \, \psi)$, which concludes the proof of the claim.

By induction, we know that $\sigma^{-1}(\varphi)$ and $\sigma^{-1}(\psi)$ are expressible in LTL(Σ), hence also $b\sigma^{-1}(\varphi)$ and $b\sigma^{-1}(\psi)$. Since both $(\Gamma b)^\infty$ (Lemma 7.1) and $(\mathrm{Min} \neq \{b\})$ are expressible in LTL(Σ), we deduce from the claim that $\sigma^{-1}(\mathbf{X}(\varphi\mathbf{U}\psi))$ is also expressible in LTL(Σ). ■

LEMMA 7.3. *Let $L \subseteq \mathbb{R}$ be recognized by h. Then $L \cap b(\Gamma b)^\infty$ is expressible in* LTL(Σ).

Proof. We define the language $K \subseteq T^\infty$ with respect to the language L by

$$K = \{\sigma(x) \in T^\infty \mid bx \in L \cap b(\Gamma b)^\infty\}.$$

We first show that $L \cap b(\Gamma b)^\infty = b\sigma^{-1}(K)$. The inclusion $\subseteq$ is clear. Conversely, let $y = y_1 b y_2 b \cdots \in \sigma^{-1}(K)$ with $y_i \in \Gamma$ and let $bx \in L \cap b(\Gamma b)^\infty$ be such that $\sigma(x) = \sigma(y)$. We write $x = x_1 b x_2 b \cdots$ with $x_i \in \Gamma$. We have the same number of factors and for all i we have $h(x_i b)\restriction_{Q'} = \sigma(x_i b) = \sigma(y_i b) = h(y_i b)\restriction_{Q'}$. If the number of factors is finite, then directly $h(bx) = h(by)$, hence $by \in L \cap b(\Gamma b)^*$. If the number of factors is infinite, then we deduce first that $h(bx_i b) = h(by_i b)$. It follows that $bx \sim_h bz \sim_h by$ with $z = x_1 b y_2 b x_3 b y_4 b \cdots$. Since $bx \in L$, we deduce that $by \in L \cap b(\Gamma b)^\omega$.

Next, we show that K is recognized by the morphism $e : T^* \to T'$. To see this let $u \in K$ and $v \in T^\infty$ be such that $u \sim_e v$. We have to show that $v \in K$. Let $x \in (\Gamma b)^\infty$ be such that $bx \in L$ and $\sigma(x) = u$. Note that u is finite if and only if $x \in (\Gamma b)^*$ if and only if x is finite.

Assume first that u and v are both finite. We choose some $y \in (\Gamma b)^*$ with $\sigma(y) = v$. We have $e(\sigma(x)) = e(u) = e(v) = e(\sigma(y))$. Thus, $h(x)\restriction_{Q'} = h(y)\restriction_{Q'}$, but then $h(bx) = h(by)$ and we conclude $by \in L$ and therefore $v = \sigma(y) \in K$.

Now, assume that u and v are both infinite. Since $u \sim_e v$, we find infinite factorizations $u = u_1 u_2 \cdots$ and $v = v_1 v_2 \cdots$ with $u_i, v_i \in T^+$ and $e(u_i) = e(v_i)$ for all i. Since $\sigma(x) = u$, we find a factorization $x = x_1 b x_2 b \cdots$ such that $x_i b \in (\Gamma b)^+$ and $\sigma(x_i b) = u_i$ for all i. Now, for all i, let $y_i b \in (\Gamma b)^+$ be such that $\sigma(y_i b) = v_i$ and let $y = y_1 b y_2 b \cdots$. For all i, we have

$$h(x_i b)\restriction_{Q'} = e \circ \sigma(x_i b) = e(u_i) = e(v_i) = e \circ \sigma(y_i b) = h(y_i b)\restriction_{Q'}$$

and therefore $h(bx_i b) = h(by_i b)$. It follows that $bx \sim_h bz \sim_h by$ with $z = x_1 b y_2 b x_3 b y_4 b \cdots$. Since $bx \in L$, we deduce that $by \in L$ and therefore $v = \sigma(y) \in K$.

