Sheffield Hallam University

On the expressiveness and monitoring of metric temporal logic

HO, H.M. <http://orcid.org/0000-0003-0387-4857>, OUAKNINE, J and WORRELL, J

Available from Sheffield Hallam University Research Archive (SHURA) at:

https://shura.shu.ac.uk/25236/

This document is the Published Version [VoR]

Citation:

HO, H.M., OUAKNINE, J and WORRELL, J (2019). On the expressiveness and monitoring of metric temporal logic. Logical Methods in Computer Science, 15 (2), 13:1-13:52. [Article]

Copyright and re-use policy

See http://shura.shu.ac.uk/information.html

Sheffield Hallam University Research Archive
http://shura.shu.ac.uk

Logical Methods in Computer Science
Volume 15, Issue 2, 2019, pp. 13:1–13:52
https://lmcs.episciences.org/

Submitted Mar. 13, 2018
Published May 10, 2019

ON THE EXPRESSIVENESS AND MONITORING OF METRIC TEMPORAL LOGIC

HSI-MING HO [a], JOËL OUAKNINE [b], AND JAMES WORRELL [a]

[a] Department of Computer Science, University of Oxford, Oxford, UK
e-mail address: hsimho@gmail.com, jbw@cs.ox.ac.uk

[b] Max Planck Institute for Software Systems, Saarland Informatics Campus, Germany
e-mail address: joel@mpi-sws.org

Abstract. It is known that *Metric Temporal Logic* (MTL) is strictly less expressive than the *Monadic First-Order Logic of Order and Metric* ($\mathsf{FO}[<,+1]$) when interpreted over timed words; this remains true even when the time domain is bounded *a priori*. In this work, we present an extension of MTL with the same expressive power as $\mathsf{FO}[<,+1]$ over bounded timed words (and also, trivially, over time-bounded signals). We then show that expressive completeness also holds in the general (time-unbounded) case if we allow the use of rational constants $q \in \mathbb{Q}$ in formulas. This extended version of MTL therefore yields a definitive real-time analogue of Kamp's theorem. As an application, we propose a *trace-length independent* monitoring procedure for our extension of MTL, the first such procedure in a dense real-time setting.

1. Introduction

Expressiveness of metric temporal logics. One of the most prominent specification formalisms used in verification is *Linear Temporal Logic* (LTL), which is typically interpreted over the non-negative integers or reals. A celebrated result of Kamp [Kam68] states that, in either case, LTL has precisely the same expressive power as the *Monadic First-Order Logic of Order* ($\mathsf{FO}[<]$). These logics, however, are inadequate to express specifications for systems whose correct behaviour depends on quantitative timing requirements. Over the last three decades, much work has therefore gone into lifting classical verification formalisms and results to the real-time setting. *Metric Temporal Logic* (MTL),[1] which extends LTL by constraining the

Key words and phrases: Computer Science, Logic in Computer Science.

Part of this work appeared in the *Proceedings of the 8th International Workshop on Reachability Problems* (2014), and in the *Proceedings of the 5th International Conference on Runtime Verification* (2014).

Joël Ouaknine was supported by ERC grant AVS-ISS (648701) and by DFG grant 389792660 as part of TRR 248 (see https://perspicuous-computing.science). James Worrell was supported by EPSRC Fellowship EP/N008197/1.

[1] In this article, we refer to the logic with constrained 'Until' and 'Since' modalities exclusively as 'MTL', and use the term 'metric temporal logics' in a broader sense to refer to temporal logics with modalities definable in $\mathsf{FO}[<,+1]$ (see below).

LOGICAL METHODS IN COMPUTER SCIENCE
DOI:10.23638/LMCS-15(2:13)2019

© H.-M. Ho, J. Ouaknine, and J. Worrell
Creative Commons

modalities by time intervals, was introduced by Koymans [Koy90] in 1990 and has emerged as a central real-time specification formalism. MTL enjoys two main semantics, depending intuitively on whether atomic formulas are interpreted as *state predicates* or as (instantaneous) *events*. In the former, the system is assumed to be under observation at every instant in time, leading to a 'continuous' semantics based on *signals*, whereas in the latter, observations of the system are taken to be (finite or infinite) sequences of timestamped snapshots, leading to a 'pointwise' semantics based on *timed words*—this is the prevalent interpretation for systems modelled as timed automata [AD94]. In both cases, the time domain is usually taken to be the non-negative real numbers. Both semantics have been extensively studied; see, e.g., [OW08] for a historical account.

Alongside these developments, researchers proposed the *Monadic First-Order Logic of Order and Metric* ($\mathsf{FO}[<, +1]$) as a natural quantitative extension of $\mathsf{FO}[<]$. Like MTL, $\mathsf{FO}[<, +1]$ can be interpreted over signals [HR04] or timed words [Wil94]. An obvious question to ask is whether MTL has the same expressive power as $\mathsf{FO}[<, +1]$, i.e., an analogue of Kamp's theorem holds in the real-time setting. Unfortunately, Hirshfeld and Rabinovich [HR07] showed that no 'finitary' extension of MTL—and *a fortiori* MTL itself—could have the same expressive power as $\mathsf{FO}[<, +1]$ over the reals.[2] Still, in the continuous semantics, MTL can be made expressively complete for $\mathsf{FO}[<, +1]$ by extending the logic with an infinite family of '*counting modalities*' [Hun13] or considering only *bounded* time domains [ORW09, OW10]. Nonetheless, and rather surprisingly, MTL with counting modalities remains strictly less expressive than $\mathsf{FO}[<, +1]$ over bounded time domains in the pointwise semantics, i.e., over timed words of bounded duration.

Monitoring of real-time specifications. In recent years, *runtime verification* (see [LS09, SHL11] for surveys) has emerged as a light-weight complementary technique to *model checking* [CE81, QS82]. It is particularly useful for systems whose internal details are either too complex to be modelled faithfully or simply not accessible. Roughly speaking, while in model checking one considers all behaviours of the model, in runtime verification one focusses on one particular behaviour—the current one. Given a specification φ and a finite timed word ρ (which we call a finite *trace* in this context), the *prefix* problem asks whether all infinite traces extending ρ satisfy φ. The *monitoring* problem, as far as we are concerned here, can be seen as an *online* version of the prefix problem where ρ grows incrementally (i.e., one event at a time): the monitoring procedure (*monitor*) for φ is executed in parallel with the system under scrutiny, and it is required to output an answer when either (i) all infinite extensions of the current trace satisfy φ, or (ii) no infinite extension of the current trace can satisfy φ.

Ideally, we would also like to require a monitoring procedure to be *trace-length independent* [Roş12, BKV13] in the sense that its time and space requirements should not depend on the length of the input trace (this is important since input traces in practical applications tend to be very long; cf., e.g., [BCE$^+$14]). In the untimed case, this is not difficult to achieve: one can translate LTL formulas into Büchi automata [GO03, DKL10] and turn them into efficient trace-length independent monitors [ABLS05]. Unfortunately, a number of obstacles hinder the application of this methodology to the real-time setting: it is known that MTL is expressively incomparable with timed automata; even though certain fragments

[2] Hirshfeld and Rabinovich's result was only stated and proved for the continuous semantics, but we believe that their approach would also carry through for the pointwise semantics. In any case, using different techniques Prabhakar and D'Souza [PD06] and Pandya and Shah [PS11] independently showed that MTL is strictly weaker than $\mathsf{FO}[<, +1]$ in the pointwise semantics.

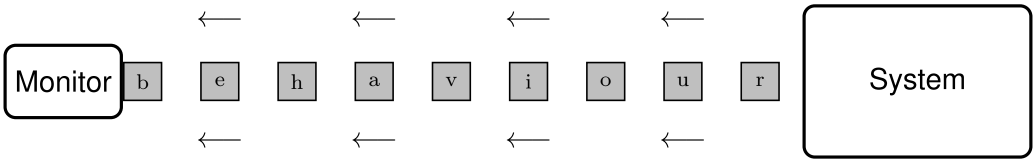


FIGURE 1. A monitor receives the trace incrementally (one event at a time).

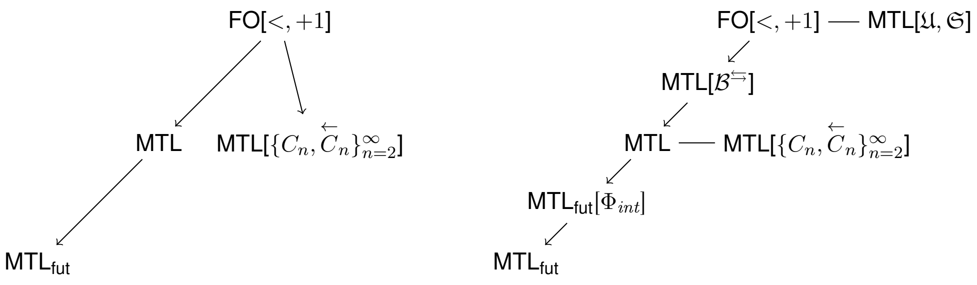
of MTL can be translated into timed automata, the latter are not always determinisable as required for the purpose of monitoring [BBBB09]. For this reason, researchers purposed automata-free monitoring procedures that work directly with metric temporal logic formulas (e.g., [MN04, BKZ11]). However, it proved difficult to maintain trace-length independence while allowing MTL in its full generality, i.e., with unbounded intervals and nesting of future and past modalities. Almost all monitoring procedures for metric temporal logics in the literature have certain syntactic or semantic limitations, e.g., only allowing bounded future modalities or assuming integer time. A notable exception is [BN12] which handles full MTL over signals, but which unfortunately fails to be trace-length independent.

Contributions. We study the expressiveness of various fragments and extensions of MTL over timed words. In particular, we highlight a fundamental deficiency in the pointwise interpretation of MTL. To amend this, we propose new (first-order definable) modalities *generalised 'Until'* ($\mathfrak{U}$) and *generalised 'Since'* ($\mathfrak{S}$). With these new modalities and the techniques developed in [PS11, ORW09, HOW13], we establish the following results:

(i). There is a strict hierarchy of metric temporal logics based on their expressiveness over bounded timed words (see Figure 2 where the arrows indicate 'strictly more expressive than' and the edges indicate 'equally expressive'). Note that this hierarchy collapses in the continuous semantics.
(ii). The metric temporal logic with the new modalities $\mathfrak{U}$ and $\mathfrak{S}$ (denoted $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$) is expressively complete for $\mathsf{FO}[<, +1]$ over bounded timed words.
(iii). The time-bounded satisfiability and model-checking problems for $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ are EXPSPACE-complete, the same as that of MTL.
(iv). Any $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula is equivalent to a *syntactically separated* one.
(v). $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ is expressively complete for $\mathsf{FO}[<, +\mathbb{Q}]$ (the rational variant of $\mathsf{FO}[<, +1]$) over unbounded (i.e., infinite non-Zeno) timed words if we allow the use of rational constants in modalities.

For monitoring, we focus on a restricted version of the monitoring problem of $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$, based on the notion of *informative prefixes* [KV01]. The main idea of our approach is to work with $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas of a special form: LTL formulas over atomic formulas comprised of bounded $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas.[3] The truth values of bounded $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas can be computed and stored efficiently with a dynamic programming algorithm; these values are then used as input to deterministic finite automata obtained from 'backbone' LTL formulas. As a result, we obtain the first trace-length independent monitoring procedure for a metric temporal logic that subsumes MTL. The procedure is free of dynamic memory allocations,

[3] It follows from the syntactic separation result that no expressiveness is sacrificed in restricting to this fragment.


$\mathsf{FO}[<,+1]$
MTL
$\mathsf{MTL}[\{C_n, \overleftarrow{C}_n\}_{n=2}^{\infty}]$
$\mathsf{MTL}_{\mathsf{fut}}$
$\mathsf{FO}[<,+1]$
$\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$
$\mathsf{MTL}[\mathcal{B}^{\leftrightarrows}]$
MTL
$\mathsf{MTL}[\{C_n, \overleftarrow{C}_n\}_{n=2}^{\infty}]$
$\mathsf{MTL}_{\mathsf{fut}}[\Phi_{int}]$
$\mathsf{MTL}_{\mathsf{fut}}$

known results our results

Figure 2. Expressiveness results over bounded timed words.

linked lists, etc., and hence can be implemented efficiently (the *amortised* running time per event is linear in the number of subformulas in all bounded formulas). To be more precise:

(vi). We give a trace-length independent monitoring procedure (which detects informative good/bad prefixes) for LTL formulas over atomic formulas comprised of bounded $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas.

(vii). For an arbitrary $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula, we show that its informative good/bad prefixes are preserved by the syntactic rewriting rules (and thus can be monitored in a trace-length independent manner).

Related work. Bouyer, Chevalier, and Markey [BCM05] showed that $\mathsf{MTL}_{\mathsf{fut}}$ (the future-only fragment of MTL) is strictly less expressive than MTL in both the continuous and pointwise semantics. This, together with the aforementioned results [HR07, PS11], form a strict hierarchy of expressiveness that holds in the both semantics:

$$\mathsf{MTL}_{\mathsf{fut}} \subsetneq \mathsf{MTL} \subsetneq \mathsf{FO}[<,+1]\,.$$

Ouaknine, Rabinovich, and Worrell [ORW09] showed that the hierarchy collapses in the continuous semantics when one considers bounded time domains of the form $[0, N)$. Our results show that this is not the case in the pointwise semantics.

Another route to expressive completeness is by allowing rational constants. In particular, counting modalities become expressible in MTL [BCM05]. Exploiting this observation, Hunter, Ouaknine, and Worrell [HOW13] showed that MTL with rational constants is expressively complete for $\mathsf{FO}[<,+\mathbb{Q}]$ in the continuous semantics. However, as can be immediately derived from a result of Prabhakar and D'Souza [PD06], this pleasant result does not hold in the pointwise semantics. On the other hand, D'Souza and Tabareau [DT04] showed that MTL with rational constants is expressively complete for $\mathsf{rec\text{-}TFO}[\diamondsuit, \overleftarrow{\diamondsuit}]$ (an 'input-determined' fragment of $\mathsf{FO}[<,+\mathbb{Q}]$) in the pointwise semantics. We complement these results by extending MTL with the new modalities $\mathfrak{U}$ and $\mathfrak{S}$ to make it expressively complete for $\mathsf{FO}[<,+\mathbb{Q}]$ in the pointwise semantics.

In a pioneering work, Thati and Roşu [TR05] proposed a rewriting-based monitoring procedure for MTL over integer-timed traces. Their procedure is trace-length independent and amenable to efficient implementations. However, trace-length independent monitoring of MTL is not possible in dense real-time settings: a monitor would have to 'remember' an infinite number of timestamps. For this reason, researchers often impose a *bounded-variability* assumption on input traces, i.e., only a bounded number of events may occur in any time unit. Under such an assumption, Nickovic and Piterman [NP10] showed that $\mathsf{MTL}_{\mathsf{fut}}$ formulas can be translated into deterministic timed automata. Unfortunately, their approach does not easily extend to full MTL.

It is known that the non-punctual fragment of MTL, called MITL, can be translated into timed automata. Since the standard constructions [AH92, AFH96] are notoriously complicated, there have been some proposals for simplified or improved constructions [MNP06, KKP11, DM13, BEG14, BGHM17]. The difficulty in using these constructions for monitoring, again, lies in the fact that timed automata cannot be determinised in general. In principle one can carry out on-the-fly determinisation for input traces of bounded variability (cf., e.g., [Tri02, BBBB09]); however, it is not clear that this approach can yield an efficient procedure.

2. Preliminaries

2.1. **Automata and logics for real-time.**

Timed words. Let the *time domain* $\mathbb{T}$ be a subinterval of $\mathbb{R}_{\geq 0}$ that contains 0. A *time sequence* $\tau = \tau_0\tau_1 \ldots$ is a non-empty finite or infinite sequence over $\mathbb{T}$ (*timestamps*) that satisfies the requirements below (we denote the length of τ by $|\tau|$):

- *Initialisation*: $\tau_0 = 0$
- *Strict monotonicity*: For all i, $0 \leq i < |\tau| - 1$, we have $\tau_i < \tau_{i+1}$.[4]

If τ is infinite we require it to be unbounded, i.e., we disallow so-called Zeno sequences. Given a finite alphabet Σ, a $\mathbb{T}$-*timed word* over Σ is a pair $\rho = (\sigma, \tau)$ where $\sigma = \sigma_0\sigma_1 \ldots$ is a non-empty finite or infinite word over Σ and τ is a time sequence over $\mathbb{T}$ of the same length. We refer to a $\mathbb{T}$-timed word simply as a *timed word* when $\mathbb{T} = \mathbb{R}_{\geq 0}$.[5] We refer the pair (σ_i, τ_i) as the i^{th} *event* in ρ, and define the *distance* between i^{th} and j^{th} $(i \leq j)$ events to be $\tau_j - \tau_i$. In this way, a timed word can be equivalently regarded as a sequence of events. We denote by $|\rho|$ the number of events in ρ. A *position* in ρ is a number i such that $0 \leq i < |\rho|$. The *duration* of ρ is defined as $\tau_{|\rho|-1}$ if ρ is finite. We write $t \in \rho$ if t is equal to one of the timestamps in ρ. For a finite alphabet Σ, we write $T\Sigma^*$ and $T\Sigma^\omega$ for the respective sets of finite and infinite timed words over Σ. A *timed (finite-word) language* over Σ is a subset of $T\Sigma^\omega$ $(T\Sigma^*)$.

Timed automata. The most popular model for real-time systems are *timed automata* [AD94], introduced by Alur and Dill in the early 1990s. Timed automata extends finite automata by real-valued variables (called *clocks*).

Definition 2.1. Given a set of clocks X, the set $\mathcal{G}(X)$ of clock constraints g is defined inductively by

$$g ::= \mathbf{true} \mid x \bowtie c \mid g_1 \wedge g_2$$

where $x \in X$, $c \in \mathbb{N}$ and $\bowtie \in \{<, \leq, >, \geq\}$.

Definition 2.2. A (non-deterministic) timed automaton $\mathcal{A}$ is a tuple $\langle \Sigma, S, S_0, X, I, E, F \rangle$ where

[4] This requirement is chosen to simplify the presentation; all the results still hold (with some minor modifications) in the case of weakly-monotonic time, i.e., requiring instead $\tau_i \leq \tau_{i+1}$ for all i, $0 \leq i < |\tau| - 1$.

