Local LTL with past constants is expressively complete for Mazurkiewicz traces*

Paul Gastin[1], Madhavan Mukund[2], and K. Narayan Kumar[2]

[1] LIAFA, Université Paris 7, 2, place Jussieu, F-75251 Paris Cedex 05
`Fax: 49 711 7816 310` — `Paul.Gastin@liafa.jussieu.fr`
[2] Chennai Mathematical Institute, 92 G N Chetty Road, Chennai 600 017, India
`Fax: 91 44 2815 7671` — `{madhavan,kumar}@cmi.ac.in`

Abstract. To obtain an expressively complete linear-time temporal logic (LTL) over Mazurkiewicz traces that is computationally tractable, we need to intepret formulas locally, at individual events in a trace, rather than globally, at configurations. Such local logics necessarily require past modalities, in contrast to the classical setting of LTL over sequences. Earlier attempts at defining expressively complete local logics have used very general past modalities as well as filters (side-conditions) that "look sideways" and talk of concurrent events. In this paper, we show that it is possible to use unfiltered future modalities in conjunction with past constants and still obtain a logic that is expressively complete over traces.
Keywords Temporal logics, Mazurkiewicz traces, concurrency

1 Introduction

Linear-time temporal logic (LTL) [17] has established itself as a useful formalism for specifying the interleaved behaviour of reactive systems. To combat the combinatorial blow-up involved in describing computations of concurrent systems in terms of interleavings, there has been a lot of interest in using temporal logic more directly on labelled partial orders.

Mazurkiewicz traces [13] are labelled partial orders generated by dependence alphabets of the form (Σ, D), where D is a *dependence* relation over Σ. If $(a, b) \notin D$, a and b are deemed to be independent actions that may occur concurrently. Traces are a natural formalism for describing the behaviour of static networks of communicating finite-state agents [26].

LTL over Σ-labelled sequences is equivalent to $\mathrm{FO}_\Sigma(<)$, the first-order logic over Σ-labelled linear orders [12] and thus defines the class of aperiodic languages over Σ. Though $\mathrm{FO}_\Sigma(<)$ permits assertions about both the past and the future, future modalities suffice for establishing the expressive completeness of LTL with respect to $\mathrm{FO}_\Sigma(<)$ [9]. From a practical point of view, a finite-state program may be checked against an LTL specification relatively efficiently [3, 22, 21].

The first expressively complete temporal logic over traces was described in [7] for finite traces and in [19] for infinite traces. The result was refined in [5] to

* Partial support of CEFIPRA-IFCPAR Project 2102-1 (ACSMV) is gratefully acknowledged.

show expressive completeness without past modalities, using an extension of the proof technique developed for LTL in [25]. Formulas in both these logics are defined at *global configurations* (maximal antichains). Unfortunately, reasoning at the level of global configurations makes the complexity of deciding satisfiability non-elementary [23]. Computational tractability seems to require interpreting formulas at *local states*—effectively at individual events.

Recently, in [11], a local temporal logic has been defined over traces and shown to be expressively complete and tractable (the satisfiability problem is in PSPACE). This logic uses both future and past modalities (similar to the *until* and *since* operators of LTL) which are further equipped with filters (side-conditions). It was also shown that for finite traces, a restricted form of past modalities suffices, but only in conjunction with filtered future modalities. Another proposal is presented in [1] and this logic also uses a version of the until and since operators.

LTL without any past operators is expressively complete over words but this cannot be the case for traces: there exist two first-order inequivalent traces that cannot be distinguished using only future modalities [24].

In this paper, we show that a very limited ability to talk about the past is sufficient to obtain expressive completeness over traces. Our logic uses *unfiltered* future modalities and a finite number of past *constants*. (In particular, there is no nesting of past operators and for that matter even future formulas cannot be nested into past formulas.) As in [4, 5, 11], we show expressive completeness using an extension to traces of the proof technique introduced in [25] for LTL over sequences. From the recent general result proved in [10], it follows that the satisfiability problem for this new logic is also in PSPACE.

The paper is organized as follows. We begin with some preliminaries about traces. In Section 3 we define our new temporal logic. Section 5 describes a syntactic partition of traces that is used in Section 6 to establish expressive completeness.

2 Preliminaries

We briefly recall some notions about Mazurkiewicz traces (see [6] for background). A *dependence alphabet* is a pair (Σ, D) where the alphabet Σ is a finite set of actions and the *dependence relation* $D \subseteq \Sigma \times \Sigma$ is reflexive and symmetric. The *independence relation* I is the complement of D. For $A \subseteq \Sigma$, the set of letters independent of A is denoted by $I(A) = \{b \in \Sigma \mid (a, b) \in I \text{ for all } a \in A\}$ and the set of letters depending on (some action in) A is denoted by $D(A) = \Sigma \setminus I(A)$.

A *Mazurkiewicz trace* is a labelled partial order $t = [V, \leq, \lambda]$ where V is a set of vertices labelled by $\lambda : V \to \Sigma$ and $\leq$ is a partial order over V satisfying the following conditions: For all $x \in V$, the downward set $\downarrow x = \{y \in V \mid y \leq x\}$ is finite, $(\lambda(x), \lambda(y)) \in D$ implies $x \leq y$ or $y \leq x$, and $x \lessdot y$ implies $(\lambda(x), \lambda(y)) \in D$, where $\lessdot \, = \, < \setminus <^2$ is the immediate successor relation in t.

The *alphabet* of a trace t is the set $\mathrm{alph}(t) = \lambda(V) \subseteq \Sigma$ and its *alphabet at infinity*, $\mathrm{alphinf}(t)$, is the set of letters occurring infinitely often in t. The set of all traces is denoted by $\mathbb{R}(\Sigma, D)$ or simply by $\mathbb{R}$. A trace t is called *finite* if V is finite. For $t = [V, \leq, \lambda] \in \mathbb{R}$, we define $\min(t) \subseteq V$ as the set of all minimal vertices of t. We can also read $\min(t) \subseteq \Sigma$ as the set of labels of the minimal vertices of t. It will be clear from the context what we actually mean.

Let $t_1 = [V_1, \leq_1, \lambda_1]$ and $t_2 = [V_2, \leq_2, \lambda_2]$ be a pair of traces such that $\mathrm{alphinf}(t_1) \times \mathrm{alph}(t_2) \subseteq I$. We then define the concatenation of t_1 and t_2 to be $t_1 \cdot t_2 = [V, \leq, \lambda]$ where $V = V_1 \cup V_2$ (assuming wlog that $V_1 \cap V_2 = \emptyset$), $\lambda = \lambda_1 \cup \lambda_2$ and $\leq$ is the transitive closure of the relation $\leq_1 \cup \leq_2 \cup (V_1 \times V_2 \cap \lambda^{-1}(D))$. The set of finite traces is then a monoid, denoted $\mathbb{M}(\Sigma, D)$ or simply $\mathbb{M}$, with the empty trace $1 = (\emptyset, \emptyset, \emptyset)$ as unit.

Here is some useful notation for subclasses of traces. For $C \subseteq \Sigma$, let $\mathbb{R}_C = \{t \in \mathbb{R} \mid \mathrm{alph}(x) \subseteq C\}$ and $\mathbb{M}_C = \mathbb{M} \cap \mathbb{R}_C$. Also, $(\mathrm{alph} = C) = \{t \in \mathbb{R} \mid \mathrm{alph}(t) = C\}$, $(\mathrm{alphinf} = C) = \{t \in \mathbb{R} \mid \mathrm{alphinf}(t) = C\}$ and $(\min = C) = \{t \in \mathbb{R} \mid \min(t) = C\}$. For $A, C \subseteq \Sigma$, we set $\mathbb{R}_C^A = \mathbb{R}_C \cap (\mathrm{alphinf} = A)$. Observe that $\mathbb{M}_C = \mathbb{R}_C^{\emptyset}$.

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

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

where $a \in \Sigma$ and $x, y \in \mathrm{Var}$ are first order variables. Given a trace $t = [V, \leq, \lambda]$ and a valuation $\sigma : \mathrm{Var} \to V$, $t, \sigma \models \varphi$ denotes that t satisfies φ under σ. We interpret each predicate P_a by the set $\{x \in V \mid \lambda(x) = a\}$ and the relation $<$ as the strict partial order relation of t. The semantics then lifts to all formulas as usual. Since the meaning of a closed formula (sentence) φ is independent of the valuation σ, we can associate with each sentence φ the language $\mathcal{L}(\varphi) = \{t \in \mathbb{R} \mid t \models \varphi\}$. We say that a trace language $L \subseteq \mathbb{R}$ is expressible in $\mathrm{FO}_\Sigma(<)$ if there exists a sentence $\varphi \in \mathrm{FO}_\Sigma(<)$ such that $L = \mathcal{L}(\varphi)$. We denote by $\mathrm{FO}_{(\Sigma,D)}(<)$ the set of trace languages $L \subseteq \mathbb{R}(\Sigma, D)$ that are expressible in $\mathrm{FO}_\Sigma(<)$. For $n > 0$, $\mathrm{FO}_\Sigma^n(<)$ denotes the set of formulas with at most n distinct variables (note that each variable may be bound and reused several times).

We use the algebraic notion of recognizability. Let $h : \mathbb{M} \to S$ be a morphism to a finite monoid S. For $t, u \in \mathbb{R}$, we say that t and u are h-similar, denoted $t \sim_h u$, if either $t, u \in \mathbb{M}$ and $h(t) = h(u)$ or t and u have infinite factorizations in non-empty finite traces $t = t_1 t_2 \cdots$, $u = u_1 u_2 \cdots$ with $h(t_i) = h(u_i)$ for all i. The transitive closure $\approx_h$ of $\sim_h$ is an equivalence relation. Since S is finite, this equivalence relation is of finite index with at most $|S|^2 + |S|$ equivalence classes. A trace language $L \subseteq \mathbb{R}$ is *recognized* by h if it is saturated by $\approx_h$ (or equivalently by $\sim_h$), i.e., $t \in L$ implies $[t]_{\approx_h} \subseteq L$ for all $t \in \mathbb{R}$.