Now, it is easy to conclude the proof. Since $|Q'| < |Q|$ and K is recognized by $e : T^* \to T' \subseteq \mathrm{Trans}(Q')$, we know by induction that K is expressible in LTL(T). Using Lemma 7.2, we deduce that $\sigma^{-1}(K)$ is expressible in LTL(Σ). Therefore, $L \cap b(\Gamma b)^\infty = b\sigma^{-1}(K)$ is expressible in LTL(Σ). ■

We are now ready to complete the proof of Proposition 7.1. Let $L \subseteq \mathbb{R}$ be a real trace language recognized by the morphism h. We claim that

$$L \cap \Delta = \bigcup_{u,v \in S} (h^{-1}(u) \cap \Pi)(h^{-1}(v) \cap \mathbb{M}_{I(b)})(L(uv) \cap b(\Gamma b)^\infty).$$

Indeed, let $t = xyz$ with $x \in h^{-1}(u)$, $y \in h^{-1}(v)$ and $z \in L(uv)$ for some $u, v \in S$. Then $h(xy) = uv$ and we deduce from Proposition 5.3 that $t = xyz \in L$. Conversely, let $t \in (L \cap \Delta)$. Using the unambiguous factorization $\Delta = \Pi \mathbb{M}_{I(b)} b(\Gamma b)^\infty$, we can write $t = xyz$ with $x \in \Pi$, $y \in \mathbb{M}_{I(b)}$ and $z \in b(\Gamma b)^\infty$. If we let $u = h(x)$ and $v = h(y)$ then we get $z \in L(uv)$ which concludes the proof of the claim.

By Proposition 5.3, we know that $L(uv)$ is recognized by h, hence by Lemma 7.3 the language $L_3 = L(uv) \cap b(\Gamma b)^\infty \subseteq (\mathrm{Min} = b)$ is expressible in LTL(Σ). By Proposition 5.1, the languages $L_1 = h^{-1}(u) \cap \mathbb{M}_B$ and $L_2 = h^{-1}(v) \cap \mathbb{M}_B$ are recognized by the restriction of h to $\mathbb{M}_B$. Hence, by induction on the size of the alphabet, they are expressible in LTL(B), and also in LTL(Σ) by Lemma 3.1. Since $\Pi \cup \mathbb{M}_{I(b)} \subseteq \mathbb{M}_B$, we have $h^{-1}(u) \cap \Pi = L_1 \cap \Pi$ and $h^{-1}(v) \cap \mathbb{M}_{I(b)} = L_2 \cap \mathbb{M}_{I(b)}$. Using Lemma 6.1, we deduce that

$$(L_1 \cap \Pi)(L_2 \cap \mathbb{M}_{I(b)})L_3 = (h^{-1}(u) \cap \Pi)(h^{-1}(v) \cap \mathbb{M}_{I(b)})(L(uv) \cap b(\Gamma b)^\infty)$$

is expressible in LTL(Σ).

Acknowledgement: We thank Alexander Rabinovich for stimulating discussions during the Computer Science Logic conference 1999 in Madrid.

REFERENCES

1. R. Alur, D. Peled, and W. Penczek. Model-checking of causality properties. In *Proc. 10th IEEE Symposium on Logic in Computer Science (LICS'95), San Diego (California), 1995*, pages 90–100, 1995.
2. V. Diekert and P. Gastin. An expressively complete temporal logic without past tense operators for Mazurkiewicz traces. In J. Flum and M. Rodríguez-Artalejo, editors, *Proc. 13th Computer Science Logic (CSL'99), Madrid (Spain) 1999*, number 1683 in Lecture Notes in Computer Science, pages 188–203. Springer, 1999.
3. V. Diekert and P. Gastin. LTL is expressively complete for Mazurkiewicz traces. In U. Montanari, J. Rolim, and I. Welzl, editors, *Proc. 27th International Colloquium on Automata, Languages and Programming (ICALP'2000), Geneva (Switzerland), 2000*, number 1853 in Lecture Notes in Computer Science, pages 211–222. Springer, 2000.
4. V. Diekert and P. Gastin. Local temporal logic is expressively complete for cograph dependence alphabets. In R. Nieuwenhuis and A. Voronkov, editors, *Proc. 8th International Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR'01), Havanna (Cuba)*, number 2250 in Lecture Notes in Artificial Intelligence, pages 55–69. Springer, 2001.
5. V. Diekert and P. Gastin. Safety and liveness properties for real traces and a direct translation from LTL to monoids. In W. Brauer, H. Ehrig, J. Karhumäki, and A. Salomaa, editors, *Lecture Notes in Computer Science*. Springer, 2002.
6. V. Diekert and G. Rozenberg, editors. *The Book of Traces*. World Scientific, Singapore, 1995.
7. W. Ebinger. *Charakterisierung von Sprachklassen unendlicher Spuren durch Logiken*. Dissertation, Institut für Informatik, Universität Stuttgart, 1994.
8. W. Ebinger and A. Muscholl. Logical definability on infinite traces. *Theoretical Computer Science*, 154:67–84, 1996.
9. D. Gabbay, I. Hodkinson, and M. Reynolds. *Temporal Logic: Mathematical Foundations and Computational Aspects*. Clarendon Press, Oxford, 1994.
10. D. Gabbay, A. Pnueli, S. Shelah, and J. Stavi. On the temporal analysis of fairness. In *Conference Record of the 12th ACM Symposium on Principles of Programming Languages (POPL'80), Las Vegas (Nevada), 1980*, pages 163–173, 1980.
11. P. Gastin, R. Meyer, and A. Petit. A (non-elementary) modular decision procedure for LTrL. In *Proc. 23rd Symposium on Mathematical Foundations of Computer Science (MFCS'98), Brno (Czech Republic), 1998*, volume 1450 of *Lecture Notes in Computer Science*, pages 356–365. Springer, 1998.
12. P. Gastin and A. Petit. Infinite traces. In V. Diekert and G. Rozenberg, editors, *The Book of Traces*, chapter 11, pages 393–486. World Scientific, Singapore, 1995.