[5] By the non-Zeno requirement, if $\mathbb{T}$ is bounded then a $\mathbb{T}$-timed word must be a finite timed word.

- Σ is a finite alphabet
- S is a finite set of locations
- $S_0 \subseteq S$ is the set of initial locations
- X is a finite set of clocks
- $I : S \mapsto \mathcal{G}(X)$ is a mapping that labels each location in S with a clock constraint in $\mathcal{G}(X)$ (an 'invariant')
- $E \subseteq S \times \Sigma \times 2^X \times \mathcal{G}(X) \times S$ is the set of edges. An edge $\langle s, a, \lambda, g, s' \rangle$ denotes an a-labelled edge from location s to location s' where g (a 'guard') specifies when the edge is enabled and $\lambda \subseteq X$ is the set of clocks to be reset with this edge
- F is the set of accepting locations.

We say that $\mathcal{A}$ is *deterministic* if it (i) has only one initial location and (ii) for each $s \in S$, $a \in \Sigma$ and every pair of edges $\langle s, a, \lambda_1, g_1, s_1 \rangle$, $\langle s, a, \lambda_2, g_2, s_2 \rangle$, g_1 and g_2 are mutually exclusive (i.e., $g_1 \wedge g_2$ is unsatisfiable). We say that $\mathcal{A}$ is *complete* if for each $s \in S$ and $a \in \Sigma$, the disjunction of the clock constraints of the a-labelled edges starting at s is a valid formula.

Assume that $\mathcal{A}$ has n clocks. We define its set of clock values as $\mathsf{Val} = [0, c_{max}] \cup \{\top\}$ where c_{max} is the maximum constant appearing in $\mathcal{A}$. A *state* of $\mathcal{A}$ as a pair $(s, \mathbf{v})$ where $s \in S$ is a location and $\mathbf{v} \in \mathsf{Val}^n$ is a *clock valuation.* Write $\mathbf{v}(x)$ for the value of clock x in $\mathbf{v}$. We denote by $Q = S \times \mathsf{Val}^n$ the set of all states of $\mathcal{A}$. A *run* of $\mathcal{A}$ on a timed word can be seen as follows: the automaton takes some edge when an event arrives, otherwise it stays in the same location as time elapses. More precisely, $\mathcal{A}$ induces a labelled transition system $\mathcal{T}_{\mathcal{A}} = \langle Q, \rightsquigarrow, \rightarrow \rangle$ where $\rightsquigarrow \subseteq Q \times \mathbb{R}_{>0} \times Q$ is the *delay-step relation* and $\rightarrow \subseteq Q \times \Sigma \times Q$ is the *discrete-step relation.* In these steps, corresponding invariants and guards must be met (define $\top > c$ for all constants c):

- For $(s, \mathbf{v}) \overset{t}{\rightsquigarrow} (s', \mathbf{v}')$, $s' = s$, $\mathbf{v}' = \mathbf{v} + t$ and $\mathbf{v} + t' \models I(s)$ for all $0 \le t' \le t$.
- For $(s, \mathbf{v}) \xrightarrow{a} (s', \mathbf{v}')$, there is an edge $\langle s, a, \lambda, g, s' \rangle \in E$ such that $\mathbf{v}' = \mathbf{v}[\lambda := 0]$ and $\mathbf{v} \models g$.

The clock valuation $\mathbf{v} + t$ maps each clock x to $\mathbf{v}(x) + t$ if $\mathbf{v}(x) + t \le c_{max}$, otherwise $\top$. $\mathbf{v}[\lambda := 0]$ maps x to $\mathbf{v}(x)$ if $x \notin \lambda$, otherwise 0. Formally, a run of $\mathcal{A}$ on $\rho = (\sigma, \tau)$ is an alternating sequence of delay steps and discrete steps

$$(s_0, \mathbf{v}_0) \xrightarrow{\sigma_0} (s_1, \mathbf{v}_1) \overset{d_0}{\rightsquigarrow} (s_2, \mathbf{v}_2) \xrightarrow{\sigma_1} (s_3, \mathbf{v}_3) \overset{d_1}{\rightsquigarrow} (s_4, \mathbf{v}_4) \xrightarrow{\sigma_2} \dots$$

where $d_i = \tau_{i+1} - \tau_i$ for $i \ge 0$, $s_0 \in S_0$ and $\mathbf{v}_0 = 0^n$. A finite timed word ρ' is *accepted* by $\mathcal{A}$ if there is an *accepting* run (i.e., ending in an accepting location) of $\mathcal{A}$ on u'. We can also equip $\mathcal{A}$ with a Büchi acceptance condition; in this case, a run is *accepting* if it visits an accepting location infinitely often, and an infinite timed word ρ is *accepted* by $\mathcal{A}$ if there is such a run of $\mathcal{A}$ on ρ. The *timed (finite-word) language* defined by $\mathcal{A}$ is the set of (finite) timed words accepted by $\mathcal{A}$. Note that timed automata are not closed under complementation; for example, the complement of the timed language accepted by the timed automaton in the example below cannot be recognised by any timed automaton.

Example 2.3 [AM04]**.** Consider the timed automaton with $\Sigma = \{a, b\}$ in Figure 3. The automaton accepts timed words containing an a event at some time t such that no event occurs at time $t + 1$.

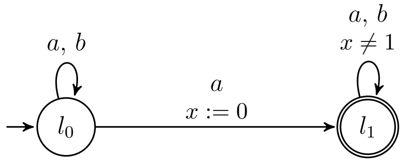


FIGURE 3. A timed automaton.

Monadic First-Order Logic of Order and Metric. We now define the *Monadic First-Order Logic of Order and Metric* ($\mathsf{FO}[<,+1]$) [Wil94] which subsumes all the other logics discussed in this article.

Definition 2.4. Given a set of monadic predicates $\mathbf{P}$, the set of $\mathsf{FO}[<,+1]$ formulas is generated by the grammar

$$\vartheta \;::=\; \mathbf{true} \mid P(x) \mid x < x' \mid d(x,x') \sim c \mid \vartheta_1 \wedge \vartheta_2 \mid \neg\vartheta \mid \exists x\, \vartheta\,,$$

where $P \in \mathbf{P}$, x, x' are variables, $\sim\, \in \{<, >\}$ and $c \in \mathbb{N}$.[6]

The fragment where $d(x,x') \sim c$ is absent is called the *Monadic First-Order Logic of Order* ($\mathsf{FO}[<]$).

Metric temporal logics. Formulas of metric temporal logics are $\mathsf{FO}[<,+1]$ formulas (with a single free variable) built from monadic predicates using Boolean connectives and *modalities* (or *operators*). A k-ary modality is defined by an $\mathsf{FO}[<,+1]$ formula $\varphi(x, X_1, \ldots, X_k)$ with a single free variable x and k free monadic predicates $X_1, \ldots, X_k$. For example, the MTL [Koy90] modality $\mathcal{U}_{(0,5)}$ is defined by the $\mathsf{FO}[<,+1]$ formula

$$\mathcal{U}_{(0,5)}(x, X_1, X_2) \;=\; \exists x' \Big(\; x < x' \wedge d(x,x') < 5 \wedge X_2(x') \\ \wedge \forall x'' \left(x < x'' \wedge x'' < x' \implies X_1(x'')\right)\Big)\,.$$

The MTL formula $\varphi_1 \,\mathcal{U}_{(0,5)}\, \varphi_2$ (usually written in infix notation) is obtained by substituting MTL formulas φ_1, φ_2 for X_1, X_2, respectively.

Definition 2.5. Given a set of monadic predicates $\mathbf{P}$, the set of MTL formulas is generated by the grammar

$$\varphi \;::=\; \mathbf{true} \mid P \mid \varphi_1 \wedge \varphi_2 \mid \neg\varphi \mid \varphi_1 \,\mathcal{U}_I\, \varphi_2 \mid \varphi_1 \,\mathcal{S}_I\, \varphi_2\,,$$

where $P \in \mathbf{P}$ and $I \subseteq (0, \infty)$ is an interval with endpoints in $\mathbb{N} \cup \{\infty\}$.

The (future-only) fragment $\mathsf{MTL}_{\mathsf{fut}}$ is obtained by disallowing subformulas of the form $\varphi_1 \,\mathcal{S}_I\, \varphi_2$. We write $|I|$ for $\sup(I) - \inf(I)$. If I is not present as a subscript then it is assumed to be $(0, \infty)$. We sometimes use pseudo-arithmetic expressions to denote intervals, e.g., '≥ 1' denotes $[1, \infty)$ and '$= 1$' denotes the singleton $\{1\}$. We also employ the usual syntactic sugar, e.g., $\mathbf{false} \equiv \neg\mathbf{true}$, $\Diamond_I\, \varphi \equiv \mathbf{true}\,\mathcal{U}_I\, \varphi$, $\overleftarrow{\Diamond}_I\, \varphi \equiv \mathbf{true}\,\mathcal{S}_I\, \varphi$, $\Box_I\, \varphi \equiv \neg \Diamond_I \neg\varphi$ and $\bigcirc_I\, \varphi \equiv \mathbf{false}\,\mathcal{U}_I\, \varphi$, etc. For convenience, we also use 'weak' temporal operators as syntactic sugar, e.g., $\varphi_1 \,\mathcal{U}^w_I\, \varphi_2 \equiv \varphi_1 \wedge (\varphi_1 \,\mathcal{U}_I\, \varphi_2)$ if $0 \notin I$ and $\varphi_1 \,\mathcal{U}^w_I\, \varphi_2 \equiv \varphi_2 \vee (\varphi_1 \wedge (\varphi_1 \,\mathcal{U}_I\, \varphi_2))$ if $0 \in I$ (we allow $0 \in I$ in the case of weak temporal operators). We denote by $|\varphi|$ the number of subformulas in φ.

[6] Note that whilst we refer to the logic as $\mathsf{FO}[<,+1]$, we adopt an equivalent definition where binary distance predicates $d(x,x') \sim c$ (as in [Wil94]) are used in place of the usual $+1$ function symbol.

The pointwise semantics. With each $\mathbb{T}$-timed word $\rho = (\sigma, \tau)$ over $\Sigma_{\mathbf{P}} = 2^{\mathbf{P}}$ we associate a structure M_ρ. Its universe U_ρ is the subset $\{\tau_i \mid 0 \leq i < |\rho|\}$ of $\mathbb{T}$. The order relation $<$ and monadic predicates in $\mathbf{P}$ are interpreted in the expected way, e.g., $P(\tau_i)$ holds in M_ρ iff $P \in \sigma_i$. The binary *distance predicate* $d(x, x') \sim c$ holds iff $|x - x'| \sim c$. The satisfaction relation is defined inductively as usual. We write $M_\rho, t_0, \ldots, t_{n-1} \models \vartheta(x_0, \ldots, x_{n-1})$ (or $\rho, t_0, \ldots, t_{n-1} \models \vartheta(x_0, \ldots, x_{n-1})$) if $t_0, \ldots, t_{n-1} \in U_\rho$ and $\vartheta(t_0, \ldots, t_{n-1})$ holds in M_ρ. We say that two $\mathsf{FO}[<, +1]$ formulas $\vartheta_1(x)$ and $\vartheta_2(x)$ are *equivalent* over $\mathbb{T}$-timed words if for all $\mathbb{T}$-timed words ρ and $t \in U_\rho$,

$$\rho, t \models \vartheta_1(x) \iff \rho, t \models \vartheta_2(x)\,.$$

We say that a metric logic L' is *expressively complete* for metric logic L over $\mathbb{T}$-timed words iff for any formula $\vartheta(x) \in L$, there is an equivalent formula $\varphi(x) \in L'$ over $\mathbb{T}$-timed words. We say that L' is *at least as expressive as* (or *more expressive than*) L over $\mathbb{T}$-timed words (written $L \subseteq L'$) iff for any formula $\vartheta \in L$, there is an *initially equivalent* formula $\varphi \in L'$ over $\mathbb{T}$-timed words (i.e., ϑ and φ evaluates to the same truth value at the beginning of any $\mathbb{T}$-timed word). If $L \subseteq L'$ but $L' \nsubseteq L$ then we say that L' is *strictly more expressive than* L (or L is *strictly less expressive than* L') over $\mathbb{T}$-timed words.

As we have seen earlier, each MTL formula can be defined as an $\mathsf{FO}[<, +1]$ formula with a single free variable. Here, for the sake of completeness we give an (equivalent) traditional inductive definition of the satisfaction relation for MTL over timed words. We write $\rho \models \varphi$ if $\rho, 0 \models \varphi$.

Definition 2.6. The satisfaction relation $\rho, i \models \varphi$ for an MTL formula φ, a timed word $\rho = (\sigma, \tau)$ and a position i in ρ is defined as follows:

- $\rho, i \models \mathbf{true}$
- $\rho, i \models P$ iff $P(\tau_i)$ holds in M_ρ
- $\rho, i \models \varphi_1 \wedge \varphi_2$ iff $\rho, i \models \varphi_1$ and $\rho, i \models \varphi_2$
- $\rho, i \models \neg\varphi$ iff $\rho, i \not\models \varphi$
- $\rho, i \models \varphi_1 \, \mathcal{U}_I \, \varphi_2$ iff there exists j, $i < j < |\rho|$ such that $\rho, j \models \varphi_2$, $\tau_j - \tau_i \in I$ and $\rho, k \models \varphi_1$ for all k with $i < k < j$
- $\rho, i \models \varphi_1 \, \mathcal{S}_I \, \varphi_2$ iff there exists j, $0 \leq j < i$ such that $\rho, j \models \varphi_2$, $\tau_i - \tau_j \in I$ and $\rho, k \models \varphi_1$ for all k with $j < k < i$.

Example 2.7. The $\mathsf{MTL}_{\mathsf{fut}}$ formula

$$\varphi = \Box(P \implies \Diamond_{<3} Q) \tag{2.1}$$

is satisfied by a timed word ρ if and only if there is a P-event in ρ (say at time t), and there is a Q-event in ρ with timestamp in $(t, t+3)$.

Safety relative to the divergence of time. Recall that we require the timestamps of any infinite timed word to be a strictly-increasing divergent sequence. Based upon this assumption, we define *safety properties* in exactly the same way as in the qualitative case [AS87]; for example, (2.1) is a safety property as any infinite timed word u' violating φ must have a prefix u such that there is a P-event in u with no Q-event in the following three time units. On the other hand, had we allowed Zeno timed words, φ would not be safety as

$$(\{P\}, 0)(\{P\}, 1)(\{P\}, 1 + \frac{1}{2})(\{P\}, 1 + \frac{1}{2} + \frac{1}{4}) \ldots$$

violates φ without having a prefix that cannot be extended into an infinite timed word satisfying φ. The notion we adopt here is called *safety relative to the divergence of time* in the literature [HMP92].

The continuous semantics. Another way to interpret metric logics is to regard time as a continuous entity; a behaviour of a system can thus be viewed as a continuous function. Formally, a $\mathbb{T}$-*signal* over finite alphabet Σ is a function $f : \mathbb{T} \mapsto \Sigma$ that is *finitely variable*, i.e., the restriction of f to a subinterval of $\mathbb{T}$ of finite length has only a finite number of discontinuities. We refer to a $\mathbb{T}$-signal simply as a *signal* when $\mathbb{T} = \mathbb{R}_{\geq 0}$. With each signal f over $\Sigma_{\mathbf{P}}$ we associate a structure M_f. Its universe U_f is $\mathbb{T}$. The order relation $<$ and monadic predicates in $\mathbf{P}$ are interpreted in the expected way, e.g., $P(x)$ holds in M_f iff $P \in f(x)$. The binary *distance predicate* $d(x, x') \sim c$ holds iff $|x - x'| \sim c$. We write $M_f, t_0, \ldots, t_{n-1} \models \vartheta(x_0, \ldots, x_{n-1})$ (or $f, t_0, \ldots, t_{n-1} \models \vartheta(x_0, \ldots, x_{n-1})$) if $t_0, \ldots, t_{n-1} \in U_f$ and $\vartheta(t_0, \ldots, t_{n-1})$ holds in M_f. The notions of equivalence of formulas, expressiveness of metric logics, etc. are defined as in the case of timed words.

The satisfaction relation for MTL over signals is defined as follows. We write $f \models \varphi$ if $f, 0 \models \varphi$.

Definition 2.8. The satisfaction relation $f, t \models \varphi$ for an MTL formula φ, a signal f and $t \in U_f$ is defined as follows:

- $f, t \models P$ iff $P(t)$ holds in M_f
- $f, t \models \mathbf{true}$
- $f, t \models \varphi_1 \wedge \varphi_2$ iff $f, t \models \varphi_1$ and $f, t \models \varphi_2$
- $f, t \models \neg\varphi$ iff $f, t \not\models \varphi$
- $f, t \models \varphi_1 \, \mathcal{U}_I \, \varphi_2$ iff there exists $t' > t$, $t' \in \mathbb{T}$ such that $f, t' \models \varphi_2$, $t' - t \in I$ and $f, t'' \models \varphi_1$ for all t'' with $t < t'' < t'$
- $f, t \models \varphi_1 \, \mathcal{S}_I \, \varphi_2$ iff there exists $t' < t$, $t' \in \mathbb{T}$ such that $f, t' \models \varphi_2$, $t - t' \in I$ and $f, t'' \models \varphi_1$ for all t'' with $t' < t'' < t$.

Relating the two semantics. Note that timed words can be regarded as a particular kind of signal: for a given $\mathbb{T}$-timed word ρ over Σ_P, we can introduce a 'silent' monadic predicate P_ϵ and construct the corresponding $\mathbb{T}$-signal f^ρ over $\Sigma_{\mathbf{P}'}$, where $\mathbf{P}' = \mathbf{P} \cup \{P_\epsilon\}$, as follows:

- $f^\rho(\tau_i) = \sigma_i$ for all i, $0 \leq i < |\rho|$
- $f^\rho(\tau_i) = \{P_\epsilon\}$.

This enables us to interpret metric logics over timed words 'continuously'. We can thus compare the expressiveness of metric logics in both semantics by restricting the models of the continuous interpretations of metric logics to signals of this form (i.e., f^ρ for some timed word ρ). For example, we say that continuous $\mathsf{FO}[<, +1]$ is at least as expressive as pointwise $\mathsf{FO}[<, +1]$ since for each $\mathsf{FO}[<, +1]$ formula $\vartheta_{pw}(x)$, there is an 'equivalent' $\mathsf{FO}[<, +1]$ formula $\vartheta_{cont}(x)$ such that $\rho, t \models \vartheta_{pw}(x)$ iff $f^\rho, t \models \vartheta_{cont}(x)$.