Let $L \subseteq \mathbb{R}$ be recognized by a morphism $h : \mathbb{M} \to S$. For $B \subseteq \Sigma$, $L \cap \mathbb{M}_B$ and $L \cap \mathbb{R}_B$ are recognized by $h\lceil_{\mathbb{M}_B}$ the restriction of h to $\mathbb{M}_B$.

A finite monoid S is *aperiodic* if there is an $n \geq 0$ such that $s^n = s^{n+1}$ for all $s \in S$. A trace language $L \subseteq \mathbb{R}$ is *aperiodic* if it is recognized by some morphism to a finite and aperiodic monoid.

First-order definability coincides with aperiodicity for traces.

Theorem 1 ([7, 8]). *A language $L \subseteq \mathbb{R}(\Sigma, D)$ is expressible in $\mathrm{FO}_\Sigma(<)$ if and only if it is aperiodic.*

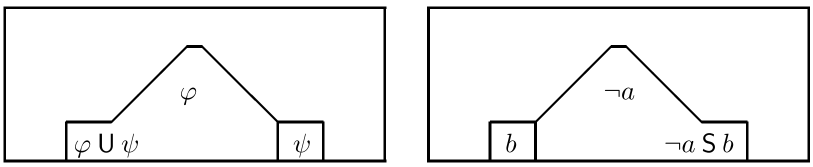
3 Local temporal logic

We denote by $\mathrm{LocTL}^{\mathrm{i}}_{\Sigma}$ the set of (internal) formulas over the alphabet Σ. They are given by the following syntax:

$$\varphi ::= a \in \Sigma \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{EX}\,\varphi \mid \varphi\,\mathsf{U}\,\varphi \mid \neg a\,\mathsf{S}\,b,\ \ a, b \in \Sigma$$

Let $t = [V, \leq, \lambda] \in \mathbb{R}$ be a finite or infinite trace and let $x \in V$ be some vertex of t. We write $t, x \models \varphi$ to denote that trace t at node x satisfies the formula $\varphi \in \mathrm{LocTL}^{\mathrm{i}}_{\Sigma}$. This is defined inductively as follows:

$$\begin{array}{lll}
t, x \models a & \text{if} & \lambda(x) = a \\
t, x \models \neg\varphi & \text{if} & t, x \not\models \varphi \\
t, x \models \varphi \vee \psi & \text{if} & t, x \models \varphi \text{ or } t, x \models \psi \\
t, x \models \mathsf{EX}\,\varphi & \text{if} & \exists y.\ x \lessdot y \text{ and } t, y \models \varphi \\
t, x \models \varphi\,\mathsf{U}\,\psi & \text{if} & \exists z \geq x.\ [t, z \models \psi \text{ and } \forall y.\ (x \leq y < z) \Rightarrow t, y \models \varphi] \\
t, x \models \neg a\,\mathsf{S}\,b & \text{if} & \exists z \leq x.\ [\lambda(z) = b \text{ and } \forall y.\ (z < y \leq x) \Rightarrow \lambda(y) \neq a]
\end{array}$$



The modality U is the "universal" *until* operator defined in [4]. The modality S is the corresponding *since* operator. Note that we only use the operator S in the very restricted form of a fixed number of past constants.

Past modalities are essential, as indicated by the following example from [24], where the dependence relation is $a - b - c - d$. These two traces are not first-order equivalent but are bisimilar at the level of events and thus cannot be distinguished by purely future modalities.

$$\begin{array}{lll}
a \to b \to c \to b \to c \cdots & \qquad & d \to c \to b \to c \to b \cdots \\
\quad\ \ \uparrow & & \quad\ \ \uparrow \\
d \to c & & a \to b
\end{array}$$

As usual, we can derive useful operators such as *universal next* $\mathsf{AX}\,\varphi = \neg\,\mathsf{EX}\,\neg\varphi$, *eventually in the future* $\mathsf{F}\,\varphi = \top\,\mathsf{U}\,\varphi$ and *always in the future* $\mathsf{G}\,\varphi = \neg\,\mathsf{F}\,\neg\varphi$. The modality $\mathsf{F}^{\infty}\,a = \mathsf{F}\,a \wedge \mathsf{G}(a \Rightarrow \mathsf{EX}\,\mathsf{F}\,a)$ expresses the existence of infinitely many vertices labelled with a above the current vertex.

Traces as models of formulas: We now turn our attention to defining when a trace satisfies a formula. For LTL over sequences, lifting satisfaction at positions to satisfaction by a word is quite simple: a word models a formula if its initial position models the formula. Since a trace, in general, does not have a unique initial position, we need to use *initial formulas* as introduced in [4]. These are boolean combinations of formulas $\mathsf{EM}\,\varphi$, each of which asserts the existence of a minimal vertex in a trace satisfying the internal formula φ. More precisely, the set LocTL_Σ of initial formulas over the alphabet Σ is defined as follows:

$$\alpha ::= \bot \mid \mathsf{EM}\,\varphi, \varphi \in \mathrm{LocTL}^{\mathrm{i}}_\Sigma \mid \neg\alpha \mid \alpha \vee \alpha$$

The semantics is given by:

$$\begin{array}{lll} t \models \mathsf{EM}\,\varphi & \text{if} & \exists x.\ (x \in \min(t) \text{ and } t, x \models \varphi) \\ t \models \neg\alpha & \text{if} & t \not\models \alpha \\ t \models \alpha \vee \beta & \text{if} & t \models \alpha \text{ or } t \models \beta \end{array}$$

An initial formula $\alpha \in \mathrm{LocTL}_\Sigma$ defines the trace language $\mathcal{L}(\alpha) = \{t \in \mathbb{R} \mid t \models \alpha\}$. We can then express various alphabetic properties using initial formulas: $\mathcal{L}(\mathsf{EM}\,a) = \{t \in \mathbb{R} \mid a \in \min(t)\}$, $\mathcal{L}(\mathsf{EM}\,\mathsf{F}\,a) = \{t \in \mathbb{R} \mid a \in \mathrm{alph}(t)\}$, and $\mathcal{L}(\mathsf{EM}\,\mathsf{F}^\infty\,a) = \{t \in \mathbb{R} \mid a \in \mathrm{alphinf}(t)\}$. Therefore, for $C \subseteq \Sigma$, trace languages such as $(\mathrm{alph} = C)$, $(\mathrm{alphinf} = C)$ and $(\min = C)$ are expressible in LocTL_Σ.

The following result is immediate from the definition of LocTL_Σ.

Proposition 2. *If a trace language is expressible in* LocTL_Σ*, then it is expressible in* $\mathrm{FO}^3_\Sigma(<)$.

We now show that the "filtered" modalities EX_b and F_b from [11], with the following semantics, are both expressible in $\mathrm{LocTL}^{\mathrm{i}}_\Sigma$.

$$\begin{array}{lll} t, x \models \mathsf{EX}_b\,\varphi & \text{if} & \exists y.\ [x \lessdot y \text{ and } t, y \models \varphi \text{ and } \forall z.\ (z \le y \wedge \lambda(z) = b) \Rightarrow z \le x] \\ t, x \models \mathsf{F}_b\,\varphi & \text{if} & \exists y.\ [x \le y \text{ and } t, y \models \varphi \text{ and } \forall z.\ (z \le y \wedge \lambda(z) = b) \Rightarrow z \le x] \end{array}$$

Proposition 3. *For any trace t over some alphabet Σ, any position x in t and any formula φ of* $\mathrm{LocTL}^{\mathrm{i}}_\Sigma$

$$t, x \models \mathsf{EX}_b\,\varphi \iff t, x \models (b \wedge \mathsf{EX}(\varphi \wedge \neg b)) \vee \bigvee_{a \neq b} (a \wedge \mathsf{EX}(\varphi \wedge \neg(\neg a\,\mathsf{S}\,b)))$$

Proof. Suppose $t, x \models \mathsf{EX}_b\,\varphi$. Then, there exists y such that $x \lessdot y$, $t, y \models \varphi$ and for all z, $(z \le y \wedge \lambda(z) = b) \Rightarrow z \le x$. Clearly $\lambda(y) \neq b$. If $\lambda(x) = b$ then $t, x \models b \wedge \mathsf{EX}(\varphi \wedge \neg b))$. Assume now that $\lambda(x) = a \neq b$ and suppose $t, y \models \neg a\,\mathsf{S}\,b$. Then, $\lambda(z) = b$ for some $z \le y$ such that $\lambda(u) \neq a$ for all $z < u \le y$. From the definition of EX_b, $z \le x$ and using $\lambda(x) \neq \lambda(z)$ we get $z < x < y$, a contradiction with $\lambda(x) = a$. Hence it must be the case that $t, y \models \neg(\neg a\,\mathsf{S}\,b)$.

For the converse, suppose $t, x \models b \wedge \mathsf{EX}(\varphi \wedge \neg b)$. Thus there is a y with $x \lessdot y$ and $t, y \models \varphi$ and $\lambda(y) \neq b$. Hence the maximal b event below y must be x. Thus, $\forall z.\ (z \le y \wedge \lambda(z) = b) \Rightarrow z \le x$ and hence $t, x \models \mathsf{EX}_b\,\varphi$.