13. J. A. W. Kamp. *Tense Logic and the Theory of Linear Order.* PhD thesis, University of California, Los Angeles (California), 1968.
14. R. Meyer and A. Petit. Expressive completeness of LTrL on finite traces: An algebraic proof. In *Proc. 15th Annual Symposium on Theoretical Aspects of Computer Science (STACS'98), Paris (France), 1998*, volume 1373 of *Lecture Notes in Computer Science*, pages 533–543. Springer, 1998.
15. M. Mukund and P. S. Thiagarajan. Linear time temporal logics over Mazurkiewicz traces. In *Proc. 21st Symposium on Mathematical Foundations of Computer Science (MFCS'96), Cracow (Poland), 1996*, volume 1113, pages 62–92, 1996.
16. P. Niebert. A ν-calculus with local views for systems of sequential agents. In J. Wiedermann and P. Hájek, editors, *Proc. 20th Symposium on Mathematical Foundations of Computer Science (MFCS'95), Prague (Czech Republic), 1995*, volume 969 of *Lecture Notes in Computer Science*, pages 563–573. Springer, 1995.
17. W. Penczek. Temporal logics for trace systems: On automated verification. *International Journal of Foundations of Computer Science*, 4:31–67, 1993.
18. W. Penczek and R. Kuiper. Traces and logic. In V. Diekert and G. Rozenberg, editors, *The Book of Traces*, chapter 10, pages 307–390. World Scientific, Singapore, 1995.
19. R. Ramanujam. Locally linear time temporal logic. In *Proc. 11th Annual IEEE Symposium on Logic in Computer Science (LICS'96), New Brunswick (New Jersey), 1996*, pages 118–128, 1996.
20. P. S. Thiagarajan. A trace based extension of linear time temporal logic. In *Proc. 9th Annual IEEE Symposium on Logic in Computer Science (LICS'94), Paris (France), 1994*, pages 438–447, 1994.
21. P. S. Thiagarajan. A trace consistent subset of PTL. In *Proc. Sixth International Conference on Concurrency Theory (CONCUR'95), Philadelphia (Pennsylvania), 1995*, number 962 in Lecture Notes in Computer Science, pages 438–452. Springer, 1995.
22. P. S. Thiagarajan and I. Walukiewicz. An expressively complete linear time temporal logic for Mazurkiewicz traces. In *Proc. 12th Annual IEEE Symposium on Logic in Computer Science (LICS'97), Warsaw (Poland)*, pages 183–194, 1997.
23. I. Walukiewicz. Difficult configurations – on the complexity of LTrL. In K. G. Larsen et al., editors, *Proc. 25th International Colloquium on Automata, Languages and Programming (ICALP'98), Aalborg (Denmark)*, number 1443 in Lecture Notes in Computer Science, pages 140–151, Berlin-Heidelberg-New York, 1998. Springer.
24. Th. Wilke. Classifying discrete temporal properties. In C. Meinel and S. Tison, editors, *Proc. 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS'99), Trier (Germany), 1999*, number 1443 in Lecture Notes in Computer Science, pages 32–46, Berlin-Heidelberg-New York, 1999. Springer. Invited Lecture.