Example 2.9. Consider the timed word ρ illustrated in Figure 4 where the red boxes denote P-events. The $\mathsf{MTL}_{\mathsf{fut}}$ formula

$$\varphi = \Diamond(\Diamond_{=1} P)$$

does not hold at the beginning of ρ in the pointwise semantics (i.e., $\rho \not\models \varphi$) since there is no event at exactly one time unit before the second event in ρ. On the other hand, φ holds at the beginning of ρ in the continuous semantics (i.e., $f^\rho \models \varphi$) since there is a point (at which

P_ϵ holds) at exactly one time unit before the second event in f^ρ. We can, however, simulate the pointwise semantics with

$$\varphi' = \Diamond\Big(\neg P_\epsilon \wedge \big(\Diamond_{=1}(\neg P_\epsilon \wedge P)\big)\Big)\,,$$

for which we have $\pi \models \varphi$ iff $f^\pi \models \varphi'$ for all timed words π.

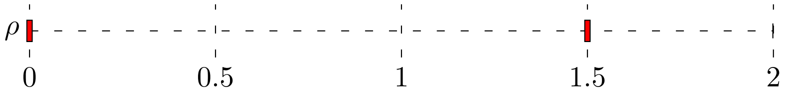


Figure 4. The timed word ρ.

As we see in the example above, the pointwise and continuous interpretations of metric logics differs in the range of first-order quantifiers. While the ability to quantify over time points *between* events appears to increase the expressiveness of metric logics, this is not the case for $\mathsf{FO}[<,+1]$ as both interpretations are indeed equally expressive (when one considers only signals of the form f^ρ) [DHV07].[7] By contrast, MTL is strictly more expressive in the continuous semantics than in the pointwise semantics [PD06].

2.2. **Model checking.** A key advantage in using $\mathsf{LTL}_{\mathsf{fut}}$ (or LTL) in verification is that its *model-checking* problem is PSPACE-complete [SC85], much better than the complexity of the same problem for $\mathsf{FO}[<]$ (non-elementary [Sto74]). Given a Büchi automaton $\mathcal{A}$ that models the system and a specification written as an $\mathsf{LTL}_{\mathsf{fut}}$ formula Φ, the corresponding model-checking problem asks whether the language defined by $\mathcal{A}$ is included in the language defined by Φ. By a fundamental result in verification—$\mathsf{LTL}_{\mathsf{fut}}$ formulas can be translated into Büchi automata [WVS83]—this reduces to the *emptiness* problem on the product Büchi automaton of $\mathcal{A}$ and the Büchi automaton $\mathcal{B}_{\neg\Phi}$ translated from $\neg\Phi$. The latter problem can be solved, e.g., by a standard fixed-point algorithm [EL86]. This is sometimes called the *automata-theoretic approach* to $\mathsf{LTL}_{\mathsf{fut}}$ model checking.

In the real-time setting, given a timed (Büchi) automaton $\mathcal{A}$ and a specification φ (e.g., a formula of some metric logic), the corresponding model-checking problem asks whether the timed (finite-word) language defined by $\mathcal{A}$ is included in the timed (finite-word) language defined by φ. By analogy with the untimed case, one may solve this problem by first translating $\neg\varphi$ into a timed automaton $\mathcal{A}_{\neg\varphi}$ and then checking the emptiness of the product of $\mathcal{A}$ and $\mathcal{A}_{\neg\varphi}$. This methodology works for certain metric logics; for example, each formula of $\mathsf{MITL}_{\mathsf{fut}}$ (the non-punctual fragment of $\mathsf{MTL}_{\mathsf{fut}}$) can be translated into a timed automaton, and the model-checking problem for timed automata against $\mathsf{MITL}_{\mathsf{fut}}$ is EXPSPACE-complete [AFH96]. However, this does not apply to $\mathsf{MTL}_{\mathsf{fut}}$ as $\mathsf{MTL}_{\mathsf{fut}}$ formulas, in general, cannot be translated into timed automata.

[7] The translation in [DHV07] also holds in a time-bounded setting with trivial modifications.

2.3. **Monitoring.** The *prefix* problem [BKV13] asks the following: given a specification Φ and a finite word u, do all infinite extensions of u satisfy Φ? If the answer is 'yes', then we say that u' is a *good prefix* for Φ. Similarly, u is a *bad prefix* for Φ if the answer to the dual problem is 'yes', i.e., none of its infinite extensions satisfies Φ. The *monitoring* problem takes instead a specification Φ and an infinite word u' as inputs. In contrast to standard decision problems, the latter input is given *incrementally*, i.e., one symbol at a time; a monitor (a procedure that 'solves' the monitoring problem) is required to continuously check whether the currently accumulated finite word u (a prefix of u') is a good/bad prefix for Φ and report as necessary.

3. Expressive completeness of $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ over bounded timed words

In this section, we study the expressiveness of MTL (and its various fragments and extensions) in a time-bounded pointwise setting, i.e., all timed words are assumed to have durations less than a positive integer N. We first recall MTL EF games [PS11], which serves as our main tool in proving expressiveness results. Then we demonstrate a strict hierarchy of metric temporal logics (based on their expressiveness over bounded timed words) as we extend $\mathsf{MTL}_{\mathsf{fut}}$ incrementally towards $\mathsf{FO}[<, +1]$. Finally, we show that MTL, equipped with both the forwards and backwards temporal modalities generalised 'Until' ($\mathfrak{U}^c_I$) and generalised 'Since' ($\mathfrak{S}^c_I$), has precisely the same expressive power as $\mathsf{FO}[<, +1]$ over bounded time domains in the pointwise semantics. For the time-bounded satisfiability and model-checking problems, we show that the relevant constructions (and hence the complexity bounds) for MTL in [ORW09] carry over to our new logic $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$.

3.1. **MTL EF games.** Ehrenfeucht-Fraïssé games are handy tools in proving the inexpressibility of certain properties in first-order logics. In many proofs in this section, we resort to (extended versions of) Pandya and Shah's MTL *EF games* on timed words [PS11], which itself is a timed generalisation of Etessami and Wilke's LTL EF games [EW96].

An m-round MTL EF game starts with round 0 and ends with round m. The game is played by two players (*Spoiler* and *Duplicator*) on a pair of timed words ρ and ρ'.[8] A *configuration* is a pair of positions (i, j), respectively in ρ and ρ'. In each round r $(0 \leq r \leq m)$, the game proceeds as follows. *Spoiler* first checks whether the two events that correspond to the current configuration (i_r, j_r) in ρ and ρ' satisfy the same set of monadic predicates. If this is not the case then he wins the game. Otherwise if $r < m$, *Spoiler* chooses an interval $I \subseteq (0, \infty)$ with endpoints in $\mathbb{N} \cup \{\infty\}$ and plays either of the following moves:

- $\mathcal{U}_I$*-move*: *Spoiler* chooses one of the two timed words (say ρ). He then picks i'_r such that $i_r < i'_r$ and $\tau_{i'_r} - \tau_{i_r} \in I$ where $\tau_{i'_r}$ and τ_{i_r} are the corresponding timestamps in ρ (if there is no such i'_r then *Duplicator* wins the game). *Duplicator* must choose a position j'_r in ρ' such that the difference of the corresponding timestamps in ρ' is in I. If she cannot find such a position then *Spoiler* wins the game. Otherwise, *Spoiler* plays either of the following 'parts':
  - $\Diamond$*-part*: The game proceeds to the next round with $(i_{r+1}, j_{r+1}) = (i'_r, j'_r)$.
  - $\mathcal{U}$*-part*: If $j'_r = j_r + 1$ the game proceeds to the next round with $(i_{r+1}, j_{r+1}) = (i'_r, j'_r)$. If $i'_r = i_r + 1$ but $j'_r \neq j_r + 1$ then *Spoiler* wins the game. Otherwise *Spoiler* picks another position j''_r in ρ' such that $j_r < j''_r < j'_r$. *Duplicator* have to choose a position i''_r in ρ

[8] We follow the convention that *Spoiler* is male and *Duplicator* is female.

such that $i_r < i''_r < i'_r$ in response. If she cannot find such a position then *Spoiler* wins the game; otherwise the game proceeds to the next round with $(i_{r+1}, j_{r+1}) = (i''_r, j''_r)$.

- $\mathcal{S}_I$*-move*: Defined symmetrically.

We say that *Duplicator* has a *winning strategy* for the m-round MTL EF game on ρ and ρ' that starts from configuration (i, j) if and only if, no matter how *Spoiler* plays, he cannot win the m-round MTL EF game on ρ and ρ' with $(i_0, i_0) = (i, j)$. If this is not the case then we say that *Spoiler* has a winning strategy.

It is obvious that the moves in MTL EF games are closely related to the semantics of modalities in MTL formulas. For example, the $\mathcal{U}_I$-move can be seen as *Spoiler*'s attempt to verify that a formula of the form $\varphi_1 \, \mathcal{U}_I \, \varphi_2$ holds at i_r in ρ if and only if it holds at j_r in ρ': the $\diamondsuit$-part and the remaining rounds verify that φ_2 holds at i'_r in ρ iff it holds at j'_r in ρ', whereas the $\mathcal{U}$-part and the remaining rounds verify that φ_1 holds at all i''_r, $i_r < i''_r < i'_r$ in ρ iff it holds at all j''_r, $j_r < j''_r < j'_r$ in ρ'. Formally, the following theorem relates the number of rounds of MTL EF games to the *modal depth* (i.e., the maximal depth of nesting of modalities) of MTL formulas.

Theorem 3.1 [PS11]**.** *For (finite) timed words ρ, ρ' and an MTL formula φ of modal depth $\leq m$, if* Duplicator *has a winning strategy for the m-round MTL EF game on ρ, ρ' with $(i_0, j_0) = (0, 0)$, then*

$$\rho \models \varphi \iff \rho' \models \varphi \, .$$

In other words, ρ, ρ' can be distinguished by an MTL formula of modal depth $\leq m$ if and only if *Spoiler* has a winning strategy for the m-round MTL EF game on ρ, ρ' with $(i_0, j_0) = (0, 0)$. Note that specialised versions of Theorem 3.1 also hold for sublogics of MTL; for example, the corresponding theorem for $\mathsf{MTL}_{\mathsf{fut}}$ is obtained by banning the $\mathcal{S}_I$-move.

Example 3.2. Consider the timed words ρ and ρ' illustrated in Figure 5 where the white, red and blue boxes represent events at which no monadic predicate holds, P-events, and Q-events, respectively. The positions are labelled above the events.

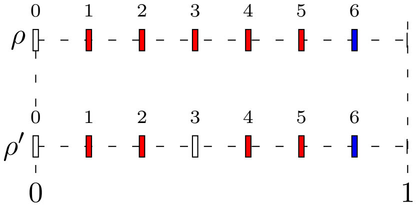


Figure 5. ρ and ρ' can be distinguished by $P \, \mathcal{U} \, Q$.

In the 1-round MTL EF game on ρ, ρ' with $(i_0, j_0) = (0, 0)$, a winning strategy for *Spoiler* can be described as follows:

1. The two events that correspond to $(i_0, j_0) = (0, 0)$ in ρ and ρ' satisfy the same set of monadic predicates, so *Spoiler* does not win here.
2. *Spoiler* chooses $I = (0, \infty)$ and $i'_0 = 6$ in ρ.
3. If *Duplicator* chooses $j'_0 \neq 6$ in ρ', she will lose at the beginning of round 1. So she chooses $j'_0 = 6$.
4. *Spoiler* plays the $\mathcal{U}$-part and chooses $j''_0 = 3$ in ρ'.
5. *Duplicator* can only choose i''_0 in ρ such that $1 \leq i''_0 \leq 5$. But she will then lose at the beginning of round 1.

It follows that there is an MTL formula of modal depth 1 that distinguishes ρ and ρ'. One such formula is $P \,\mathcal{U}\, Q$, which can be obtained from *Spoiler*'s winning strategy above.

3.2. A hierarchy of expressiveness.

We now present a sequence of successively more expressive extensions of $\mathsf{MTL}_{\mathsf{fut}}$ over bounded timed words. The technique we use here is to construct two *families* of models—parametrised by m—such that there is a certain formula of the more expressive logic telling them apart for all m, yet they cannot be distinguished by any formula of the less expressive logic with modal depth $\leq m$ (i.e., *Duplicator* has a winning strategy in the corresponding m-round MTL EF game). Along the way we highlight the key features that give rise to the differences in expressiveness. The necessity of new modalities is justified by the fact that no known extension can lead to expressive completeness.

Definability of the beginning of time. Recall that $\mathsf{MTL}_{\mathsf{fut}}$ and $\mathsf{FO}[<, +1]$ have the same expressiveness over $[0, N)$-signals [ORW09]. This result fails in the pointwise semantics.

Proposition 3.3 (Corollary of [PD06, Section 8])**.** MTL *is strictly more expressive than* $\mathsf{MTL}_{\mathsf{fut}}$ *over* $[0, N)$*-timed words.*[9]

To explain this discrepancy between the two semantics, observe that a distinctive feature of the continuous interpretation of $\mathsf{MTL}_{\mathsf{fut}}$ is exploited in [ORW09]: in any $[0, N)$-signal, the formula $\Diamond_{=(N-1)}\, \mathbf{true}$ holds in $[0, 1)$ and nowhere else. One can make use of conjunctions of similar formulas to determine the integer part of the current instant (where the relevant formula is being evaluated). Unfortunately, since the duration of a given bounded timed word is not known *a priori*, this trick does not work for $\mathsf{MTL}_{\mathsf{fut}}$ in the pointwise semantics. For example, the formula $\Diamond_{=1}\, \mathbf{true}$ does not hold at any position in the $[0, 2)$-timed word $\rho = (\sigma_0, 0)(\sigma_1, 0.5)$. However, the same effect can be achieved in MTL by using past modalities. Let

$$\varphi_{i,i+1} = \diamondminus_{[i,i+1)}(\neg \diamondminus\, \mathbf{true})$$

and $\Phi_{int} = \{\varphi_{i,i+1} \mid i \in \mathbb{N}\}$. Note that the subformula $\neg \diamondminus\, \mathbf{true}$ can only hold at the very first event (with timestamp 0), thus $\varphi_{i,i+1}$ holds only at events with timestamps in $[i, i + 1)$. Denote by $\mathsf{MTL}_{\mathsf{fut}}[\Phi_{int}]$ the extension of $\mathsf{MTL}_{\mathsf{fut}}$ obtained by allowing these formulas as atomic formulas. It turned out that this very restrictive use of past modalities strictly increases the expressiveness of $\mathsf{MTL}_{\mathsf{fut}}$ over bounded timed words. Indeed, the main result of this section (Theorem 3.15) crucially depends on the use of these formulas.

Proposition 3.4. $\mathsf{MTL}_{\mathsf{fut}}[\Phi_{int}]$ *is strictly more expressive than* $\mathsf{MTL}_{\mathsf{fut}}$ *over* $[0, N)$*-timed words.*

Proof. For a given $m \in \mathbb{N}$, we construct the following models:

$$\begin{aligned}
\mathcal{A}_m &= (\emptyset, 0)(\emptyset, 1 - \tfrac{1.5}{2m+5})(\emptyset, 1 - \tfrac{0.5}{2m+5}) \ldots (\emptyset, 1 + \tfrac{m+2.5}{2m+5})\,, \\
\mathcal{B}_m &= (\emptyset, 0)(\emptyset, 1 - \tfrac{0.5}{2m+5})(\emptyset, 1 + \tfrac{0.5}{2m+5}) \ldots (\emptyset, 1 + \tfrac{m+3.5}{2m+5})\,.
\end{aligned}$$

The models are illustrated in Figure 6, where each white box represents an event (at which no monadic predicate holds). We play an m-round MTL EF game on $\mathcal{A}_m$, $\mathcal{B}_m$ and allow only $\mathcal{U}_I$-move. After round 0, either (i) $i_1 = j_1 \geq 1$ (in which case *Duplicator* can, obviously, win the remaining rounds) or (ii) $(i_1, j_1) = (2, 1)$ (*Spoiler* chooses position 2 in $\mathcal{A}_m$) or $(i_1, j_1) = (3, 2)$ (*Spoiler* chooses position 2 in $\mathcal{B}_m$). In the latter case, it is easy to verify that in

[9] The models constructed in [PD06, Section 8] are bounded timed words.

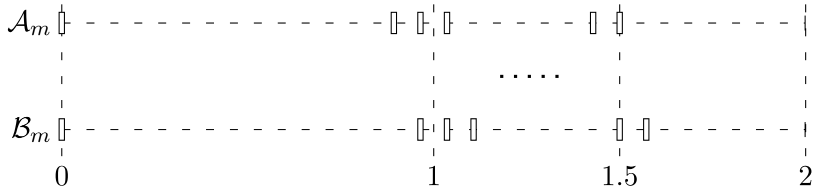


Figure 6. Models $\mathcal{A}_m$ and $\mathcal{B}_m$.

any remaining round r, *Duplicator* can make $i_{r+1} = j_{r+1} \geq 1$ or $(i_{r+1}, j_{r+1}) = (i_r + 1, j_r + 1)$. It follows from the MTL EF Theorem that no $\mathsf{MTL}_{\mathsf{fut}}$ formula of modal depth $\leq m$ can distinguish $\mathcal{A}_m$ and $\mathcal{B}_m$; however, the formula

$$\Diamond_{(0,1)}(\varphi_{0,1} \wedge \bigcirc \varphi_{0,1}),$$

which says "in the next time unit there are two events with timestamps in $[0, 1)$", distinguishes $\mathcal{A}_m$ and $\mathcal{B}_m$ for any $m \in \mathbb{N}$ (when evaluated at position 0). □

Past modalities. The conservative extension above uses past modalities in a very restricted way. This is not sufficient for obtaining the full expressiveness of MTL: the following proposition says that non-trivial nesting of future modalities and past modalities gives more expressiveness.