Suppose, $t, x \models a \wedge \mathsf{EX}(\varphi \wedge \neg(\neg a\,\mathsf{S}\,b))$ for some $a \neq b$. Let y with $x \lessdot y$ and $t, y \models \varphi \wedge \neg(\neg a\,\mathsf{S}\,b)$. Suppose $z \le y$ and $\lambda(z) = b$. Since $t, y \models \neg(\neg a\,\mathsf{S}\,b))$, there must be a u with $z < u \le y$ with $\lambda(u) = a = \lambda(x)$. Hence either $u \le x$ and we get $z < x$ or $x < u$ and since $x \lessdot y$ we must have $u = y$. Then $\lambda(y) = \lambda(x)$ and from $x \lessdot y$ and $z < y$ we deduce $z \le x$. $\square$

Let the formula Safe_b be defined as follows:

$$\mathsf{Safe}_b = (b \wedge \mathsf{AX}\,\neg b) \vee \bigvee_{a \neq b} (a \wedge \mathsf{AX}\,\neg(\neg a\,\mathsf{S}\,b)).$$

Proposition 4. *For any trace t over some alphabet Σ and any position x in t, $t, x \models \mathsf{Safe}_b$ if and only if $\forall y.\ (x \lessdot y \Rightarrow (\forall z.\ (z \leq y$ and $\lambda(z) = b) \Rightarrow z \leq x))$.*

Proof. Suppose $t, x \models \mathsf{Safe}_b$ and let $x \lessdot y$. We have $\lambda(y) \neq b$. Let $z \leq y$ with $\lambda(z) = b$. If $\lambda(x) = b$ then, clearly $z \leq x$. Otherwise, $\lambda(x) = a \neq b$ and $t, y \models \neg(\neg a\,\mathsf{S}\,b)$. Thus, there is a u with $z < u \leq y$ and $\lambda(u) = a = \lambda(x)$. Thus, either $u \leq x$ and we get $z < x$ or $x < u$ and since $x \lessdot y$ we must have $u = y$. Then $\lambda(y) = \lambda(x)$ and from $x \lessdot y$ and $z < y$ we deduce $z \leq x$.

For the converse, suppose $\forall y.\ (x \lessdot y \Rightarrow (\forall z.\ (z \leq y$ and $\lambda(z) = b) \Rightarrow z \leq x))$. Thus, $t, x \models \mathsf{AX}\,\neg b$ and if $\lambda(x) = b$ we get $t, x \models b \wedge \mathsf{AX}\,\neg b$. Assume now that $\lambda(x) = a \neq b$ and let $x \lessdot y$. For any $z \leq y$ with $\lambda(z) = b$ we have $z \leq x$. Since $\lambda(x) = a$, we obtain $t, y \models \neg(\neg a\,\mathsf{S}\,b)$. Hence, $t, x \models a \wedge \mathsf{AX}\,\neg(\neg a\,\mathsf{S}\,b)$. □

Proposition 5. *Let $t \in \mathbb{R}(\Sigma, D)$ and x be a position in t. If $t, x \models \mathsf{Safe}_b\,\mathsf{U}\,\varphi$ then $t, x \models \mathsf{F}_b\,\varphi$.*

Proof. By definition of U, there is a $y \geq x$ with $t, y \models \varphi$ and $t, z \models \mathsf{Safe}_b$ for all z with $x \leq z < y$. If $y = x$ then we are done. Otherwise we find a chain $x = x_1 \lessdot x_2 \lessdot \ldots \lessdot x_k = y$ with $k > 1$. We have $t, x_i \models \mathsf{Safe}_b$ for $1 \leq i < k$. Thus, by Proposition 4 for each $1 \leq i < k$ and z with $\lambda(z) = b$, $z \leq x_{i+1}$ implies that $z \leq x_i$. □

Proposition 6. *Let $t \in \mathbb{R}(\Sigma, D)$ and x, y be positions in t such that $x \leq y$ and for any b labelled event z, if $z \leq y$ then $z \leq x$. Then, there are at most $|\Sigma|$ events between x and y that do not satisfy Safe_b. Further, if $x \leq u < y$ and $t, u \not\models \mathsf{Safe}_b$ then, there is a v with $u \lessdot v \leq y$ and either $t, v \models \neg(\neg\lambda(u)\,\mathsf{S}\,b)$ or $\lambda(u) = b$ and $\lambda(v) \neq b$.*

Proof. Suppose there are more than $|\Sigma|$ positions between x and y that do not satisfy Safe_b. Then, there must be two such positions $x \leq u < u' \leq y$ with $\lambda(u) = \lambda(u') = a$. The hypothesis on x, y implies $a \neq b$. Since u does not satisfy Safe_b, we deduce that $t, u \models \neg\,\mathsf{AX}\,\neg(\neg a\,\mathsf{S}\,b) = \mathsf{EX}(\neg a\,\mathsf{S}\,b)$. Thus, $t, v \models \neg a\,\mathsf{S}\,b$ for some v with $u \lessdot v$. Thus, there is a b-labelled event z with $z \leq v$ and $z \not\leq u$. If $z \leq y$ then $z \leq x \leq u$, a contradiction. Since $z \leq v$ and $u' \leq y$, we deduce that $v \not\leq u'$. But $\lambda(u') = \lambda(u)$ and $\lambda(v)$ are dependent and thus, it must the case that $u' < v$. This contradicts the fact that $u \lessdot v$ and completes the proof of the first part of the proposition.

Now suppose, $t, u \models \neg\mathsf{Safe}_b$. Since $x \leq u < y$, there is a v with $u \lessdot v \leq y$. If $\lambda(u) = b$ then $\lambda(v) \neq b$ since every b-labelled event below y must be below x. Otherwise, $\lambda(u) = a \neq b$. Let z be any b-labelled event below v. Then, z is a b-labelled event below y and hence below x and thus it must be strictly below u. Thus, $t, v \models \neg(\neg a\,\mathsf{S}\,b)$. □

Let $\mathsf{F}_b^0\,\varphi = \mathsf{Safe}_b \,\mathsf{U}\, \varphi$ and $\mathsf{F}_b^{k+1}\,\varphi = \mathsf{Safe}_b \,\mathsf{U}\, \mathsf{EX}_b(\mathsf{F}_b^k\,\varphi)$.

Proposition 7. *For any trace $t \in \mathbb{R}(\Sigma, D)$, any position x in t and any formula φ of $\mathrm{LocTL}^{\mathrm{i}}_{\Sigma}$*

$$t, x \models \mathsf{F}_b\,\varphi \iff t, x \models \bigvee_{k \le |\Sigma|} \mathsf{F}_b^k\,\varphi$$

Proof. Suppose $t, x \models \mathsf{F}_b^k\,\varphi$. We show by induction on k that $t, x \models \mathsf{F}_b\,\varphi$. When $k = 0$, this is just Proposition 5. Assume now that $t, x \models \mathsf{F}_b^{k+1}\,\varphi$ for some $k \ge 0$. Then, Proposition 5 implies that there is a $u \ge x$ with $t, u \models \mathsf{EX}_b(\mathsf{F}_b^k\,\varphi)$ and for any b-labelled z, if $z \le u$ then $z \le x$. By the definition of EX_b, there is a v, $u \lessdot v$ and $t, v \models \mathsf{F}_b^k\,\varphi$ and for any b-labelled z, if $z \le v$ then $z \le u$ and hence $z \le x$. Finally, by induction hypothesis, there must be a y such that $v \le y$ (and hence $x \le y$) and $t, y \models \varphi$ and for any b-labelled z, if $z \le y$ then $z \le v$ and hence $z \le x$. Thus, $t, x \models \mathsf{F}_b\,\varphi$.

For converse, suppose $t, x \models \mathsf{F}_b\,\varphi$. Therefore, there is a $y \ge x$ such that $t, y \models \varphi$ and for any b-labelled event z, if $z \le y$ then $z \le x$. If $t, x \models \mathsf{Safe}_b \,\mathsf{U}\, \varphi = \mathsf{F}_b^0\,\varphi$ then we are done. Otherwise, we consider a maximal chain $x \le x_1 < \ldots < x_k < y$ where each x_i does not satisfy Safe_b. From Proposition 6 we know that $k \le |\Sigma|$.

Observation: Since this is a maximal such chain it follows that $t, z \models \mathsf{Safe}_b$ for all z with $x \le z < x_1$ or $x_k \le z < y$ or $x_i < z < x_{i+1}$ with $1 \le i < k$.

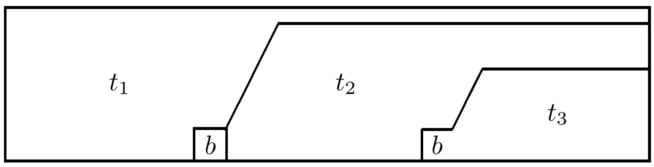
Thus, $t, x_k \models \mathsf{EX}_b\,\mathsf{F}_b^0\,\varphi$. Suppose, $t, x_i \models \mathsf{EX}_b(\mathsf{F}_b^{k-i}\,\varphi)$. Let $x_{i-1} \lessdot v \le x_i$. Then, $t, v \models \mathsf{Safe}_b \,\mathsf{U}\, \mathsf{EX}_b\,\mathsf{F}_b^{k-i}\,\varphi$ and we obtain $t, x_{i-1} \models \mathsf{EX}_b\,\mathsf{F}_b^{k-i+1}\,\varphi$. By induction we get $t, x_1 \models \mathsf{EX}_b\,\mathsf{F}_b^{k-1}$. Hence $t, x \models \mathsf{Safe}_b \,\mathsf{U}\, \mathsf{EX}_b\,\mathsf{F}_b^{k-1}\,\varphi = \mathsf{F}_b^k\,\varphi$. □

4 The Lifting Lemmas

In this section we establish some important lemmas that are critical in proving the expressive completeness of LocTL_{Σ}.