Proposition 3.5. MTL *is strictly more expressive than* $\mathsf{MTL}_{\mathsf{fut}}[\Phi_{int}]$ *over* $[0, N)$*-timed words.*

Proof. For a given $m \in \mathbb{N}$, we construct

$$\mathcal{C}_m \quad = \quad (\emptyset, 0)(\emptyset, \tfrac{0.5}{2m+3})(\emptyset, \tfrac{1.5}{2m+3}) \ldots (\emptyset, 2 - \tfrac{0.5}{2m+3}).$$

$\mathcal{D}_m$ is constructed as $\mathcal{C}_m$ except that the event at time $\frac{m+1.5}{2m+3} = 0.5$ is missing.

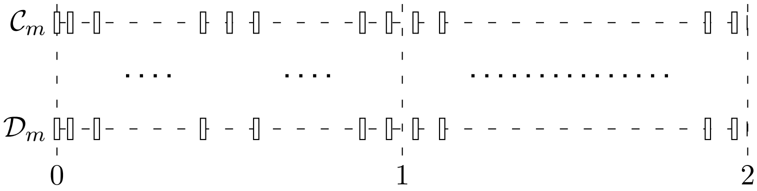


Figure 7. Models $\mathcal{C}_m$ and $\mathcal{D}_m$.

The models are illustrated in Figure 7, where each white box represents an event (at which no monadic predicate holds). We play an m-round MTL EF game on $\mathcal{C}_m$ and $\mathcal{D}_m$, allowing only $\mathcal{U}_I$-move. For simplicity, assume that we can use special monadic predicates to refer to formulas in Φ_{int}. In each round r, *Duplicator* can either make (i) $i_{r+1} = j_{r+1} + 1$ and $i_{r+1} \geq m + 3$ (in which case she can win the remaining rounds) or (ii) $i_{r+1} = j_{r+1}$ and i_{r+1} is not equal to $2m + 2$, $2m + 3$ or $4m + 5$. It follows from the MTL EF Theorem that no $\mathsf{MTL}_{\mathsf{fut}}[\Phi_{int}]$ formula of modal depth $\leq m$ can distinguish $\mathcal{C}_m$ and $\mathcal{D}_m$; but the formula

$$\Box_{(1,2)}(\diamondminus_{=1} \mathbf{true}),$$

which says "for each event in $(1, 2)$ from now, there is a corresponding event exactly 1 time unit earlier", distinguishes $\mathcal{C}_m$ and $\mathcal{D}_m$ for any $m \in \mathbb{N}$ (when evaluated at position 0). □

Counting modalities. The modality $C_n(x, X)$ asserts that X holds at least at n points in the open interval $(x, x+1)$. The modalities C_n for $n \geq 2$ are called *counting modalities*. It is well-known that these modalities are not expressible in MTL over signals [HR07]. For this reason, they (and variants thereof) are often used to prove inexpressiveness results for various metric logics. For example, the following property

- P holds at an event at time y in the future
- Q holds at an event at time $y' > y$
- R holds at an event at time $y'' > y' > y$
- Both the Q-event and the R-event are within $(1, 2)$ from the P-event

can be expressed as the $\mathsf{FO}[<, +1]$ formula

$$\begin{aligned}\vartheta_{pqr}(x) = \exists y \Big(x < y \wedge P(y) \wedge \exists y' \big(& y < y' \wedge d(y, y') > 1 \wedge d(y, y') < 2 \wedge Q(y') \\ & \wedge \exists y'' \left(y' < y'' \wedge d(y, y'') > 1 \wedge d(y, y'') < 2 \wedge R(y'') \right) \big) \Big) \,,\end{aligned}$$

yet it has no equivalent in MTL over timed words [PS11]. The difficulty here is that while we can easily write 'there is a Q-event within $(1, 2)$ from a P-event in the future' as $\Diamond(P \wedge \Diamond_{(1,2)} Q)$, it is not possible to express 'there is a R-event after the Q-event' and 'that R-event is within $(1, 2)$ from the P-event' simultaneously in MTL. Indeed, it was shown recently that in the continuous semantics, MTL extended with counting modalities and their past counterparts (which we denote by $\mathsf{MTL}[\{C_n, \overleftarrow{C}_n\}_{n=2}^{\infty}]$) is expressively complete for $\mathsf{FO}[<, +1]$ [Hun13]. In other words, counting modalities are exactly what separates the expressiveness of MTL and $\mathsf{FO}[<, +1]$ in the continuous semantics. In the time-bounded pointwise case, however, they add no expressiveness to MTL. To see this, observe that the following formula is equivalent to $\vartheta_{pqr}(x)$ over $[0, N)$-timed words (we make use of the formulas in Φ_{int} defined earlier):

$$\begin{aligned}\Diamond\Bigg(\bigvee_{0 \leq i \leq N-1} \Bigg(P \wedge \varphi_{i,i+1} \wedge \Big(& \underbrace{\Diamond_{>1}\big(Q \wedge \Diamond(R \wedge \varphi_{i+1,i+2})\big)}_{\text{Case (i)}} \\ & \vee \underbrace{\Diamond_{<2}\big(R \wedge \varphi_{i+2,i+3} \wedge \diamondminus(Q \wedge \varphi_{i+2,i+3})\big)}_{\text{Case (ii)}} \\ & \vee \underbrace{\big(\Diamond_{>1}(Q \wedge \varphi_{i+1,i+2}) \wedge \Diamond_{<2}(R \wedge \varphi_{i+2,i+3})\big)}_{\text{Case (iii)}} \Big) \Bigg) \Bigg) \,.\end{aligned}$$

The three cases that correspond to the subformulas are illustrated in Figure 8 where time is measured relative to the very first event (with timestamp 0). Note how we use the 'integer boundaries' as an alternative distance measure and thus ensure that both the Q-event and the R-event are within $(1, 2)$ from the P-event.

The same idea can readily be generalised to handle counting modalities and their past counterparts. We therefore have the following proposition.

Proposition 3.6. MTL *is expressively complete for* $\mathsf{MTL}[\{C_n, \overleftarrow{C}_n\}_{n=2}^{\infty}]$ *over* $[0, N)$*-timed words.*

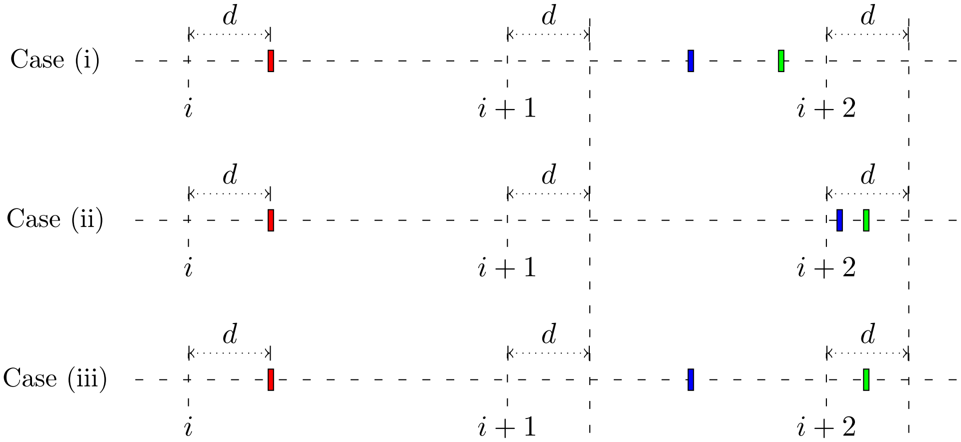


Figure 8. Counting modalities is expressible in MTL over $[0, N)$-timed words. The red, blue, and green boxes represent P-events, Q-events, and R-events respectively.

Non-local properties (one reference point). Proposition 3.6 shows that a part of the expressiveness hierarchy of metric logics over ($\mathbb{R}_{\geq 0}$-)timed words collapses in a time-bounded pointwise setting. Nonetheless, MTL is still not expressive enough to capture the whole of $\mathsf{FO}[<, +1]$ in such a setting. Recall that another feature of the continuous interpretation of $\mathsf{MTL}_{\mathsf{fut}}$ used in the proof in [ORW09] is that $\Diamond_{=k}\,\varphi$ holds at t *iff* φ holds at $t + k$. Suppose that we want to specify the following property over $\mathbf{P} = \{P, Q\}$ for some positive integer a (let the current instant be t_1):

- There is an event at time $t_2 > t_1 + a$ where Q holds
- P holds at all events in $(t_1 + a, t_2)$.

In the continuous semantics, the property can easily be expressed as the following $\mathsf{MTL}_{\mathsf{fut}}$ formula

$$\varphi_{\mathit{cont1}} = \Diamond_{=a}\big((P \vee P_\epsilon)\,\mathcal{U}\,Q\big)$$

over signals of the form f^ρ (over $\Sigma_{\mathbf{P}'}$ where $\mathbf{P}' = \mathbf{P} \cup \{P_\epsilon\}$); see Figure 9 for an example where the formula φ_{cont1} holds at t_1 in the continuous semantics.

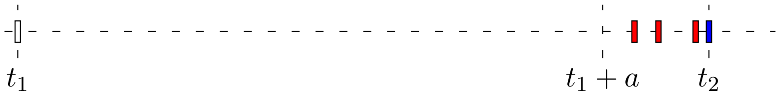


Figure 9. φ_{cont1} holds at t_1 in the continuous semantics. The red boxes denote P-events and the blue boxes denote Q-events.

Essentially, when the current instant is t_1, the continuous interpretation of MTL allows one to speak of events 'from' $t_1 + a$, regardless of whether there is an actual event at $t_1 + a$. As we will show, it is not possible to do the same with the pointwise interpretation of MTL when there is no event at $t_1 + a$. To remedy this issue within the pointwise semantic framework, we introduce a simple family of modalities $\mathcal{B}^{\rightarrow}_I$ ('Beginning') and their past versions $\mathcal{B}^{\leftarrow}_I$. They can be used to refer to the *first* (earliest or latest, respectively) event in a given interval. For example, we define the modality that asserts "X holds at the first event in (a, b) relative to the current instant" as the following $\mathsf{FO}[<, +1]$ formula:

$$\mathcal{B}^{\rightarrow}_{(a,b)}(x, X) = \exists x' \Big(x < x' \land d(x, x') > a \land d(x, x') < b \land X(x') \\ \land \nexists x'' \left(x < x'' \land x'' < x' \land d(x, x'') > a\right)\Big) .$$

The property above can now be written as $\mathcal{B}^{\rightarrow}_{(a,\infty)}\left(Q \lor (P \,\mathcal{U}\, Q)\right)$ in the pointwise semantics. We refer to the extension of MTL with $\mathcal{B}^{\rightarrow}_I, \mathcal{B}^{\leftarrow}_I$ as $\mathsf{MTL}[\mathcal{B}^{\leftrightarrows}]$.[10] The following proposition states that this extension is indeed non-trivial.

Proposition 3.7. $\mathsf{MTL}[\mathcal{B}^{\leftrightarrows}]$ *is strictly more expressive than* MTL *over* $[0, N)$*-timed words.*

Proof. The proof we give here is inspired by a proof in [PS11, Section 5]. Given $m \in \mathbb{N}$, we describe models $\mathcal{E}_m$ and $\mathcal{F}_m$ that are indistinguishable by MTL formulas of modal depth $\leq m$ but distinguishing in $\mathsf{MTL}[\mathcal{B}^{\leftrightarrows}]$.

We first describe $\mathcal{F}_m$. Let $g = \frac{1}{2m+6}$ and pick positive $\varepsilon < \frac{g}{\frac{1}{g}-1}$. The first event (at time 0) satisfies $\neg P \land \neg Q$. Then, a sequence of overlapping segments (arranged as described below) starts at time $\frac{0.5}{2m+5}$; see Figure 10 for an illustration of a segment. Each segment consists of an event satisfying $P \land \neg Q$ and an event satisfying $\neg P \land Q$ (we refer to them as P-events and Q-events, respectively). If the P-event in the i^{th} segment is at time t, then its Q-event is at time $t + 2g + \frac{1}{2}\varepsilon$. All P-events in neighbouring segments are separated by $g - \frac{g}{\frac{1}{g}-1}$. We put a total of $4m + 12$ segments.

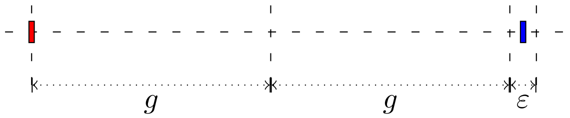


FIGURE 10. A single segment in $\mathcal{F}_m$. The red box denotes a P-event and the blue box denotes a Q-event.

$\mathcal{E}_m$ is almost identical to $\mathcal{F}_m$ except the $(3m + 9)^{th}$ segment. Let this segment start at t_{3m+9}. In $\mathcal{E}_m$, we move the corresponding Q-event to $t + 2g - \frac{1}{2}\varepsilon$ (see Figure 11). Note in particular that there are P-events at time 0.5 in both models (in their $(m + 4)^{th}$ segment).

The only difference in two models is a pair of Q-events, which we denote by x and y respectively and write their corresponding timestamps as t_x and t_y (see Figure 11). It is easy to verify that no two events are separated by an integer distance. We say a configuration (i, j) is *identical* if $i = j$. For $i \geq 1$, we denote by $seg(i)$ the segment that the i^{th} event belongs to, and we write $P(i)$ if the i^{th} event is a P-event and $Q(i)$ if its a Q-event.

Proposition 3.8. Duplicator *has a winning strategy for m-round* MTL *EF game on $\mathcal{E}_m$ and $\mathcal{F}_m$ with $(i_0, j_0) = (0, 0)$. In particular, she has a winning strategy such that for each round $0 \leq r \leq m$, the i_r^{th} event in $\mathcal{E}_m$ and the j_r^{th} event in $\mathcal{F}_m$ satisfy the same set of monadic predicates and*

[10] Readers may find the modalities $\mathcal{B}^{\rightarrow}_I$ similar to the modalities $\rhd_I$ in *Event-Clock Logic* [HRS98]. The difference is that the formula $\mathcal{B}^{\rightarrow}_I \varphi$ requires φ to hold at the *first* event in I, whereas the formula $\rhd_I \varphi$ requires (i) φ to hold at *some* event in I and that (ii) φ does not hold at any other event between the current instant and the time of that event.

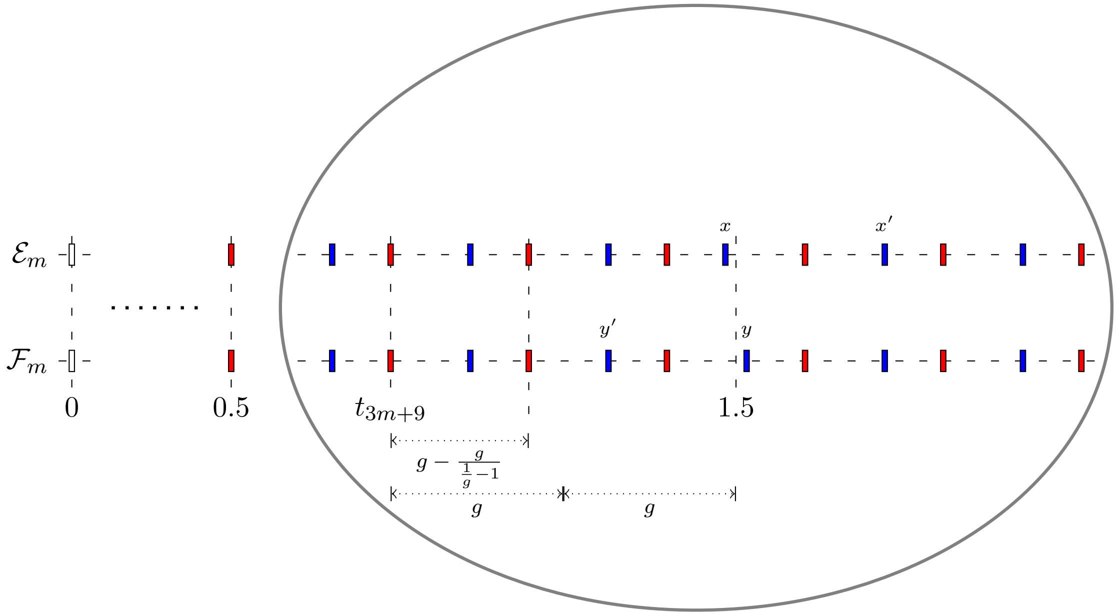


FIGURE 11. A close-up near the $(3m+9)^{th}$-segments in $\mathcal{E}_m$ and $\mathcal{F}_m$.

- *if* $i_r \neq j_r$, *then*
  - $seg(i_r) - seg(j_r) < r$
  - $(m+1-r) < seg(i_r), seg(j_r) < (m+5+r)$ *or* $(3m+8-r) < seg(i_r), seg(j_r) < (3m+12+r)$.

We prove the proposition by induction on r. The idea is to try to make the resulting configurations identical. When this is not possible *Duplicator* simply imitates what *Spoiler* does.

- *Base step.* The proposition holds trivially for $(i_0, j_0) = (0, 0)$.
- *Induction step.* Suppose that the claim holds for $r < m$. We prove it also holds for $r+1$.
  - $(i_r, j_r) = (0, 0)$:
    Duplicator can always make (i_{r+1}, j_{r+1}) identical.
  - $(i_r, j_r) \neq (0, 0)$ is identical:
    Duplicator tries to make (i'_r, j'_r) identical. This may only fail when
    - $P(i_r)$, $P(j_r)$ and $seg(i_r) = seg(j_r) = m+4$.
    - $Q(i_r)$, $Q(j_r)$ and $seg(i_r) = seg(j_r) = 3m+9$, i.e., x and y.