Lemma 8. *Let $A \subsetneq \Sigma$ and $b \in \Sigma$ with $b \notin A$. For all $\varphi \in \mathrm{LocTL}^{\mathrm{i}}_{A}$, there is a formula $\widetilde{\varphi} \in \mathrm{LocTL}^{\mathrm{i}}_{A\cup\{b\}}$ such that for all $t = t_1 b t_2 t_3 \in \mathbb{R}$ with $t_1 \in \mathbb{R}$, $t_2 \in \mathbb{R}_A$, $\min(t_2) \subseteq D(b)$ and $\min(t_3) \subseteq \{b\}$ and for all $x \in bt_2$ we have*

$$bt_2, x \models \varphi \text{ iff } t, x \models \widetilde{\varphi}.$$



Proof. Both the definition of $\widetilde{\varphi}$ as well as the proof of the Lemma proceeds by structural induction on φ. We let $\widetilde{a} = a$, $\widetilde{\neg\varphi} = \neg\widetilde{\varphi}$, $\widetilde{\varphi \vee \psi} = \widetilde{\varphi} \vee \widetilde{\psi}$, $\widetilde{\mathsf{EX}\,\varphi} = \mathsf{EX}_b\,\widetilde{\varphi}$, $\widetilde{\varphi \,\mathsf{U}\, \psi} = \bigvee_{d \in A\cup\{b\}} ((\widetilde{\varphi} \,\mathsf{U}\, (d \wedge \widetilde{\psi})) \wedge \mathsf{F}_b(d \wedge \widetilde{\psi}))$ and $\widetilde{\neg c \,\mathsf{S}\, d} = (\neg c \,\mathsf{S}\, d) \wedge \neg(\neg d \,\mathsf{S}\, b)$.

The proof for the operators $\neg$ and $\vee$ and for $\varphi = a$ is quite obvious. In what follows we write v for $\min(bt_2t_3)$ (and hence $\lambda(v) = b$). Let $x \in bt_2$.

$bt_2, x \models \mathsf{EX}\,\varphi$ if and only if there is a $y \in t_2$ with $x \lessdot y$ and $bt_2, y \models \varphi$. Since $y \in t_2$, by the induction hypothesis, $t, y \models \widetilde{\varphi}$. Further, since $y \in t_2$, if $z \in t$, $z \leq y$ and $\lambda(z) = b$ then $z \leq v$ and hence $z \leq x$. Thus, $t, x \models \mathsf{EX}_b\,\widetilde{\varphi}$. For the converse, suppose $t, x \models \mathsf{EX}_b\,\widetilde{\varphi}$. Thus, there is a $y \in t$ with $x \lessdot y$ and $t, y \models \widetilde{\varphi}$ and if $z \in t$ with $\lambda(z) = b$ and $z \leq y$ then $z \leq x$. Thus, $y \notin t_3$ and hence $y \in t_2$. So, by induction hypothesis, $bt_2, y \models \varphi$ and hence $bt_2, x \models \mathsf{EX}\,\varphi$.

Suppose, $bt_2, x \models \varphi \,\mathsf{U}\, \psi$. Then, there is a $y \in bt_2$ with $x \leq y$ and $bt_2, y \models \psi$ and if $x \leq z < y$ then $bt_2, z \models \varphi$. By the induction hypothesis, $t, y \models \widetilde{\psi}$ and if $x \leq z < y$ then $t, z \models \widetilde{\varphi}$. Since $y \in bt_2$, $\lambda(y) = d \in A \cup \{b\}$ and thus $t, x \models \widetilde{\varphi} \,\mathsf{U}\, (d \wedge \widetilde{\psi})$. Further, if $z \in t$ and $\lambda(z) = b$ and $z \leq y$ then, since $y \in bt_2$, $z \leq v$ and hence $z \leq x$. Thus, by the definition of F_b, $t, x \models \mathsf{F}_b(d \wedge \widetilde{\psi})$. For the converse, suppose there is some $d \in A \cup \{b\}$ and $t, x \models \widetilde{\varphi} \,\mathsf{U}\, (d \wedge \widetilde{\psi})$ and $t, x \models \mathsf{F}_b(d \wedge \widetilde{\psi})$. Since $t, x \models \widetilde{\varphi} \,\mathsf{U}\, (d \wedge \widetilde{\psi})$, there is a $y \in t$, $y \geq x$, $\lambda(y) = d$ and $t, y \models \widetilde{\psi}$ and $t, z \models \widetilde{\varphi}$ for any $x \leq z < y$. Let y_0 be the minimum d-labelled event above x that satisfies $\widetilde{\psi}$. Then, clearly $t, z \models \widetilde{\varphi}$ for any z with $x \leq z < y_0$. Since $t, x \models \mathsf{F}_b(d \wedge \widetilde{\psi})$ there is a y', $x \leq y'$, $t, y' \models d \wedge \widetilde{\psi}$ and if $z \in t$, $\lambda(z) = b$ and $z \leq y'$ then $z \leq x$. Now, since $x \in bt_2$, $y' \in t_2t_3$. However, since $\min(t_3) \subseteq \{b\}$ it follows that y' must be in bt_2. Thus, $y_0 \in bt_2$. Thus, by induction hypothesis, $bt_2, y_0 \models \psi$ and if $x \leq z < y$ then ($z \in bt_2$ and thus) $bt_2, z \models \varphi$. Thus, $bt_2, x \models \varphi \,\mathsf{U}\, \psi$.

Finally, if $bt_2, x \models \neg c \,\mathsf{S}\, d$, $c, d \in A$ then there is a $y \in bt_2$, $y \leq x$, $\lambda(y) = d$ and for any $z \in bt_2$ if $y < z \leq x$ then $\lambda(z) \neq c$. $y \in t$ and if $z \in t$ and $y < z \leq x$ then $z \in t_2$ and $y < z \leq x$ and thus $\lambda(z) \neq c$ and so $t, x \models \neg c \,\mathsf{S}\, d$. We still need to show that $t, x \models \neg(\neg d \,\mathsf{S}\, b)$. Suppose $t, x \models \neg d \,\mathsf{S}\, b$. Then, since v is the maximal b-labelled event below x, it must be the case that if $v < z \leq x$ then $\lambda(z) \neq d$. However, $y \in t_2$ and $v < y$ (since $\lambda(y) = d$ and $d \neq b$) and $y \leq x$ and $\lambda(y) = d$. A contradiction. Thus, $t, x \models \neg(\neg d \,\mathsf{S}\, b)$ and hence $t, x \models \widetilde{\neg c \,\mathsf{S}\, d}$. For the converse, suppose $t, x \models \neg c \,\mathsf{S}\, d$ and $t, x \models \neg(\neg d \,\mathsf{S}\, b)$. Since $t, x \models \neg c \,\mathsf{S}\, d$, there is a $y \in t$, $y \leq x$, $\lambda(y) = d$ and for any z with $y < z \leq x$, $\lambda(z) \neq c$. In particular if y_0 is the maximal d-labelled event below x in t then, for any z with $y_0 < z \leq x$, $\lambda(z) \neq c$. Since $t, x \models \neg(\neg d \,\mathsf{S}\, b)$ and $t, v \models b$, there is a y' with $v < y' \leq x$ and $\lambda(y') = d$. Thus, there is a $y' \in t_2$ with $y' \leq x$ and $\lambda(y') = d$. Thus, $y_0 \in t_2$ and thus $bt_2, x \models \neg c \,\mathsf{S}\, d$. $\square$

Lemma 9. *Let $A \subseteq \Sigma$ and $b \in \Sigma$ with $b \notin A$. For all $\alpha \in \mathrm{LocTL}_A$, there exists a formula $\overline{\alpha} \in \mathrm{LocTL}^{\mathrm{i}}_{A \cup \{b\}}$ such that for all $t = t_1bt_2t_3 \in \mathbb{R}$ with $t_1 \in \mathbb{R}$, $t_2 \in \mathbb{R}_A$, $\min(t_2) \subseteq D(b)$ and $\min(t_3) \subseteq \{b\}$, we have $t_2 \models \alpha$ if and only if $t, \min(bt_2t_3) \models \overline{\alpha}$.*

Proof. First, we claim that for all $\xi \in \mathrm{LocTL}^{\mathrm{i}}_A$ and all y in t_2, we have $t_2, y \models \xi$ if and only if $bt_2, y \models \xi$. The proof is an easy induction on ξ. The only intersting case is when $\xi = \neg c \,\mathsf{S}\, d$ and for this we use $d \neq b$.

Now, the lemma is proved by structural induction on α. We claim that $\overline{\neg\alpha} = \neg\overline{\alpha}$, $\overline{\alpha \vee \beta} = \overline{\alpha} \vee \overline{\beta}$ and $\overline{\mathsf{EM}\,\varphi} = \mathsf{EX}(\widetilde{\varphi} \wedge \neg b)$ where $\widetilde{\varphi}$ is the formula given by Lemma 8. The only non trivial case is $\mathsf{EM}\,\varphi$. We let $x = \min(bt_2t_3)$.

Assume that $t_2, y \models \varphi$ for some minimal position y in t_2. Since $t_2 \in \mathbb{R}_A$ we have $\lambda(y) \neq b$. From the claim we get $bt_2, y \models \varphi$ and Lemma 8 implies $t, x \models \widetilde{\varphi}$. Since $\lambda(x) = b$ and $\min(t_2) \subseteq D(b)$ we deduce that $x \lessdot y$. Therefore, $t, x \models \mathsf{EX}(\widetilde{\varphi} \wedge \neg b)$.

Conversely, assume that $t, x \models \mathsf{EX}(\widetilde{\varphi} \wedge \neg b)$ and let y be such that $x \lessdot y$, $\lambda(y) \neq b$ and $t, y \models \varphi$. Since $\min(t_3) \subseteq \{b\}$, the vertex y cannot be in t_3, therefore it is minimal in t_2. From Lemma 8 we get $bt_2, y \models \varphi$ and the claim implies $t_2, y \models \varphi$. Therefore, $t_2 \models \mathsf{EM}\,\varphi$. $\square$

Lemma 10. *Let $A \subseteq \Sigma$ and $b \in \Sigma$ with $b \notin A$. For all $\alpha \in \mathrm{LocTL}_A$, there exists a formula $\widetilde{\alpha} \in \mathrm{LocTL}_{A \cup \{b\}}$ such that for all $t = t_1 t_2$ with $t_1 \in \mathbb{R}_A$ and $\min(t_2) \subseteq \{b\}$, we have $t_1 \models \alpha$ if and only if $t \models \widetilde{\alpha}$.*

Proof. We prove first that for all $\varphi \in \mathrm{LocTL}^{\mathrm{i}}_A$, there exists a formula $\widetilde{\varphi} \in \mathrm{LocTL}^{\mathrm{i}}_{A \cup \{b\}}$ such that for all $t = t_1 t_2$ with $t_1 \in \mathbb{R}_A$, $\min(t_2) \subseteq \{b\}$ and for all $x \in t_1$, we have $t_1, x \models \varphi$ if and only if $t, x \models \widetilde{\varphi}$.

As for Lemma 8, we proceed by structural induction on φ but the formulas for the temporal modalities are simpler: $\widetilde{\mathsf{EX}\,\varphi} = \mathsf{EX}(\widetilde{\varphi} \wedge \neg(\neg b \mathbin{\mathsf{S}} b))$, $\widetilde{\varphi \mathbin{\mathsf{U}} \psi} = \widetilde{\varphi} \mathbin{\mathsf{U}} (\widetilde{\psi} \wedge \neg(\neg b \mathbin{\mathsf{S}} b))$ and $\widetilde{\neg c \mathbin{\mathsf{S}} d} = \neg c \mathbin{\mathsf{S}} d$. The proof is simpler than for Lemma 8 and is omitted.

For initial formulas, we have $\widetilde{\mathsf{EM}\,\varphi} = \mathsf{EM}(\widetilde{\varphi} \wedge \neg b)$ and indeed, $\widetilde{\neg\alpha} = \neg\widetilde{\alpha}$ and $\widetilde{\alpha \vee \beta} = \widetilde{\alpha} \vee \widetilde{\beta}$. The proof is easy. $\square$

5 Decomposition of traces

The proof of our main result is a case analysis based on partitioning the set of traces according to the structure of the trace. Fix a letter $b \in \Sigma$ and set $B = \Sigma \setminus \{b\}$. Using the notation introduced in Section 2, let $\Gamma^A = \{t \in \mathbb{R}^A_B \mid \min(t) \subseteq D(b)\}$, $\Gamma = \Gamma^\emptyset$, and $\Omega_A = \{t \in \mathbb{R}_{I(A)} \mid \min(t) \subseteq \{b\}\}$.

Each trace $t \in \mathbb{R}$ has a unique finite or infinite factorization $t = t_0 b t_1 b t_2 \cdots$ with $t_0 \in \mathbb{R}_B$ and $t_i \in \mathbb{R}_B \cap (\min \subseteq D(b))$ for all $i > 0$. In particular, we have

$$(\min = \{b\}) = (b\Gamma)^+ \uplus (b\Gamma)^\omega \uplus \bigcup_{\emptyset \neq A \subseteq B} (b\Gamma)^* b \Gamma^A \Omega_A$$

The following two results will allow us to use this decomposition effectively in proving the expressive completeness of our logic. For this, we use $\mathsf{F}^\infty_b\, a = \mathsf{F}_b\, a \wedge \neg\, \mathsf{F}_b(a \wedge \neg\, \mathsf{EX}_b\, \mathsf{F}_b\, a)$.

Lemma 11. *Let $t = t_0 t'$ with $t_0, t' \in \mathbb{R}$ and $\min(t') = \{b\}$. Then,*

1. *$t' \in (b\Gamma)^\infty \setminus \{1\}$ if and only if $t, \min(t') \models \beta$ with*

$$\beta = \bigvee_C \left(\bigwedge_{c \in C} \mathsf{F}^\infty\, c \wedge \bigwedge_{c \notin C} \neg\, \mathsf{F}^\infty\, c \right)$$

where C ranges over connected subsets of Σ such that $b \in C$ if $C \neq \emptyset$.

2. $t' \in (b\Gamma)^* b\Gamma^A \Omega_A$ *if and only if* $t, \min(t') \models \gamma$ *with*

$$\gamma = \bigvee_{C\subseteq\Sigma} \left(\bigwedge_{c\notin C} \neg \mathsf{F}^\infty c \right) \wedge \mathsf{F}\left(b \wedge \bigwedge_{c\in C} \mathsf{F}^\infty c \wedge \bigwedge_{a\in A} \mathsf{F}_b^\infty a \wedge \bigwedge_{a\notin A} \neg \mathsf{F}_b^\infty a \right).$$

Note that "the" b *in* $b\Gamma^A\Omega_A$ *is characterized by the formula* $b \wedge \mathsf{F}_b^\infty a$*, where* a *is any letter in* A*.*

Proof. (1) Let $C = \mathrm{alphinf}(t')$. Since $t' \in (\min = \{b\})$, we have $t' \in (b\Gamma)^\infty \setminus \{1\}$ if and only if either $C = \emptyset$ or C is connected and $b \in C$. This is precisely what β states.

(2) Assume that $t' = t_1 b t_2 t_3$ with $t_1 \in (b\Gamma)^*$, $t_2 \in \Gamma^A$ and $t_3 \in \Omega_A$. Let $C = \mathrm{alphinf}(t')$. We have $t, \min(t') \models \bigwedge_{c\notin C} \neg \mathsf{F}^\infty c$. Let $x = \min(bt_2t_3) \in t$. Since $t_1 \in (b\Gamma)^*$, we have $C = \mathrm{alphinf}(bt_2t_3)$ and we get $t, x \models \bigwedge_{c\in C} \mathsf{F}^\infty c$. We have, $a \in \mathrm{alphinf}(t_2)$ iff $t, x \models \mathsf{F}_b^\infty a$. We deduce that $t, x \models \bigwedge_{a\in A} \mathsf{F}_b^\infty a \wedge \bigwedge_{a\notin A} \neg \mathsf{F}_b^\infty a$. Therefore, $t, \min(t') \models \gamma$.

Conversely, assume that $t, \min(t') \models \gamma$ and let $C = \mathrm{alphinf}(t')$. We have $t, \min(t') \models \bigwedge_{c\notin C} \neg \mathsf{F}^\infty c$. Let $x \in t'$ with $t, x \models b \wedge \bigwedge_{c\in C} \mathsf{F}^\infty c \wedge \bigwedge_{a\in A} \mathsf{F}_b^\infty a \wedge \bigwedge_{a\notin A} \neg \mathsf{F}_b^\infty a$. Let t_1 be the prefix of $t' = [V', \le, \lambda]$ defined by the set of vertices $\{y \in V' \mid x \not\le y\}$. Since $\mathrm{alphinf}(t') = C$ and $t, x \models \bigwedge_{c\in C} \mathsf{F}^\infty c$ we deduce that $t_1 \in \mathbb{M}$ and hence $t_1 \in (b\Gamma)^*$. Write $t' = t_1 b t_2 t_3$ with $t_2 \in \mathbb{R}_B$ and $\min(t_3) \subseteq \{b\}$. Then, $a \in \mathrm{alphinf}(t_2)$ iff $t, x \models \mathsf{F}_b^\infty a$. Therefore, $\mathrm{alphinf}(t_2) = A$ and $t_2 \in \Gamma^A$. Finally, $\mathrm{alphinf}(t_2) \times \mathrm{alph}(t_3) \subseteq I$, hence $t_3 \in \Omega_A$. □

Lemma 12. *Let* $A \subseteq \Sigma$ *and let* $L \subseteq \mathbb{R}$ *be a trace language recognized by a morphism* h *from* $\mathbb{M}$ *into a finite monoid* S*. Then,*

$$L \cap (b\Gamma)^* b\Gamma^A\Omega_A \;=\; \bigcup_{\textit{finite}} (L_1 \cap (b\Gamma)^*) b (L_2 \cap \Gamma^A)(L_3 \cap \Omega_A)$$

where the trace languages $L_i \subseteq \mathbb{R}$ *are recognized by* h*.*

Proof. Follows from the formula:

$$L \cap (b\Gamma)^* b\Gamma^A\Omega_A = \bigcup ([x]_{\approx_h} \cap (b\Gamma)^*) b([y]_{\approx_h} \cap \Gamma^A)([z]_{\approx_h} \cap \Omega_A)$$

where the union ranges over all $xbyz \in L$ with $x \in (b\Gamma)^*$, $y \in \Gamma^A$ and $y \in \Omega_A$.

The inclusion $\subseteq$ is clear. Conversely, let $x, x' \in (b\Gamma)^*$, $y, y' \in \Gamma^A$ and $z, z' \in \Omega_A$ with $xbyz \in L$, $x \sim_h x'$, $y \sim_h y'$ and $z \sim_h z'$. The cases y or z finite are simpler, hence we assume y, z infinite. We have $y = y_1 y_2 \cdots$, $y' = y'_1 y'_2 \cdots$ with $h(y_i) = h(y'_i)$ for all $i \ge 1$ and wlog $\mathrm{alph}(y_i) \subseteq A$ and $\mathrm{alph}(y'_i) \subseteq A$ for all $i > 1$. Similarly, we can write $z = z_1 z_2 \cdots$, $z' = z'_1 z'_2 \cdots$ with $h(z_i) = h(z'_i)$ for all $i \ge 1$.