    In these cases, *Duplicator* chooses another event in a neighbouring segment that minimises $|seg(i'_r) - seg(j'_r)|$. For example, if (i_r, j_r) corresponds to x and y and *Spoiler* chooses j'_r such that $P(j'_r)$ and $seg(j'_r) = m+4$ in a $\mathcal{S}_{(1,\infty)}$-move, *Duplicator* chooses i'_r with $seg(i'_r) = m+3$. If *Spoiler* then plays $\diamondminus$-part, the resulting configuration $(i_{r+1}, j_{r+1}) = (i'_r, j'_r)$ clearly satisfy the claim. If she plays $\mathcal{S}$-part, *Duplicator* makes (i''_r, j''_r) identical whenever possible. Otherwise she chooses a suitable event that minimises $|seg(i''_r) - seg(j''_r)|$. For instance, if $Q(i''_r)$ and $seg(i''_r) = m+1$, *Duplicator* chooses j''_r such that $Q(j''_r)$ and $seg(j''_r) = m+2$. The resulting configuration $(i_{r+1}, j_{r+1}) = (i''_r, j''_r)$ clearly satisfies the claim.
  - (i_r, j_r) is not identical:
    Duplicator tries to make (i'_r, j'_r) identical. If this is not possible, then *Duplicator* chooses

an event that minimises $|seg(i'_r) - seg(j'_r)|$. For example, consider $seg(i_r) = m+4$, $seg(j_r) = m+3$ such that $P(i_r)$ and $P(j_r)$, and *Spoiler* chooses x in an $\mathcal{U}_{(0,1)}$-move. In this case, *Duplicator* cannot choose y', but she may choose the first Q-event that happens before y'. *Duplicator* responds to $\mathcal{U}$-parts and $\mathcal{S}$-parts in similar ways as before. It is easy to see that the claim holds.

Proposition 3.7 now follows from Proposition 3.8, the MTL EF Theorem, and the fact that $\mathcal{E}_m \models \Diamond(P \wedge \mathcal{B}^{\rightarrow}_{(1,2)}P)$ but $\mathcal{F}_m \not\models \Diamond(P \wedge \mathcal{B}^{\rightarrow}_{(1,2)}P)$. □

Non-local properties (two reference points). Adding modalities $\mathcal{B}^{\rightarrow}_I, \mathcal{B}^{\leftarrow}_I$ to MTL allows one to specify properties with respect to a distant time point even when there is no event at that point. However, the following proposition shows that this is still not enough for expressive completeness.

Proposition 3.9. FO[$<, +1$] *is strictly more expressive than* MTL[$\mathcal{B}^{\leftrightarrows}$] *over* $[0, N)$*-timed words.*

Proof. This is similar to a proof in [PD06, Section 7]. Given $m \in \mathbb{N}$, we construct two models as follows. Let

$$\begin{aligned}\mathcal{G}_m = (\emptyset, 0)(\emptyset, \tfrac{0.5}{2m+3})(\emptyset, \tfrac{1.5}{2m+3}) \dots (\emptyset, 1 - \tfrac{0.5}{2m+3}) \\ (\emptyset, 1 + \tfrac{0.5}{2m+2})(\emptyset, 1 + \tfrac{1.5}{2m+2}) \dots\dots (\emptyset, 2 - \tfrac{0.5}{2m+2})\,.\end{aligned}$$

$\mathcal{H}_m$ is constructed as $\mathcal{G}_m$ except that the event at time $\frac{m+1.5}{2m+3} = 0.5$ is missing.

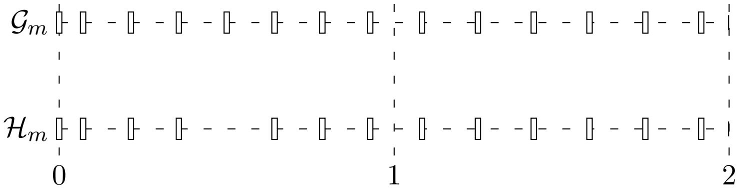


FIGURE 12. Models $\mathcal{G}_m$ and $\mathcal{H}_m$ for $m = 2$.

Figure 12 illustrates the models for the case $m = 2$ where white boxes represent events at which no monadic predicate holds. Observe that no two events are separated by an integer distance. We say that a configuration (i, j) is *synchronised* if they correspond to events with the same timestamp. Here we extend MTL EF games with the following moves to obtain MTL[$\mathcal{B}^{\leftrightarrows}$] EF games:

- $\mathcal{B}^{\rightarrow}_I$*-move*: *Spoiler* chooses one of the two timed words (say ρ) and picks i'_r such that (i) $\tau_{i'_r} - \tau_{i_r} \in I$ in ρ and (ii) there is no position $i' < i'_r$ in ρ such that $\tau_{i'} - \tau_{i_r} \in I$. *Duplicator* must choose a position j'_r in ρ' such that j'_r is the first position in I relative to j_r in ρ'. If she cannot find such a position then *Spoiler* wins the game.
- $\mathcal{B}^{\leftarrow}_I$*-move*: Defined symmetrically.

Theorem 3.10 (MTL[$\mathcal{B}^{\leftrightarrows}$] EF Theorem)**.** *For (finite) timed words* ρ, ρ' *and an* MTL[$\mathcal{B}^{\leftrightarrows}$] *formula* φ *of modal depth* $\leq m$, *if* Duplicator *has a winning strategy for the* m*-round* MTL[$\mathcal{B}^{\leftrightarrows}$] *EF game on* ρ, ρ' *with* $(i_0, j_0) = (0, 0)$, *then*

$$\rho \models \varphi \iff \rho' \models \varphi\,.$$

Proposition 3.11. Duplicator *has a winning strategy for m-round* $\mathsf{MTL}[\mathcal{B}^{\leftrightarrows}]$ *EF game on* $\mathcal{G}_m$ *and* $\mathcal{H}_m$ *with* $(i_0, j_0) = (0, 0)$*. In particular, she has a winning strategy such that for each round* $0 \leq r \leq m$*, the* i_r^{th} *event in* $\mathcal{G}_m$ *and the* j_r^{th} *event in* $\mathcal{H}_m$ *satisfy the same set of monadic predicates and*

- *if* (i_r, j_r) *is not synchronised, then*
  - $|i_r - j_r| = 1$
  - $(m + 1 - r) < i_r, j_r < (m + 3 + r)$ *or* $(3m + 4 - r) < i_r, j_r < (3m + 5 + r)$*.*

We prove the proposition by induction on r. The idea, again, is to try to make the resulting configurations identical.

- *Base step.* The proposition holds trivially for $(i_0, j_0) = (0, 0)$.
- *Induction step.* Suppose that the claim holds for $r < m$. We prove it also holds for $r + 1$.
  - $(i_r, j_r) = (0, 0)$:
    Duplicator tries to make (i'_r, j'_r) synchronised. If *Spoiler* chooses $i'_r = m + 2$, *Duplicator* chooses either $j'_r = m + 1$ or $j'_r = m + 2$.
  - $(i_r, j_r) \neq (0, 0)$ is synchronised:
    Duplicator tries to make (i'_r, j'_r) synchronised. If this is not possible then *Duplicator* chooses a suitable event that minimises $|i'_r - j'_r|$. It is easy to see that the resulting configuration (i_{r+1}, j_{r+1}) satisfies the claim regardless of how *Spoiler* plays.
  - (i_r, j_r) is not synchronised:
    The strategy of *Duplicator* is same as the case above.

Proposition 3.9 now follows from Proposition 3.11, Theorem 3.10, and the fact that the $\mathsf{FO}[<, +1]$ formula

$$\begin{aligned}\exists x' \Big(d(x, x') > 1 \wedge d(x, x') < 2 \wedge \exists x'' \Big(& x' < x'' \wedge \nexists y' \, (x' < y' \wedge y' < x'') \\ & \wedge d(x, x'') > 1 \wedge d(x, x'') < 2 \\ & \wedge \nexists y'' \, \big(d(x', y'') < 1 \wedge d(x'', y'') > 1 \big) \Big) \Big)\end{aligned}$$

distinguishes $\mathcal{G}_m$ and $\mathcal{H}_m$ for any $m \in \mathbb{N}$ (when evaluated at position 0). This formula asserts that there is a pair of neighbouring events in $(1, 2)$ such that there is no event between them if they are both mapped to exactly one time unit earlier. □

One way to understand why $\mathsf{MTL}[\mathcal{B}^{\leftrightarrows}]$ is still less expressive than $\mathsf{FO}[<, +1]$ is to consider the arity of modalities. Let the current instant be t_1, and suppose that we want to specify the following property for some positive integers a and c ($a > c$):[11]

- There is an event at $t_2 > t_1 + a$ where Q holds
- P holds at all events in $\big(t_1 + c, t_1 + c + (t_2 - t_1 - a)\big)$.

See Figure 13 for an example. In the continuous semantics, this property can be expressed as the following simple formula over signals of the form f^ρ:

$$\varphi_{\mathit{cont2}} = \big(\Diamond_{=c}(P \vee P_\epsilon)\big) \, \mathcal{U} \, (\Diamond_{=a} Q) \, .$$

Observe how this formula talks about events from two (instead of just one) time points: $t_1 + c$ and $t_1 + a$. In the same vein, the following formula can be used to distinguish $\mathcal{G}_m$ and

[11] We remark that a closely related yet different property is used in [LW08] to show that one-clock alternating timed automata and timed automata are expressively incomparable.

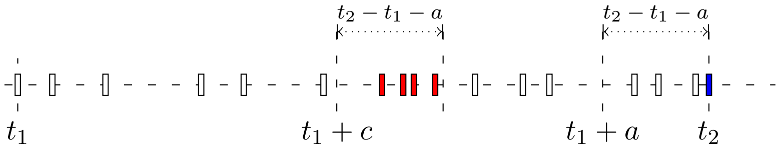


FIGURE 13. φ_{cont2} holds at t_1 in the continuous semantics. The red boxes denote P-events and the blue boxes denote Q-events.

$\mathcal{H}_m$ (defined in the proof of Proposition 3.9) in the continuous semantics:

$$\varphi_{cont3} = \Diamond_{(1,2)}\big(\neg P_\epsilon \wedge (\diamondminus_{=1} P_\epsilon)\, \mathcal{U}\, (\neg P_\epsilon)\big)\,.$$

Indeed, to express such properties in the pointwise semantics, we need *binary* variants of $\mathcal{B}^{\rightarrow}_I, \mathcal{B}^{\leftarrow}_I$, which are exactly what we propose in the next section.

3.3. New modalities.

We define a family of modalities which can be understood as generalisations of the usual 'Until' and 'Since' modalities. Intuitively, these new modalities closely mimic the meanings of formulas of the form $(\Diamond_{=k_1} \varphi_1)\, \mathcal{U}_{<k_3}\, (\Diamond_{=k_2} \varphi_2)$ or $(\Diamond_{=k_1} \varphi_1)\, \mathcal{S}_{<k_3}\, (\Diamond_{=k_2} \varphi_2)$ in the continuous semantics.

Generalised 'Until' and 'Since'. Let $I \subseteq (0, \infty)$ be an interval with endpoints in $\mathbb{N} \cup \{\infty\}$ and $c \in \mathbb{N}$, $c \leq \inf(I)$. The formula $\varphi_1 \, \mathfrak{U}^c_I \, \varphi_2$, when imposed at t_1, asserts that

- There is an event at t_2 where φ_2 holds and $t_2 - t_1 \in I$
- φ_1 holds at all events in the open interval $\Big(t_1 + c, t_1 + c + \Big(t_2 - \big(t_1 + \inf(I)\big)\Big)\Big)$.

For example, the formula $P\, \mathfrak{U}^c_{(a,\infty)}\, Q$ (which is 'equivalent' to φ_{cont2} when the latter is interpreted over signals of the form f^ρ) holds at time t_1 in Figure 13. Formally, for $I = (a, b) \subseteq (0, \infty)$, $a \in \mathbb{N}$, $b \in \mathbb{N} \cup \{\infty\}$ and $c \in \mathbb{N}$ with $c \leq a$, we define the *generalised 'Until'* modality $\mathfrak{U}^c_{(a,b)}$ by the following $\mathsf{FO}[<, +1]$ formula:

$$\begin{aligned}\mathfrak{U}^c_{(a,b)}(x, X_1, X_2) = \exists x' \Big(\; & x < x' \;\wedge d(x,x') > a \wedge d(x,x') < b \wedge X_2(x') \\ & \wedge \forall x'' \;\big(x < x'' \wedge d(x,x'') > c \wedge x'' < x' \\ & \qquad \wedge d(x',x'') > (a - c) \implies X_1(x'')\big)\Big)\,.\end{aligned}$$

Symmetrically, we define the *generalised 'Since'* modality $\mathfrak{S}^c_{(a,b)}$ as

$$\begin{aligned}\mathfrak{S}^c_{(a,b)}(x, X_1, X_2) = \exists x' \Big(\; & x' < x \;\wedge d(x,x') > a \wedge d(x,x') < b \wedge X_2(x') \\ & \wedge \forall x'' \;\big(x'' < x \wedge d(x,x'') > c \wedge x' < x'' \\ & \qquad \wedge d(x',x'') > (a - c) \implies X_1(x'')\big)\Big)\,.\end{aligned}$$

We also define the modalities for $I \subseteq (0, \infty)$ being a half-open interval or a closed interval in the expected way and refer to the logic obtained by adding these modalities to MTL as $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$. Note that the usual 'Until' and 'Since' modalities can be written in terms of the generalised modalities. For instance,

$$\varphi_1\, \mathcal{U}_{(a,b)}\, \varphi_2 = \varphi_1\, \mathfrak{U}^a_{(a,b)}\, \varphi_2 \wedge \neg \Big(\mathbf{true}\, \mathfrak{U}^0_{(0,a]}\, (\neg\varphi_1)\Big)\;.$$

More liberal bounds. In defining modalities $\mathfrak{U}^{c}_{(a,b)}$ and $\mathfrak{S}^{c}_{(a,b)}$ we required that $0 \leq c \leq a$. We now show that more liberal uses of bounds (constraining intervals and superscript 'c') are indeed syntactic sugars, and we therefore allow them in the rest of this section. For instance, suppose that we want to to assert the following property (which translates to $\big(\Diamond_{=10}(\varphi_1 \vee P_\epsilon)\big)\,\mathcal{U}_{<3}\,(\Diamond_{=2}\,\varphi_2)$ in the continuous semantics) at t_1:

- There is an event at t_2 where φ_2 holds and $t_2 - t_1 \in (2, 5)$
- φ_1 holds at all events in $\big(t_1 + 10, t_1 + 10 + (t_2 - t_1 - 2)\big)$.

This can be expressed in $\mathsf{FO}[<, +1]$ as

$$\begin{aligned}\exists x' \Big(\, & x < x' \;\wedge d(x, x') > 2 \wedge d(x, x') < 5 \wedge X_2(x') \\ & \wedge \forall x'' \;\big(x < x'' \wedge d(x, x'') > 10 \wedge d(x', x'') < 8 \implies X_1(x'')\big)\Big)\end{aligned}$$

where X_1, X_2 are to be substituted with φ_1, φ_2. While we could define a modality

$$\mathfrak{U}^{10}_{(2,5)}(x, X_1, X_2)$$

by this formula, this is not necessary as the formula is indeed equivalent to

$$\Diamond_{(2,5)}\,\varphi_2 \wedge \neg\Big((\neg\varphi_2)\,\mathfrak{U}^{2}_{(10,13)}\,\big(\neg\varphi_1 \wedge \neg(\diamondminus_{=8}\,\varphi_2)\big)\Big)\,.$$

In the continuous semantics we can, of course, also refer points in the past in such formulas, e.g., $(\diamondminus_{=k_1}\,\varphi_1)\,\mathcal{U}_{<k_3}\,(\Diamond_{=k_2}\,\varphi_2)$. We now generalise the idea above to handle these cases.

Proposition 3.12. *Let the current instant be t_1. The property (and its past counterpart):*

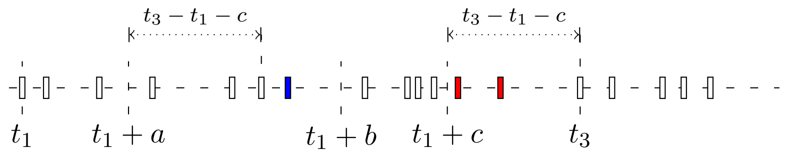
- *There is an event at t_2 where φ_2 holds and $t_2 - t_1 \in I$*
- *φ_1 holds at all events in $\Big(t_1 + c, t_1 + c + \Big(t_2 - \big(t_1 + \inf(I)\big)\Big)\Big)$*

where $I \subseteq (-\infty, \infty)$, $\inf(I) \in \mathbb{Z}$, $\sup(I) \in \mathbb{Z} \cup \{\infty\}$ and $c \in \mathbb{Z}$ can be expressed with the modalities defined earlier (i.e., $\mathfrak{U}^{c}_{I}, \mathfrak{S}^{c}_{I}$ with $I \subseteq (0, \infty)$ and $c \leq \inf(I)$).

Proof. Without loss of generality, we shall only focus on expressing the future version of the property for the case of I being an open interval. To ease the presentation, we use the following convention in all the illustrations in this proof: the red boxes denote φ_1-events, blue boxes denote φ_2-events, and white boxes denote events where neither φ_1 nor φ_2 hold. We prove the claim in each of the following cases:

- *$a \geq 0$ and $0 \leq c \leq a$*: This corresponds to the standard version of $\mathfrak{U}^{c}_{I}$ that we have already defined.
- *$a \geq 0$ and $c > a$*: $\varphi_1\,\mathfrak{U}^{c}_{(a,b)}\,\varphi_2$ does not hold at t_1 if and only if one of the following holds at t_1:
  - *There is no φ_2-event in $(t_1 + a, t_1 + b)$*: This can be enforced by
  $$\neg(\Diamond_{(a,b)}\,\varphi_2)\,.$$
  - *$\neg\varphi_1$ holds at an event at $t_3 \in \big(t_1 + c, t_1 + c + (b - a)\big)$ and there is no φ_2-event in $(t_1 + a, t_1 + a + (t_3 - t_1 - c)]$*: This can be enforced by
  $$(\neg\varphi_2)\,\mathfrak{U}^{a}_{\left(c,c+(b-a)\right)}\,\big(\neg\varphi_1 \wedge \underbrace{\neg(\diamondminus_{=(c-a)}\,\varphi_2)}_{\psi}\big)\,.$$

  We need the subformula ψ to ensure that there is no φ_2-event at $t_1 + a + (t_3 - t_1 - c)$. The desired formula is the conjunction of the negations of these two formulas.



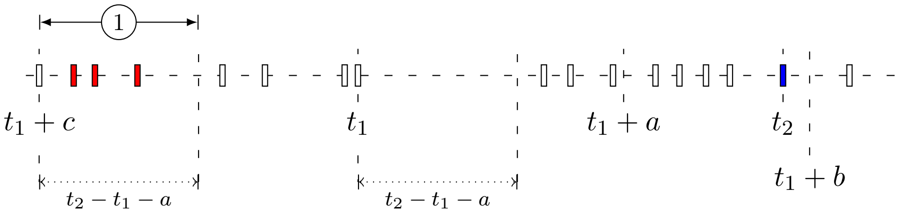
- $a \geq 0$ *and* $c < 0$: Let t_2 be the first time instant in $(t_1 + a, t_1 + b)$ where there is a φ_2-event. Consider the following subcases:
  - *There is no event in* $\big(t_1, t_1 + (t_2 - t_1 - a)\big)$: This can be enforced by

    $$\varphi = \mathbf{false}\ \mathfrak{U}^{0}_{(a,b)}\ \varphi_2\,.$$

    Then we can enforce that φ_1 holds at all events in ① in the illustration below by

    $$\varphi' = (\neg\varphi_2)\,\mathfrak{U}^{a}_{(a,b)}\,\big(\varphi_2 \wedge \underbrace{(\varphi_1\ \mathfrak{S}^{a+|c|}_{(a,b)}\ \mathbf{true})}_{\psi'}\big)\,.$$

    Observe that the subformula ψ' must hold at t_2 if φ_1 holds at all events in ①. This is because, by assumption, there must be an event at t_1.



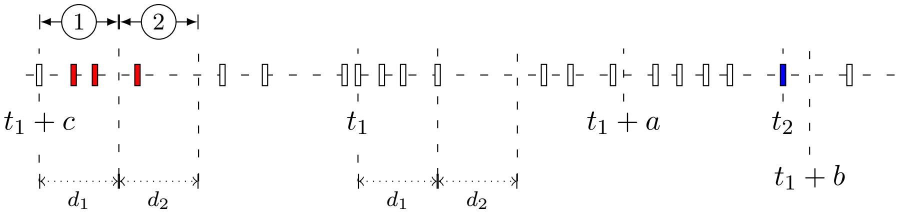
  - *There are events in* $\big(t_1, t_1 + (t_2 - t_1 - a)\big)$: In this case, φ' can only ensure that φ_1 holds at all events in ② (see the illustration below where $d_1 + d_2 = t_2 - t_1 - a$). We can enforce that φ_1 holds at all events in ① by

    $$\varphi'' = \psi''\,\mathcal{U}\,(\varphi \wedge \psi'')$$

    where

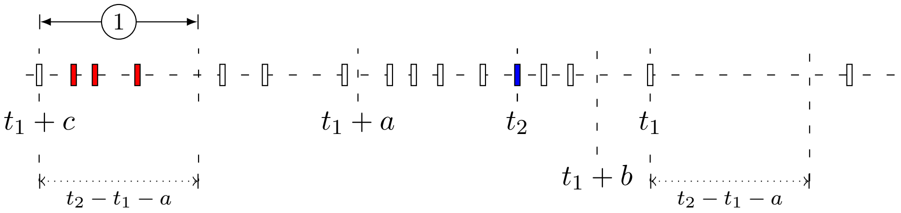
    $$\psi'' = \underbrace{(\varphi_1\ \mathfrak{S}^{|c|}_{(0,b-a)}\ \mathbf{true})}_{\psi'''} \wedge\neg(\Diamond_{=|c|}\,\neg\varphi_1)\,.$$

    It is easy to see that φ must hold at the last event in $\big(t_1, t_1 + (t_2 - t_1 - a)\big)$. The correctness of our use of the subformula ψ''' here again depends on the fact that there is an event at t_1.

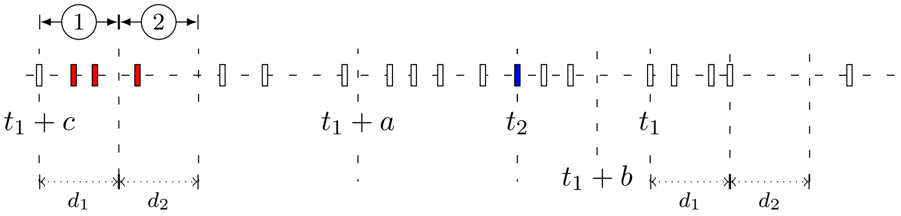


The desired formula is $\varphi' \wedge (\varphi \vee \varphi'')$.

- $a < 0$ *and* $c \geq 0$: Without loss of generality we assume $a < b < 0$. Similar to the case $a \geq 0$ *and* $c > a$ above, the desired formula is
$$\diamondminus_{(|b|,|a|)}\, \varphi_2 \wedge \neg\Big((\neg\varphi_2)\, \mathfrak{U}^{a}_{\left(c,c+(b-a)\right)}\, \big(\neg\varphi_1 \wedge \neg(\diamondminus_{=(c-a)}\, \varphi_2)\big)\Big)\,.$$
- $a < 0$ *and* $c \leq a$: Without loss of generality we assume $a < b < 0$. Let t_2 be the first time instant in $(t_1 + a, t_1 + b)$ where there is a φ_2-event. Similar to the case $a \geq 0$ and $c < 0$ above, consider the following subcases:
  - *There is no event in* $\big(t_1, t_1 + (t_2 - t_1 - a)\big)$: We enforce that φ_1 holds at all events in ① in the illustration below by
$$\varphi''' = \textbf{false}\ \mathfrak{U}^{0}_{(a,b)}\ \big(\varphi_2 \wedge (\varphi_1\, \mathfrak{S}^{a+|c|}_{(a,b)}\ \textbf{true})\big)\,.$$



  - *There are events in* $\big(t_1, t_1 + (t_2 - t_1 - a)\big)$: We enforce that φ_1 holds at all events in ① and ② in the illustration below (in which $d_1 + d_2 = t_2 - t_1 - a$) by
$$\diamondminus_{(|b|,|a|)}\, \varphi_2 \wedge \big(\psi''\, \mathcal{U}\, (\varphi''' \wedge \psi'')\big)\,,$$
where ψ'' is defined in the case $a \geq 0$ and $c < 0$ above.



The desired formula is the disjunction of these two formulas.
- $a < 0$ *and* $a < c < 0$: Without loss of generality we assume $a < b < 0$. The desired formula is identical to the formula in the case $a < 0$ and $c \geq 0$ above. □

We can now give an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula that distinguishes, in the pointwise semantics, the models $\mathcal{G}_m$ and $\mathcal{H}_m$ in the proof of Proposition 3.9:
$$\Diamond_{(1,2)}\big(\textbf{true} \wedge (\textbf{false}\, \mathfrak{U}^{-1}_{>0}\ \textbf{true})\big)\,.$$
This formula is 'equivalent' to the formula φ_{cont3} which distinguishes $\mathcal{G}_m$ and $\mathcal{H}_m$ in the continuous semantics.

3.4. **The translation.** We now give a translation from an arbitrary $\mathsf{FO}[<, +1]$ formula with a single free variable into an equivalent $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula over $[0, N)$-timed words. Our proof strategy closely follows [ORW09]: first convert the formula into a non-metric formula, then translate this formula into LTL, and finally construct an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula equivalent to the original formula. The crux of the translation is a *'stacking' bijection* between $[0, N)$-timed words over $\Sigma_{\mathbf{P}}$ and a set of $[0, 1)$-timed words over an extended alphabet. Roughly speaking, since the time domain is bounded, we can encode the integer parts of timestamps with a bounded number of new monadic predicates. This enables us to work instead with 'stacked' $[0, 1)$-timed words, in which only the ordering of events are relevant.

Stacking bounded timed words. For each monadic predicate $P \in \mathbf{P}$, we introduce fresh monadic predicates P_i, $0 \leq i \leq N-1$ and let the set of all these new monadic predicates be $\overline{\mathbf{P}}$. The intended meaning is that for $x \in [0, 1)$, $P_i(x)$ holds in a stacked $[0, 1)$-timed word iff P holds at time $i + x$ in the corresponding $[0, N)$-timed word. We also introduce $\overline{\mathbf{Q}} = \{Q_i \mid 0 \leq i \leq N-1\}$ such that for $x \in [0, 1)$, $Q_i(x)$ holds in a stacked $[0, 1)$-timed word iff there is an event at time $i + x$ in the corresponding $[0, N)$-timed word, regardless of whether any $P \in \mathbf{P}$ holds there. Let

$$\vartheta_{event} = \forall x \left(\bigvee_{0 \leq i \leq N-1} Q_i(x) \right) \land \forall x \left(\bigwedge_{0 \leq i \leq N-1} \big(P_i(x) \implies Q_i(x) \big) \right)$$

and $\vartheta_{init} = \exists x \left(\nexists x' \, (x' < x) \land Q_0(x) \right)$. There is an obvious 'stacking' bijection (indicated by overlining) between $[0, N)$-timed words over $\Sigma_{\mathbf{P}}$ and $[0, 1)$-timed words over $\Sigma_{\overline{\mathbf{P}} \cup \overline{\mathbf{Q}}}$ satisfying $\vartheta_{event} \land \vartheta_{init}$. For a concrete example, the stacked counterpart of the $[0, 2)$-timed word

$$\rho = (\{A\}, 0)(\{A, C\}, 0.3)(\{B\}, 1)(\{B, C\}, 1.5)$$

with $\mathbf{P} = \{A, B, C\}$ is the $[0, 1)$-timed word:

$$\overline{\rho} = (\{Q_0, Q_1, A_0, B_1\}, 0)(\{Q_0, A_0, C_0\}, 0.3)(\{Q_1, B_1, C_1\}, 0.5)\,.$$

Stacking $\mathsf{FO}[<, +1]$ *formulas.* Let $\vartheta(x)$ be an $\mathsf{FO}[<, +1]$ formula with a single free variable x where each quantifier uses a fresh new variable. Without loss of generality, we assume that $\vartheta(x)$ contains only existential quantifiers (this can be achieved by syntactic rewriting). Replace the formula by

$$\big(Q_0(x) \land \vartheta[x/x]\big) \lor \big(Q_1(x) \land \vartheta[x + 1/x]\big) \lor \ldots \lor \big(Q_{N-1}(x) \land \vartheta[x + (N-1)/x]\big)$$

where $\vartheta[e/x]$ denotes the formula obtained by substituting all free occurrences of x in ϑ by (an expression) e. Then, similarly, recursively replace every subformula $\exists x' \, \theta$ by

$$\exists x' \Big(\big(Q_0(x') \land \theta[x'/x']\big) \lor \ldots \lor \big(Q_{N-1}(x') \land \theta[x' + (N-1)/x']\big) \Big)\,.$$

Note that we do not actually have the $+k$ functions in our pointwise version of $\mathsf{FO}[<, +1]$; they are only used as annotations here and will be removed later, e.g., $x' + k$ means that $Q_k(x')$ holds. We then carry out the following syntactic substitutions:

- For each inequality of the form $x_1 + k_1 < x_2 + k_2$, replace it with
  - $x_1 < x_2$ if $k_1 = k_2$
  - **true** if $k_1 < k_2$
  - **false** if $k_1 > k_2$
- For each distance formula, e.g., $d(x_1 + k_1, x_2 + k_2) < 2$, replace it with

  - **true** if $|k_1 - k_2| \leq 1$
  - $x_2 < x_1$ if $k_2 - k_1 = 2$
  - $x_1 < x_2$ if $k_1 - k_2 = 2$
  - **false** if $|k_1 - k_2| > 2$
- Replace terms of the form $P(x_1 + k)$ with $P_k(x_1)$.

This gives a non-metric first-order formula $\overline{\vartheta}(x)$ over $\overline{\mathbf{P}} \cup \overline{\mathbf{Q}}$. Denote by *frac*$(t)$ the fractional part of a non-negative real t. It is not hard to see that for each $[0, N)$-timed word $\rho = (\sigma, \tau)$ over $\Sigma_{\mathbf{P}}$ and its stacked counterpart $\overline{\rho}$, the following holds:

- $\rho, t \models \vartheta(x)$ implies $\overline{\rho}, \overline{t} \models \overline{\vartheta}(x)$ where $\overline{t} = \mathit{frac}(t)$
- $\overline{\rho}, \overline{t} \models \overline{\vartheta}(x)$ implies there exists $t \in \rho$ with $\mathit{frac}(t) = \overline{t}$ such that $\rho, t \models \vartheta(x)$.

Moreover, if $\rho, t \models \vartheta(x)$, then the integer part of t indicates which disjunct in $\overline{\vartheta}(x)$ is satisfied when x is substituted with $\overline{t} = \mathit{frac}(t)$, and vice versa. By Kamp's theorem [Kam68] (applied individually on each $\vartheta[x + i/x]$), $\overline{\vartheta}(x)$ is equivalent to an LTL formula $\overline{\varphi}$ of the following form:

$$(Q_0 \wedge \overline{\varphi}_0) \vee (Q_1 \wedge \overline{\varphi}_1) \vee \ldots \vee (Q_{N-1} \wedge \overline{\varphi}_{N-1}) \,.$$

Unstacking. We construct inductively an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula ψ for each subformula $\overline{\psi}$ of $\overline{\varphi}_i$ (for some $i \in \{0, \ldots, N-1\}$). Again, we make use of the formulas in Φ_{int} defined earlier.

- $\overline{\psi} = P_j$: Let
$$\psi = (\varphi_{0,1} \wedge \Diamond_{=j} P) \vee \ldots \vee (\varphi_{j,j+1} \wedge P) \vee \ldots \vee (\varphi_{N-1,N} \wedge \diamondminus_{=((N-1)-j)} P) \,.$$
- $\overline{\psi} = Q_j$: Similarly, let
$$\psi = (\varphi_{0,1} \wedge \Diamond_{=j} \mathbf{true}) \vee \ldots \vee (\varphi_{j,j+1} \wedge \mathbf{true}) \vee \ldots \vee (\varphi_{N-1,N} \wedge \diamondminus_{=((N-1)-j)} \mathbf{true}) \,.$$
- $\overline{\psi} = \overline{\psi}_1 \, \mathcal{U} \, \overline{\psi}_2$: Let $\psi^{j,k,l} = \psi_1 \, \mathfrak{U}^{k}_{(j,j+1)} \, (\psi_2 \wedge \varphi_{l,l+1})$. The desired formula is
$$\psi = \bigvee_{0 \leq i \leq N-1} \left(\varphi_{i,i+1} \wedge \bigvee_{\substack{-i \leq j \leq (N-1)-i \\ l = i+j}} \left(\bigwedge_{-i \leq k \leq (N-1)-i} \psi^{j,k,l} \right) \right) .$$
- $\overline{\psi} = \overline{\psi}_1 \, \mathcal{S} \, \overline{\psi}_2$: This is symmetric to the case of $\overline{\psi}_1 \, \mathcal{U} \, \overline{\psi}_2$.

The construction for the other cases are trivial and therefore omitted.

Proposition 3.13. *Let $\overline{\psi}$ be a subformula of $\overline{\varphi}_i$ for some $i \in \{0, \ldots, N-1\}$. There is an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula ψ such that for any $[0, N)$-timed word ρ, $t \in \rho$ and $\mathit{frac}(t) = \overline{t} \in \overline{\rho}$, we have*
$$\overline{\rho}, \overline{t} \models \overline{\psi} \iff \rho, t \models \psi \,.$$

Proof. Induction on the structure of $\overline{\psi}$ and ψ, where the latter is constructed as described above.

- $\overline{\psi} = P_j$: Assume $\overline{\rho}, \overline{t} \models \overline{\psi}$. If $t = j + \overline{t}$, the disjunct $(\varphi_{j,j+1} \wedge P)$ of ψ clearly holds at t in ρ. If $t = j' + \overline{t}$ where $j' \neq j$, since there is a P-event at time $j + \overline{t}$ in ρ, the j'-th disjunct of ψ must hold at t in ρ. The proof for the other direction is similar.
- $\overline{\psi} = \overline{\psi}_1 \, \mathcal{U} \, \overline{\psi}_2$: Assume $\overline{\rho}, \overline{t} \models \overline{\psi}$ and let the 'witness' (i.e., where $\overline{\psi}_2$ holds) be at $\overline{t'}$. By construction and the induction hypothesis, there is an event at $t' = l + \overline{t}$ in ρ for some $l \in \{0, \ldots, N-1\}$ such that $\rho, t' \models \psi_2$. Moreover, since we have $\overline{\rho}, \overline{t''} \models \overline{\psi_1}$ for all $\overline{t''}$, $\overline{t} < \overline{t''} < \overline{t'}$, we must have $\rho, t'' \models \psi_1$ for all $t'' \in \rho$ with $t'' = k' + \overline{t''}$ for some $\overline{t} < \overline{t''} < \overline{t'}$

and $0 \leq k' \leq N-1$. Now let $t = i + \overline{t}$ for some $i \in \{0, \dots, N-1\}$ be a timestamp in ρ and let $j = l - i$. It is clear that $\rho, t \models \varphi_{i,i+1}$ and

$$\rho, t \models \bigwedge_{\substack{0 \leq k' \leq N-1 \\ k = k' - i}} \psi^{j,k,l} \,,$$

as required. For the other direction, let $t = i + \overline{t}$ for some $i \in \{0, \dots, N-1\}$ and let

$$\rho, t \models \bigwedge_{-i \leq k \leq (N-1)-i} \psi^{j,k,l}$$

for some $j \in \{-i, \dots, (N-1)-i\}$ and $l = i + j$. It follows that there is a (minimal) $\overline{t'} > \overline{t}$ such that $\rho, l + \overline{t'} \models \psi_2$ and $\rho, k' + \overline{t''} \models \psi_1$ for all $t'' \in \rho$ with $t'' = k' + \overline{t''}$ for some $\overline{t} < \overline{t''} < \overline{t'}$ and $0 \leq k' \leq N-1$. The claim follows by construction and the induction hypothesis.

The other cases are trivial or symmetric. □

Using the construction above, we obtain an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ_i for each $\overline{\varphi}_i$. Substitute them into $\overline{\varphi}$ and replace all remaining Q_i by $\varphi_{i,i+1}$ to obtain our final formula φ, which is equivalent to the original $\mathsf{FO}[<, +1]$ formula $\vartheta(x)$ over $[0, N)$-timed words.

Proposition 3.14. *For any $[0, N)$-timed word ρ and $t \in \rho$, we have*

$$\rho, t \models \varphi(x) \iff \rho, t \models \vartheta(x) \,.$$

We are now ready to state the main result of this section.

Theorem 3.15. $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ *is expressively complete for* $\mathsf{FO}[<, +1]$ *over* $[0, N)$*-timed words.*

3.5. Time-bounded verification.

We claim that the *timed-bounded satisfiability* and *time-bounded model-checking* problems for $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ are EXPSPACE-complete in both the pointwise and continuous semantics.