We have $xbyz = (xby_1z_1)(y_2z_2)\cdots$ and $x'by'z' = (x'by'_1z'_1)(y'_2z'_2)\cdots$. Moreover, $h(xby_1z_1) = h(x'by'_1z'_1)$, and $h(y_iz_i) = h(y'_iz'_i)$ for all $i > 1$. Therefore, $x'by'z' \sim_h xbyz \in L$ and $x'by'z' \in L$. Since $\approx_h$ is the transitive closure of $\sim_h$, the result follows easily by induction. □

6 Expressive Completeness

If T is a finite alphabet, we define the linear temporal logic $\mathrm{LTL}_T(\mathsf{XU})$ by the syntax:

$$f ::= u \in T \mid \neg f \mid f \vee f \mid f \,\mathsf{XU}\, f.$$

The length of a finite or infinite word $w = w_1 w_2 \cdots \in T^\infty$ is $|w| \in \mathbb{N} \cup \{\omega\}$. The semantics of $\mathrm{LTL}_T(\mathsf{XU})$ is given for $w = w_1 w_2 \cdots \in T^\infty$ by

$$\begin{array}{ll} w \models u & \text{if } |w| > 0 \text{ and } w_1 = u \\ w \models \neg f & \text{if } w \not\models f \\ w \models f \vee g & \text{if } w \models f \text{ or } w \models g \\ w \models f \,\mathsf{XU}\, g & \text{if } \exists j \in \mathbb{N} \text{ with } 1 < j \leq |w| + 1 \text{ and} \\ & \qquad w_j w_{j+1} \cdots \models g \text{ and } w_k w_{k+1} \cdots \models f, \forall 1 < k < j. \end{array}$$

Note that if $w \models f \,\mathsf{XU}\, g$ then w is nonempty. A formula $f \in \mathrm{LTL}_T(\mathsf{XU})$ defines the word language $\mathcal{L}(f) = \{w \in T^\infty \mid w \models f\}$. We use the following proposition which is a consequence of several results on the equivalence between aperiodic word languages, star-free word languages, first order word languages and word languages expressible in $\mathrm{LTL}_T(\mathsf{XU})$ [18, 12, 14, 20, 9, 15, 16, 2].

Proposition 13. *Every aperiodic word language $K \in T^\infty$ is expressible in* $\mathrm{LTL}_T(\mathsf{XU})$.

We fix $T = h(b\Gamma)$ and we define the mapping $\sigma : (b\Gamma)^\infty \to T^\infty$ by $\sigma(t) = h(bt_1)h(bt_2) \cdots$ if $t = bt_1 bt_2 \cdots$ with $t_i \in \Gamma$ for $i \geq 1$. Note that the mapping σ is well-defined since each trace $t \in (b\Gamma)^\infty$ has a unique factorization $t = bt_1 bt_2 \cdots$ with $t_i \in \Gamma$ for $i \geq 1$.

Lemma 14. *Let $L \subseteq \mathbb{R}$ be recognized by h. Then,*

1. *$L \cap (b\Gamma)^\omega = \sigma^{-1}(K)$ for some K expressible in* $\mathrm{LTL}_T(\mathsf{XU})$.
2. *$L \cap (b\Gamma)^+ = \sigma^{-1}(K)$ for some K expressible in* $\mathrm{LTL}_T(\mathsf{XU})$.

Proof. 1. Let $K = \sigma(L \cap (b\Gamma)^\omega)$. We first show that $L \cap (b\Gamma)^\omega = \sigma^{-1}(K)$. The inclusion $\subseteq$ is clear. Conversely, let $t = bt_1 bt_2 \cdots \in L \cap (b\Gamma)^\omega$ and $s \in \sigma^{-1}(\sigma(t))$. Then, $s = bs_1 bs_2 \cdots \in (b\Gamma)^\omega$ with $h(bt_i) = h(bs_i)$ for all i. Therefore, $s \sim_h t$ and $t \in L$ implies $s \in L$.

Next, we show that K is recognized by the evaluation morphism $e : T^* \to S$ defined by $e(u) = u$ for all $u \in T$.

Let $v, w \in T^\omega$ with $v \sim_e w$ and $v \in K$. Write $v = v_1 v_2 \cdots \in T^\omega$ and $w = w_1 w_2 \cdots \in T^\omega$. Since $T = h(b\Gamma)$ we find $t = bt_1 bt_2 \cdots \in L$ with $t_i \in \Gamma$ and $h(bt_i) = v_i$ for all $i > 1$; and $s = bs_1 bs_2 \cdots$ with $s_i \in \Gamma$ and $h(bs_i) = w_i$ for all $i > 1$. From $v \sim_e w$ we deduce that $t \sim_h s$. And since $t \in L$, we get $s \in L$ and then $w = \sigma(s) \in K$.

Since S is an aperiodic monoid, $K \subseteq T^\omega$ is an aperiodic word language over T and by Proposition 13 we get K expressible in $\mathrm{LTL}_T(\mathsf{XU})$. Note that this is the only place where we use S aperiodic.

2. Similar. ☐

Next we show show how to lift an $\mathrm{LTL}_T(\mathsf{XU})$ formula for $K \subseteq T^\infty$ to a $\mathrm{LocTL}^{\mathrm{i}}$ formula for $\sigma^{-1}(K) \in (b\Gamma)^\infty$.

Lemma 15. *Suppose that any aperiodic trace language over B is expressible in LocTL_B. Then, for all $f \in \mathrm{LTL}_T(\mathsf{XU})$ there exists $\widetilde{f} \in \mathrm{LocTL}^{\mathrm{i}}_\Sigma$ such that for all $t = t_1 t'$ with $t_1 \in \mathbb{R}$ and $t' \in (b\Gamma)^\infty \setminus \{1\}$, we have*

$$\sigma(t') \models f \text{ iff } t, \min(t') \models \widetilde{f}.$$

Proof. We use an induction on f. Clearly, $\widetilde{\neg f} = \neg \widetilde{f}$ and $\widetilde{f \vee g} = \widetilde{f} \vee \widetilde{g}$. Now, we consider the case $f \,\mathsf{XU}\, g$ and we claim that

$$\widetilde{f \,\mathsf{XU}\, g} = \mathsf{EX}\big((\neg b \vee \widetilde{f}) \,\mathsf{U}\, (b \wedge \widetilde{g})\big).$$

Let $t' = t_2 t_3 \cdots$ with $t_i \in b\Gamma$ for all $i > 1$ and let $x = \min(t')$.

Assume that $\sigma(t') \models f \,\mathsf{XU}\, g$ and let $j > 2$ be such that $\sigma(t_j t_{j+1} \cdots) \models g$ and $\sigma(t_k t_{k+1} \cdots) \models f$ for all $2 < k < j$. Let $x' \in t$ with $x \lessdot x'$ and $z = \min(t_j)$. We have $x' \leq z$, $\lambda(z) = b$ and $t, z \models \widetilde{g}$ by induction. Now, let $y \in t$ with $x' \leq y < z$. Either $\lambda(y) \neq b$ and $t, y \models \neg b$. Or there exists $2 < k < j$ with $y = \min(t_k)$ and by induction $t, y \models \widetilde{f}$. Therefore, $t, x' \models (\neg b \vee \widetilde{f}) \,\mathsf{U}\, (b \wedge \widetilde{g})$ and $t, x \models \mathsf{EX}\big((\neg b \vee \widetilde{f}) \,\mathsf{U}\, (b \wedge \widetilde{g})\big)$.

Conversely, assume that $t, x \models \mathsf{EX}((\neg b \vee \widetilde{f}) \,\mathsf{U}\, (b \wedge \widetilde{g}))$. Let $x', z \in t$ with $x \lessdot x' \leq z$, $t, z \models b \wedge \widetilde{g}$ and $t, y \models \neg b \vee \widetilde{f}$ for all $x' \leq y < z$. Since $\lambda(z) = b$, there exists $j > 2$ with $z = \min(t_j)$ and by induction, we get $\sigma(t_j t_{j+1} \cdots) \models g$. Now, let $2 < k < j$ and $y = \min(t_k)$. We have $x' \leq y < z$ and since $\lambda(y) = b$ we get $t, y \models \widetilde{f}$ and $\sigma(t_k t_{k+1} \cdots) \models f$ by induction. Therefore, $\sigma(t') \models f \,\mathsf{XU}\, g$.

It remains to deal with a formula of the form $f = s \in T$. For all $r \in S$, the trace language $h^{-1}(r) \cap \mathbb{M}_B$ is aperiodic and therefore expressible in LocTL_B by the hypothesis of the lemma: we find $\alpha_r \in \mathrm{LocTL}_B$ such that for all $t'' \in \mathbb{M}_B$, $h(t'') = r$ if and only if $t'' \models \alpha_r$. Let $\overline{\alpha_r} \in \mathrm{LocTL}^{\mathrm{i}}_\Sigma$ be the formula obtained using Lemma 9. We claim that

$$\widetilde{s} = \bigvee_{h(b) \cdot r = s} \overline{\alpha_r}.$$

We write $t' = b t_2 t_3$ with $t_2 \in \Gamma$ and $t_3 \in (b\Gamma)^\infty$. Let $r = h(t_2)$. Then, $t_2 \models \alpha_r$ and by Lemma 9 we obtain $t, \min(t') \models \widetilde{\alpha_r}$. Now, if $\sigma(t') \models s$ then $s = h(bt_2) = h(b)r$ and we get $t, \min(t') \models \widetilde{s}$.