Theorem 3.16. *The time-bounded satisfiability problem for* $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ *(in both the pointwise and continuous semantics) is* EXPSPACE*-complete.*

Proof. First note that for each $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula over timed words one can construct, in linear time, an 'equivalent' $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula over signals of the form f^ρ. Then, in the continuous semantics, one can replace all subformulas of the form $\varphi_1 \, \mathfrak{U}^c_{(a,b)} \, \varphi_2$ in an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula by

$$(\Diamond_{=c} \, \varphi_1) \, \mathcal{U}_{<b} \, (\Diamond_{=a} \, \varphi_2)$$

(this can incur at most a linear blow-up). The claim therefore follows from [ORW09]. However, for the sake of completeness, we give a direct proof (for the case of pointwise semantics) along the lines of [ORW09] here; see Section 6 for a discussion on the practical implication.

For each subformula ψ of a given $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ and every $i \in \{0, \dots, N\}$, we introduce a monadic predicate F_i^ψ. We then add suitable subformulas into $\overline{\varphi}$ to ensure that F_i^ψ holds at $\overline{t}$ in $\overline{\rho}$ iff ψ holds at $t = \overline{t} + i$ in ρ. As an example, let $A \, \mathfrak{U}^1_{(2,3)} \, B$ be a subformula of φ. We require the following formula to hold at every point in time (assume that $i \leq N-4$):

$$\begin{aligned} F_i^{A \mathfrak{U}^1_{(2,3)} B} \iff & \big((F^Q_{i+1} \implies F^A_{i+1}) \, \mathcal{U} \, F^B_{i+2}\big) \\ & \vee \Big(\Box(F^Q_{i+1} \implies F^A_{i+1}) \wedge \diamondminus\big(F^B_{i+3} \wedge \boxminus(F^Q_{i+2} \implies F^A_{i+2})\big)\Big) \,. \end{aligned}$$

We also add the $\mathsf{LTL}_{\mathsf{fut}}$ equivalents of ϑ_{event} and ϑ_{init} into $\overline{\varphi}$ as conjuncts. It is clear that $\overline{\varphi}$ is of size exponential in the size of φ. EXPSPACE-hardness follows from the corresponding result of Bounded-MTL (in the pointwise semantics) in [BMOW07]. □

Since the time-bounded model-checking problem and satisfiability problem are inter-reducible in both the pointwise and continuous semantics [Wil94, HRS98], we have the following theorem.

Theorem 3.17. *The time-bounded model-checking problem for timed automata against* $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ *(in both the pointwise and continuous semantics) is* EXPSPACE*-complete.*

4. Expressive completeness of $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ over unbounded timed words

Recall that the counting modality $C_2(x, X)$ asserts that X holds at at least two points in $(x, x+1)$. While the modality is not expressible in MTL, it is equivalent to the following MTL formula with *rational* constants:

$$\Diamond_{(0,\frac{1}{2})}(X \wedge \Diamond_{(0,\frac{1}{2})} X) \vee \Diamond_{(\frac{1}{2},1)}(X \wedge \overleftarrow{\Diamond}_{(0,\frac{1}{2})} X) \vee (\Diamond_{(0,\frac{1}{2})} X \wedge \Diamond_{(\frac{1}{2},1)} X)\,.$$

Indeed, MTL with rational constants is expressively complete for $\mathsf{FO}[<, +\mathbb{Q}]$ (the rational version of $\mathsf{FO}[<, +1]$) over signals [HOW13]. Unfortunately, even with rational endpoints, MTL is still less expressive than $\mathsf{FO}[<, +1]$ in the pointwise semantics [PD06]. We show in this section that expressive completeness of MTL over (infinite) timed words can be recovered by adding (the rational versions of) the modalities generalised 'Until' ($\mathfrak{U}^c_I$) and generalised 'Since' ($\mathfrak{S}^c_I$) we introduced in the last section.

Our presentation in this section essentially follows [HOW13]. We first give a set of rewriting rules that 'extract' unbounded temporal operators from the scopes of bounded temporal operators. Then we invoke Gabbay's separation theorem [GPSS80] to obtain a syntactic separation result for $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ in the pointwise semantics. Exploiting a normal form for $\mathsf{FO}[<, +1]$ in [GPSS80], we show that any bounded $\mathsf{FO}[<, +\mathbb{Q}]$ formula can be rewritten into an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula. Finally, these ideas are combined to obtain the desired result.

4.1. Syntactic separation of $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas.

We present a series of logical equivalence rules that can be used to rewrite a given $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula into an equivalent formula in which no unbounded temporal operators occurs within the scope of a bounded temporal operator. Only the rules for open intervals are given, as the rules for other types of intervals are straightforward variants.

A normal form for $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$. We say an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula is in *normal form* if it satisfies:

(i). All occurrences of unbounded temporal operator are of the form $\mathcal{U}_{(0,\infty)}$, $\mathcal{S}_{(0,\infty)}$, $\Box_{(0,\infty)}$, $\overleftarrow{\Box}_{(0,\infty)}$.

(ii). All other occurrences of temporal operators are of the form $\mathcal{U}_I$, $\mathcal{S}_I$, $\mathfrak{U}^c_I$, $\mathfrak{S}^c_I$ with bounded I.

(iii). Negation is only applied to monadic predicates or bounded temporal operators.

(iv). In any subformula of the form $\varphi_1 \, \mathcal{U}_I \, \varphi_2$, $\varphi_1 \, \mathcal{S}_I \, \varphi_2$, $\Diamond_I \, \varphi_2$, $\overleftarrow{\Diamond}_I \, \varphi_2$, $\varphi_1 \, \mathfrak{U}^c_I \, \varphi_2$, $\varphi_1 \, \mathfrak{S}^c_I \, \varphi_2$ where I is bounded, φ_1 is a disjunction of subformulas and φ_2 is a conjunction of subformulas.

We now describe how to rewrite a given formula into normal form. To satisfy (i) and (ii), apply the usual rules (e.g., $\Box_I \varphi \iff \neg \Diamond_I \neg\varphi$) and the rules:

$$\begin{aligned}
\varphi_1 \mathcal{U}_{(a,\infty)} \varphi_2 &\iff \varphi_1 \mathcal{U} \varphi_2 \land \Box_{(0,a]}(\varphi_1 \land \varphi_1 \mathcal{U} \varphi_2)\\
\varphi_1 \mathfrak{U}^c_{(a,\infty)} \varphi_2 &\iff \varphi_1 \mathfrak{U}^c_{(a,2a]} \varphi_2 \lor \Big(\Diamond^w_{[0,c]} \big(\varphi_1 \mathcal{U}_{(a,\infty)} (\varphi_2 \lor \Diamond_{\leq a-c} \varphi_2)\big)\Big)\,.
\end{aligned}$$

To satisfy (iii), use the usual rules and the rule:

$$\neg(\varphi_1 \mathcal{U} \varphi_2) \iff \Box\neg\varphi_2 \lor \big(\neg\varphi_2 \mathcal{U} (\neg\varphi_2 \land \neg\varphi_1)\big)\,.$$

For (iv), use the usual rules of Boolean algebra and the rules below:

$$\begin{aligned}
\phi \mathcal{U}_I (\varphi_1 \lor \varphi_2) &\iff (\phi \mathcal{U}_I \varphi_1) \lor (\phi \mathcal{U}_I \varphi_2)\\
(\varphi_1 \land \varphi_2) \mathcal{U}_I \phi &\iff (\varphi_1 \mathcal{U}_I \phi) \land (\varphi_2 \mathcal{U}_I \phi)\\
\phi \mathfrak{U}^c_I (\varphi_1 \lor \varphi_2) &\iff (\phi \mathfrak{U}^c_I \varphi_1) \lor (\phi \mathfrak{U}^c_I \varphi_2)\\
(\varphi_1 \land \varphi_2) \mathfrak{U}^c_I \phi &\iff (\varphi_1 \mathfrak{U}^c_I \phi) \land (\varphi_2 \mathfrak{U}^c_I \phi)\,.
\end{aligned}$$

The rules for past temporal operators are as symmetric. We prove one of these rules as the others are simpler.

Proposition 4.1. *The following equivalence holds over infinite timed words:*

$$\varphi_1 \mathfrak{U}^c_{(a,\infty)} \varphi_2 \iff \varphi_1 \mathfrak{U}^c_{(a,2a]} \varphi_2 \lor \Big(\Diamond^w_{[0,c]} \big(\varphi_1 \mathcal{U}_{(a,\infty)} (\varphi_2 \lor \Diamond_{\leq a-c} \varphi_2)\big)\Big)\,.$$

Proof. Let the current position be i and the witness be at position w. Consider the following cases:

- $\tau_w \in (\tau_i + a, \tau_i + 2a]$: $\varphi_1 \mathfrak{U}^c_{(a,2a]} \varphi_2$ clearly holds.
- $\tau_w \in (\tau_i + 2a, \infty)$: Consider the following subcases:
  - φ_1 holds at all positions $j < w$ such that $\tau_j > \tau_i + c$: $\varphi_1 \mathcal{U}_{(a,\infty)} \varphi_2$ holds at the maximal position j' such that $\tau_{j'} \in [\tau_i, \tau_i + c]$.
  - φ_1 holds at all positions $j < w$ such that $\tau_j > \tau_i + c$ and $\tau_w - \tau_j > a - c$: By assumption, there is a position j' at which φ_1 does not hold and $\tau_w - \tau_{j'} \leq a - c$. Since $\tau_w > \tau_i + 2a$, we have $\tau_{j'} > \tau_i + a + c$. It follows that $\varphi_1 \mathcal{U}_{(a,\infty)} (\Diamond_{\leq a-c} \varphi_2)$ holds at the maximal position in $[\tau_i, \tau_i + c]$.

The other direction is obvious. □

Extracting unbounded operators from bounded operators. We now provide a set of rewriting rules that extract unbounded temporal operators from the scopes of bounded temporal operators. In what follows, let $\varphi_{xlb} = \mathbf{false}\, \mathcal{U}_{(0,b)}\, \mathbf{true}$, $\varphi_{ylb} = \mathbf{false}\, \mathcal{S}_{(0,b)}\, \mathbf{true}$ and

$$\begin{aligned}
\varphi_{ugb} = &\Big((\varphi_{xlb} \implies \Box_{(b,2b)} \varphi_1) \land \big(\neg\varphi_{ylb} \implies (\varphi_1 \land \Box_{(0,b]} \varphi_1)\big)\Big)\\
&\quad \mathcal{U}\Bigg(\big(\varphi_1 \land (\varphi_1 \mathcal{U}_{(b,2b)} \varphi_2)\big) \lor \Bigg(\neg\varphi_{ylb} \land \Big(\varphi_2 \lor \big(\varphi_1 \land (\varphi_1 \mathcal{U}_{(0,b]} \varphi_2)\big)\Big)\Bigg)\Bigg)\,,\\
\varphi_{ggb} = &\,\Box \Big((\varphi_{xlb} \implies \Box_{(b,2b)} \varphi_1) \land \big(\neg\varphi_{ylb} \implies (\varphi_1 \land \Box_{(0,b]} \varphi_1)\big)\Big)\,.
\end{aligned}$$

The intended meanings of the formulas φ_{ugb} and φ_{ggb} are similar (yet not identical) to $\varphi_1 \mathfrak{U}^b_{>b} \varphi_2$ and $\neg\big(\mathbf{true}\, \mathfrak{U}^b_{>b} (\neg\varphi_1)\big)$, respectively. Indeed, the equivalences in the following proposition still hold if we replace all occurrences of φ_{ugb} and φ_{ggb} by these simpler formulas. We, however, have to use these complicated formulas here as we aim to pull the unbounded 'Until' operator to the outermost level. The subformulas $\Box_{(b,2b)} \varphi_1$ and $\Box_{(0,b]} \varphi_1$ assert that

φ_1 holds continuously in short 'strips', and we use the subformulas φ_{xlb} and φ_{ylb} to ensure that each event before the point where φ_2 holds is covered by such a strip.

Proposition 4.2. *The following equivalences hold over infinite timed words:*

$$\begin{aligned}
\theta\,\mathcal{U}_{(a,b)}\,\big((\varphi_1\,\mathcal{U}\,\varphi_2)\wedge\chi\big) &\iff \theta\,\mathcal{U}_{(a,b)}\,\big((\varphi_1\,\mathcal{U}_{(0,2b)}\,\varphi_2)\wedge\chi\big)\vee\Big(\big(\theta\,\mathcal{U}_{(a,b)}\,(\Box_{(0,2b)}\,\varphi_1\wedge\chi)\big)\wedge\varphi_{ugb}\Big)\\
\theta\,\mathcal{U}_{(a,b)}\,(\Box\varphi\wedge\chi) &\iff \big(\theta\,\mathcal{U}_{(a,b)}\,(\Box_{(0,2b)}\,\varphi\wedge\chi)\big)\wedge\varphi_{ggb}\\
\theta\,\mathcal{U}_{(a,b)}\,\big((\varphi_1\,\mathcal{S}\,\varphi_2)\wedge\chi\big) &\iff \theta\,\mathcal{U}_{(a,b)}\,\big((\varphi_1\,\mathcal{S}_{(0,b)}\,\varphi_2)\wedge\chi\big)\vee\Big(\big(\theta\,\mathcal{U}_{(a,b)}\,(\boxminus_{(0,b)}\,\varphi_1\wedge\chi)\big)\wedge\varphi_1\,\mathcal{S}\,\varphi_2\Big)\\
\theta\,\mathcal{U}_{(a,b)}\,(\boxminus\varphi\wedge\chi) &\iff \big(\theta\,\mathcal{U}_{(a,b)}\,(\boxminus_{(0,b)}\,\varphi\wedge\chi)\big)\wedge\boxminus\varphi\\
\big((\varphi_1\,\mathcal{U}\,\varphi_2)\vee\chi\big)\,\mathcal{U}_{(a,b)}\,\theta &\iff \big((\varphi_1\,\mathcal{U}_{(0,2b)}\,\varphi_2)\vee\chi\big)\,\mathcal{U}_{(a,b)}\,\theta\\
&\qquad\vee\Bigg(\Big(\big((\varphi_1\,\mathcal{U}_{(0,2b)}\,\varphi_2)\vee\chi\big)\,\mathcal{U}_{(0,b)}\,(\Box_{(0,2b)}\,\varphi_1)\Big)\wedge\Diamond_{(a,b)}\,\theta\wedge\varphi_{ugb}\Bigg)\\
\big((\Box\varphi)\vee\chi\big)\,\mathcal{U}_{(a,b)}\,\theta &\iff \chi\,\mathcal{U}_{(a,b)}\,\theta\vee\big(\chi\,\mathcal{U}_{(0,b)}\,(\Box_{(0,2b)}\,\varphi_1)\wedge\Diamond_{(a,b)}\,\theta\wedge\varphi_{ggb}\big)\\
\big((\varphi_1\,\mathcal{S}\,\varphi_2)\vee\chi\big)\,\mathcal{U}_{(a,b)}\,\theta &\iff \big((\varphi_1\,\mathcal{S}_{(0,b)}\,\varphi_2)\vee\chi\big)\,\mathcal{U}_{(a,b)}\,\theta\\
&\qquad\vee\Bigg(\Big(\big(\boxminus_{(0,b)}\,\varphi_1\vee(\varphi_1\,\mathcal{S}_{(0,b)}\,\varphi_2)\vee\chi\big)\,\mathcal{U}_{(a,b)}\,\theta\Big)\wedge\varphi_1\,\mathcal{S}\,\varphi_2\Bigg)\\
\big((\boxminus\varphi)\vee\chi\big)\,\mathcal{U}_{(a,b)}\,\theta &\iff \chi\,\mathcal{U}_{(a,b)}\,\theta\vee\Big(\big((\boxminus_{(0,b)}\,\varphi\vee\chi)\,\mathcal{U}_{(a,b)}\,\theta\big)\wedge\boxminus\varphi\Big).
\end{aligned}$$

Proof. We sketch the proof for the first rule. In what follows, let the current position be i.

For the forward direction, let the witness be at position w. If $\tau_w < \tau_j + 2b$ for some j such that $\tau_j \in (\tau_i + a, \tau_i + b)$, the subformula $\varphi_1\,\mathcal{U}_{(0,2b)}\,\varphi_2$ clearly holds at j and we are done. Otherwise, let j be the maximal position such that $\tau_j \in (\tau_i + a, \tau_i + b)$. We know that $\Box_{(0,2b)}\,\varphi_1$ must hold at j, so $(\varphi_{xlb} \implies \Box_{(b,2b)}\,\varphi_1)$, φ_{ylb}, and hence $\big(\neg\varphi_{ylb} \implies (\varphi_1 \wedge \Box_{(0,b]}\,\varphi_1)\big)$ must hold at all positions j', $i < j' < j$. Let $l > j$ be the minimal position such that $\tau_w \in (\tau_l + b, \tau_l + 2b)$. Consider the following cases:

- There exists such l: It is clear that $\big(\varphi_1 \wedge (\varphi_1\,\mathcal{U}_{(b,2b)}\,\varphi_2)\big)$ holds at l. Since $\Box_{(b,2b)}\,\varphi_1$ holds at all positions j'', $j \le j'' < l$ by the minimality of l, $(\varphi_{xlb} \implies \Box_{(b,2b)}\,\varphi_1)$ also holds at these positions. For the other conjunct, note that φ_{ylb} holds at j and $\varphi_1 \wedge \Box_{(0,b]}\,\varphi_1$ holds at all positions j''', $j < j''' < l$.
- There is no such l: Consider the following cases:
  - $\neg\varphi_{ylb}$ and $\neg\diamondminus_{=b}\,\mathbf{true}$ hold at w: By assumption, there is no event in $(\tau_w - 2b, \tau_w)$. The proof is similar to the case where l exists.
  - $\neg\varphi_{ylb}$ and $\diamondminus_{=b}\,\mathbf{true}$ hold at w: Let l' be the position such that $\tau_{l'} = \tau_w - b$. By assumption, there is no event in $(\tau_{l'} - b, \tau_{l'})$. It follows that $\neg\varphi_{ylb}$ and $\big(\varphi_1 \wedge (\varphi_1\,\mathcal{U}_{(0,b]}\,\varphi_2)\big)$ hold at l'. The proof is similar.
  - φ_{ylb} holds at w: By assumption, there is no event in $(\tau_w - 2b, \tau_w - b)$. It is easy to see that there is a position such that $\neg\varphi_{ylb} \wedge \big(\varphi_1 \wedge (\varphi_1\,\mathcal{U}_{(0,b]}\,\varphi_2)\big)$ holds. The proof is again similar.