Conversely, assume that $t, \min(t') \models \overline{\alpha_r}$ for some r with $s = h(b)r$. By Lemma 9 we get $t_2 \models \alpha_r$ and $h(t_2) = r$. Hence, $h(bt_2) = s$ and $\sigma(t') \models s$. ☐

Lemma 16. *Suppose that any aperiodic trace language over B is expressible in LocTL_B. Let $A \subseteq \Sigma$ be non-empty and let $f \in \mathrm{LTL}_T(\mathsf{XU})$. There exists $\widetilde{f} \in \mathrm{LocTL}^{\mathrm{i}}_\Sigma$ such that for all $t = t_1 t_2 t_3$ with $t_1 \in \mathbb{R}$, $t_2 \in (b\Gamma)^*$, $t_3 \in b\Gamma^A \Omega_A$, we have*

$$\sigma(t_2) \models f \text{ iff } t, \min(t_2 t_3) \models \widetilde{f}.$$

Proof. Similar to that of Lemma 15. For the case $f \,\mathsf{XU}\, g$, we need to ensure that we stay within t_2. For this, we chose some letter $a \in A$ and we consider the formula $b \wedge \mathsf{F}_b^\infty a$ which characterizes the vertex $\min(t_3)$ within the trace t. We claim that

$$\widetilde{f \,\mathsf{XU}\, g} = \mathsf{EX}((\neg b \vee \widetilde{f}) \,\mathsf{U}\, (b \wedge \widetilde{g} \wedge \mathsf{F}(b \wedge \mathsf{F}_b^\infty a))).$$

Write $t_2 = s_1 \cdots s_n$ with $n \geq 0$ and $s_i \in b\Gamma$ and let $x = \min(t_2 t_3)$. Assume that $\sigma(t_2) \models f \,\mathsf{XU}\, g$, then we must have $n > 0$ and we find $1 < j \leq n+1$ with $\sigma(s_j \cdots s_n) \models g$ and $\sigma(s_k \cdots s_n) \models f$ for all $1 < k < j$. Let $y = \min(s_j \cdots s_n t_3)$. We have $t, y \models b \wedge \mathsf{F}(b \wedge \mathsf{F}_b^\infty a)$ and by induction $t, y \models \widetilde{g}$. Also $j > 1$ implies $x < y$ and we find $x \lessdot x' \leq y$. Finally, if $x' \leq z < y$ and $\lambda(z) = b$ then $z = \min(s_k \cdots s_n t_3)$ for some $1 < k < j$ and we get $t, z \models \widetilde{f}$ by induction. Therefore, $t, x \models \widetilde{f \,\mathsf{U}\, g}$.

Conversely, assume that $t, x \models \widetilde{f \,\mathsf{U}\, g}$. Let $x \lessdot x' \leq y$ with $t, y \models b \wedge \widetilde{g} \wedge \mathsf{F}(b \wedge \mathsf{F}_b^\infty a)$ and $t, z \models \neg b \vee \widetilde{f}$ for all $x' \leq z < y$. Since $t, y \models \mathsf{F}(b \wedge \mathsf{F}_b^\infty a)$ we have $y \leq \min(t_3)$ and therefore $x \in t_2$ is nonempty. Let j be such that $y = \min(s_j \cdots s_n t_3)$. Then, $1 < j \leq n+1$ and by induction we have $\sigma(s_j \cdots s_n) \models g$. Let $1 < k < j$ and $z = \min(s_k \cdots s_n t_3)$. We have $\lambda(z) = b$ and $x' \leq z < y$. Hence $\sigma(s_k \cdots s_n) \models f$ by induction. Therefore, $\sigma(t_2) = \sigma(s_1 \cdots s_n) \models f \,\mathsf{XU}\, g$.

Now, we consider the case $f = s \in T$. For all $r \in S$, the trace language $h^{-1}(r) \cap \mathbb{M}_B$ is aperiodic and therefore expressible in LocTL_B by the hypothesis of the lemma: we find $\alpha_r \in \mathrm{LocTL}_B$ such that for all $t'' \in \mathbb{M}_B$, $h(t'') = r$ if and only if $t'' \models \alpha_r$. Let $\overline{\alpha_r} \in \mathrm{LocTL}_\Sigma^{\mathrm{i}}$ be the formula obtained using Lemma 9. We prove as in Lemma 15 that

$$\widetilde{s} = \mathsf{EX}\,\mathsf{F}(b \wedge \mathsf{F}_b^\infty a) \wedge \bigvee_{h(b)\cdot r = s} \overline{\alpha_r}.$$

If $\sigma(t_2) \models s$ then $t_2 \in (b\Gamma)^+$ and $t, \min(t_2 t_3) \models \mathsf{EX}\,\mathsf{F}(b \wedge \mathsf{F}_b^\infty a)$. Now, write $t_2 = bt't''$ with $t' \in \Gamma$, $t'' \in (b\Gamma)^*$ and let $r = h(t')$. We have $h(b)r = s$. Since $h(t') = r$, we have $t' \models \alpha_r$ and by Lemma 9 we get $t, \min(t_2 t_3) \models \overline{\alpha_r}$. Therefore, $t, \min(t_2 t_3) \models \widetilde{s}$.

Conversely, if $t, \min(t_2 t_3) \models \mathsf{EX}\,\mathsf{F}(b \wedge \mathsf{F}_b^\infty a)$ then $t_2 \in (b\Gamma)^+$ and we can write $t_2 = bt't''$ with $t' \in \Gamma$ and $t'' \in (b\Gamma)^*$. Assume that $t, \min(bt't''t_3) \models \overline{\alpha_r}$ for some $r \in S$ with $s = h(b)r$. From Lemma 9 we get $t' \models \alpha_r$ and $h(t') = r$. We deduce $h(bt') = s$ and $\sigma(t_2) \models s$. $\square$

Lemma 17. *Suppose that for any proper subset A of Σ, any aperiodic trace language over A is expressible in* LocTL_A. *Let $L \subseteq \mathbb{R}$ be an aperiodic trace language over Σ. Then, for all $b \in \Sigma$, there exists $\varphi \in \mathrm{LocTL}_\Sigma^{\mathrm{i}}$ such that for all $t = t_0 t'$ with $t_0, t' \in \mathbb{R}$ and $\min(t') = \{b\}$, $t' \in L$ iff $t, \min(t') \models \varphi$.*

Proof. We prove this lemma by induction on the size of the alphabet Σ. If $\Sigma = \emptyset$ then there is nothing to prove. Now, suppose that $\Sigma \neq \emptyset$ and let $b \in \Sigma$. We assume that L is recognized by the aperiodic morphism $h : \mathbb{M} \to S$. Now, $L \cap (\min = \{b\})$ can be written as

$$(L \cap ((b\Gamma)^\infty \setminus \{1\})) \uplus \bigcup_{\emptyset \neq A \subseteq B} (L \cap (b\Gamma)^* b \Gamma^A \Omega_A).$$

By Lemma 14 we get $L \cap ((b\Gamma)^\infty \setminus \{1\}) = \sigma^{-1}(\mathcal{L}(f))$ for some $f \in \mathrm{LTL}_T(\mathsf{XU})$. From the hypothesis, aperiodic languages over B are expressible in LocTL_B. Hence, we can apply Lemma 15 and we get $\widetilde{f}$ such that for all $t = t_0 t'$ with $t_0 \in \mathbb{R}$ and $t' \in (b\Gamma)^\infty \setminus \{1\}$, we have $\sigma(t') \models f$ iff $t, \min(t') \models \widetilde{f}$. We conclude this case taking $\varphi = \beta \wedge \widetilde{f}$ where β is defined in Lemma 11.

Now, we consider $L \cap (b\Gamma)^* b\Gamma^A \Omega_A$ where $\emptyset \neq A \subseteq B$. By Lemma 12,

$$L \cap (b\Gamma)^* b\Gamma^A \Omega_A \;=\; \bigcup_{\text{finite}} (L_1 \cap (b\Gamma)^*) b (L_2 \cap \Gamma^A)(L_3 \cap \Omega_A)$$

where each L_i is an aperiodic language recognized by h. Thus, it suffices to show that for aperiodic languages L_1, L_2 and L_3 recognized by h, there is a formula φ such that for all $t = t_0 t'$ with $t_0, t' \in \mathbb{R}$ and $\min(t') = \{b\}$, we have $t, \min(t') \models \varphi$ if and only if $t' \in (L_1 \cap (b\Gamma)^*) b (L_2 \cap \Gamma^A)(L_3 \cap \Omega_A)$.

By Lemma 14 we get $L_1 \cap (b\Gamma)^* = \sigma^{-1}(\mathcal{L}(f_1))$ for some $f_1 \in \mathrm{LTL}_T(\mathsf{XU})$. From the hypothesis, aperiodic languages over B are expressible in LocTL_B. Hence, we can apply Lemma 16 and we get $\widetilde{f_1}$ such that for all $t = t_0 t_1 t''$ with $t_0 \in \mathbb{R}$ and $t_1 \in (b\Gamma)^*$, and $t'' \in b\Gamma^A \Omega_A$, we have $t_1 \in L_1$ iff $t, \min(t_1 t'') \models \widetilde{f_1}$.

Using again the hypothesis of the lemma, we get some formula $\alpha_2 \in \mathrm{LocTL}_B$ such that $L_2 \cap \mathbb{R}_B = \mathcal{L}(\alpha_2)$. By Lemma 9 we find $\overline{\alpha_2} \in \mathrm{LocTL}^{\mathrm{i}}_\Sigma$ such that for all $t = t_0 t_1 b t_2 t_3$ with $t_0 \in \mathbb{R}$ and $t_1 \in (b\Gamma)^*$, $t_2 \in \Gamma^A$ and $t_3 \in \Omega_A$, we have $t_2 \in L_2$ iff $t, \min(b t_2 t_3) \models \overline{\alpha_2}$.

Finally, L_3 is an aperiodic trace language over a smaller alphabet (since $A \neq \emptyset$, $I(A)$ is a proper subset of Σ) and hence by induction hypothesis there is a formula φ_3 such that for all $t = t_0 t_1 b t_2 t_3$ with $t_0 \in \mathbb{R}$ and $t_1 \in (b\Gamma)^*$, $t_2 \in \Gamma^A$ and $t_3 \in \Omega_A$ with $t_3 \neq 1$, we have $t_3 \in L_3$ iff $t, \min(t_3) \models \varphi_3$.

Putting these three pieces together we let

$$\psi \;=\; \widetilde{f_1} \wedge \mathsf{F}(b \wedge \mathsf{F}_b^\infty\, a \wedge \overline{\alpha_2} \wedge (\varphi_4 \vee \mathsf{F}_b\, \mathsf{EX}(b \wedge \varphi_3)))$$

with $\varphi_4 = \bot$ if $1 \notin L_3$ and $\varphi_4 = \neg\, \mathsf{EX}\, \mathsf{F}\, b$ otherwise. Then, for all $t = t_0 t_1 b t_2 t_3$ with $t_0 \in \mathbb{R}$ and $t_1 \in (b\Gamma)^*$, $t_2 \in \Gamma^A$ and $t_3 \in \Omega_A$, we get from the above discussion and using the fact that $t, x \models \mathsf{F}_b\, \mathsf{EX}(b \wedge \varphi)$ if and only if $t, y \models b \wedge \varphi$ for some $y > x$ such that for all $z < y$, $\lambda(z) = b$ implies $z \leq x$ that $t_1 b t_2 t_3 \in L_1 b L_2 L_3$ if and only if $t, \min(t_1 b t_2 t_3) \models \psi$.

We complete the proof with $\varphi = \gamma \wedge \psi$ where γ is the formula defined in Lemma 11. $\square$

Theorem 18. *Any aperiodic real trace language over* $\mathbb{R}(\Sigma, D)$ *is expressible in* LocTL_Σ.

Proof. The proof proceeds by induction on the size of Σ. The base case is when $\Sigma = \{a\}$ is a singleton. Then L is either a finite set or the union of a finite set and a set of the form $a^n a^*$ for some $n \geq 0$. In both cases, we see that L is expressible in LocTL_Σ. For instance, $a^3 a^*$ is expressed by the formula $\mathsf{EM}\, \mathsf{EX}\, \mathsf{EX}\, \top$. Moreover, $\{a^3\} = a^3 a^* \setminus a^4 a^*$.

For the inductive step, assume that the theorem holds for any aperiodic language over any proper subset of Σ. Let L be recognized by an aperiodic morphism $h : \mathbb{M} \to S$. Let $b \in \Sigma$ and $B = \Sigma \setminus \{b\}$ as usual. We can show as in Lemma 12 that L can be written as follows:

$$L \;=\; \bigcup_{\text{finite}} (L_1 \cap \mathbb{R}_B)(L_2 \cap (\min \subseteq \{b\}))$$

where L_1 and L_2 are languages recognized by the same aperiodic morphism h.

Since the decomposition of any trace $t \in \mathbb{R}$ as $t_1 t_2$ with $t_1 \in \mathbb{R}_B$ and $t_2 \in (\min \subseteq \{b\})$ is unique, it follows that the above decomposition can be rewritten as

$$L \;=\; \bigcup_{\text{finite}} (L_1 \cap \mathbb{R}_B)(\min \subseteq \{b\}) \cap (\mathbb{R}_B(L_2 \cap (\min \subseteq \{b\})))$$

Thus, it suffices to show that $(L_1 \cap \mathbb{R}_B)(\min \subseteq \{b\})$ and $\mathbb{R}_B(L_2 \cap (\min \subseteq \{b\}))$ are expressible in LocTL_Σ.

Now, by the induction hypothesis, there is formula α_1 in LocTL_B such that for $t_1 \in \mathbb{R}_B$, $t_1 \models \alpha_1$ if and only if $t_1 \in L_1$. Thus, by Lemma 10, there is a formula $\widetilde{\alpha_1}$ in LocTL_Σ such that $t \models \widetilde{\alpha_1}$ if and only if $t_1 \models \alpha_1$ whenever $t = t_1 t_2$ with $t_1 \in \mathbb{R}_B$ and $\min(t_2) \subseteq \{b\}$. Thus, $(L_1 \cap \mathbb{R}_B)(\min \subseteq \{b\}) = \mathcal{L}(\widetilde{\alpha_1})$ is expressible in LocTL_Σ.

Now, since we have assumed expressive completeness for every proper subset of Σ, we may appeal to Lemma 17 to conclude that there is a formula φ_2 such that for any $t = t_1 t_2$ with $\min(t_2) = b$, $t_2 \in L_2$ if and only if $t, \min(t_2) \models \varphi_2$. Consider the formula

$$\alpha = \alpha' \vee \mathsf{EM}((b \wedge \varphi_2) \vee (\neg b \wedge \mathsf{F}_b\, \mathsf{EX}(b \wedge \varphi_2)))$$

where $\alpha' = \perp$ if $1 \notin L_2$ and $\alpha' = \neg\, \mathsf{EM}\,\mathsf{F}\, b$ otherwise. Then, $t \models \alpha$ if and only if either $t \in \mathbb{R}_B$ and $1 \in L_2$, or there is a minimal b event x in the trace t and $t, x \models \varphi_2$. That is $t = t_1 t_2$ with $t_1 \in \mathbb{R}_B$, $\min(t_2) = \{b\}$ and $t_2 \in L_2$. Thus $\mathbb{R}_B(L_2 \cap (\min \subseteq \{b\})) = \mathcal{L}(\alpha)$ is also expressible in LocTL_Σ. $\square$

References

1. B. Adsul and M. Sohoni. Complete and tractable local linear time temporal logics over traces. In *Proc. of ICALP'02*, number 2380 in LNCS, pages 926–937. Springer Verlag, 2002.
2. J. Cohen, D. Perrin, and J.-E. Pin. On the expressive power of temporal logic. *Journal of Computer and System Sciences*, 46:271–295, 1993.
3. C. Courcoubetis, M.Y. Vardi, P. Wolper, and M. Yannakakis. Memory efficient algorithms for the verification of temporal properties. *formal Methods in System Design*, 1:275–288, 1992.
4. V. Diekert and P. Gastin. Local temporal logic is expressively complete for cograph dependence alphabets. In *Proc. of LPAR'01*, number 2250 in LNAI, pages 55–69. Springer Verlag, 2001.
5. V. Diekert and P. Gastin. LTL is expressively complete for Mazurkiewicz traces. *Journal of Computer and System Sciences*, 64:396–418, 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, A. Pnueli, S. Shelah, and J. Stavi. On the temporal analysis of fairness. In *Proc. of PoPL'80*, pages 163–173, Las Vegas, Nev., 1980.
10. P. Gastin and D. Kuske. Satisfiability and model checking for MSO-definable temporal logics are in PSPACE. Tech. Rep. LIAFA, Université Paris 7 (France), 2003. http://www.liafa.jussieu.fr/∼gastin/Articles/gaskus03.html.
11. 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 Verlag, 2002.
12. J.A.W. Kamp. *Tense Logic and the Theory of Linear Order*. PhD thesis, University of California, Los Angeles, California, 1968.
13. A. Mazurkiewicz. Concurrent program schemes and their interpretations. DAIMI Rep. PB 78, Aarhus University, Aarhus, 1977.
14. R. McNaughton and S. Papert. *Counter-Free Automata*. MIT Press, 1971.
15. D. Perrin. Recent results on automata and infinite words. In *Proc. of MFCS'84*, number 176 in LNCS, pages 134–148. Springer Verlag, 1984.
16. D. Perrin and J.-E. Pin. First order logic and star-free sets. *Journal of Computer and System Sciences*, 32:393–406, 1986.
17. A. Pnueli. The temporal logic of programs. In *Proc. of FOCS'77*, pages 46–57, 1977.
18. M.-P. Schützenberger. On finite monoids having only trivial subgroups. *Information and Control*, 8:190–194, 1965.
19. 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.
20. W. Thomas. Star-free regular sets of ω-sequences. *Information and Control*, 42:148–156, 1979.
21. M.Y. Vardi. An automata-theoretic approach to linear temporal logic. In *Logics for Concurrency: Structure versus Automata*, number 1043 in LNCS, pages 238–266. Springer Verlag, 1996.
22. M.Y. Vardi and P. Wolper. An automata-theoretic approach to automatic program verification. In *Proc. of LICS'86*, pages 332–344, 1986.
23. I. Walukiewicz. Difficult configurations – on the complexity of LTrL. In *Proc. of ICALP'98*, number 1443 in LNCS, pages 140–151. Springer Verlag, 1998.
24. I. Walukiewicz. Local logics for traces. *Journal of Automata, Languages and Combinatorics*, 7(2):259–290, 2002.
25. Th. Wilke. Classifying discrete temporal properties. In *Proc. of STACS'99*, number 1563 in LNCS, pages 32–46. Springer Verlag, 1999.
26. W. Zielonka. Notes on finite asynchronous automata. *R.A.I.R.O. — Informatique Théorique et Applications*, 21:99–135, 1987.