We prove the other direction by contraposition. Consider the interesting case where $\Box_{(0,2b)}\,\varphi_1$ holds at the maximal position j such that $j \in (\tau_i + a, \tau_i + b)$, yet $\varphi_1\,\mathcal{U}\,\varphi_2$ does not hold at j. By assumption, there is no φ_2-event in $(\tau_j, \tau_j + 2b)$. If φ_2 never holds in $[\tau_j + 2b, \infty)$ then we are done. Otherwise, let $l > j$ be the minimal position such that both φ_1 and φ_2 do not hold at l (note that $\tau_l \ge \tau_j + 2b$). It is clear that

$$\Bigg(\big(\varphi_1 \wedge (\varphi_1\,\mathcal{U}_{(b,2b)}\,\varphi_2)\big) \vee \Big(\neg\varphi_{ylb} \wedge \Big(\varphi_2 \vee \big(\varphi_1 \wedge (\varphi_1\,\mathcal{U}_{(0,b]}\,\varphi_2)\big)\Big)\Big)\Bigg)$$

does not hold at all positions j', $i < j' \leq l$. Consider the following cases:

- $\neg\varphi_{ylb}$ holds at l: $\varphi_1 \wedge \Box_{(0,b]}\,\varphi_1$ does not hold at l, and therefore φ_{ugb} fails to hold at i.
- φ_{ylb} holds at l: Consider the following cases:
  - There is an event in $(\tau_l - 2b, \tau_l - b)$: Let j'' be the maximal position of such an event. We have $j'' + 1 < l$, $\tau_{j''+1} - \tau_{j''} \geq b$ and $\tau_l - \tau_{j''+1} < b$. However, it follows that φ_{ylb} does not hold at $j'' + 1$ and $\varphi_1 \wedge \Box_{(0,b]}\,\varphi_1$ holds at $j'' + 1$, which is a contradiction.
  - There is no event in $(\tau_l - 2b, \tau_l - b)$: Let j'' be the minimal position such that $\tau_{j''} \in [\tau_l - b, \tau_l)$. It is clear that φ_{ylb} does not hold at j'' and $\varphi_1 \wedge \Box_{(0,b]}\,\varphi_1$ must hold at j'', which is a contradiction. □

Proposition 4.3. *The following equivalences hold over infinite timed words:*

$$\begin{aligned}
\theta\,\mathfrak{U}^c_{(a,b)}\,\big((\varphi_1\,\mathcal{U}\,\varphi_2)\wedge\chi\big) &\iff \theta\,\mathfrak{U}^c_{(a,b)}\,\big((\varphi_1\,\mathcal{U}_{(0,2b)}\,\varphi_2)\wedge\chi\big)\vee\Big(\big(\theta\,\mathfrak{U}^c_{(a,b)}\,(\Box_{(0,2b)}\,\varphi_1\wedge\chi)\big)\wedge\varphi_{ugb}\Big)\\
\theta\,\mathfrak{U}^c_{(a,b)}\,(\Box\varphi\wedge\chi) &\iff \big(\theta\,\mathfrak{U}^c_{(a,b)}\,(\Box_{(0,2b)}\,\varphi\wedge\chi)\big)\wedge\varphi_{ggb}\\
\theta\,\mathfrak{U}^c_{(a,b)}\,\big((\varphi_1\,\mathcal{S}\,\varphi_2)\wedge\chi\big) &\iff \theta\,\mathfrak{U}^c_{(a,b)}\,\big((\varphi_1\,\mathcal{S}_{(0,b)}\,\varphi_2)\wedge\chi\big)\vee\Big(\big(\theta\,\mathfrak{U}^c_{(a,b)}\,(\boxminus_{(0,b)}\,\varphi_1\wedge\chi)\big)\wedge\varphi_1\,\mathcal{S}\,\varphi_2\Big)\\
\theta\,\mathfrak{U}^c_{(a,b)}\,(\boxminus\varphi\wedge\chi) &\iff \big(\theta\,\mathfrak{U}^c_{(a,b)}\,(\boxminus_{(0,b)}\,\varphi\wedge\chi)\big)\wedge\boxminus\varphi\\
\big((\varphi_1\,\mathcal{U}\,\varphi_2)\vee\chi\big)\,\mathfrak{U}^c_{(a,b)}\,\theta &\iff \big((\varphi_1\,\mathcal{U}_{(0,2b)}\,\varphi_2)\vee\chi\big)\,\mathfrak{U}^c_{(a,b)}\,\theta\\
&\qquad\vee\Bigg(\bigg(\Big((\varphi_1\,\mathcal{U}_{(0,2b)}\,\varphi_2)\vee\chi\Big)\,\mathfrak{U}^c_{\big(c,c+(b-a)\big)}\,(\Box_{(0,2b)}\,\varphi_1)\bigg)\wedge\Diamond_{(a,b)}\,\theta\wedge\varphi_{ugb}\Bigg)\\
\big((\Box\varphi)\vee\chi\big)\,\mathfrak{U}^c_{(a,b)}\,\theta &\iff \chi\,\mathfrak{U}^c_{(a,b)}\,\theta\vee\big(\chi\,\mathfrak{U}^c_{\big(c,c+(b-a)\big)}\,(\Box_{(0,2b)}\,\varphi_1)\wedge\Diamond_{(a,b)}\,\theta\wedge\varphi_{ggb}\big)\\
\big((\varphi_1\,\mathcal{S}\,\varphi_2)\vee\chi\big)\,\mathfrak{U}^c_{(a,b)}\,\theta &\iff \big((\varphi_1\,\mathcal{S}_{(0,b)}\,\varphi_2)\vee\chi\big)\,\mathfrak{U}^c_{(a,b)}\,\theta\\
&\qquad\vee\Bigg(\bigg(\Big(\boxminus_{(0,b)}\,\varphi_1\vee(\varphi_1\,\mathcal{S}_{(0,b)}\,\varphi_2)\vee\chi\Big)\,\mathfrak{U}^c_{(a,b)}\,\theta\bigg)\wedge\varphi_1\,\mathcal{S}\,\varphi_2\Bigg)\\
\big((\boxminus\varphi)\vee\chi\big)\,\mathfrak{U}^c_{(a,b)}\,\theta &\iff \chi\,\mathfrak{U}^c_{(a,b)}\,\theta\vee\Big(\big((\boxminus_{(0,b)}\,\varphi\vee\chi)\,\mathfrak{U}^c_{(a,b)}\,\theta\big)\wedge\boxminus\varphi\Big)\,.
\end{aligned}$$

Lemma 4.4. *For any* $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ *formula* φ*, we can use the rules above to obtain an equivalent* $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ *formula* $\hat{\varphi}$ *in which no unbounded temporal operator appears in the scope of a bounded temporal operator. In particular, all occurrences of* $\mathfrak{U}^c_I, \mathfrak{S}^c_I$ *have* I *bounded.*

Proof. Define the *unbounding depth* $ud(\varphi)$ of an $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ formula φ to be the modal depth of φ counting only unbounded operators. We demonstrate a rewriting process on φ which terminates in an equivalent formula $\hat{\varphi}$ such that any subformula $\hat{\psi}$ of $\hat{\varphi}$ with outermost operator bounded has $ud(\hat{\psi}) = 0$.

Assume that the input formula φ is in normal form. Let k be the largest unbounding depth among all subformulas of φ with bounded outermost operators. We pick all minimal (w.r.t. subformula order) such subformulas ψ with $ud(\psi) = k$. By applying the rules in Section 4.1, we can rewrite ψ into ψ' where all subformulas of ψ' with bounded outermost operators have unbounded depths strictly less than k. We then substitute these ψ' back into φ to obtain φ'. We repeat this step until there remain no bounded temporal operators with unbounding depth k. The rules that rewrite a formula into normal form are used whenever necessary on relevant subformulas—this never affects their unbounding depths, and note that we never introduce $\mathfrak{U}^c_I$ or $\mathfrak{S}^c_I$. It is easy to see that we will eventually obtain such a formula φ^*. Now rewrite φ^* into normal form and start over again. This is to be repeated until we reach $\hat{\varphi}$. □

Completing the separation. We now have an $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ formula $\hat{\varphi}$ in which no unbounded temporal operator appears in the scope of a bounded temporal operator. If we regard each bounded subformula as a new monadic predicate, the formula $\hat{\varphi}$ can be seen as an LTL formula Φ, on which Gabbay's separation theorem is applicable.

Theorem 4.5 [GPSS80, Theorem 3]**.** *Every* LTL *formula is equivalent (over discrete complete models) to a Boolean combination of*

- *atomic formulas*
- *formulas of the form* $\varphi_1 \,\mathcal{U}\, \varphi_2$ *such that* φ_1 *and* φ_2 *use only* $\mathcal{U}$
- *formulas of the form* $\varphi_1 \,\mathcal{S}\, \varphi_2$ *such that* φ_1 *and* φ_2 *use only* $\mathcal{S}$.

Lemma 4.6. *Every* $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ *formula is equivalent to a Boolean combination of*

- *bounded* $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ *formulas*
- *formulas that use arbitrary* $\mathcal{U}_I$ *but only bounded* $\mathcal{S}_I$, $\mathfrak{U}^c_I$, $\mathfrak{S}^c_I$
- *formulas that use arbitrary* $\mathcal{S}_I$ *but only bounded* $\mathcal{U}_I$, $\mathfrak{U}^c_I$, $\mathfrak{S}^c_I$.

We now prove the main theorem of this subsection: each $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ formula is equivalent to a *syntactically separated* $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ formula.

Theorem 4.7. *Every* $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ *formula can be written as a Boolean combination of*

- *bounded* $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ *formulas*
- *formulas of the form* $\mathbf{false}\,\mathfrak{U}^M_{\geq M}\,\varphi$ *where* $M \in \mathbb{N}$
- *formulas of the form* $\mathbf{false}\,\mathfrak{S}^M_{\geq M}\,\varphi$ *where* $M \in \mathbb{N}$.

Proof. Suppose that we have an $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ formula φ with no unbounded $\mathcal{S}$. If φ is bounded then we are done. Otherwise we can apply Lemma 4.4 (note in particular that it does not introduce new unbounded $\mathcal{U}$ operators) and further assume that $\varphi = \varphi_1 \,\mathcal{U}\, \varphi_2$. Then, for any $M \in \mathbb{N}$, we can rewrite φ into

$$\varphi_1 \,\mathcal{U}_{<M}\, \varphi_2 \vee \left(\Box_{<M}\, \varphi_1 \wedge \left(\mathbf{false}\,\mathfrak{U}^M_{\geq M}\, \Big(\varphi_2 \vee \big(\varphi_1 \wedge (\varphi_1 \,\mathcal{U}\, \varphi_2)\big)\Big)\right)\right).$$

It is clear that φ_1 and φ_2, and therefore $\varphi_1 \mathcal{U}_{<M} \varphi_2$ and $\Box_{<M}\,\varphi_1$, have strictly fewer unbounded $\mathcal{U}$ operators than φ. By the induction hypothesis, φ is equivalent to a syntactically separated $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ formula. The case of formulas with no unbounded $\mathcal{U}$ is symmetric. □

4.2. Expressing bounded $\mathsf{FO}[<,+1]$ formulas.

In this section, we describe how to express bounded $\mathsf{FO}[<,+1]$ formulas with a single free variable in $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$. The use of rational constants is crucial here; by [HR07], $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ cannot express all counting modalities (which can be written as bounded $\mathsf{FO}[<,+1]$ formulas) if only integer constants are allowed. As some techniques here are exactly similar to that of Section 3.4, we omit certain explanations.

Suppose that we are given such a formula $\vartheta(x)$. As before, we assume that each quantifier in $\vartheta(x)$ uses a fresh new variable and $\vartheta(x)$ contains only existential quantifiers. We say that $\vartheta(x)$ is *N-bounded* if each subformula $\exists x'\,\psi$ of $\vartheta(x)$ is of the form

$$\exists x' \big((x' > x \implies d(x,x') < N) \wedge (x' < x \implies d(x,x') \leq N) \wedge \dots\big).$$

Namely, $\vartheta(x)$ only refers to the events in the half-open interval $[x - N, x + n)$. Similarly, we say that $\vartheta(x)$ is a *unit formula* if each subformula $\exists x'\,\psi$ of $\vartheta(x)$ is of the form

$$\exists x' \big(x' \geq x \wedge d(x,x') < 1 \wedge \dots\big).$$

In this case, $\vartheta(x)$ only refers to the events in $[x, x+1)$.

Stacking events around a point. Let ρ be an infinite timed word over $\Sigma_{\mathbf{P}}$, $\overline{\mathbf{P}} = \{P_i \mid P \in \mathbf{P}, -N \le i < N\}$ and $\overline{\mathbf{Q}} = \{Q_i \mid N \le i < N\}$. For each $t \in \rho$, we can construct a (finite) $[0,1)$-timed word $\overline{\rho_t}$ over $\Sigma_{\overline{\mathbf{P}} \cup \overline{\mathbf{Q}}}$ that satisfies the following:

- For all $\overline{t} \in [0,1)$ and $-N \le i < N$, P_i holds at $\overline{t} \in \overline{\rho_t}$ iff P holds at $i + \overline{t} \in \rho$.
- For all $\overline{t} \in [0,1)$ and $-N \le i < N$, Q_i holds at $\overline{t} \in \overline{\rho_t}$ iff $i + \overline{t} \in \rho$.

Stacking N-bounded $\mathsf{FO}[<,+1]$ *formulas.* Now let $\vartheta(x)$ be an N-bounded $\mathsf{FO}[<,+1]$ formula. Recursively replace every subformula $\exists x'\, \theta$ by

$$\exists x' \Big(\big(Q_{-N}(x') \land \theta[x' + (-N)/x']\big) \lor \ldots \lor \big(Q_{N-1}(x') \land \theta[x' + (N-1)/x']\big) \Big)$$

where $\vartheta[e/x]$ denotes the formula obtained by substituting all free occurrences of x in ϑ by e. We then carry out the following syntactic substitutions:

- For each inequality of the form $x_1 + k_1 < x_2 + k_2$, replace it with
  - $x_1 < x_2$ if $k_1 = k_2$
  - **true** if $k_1 < k_2$
  - **false** if $k_1 > k_2$
- For each distance formula, e.g., $d(x_1 + k_1, x_2 + k_2) < 2$, replace it with
  - **true** if $|k_1 - k_2| \le 1$
  - $x_2 < x_1$ if $k_2 - k_1 = 2$
  - $x_1 < x_2$ if $k_1 - k_2 = 2$
  - **false** if $|k_1 - k_2| > 2$
- Replace terms of the form $P(x_1 + k)$ with $P_k(x_1)$.

Finally, recursively replace every subformula $\exists x'\, \theta$ by $\exists x' \big(x' \ge x \land d(x, x') < 1 \land \theta\big)$. This gives a unit formula $\overline{\vartheta}(x)$ such that for each $t \in \rho$,

$$\rho, t \models \vartheta(x) \iff \overline{\rho_t}, 0 \models \overline{\vartheta}(x)\,.$$

Unstacking. For each $\overline{\rho_t}$, we add an event at time 1 (at which no monadic predicate holds) and call the resulting $[0,1)$-timed word $\overline{\rho_t}'$. It is clear that

$$\overline{\rho_t}, 0 \models \overline{\vartheta}(x) \iff \overline{\rho_t}', 0, 1 \models \overline{\vartheta}'(x, y)$$

where $\overline{\vartheta}'(x, y)$ is a non-metric $\mathsf{FO}[<]$ formula obtained by replacing all distance formulas of the form $d(x, x') < 1$ with $x' < y$ in $\overline{\vartheta}(x)$. We now invoke a normal form lemma from [GPSS80] to rewrite $\overline{\vartheta}'(x, y)$ into a disjunction of *decomposition formulas*.

Lemma 4.8 [GPSS80]**.** *Every* $\mathsf{FO}[<]$ *formula* $\theta(x, y)$ *in which all quantifications are of the form* $\exists x' \, (x' \ge x \land x' < y \land \ldots)$ *is equivalent to a disjunction of decomposition formulas, i.e.,* $\mathsf{FO}[<]$ *formulas of the form*

$$\begin{aligned} x < y \ &\land \exists z_0 \ldots \exists z_n \, (x = z_0 < \cdots < z_n = y) \\ &\land \textstyle\bigwedge \{\Phi_i(z_i) : 0 \le i < n\} \\ &\land \textstyle\bigwedge \{\forall u \, \big(z_i < u < z_{i+1} \implies \Psi_i(u)\big) : 0 \le i < n\} \end{aligned}$$

where Φ_i *and* Ψ_i *are* $\mathsf{LTL}_{\mathsf{fut}}$ *formulas.*[12]

[12] This version of the lemma follows from Lemma 4 and Main Lemma in [GPSS80].

In fact, when the underlying order is discrete (as is the case here), we can further postulate that Φ_i and Ψ_i are simply Boolean combinations of atomic formulas [Dam94]. It follows that $\overline{\vartheta}(x)$ is equivalent to a disjunction of unit formulas $\overline{\delta}(x)$ of the form

$$\begin{aligned}\exists z_0 \ldots \exists z_{n-1}\, (x = z_0 < \cdots < z_{n-1}) \;&\wedge d(x, z_{n-1}) < 1 \\ &\wedge \bigwedge\{\Phi_i(z_i) : 0 \leq i < n\} \\ &\wedge \bigwedge\{\forall u \left(z_i < u < z_{i+1} \implies \Psi_i(u)\right) : 0 \leq i < n-1\} \\ &\wedge \forall u \left(z_{n-1} < u \wedge d(x, u) < 1 \implies \Psi_{n-1}(u)\right)\end{aligned}$$

where Φ_i and Ψ_i are Boolean combinations of atomic formulas.

It remains to show that for each such unit formula $\overline{\delta}(x)$ and each $t \in \rho$, we can construct an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ such that

$$\overline{\rho_t}, 0 \models \overline{\delta}(x) \iff \rho, t \models \varphi\,.$$

For later convenience, we prove a stronger claim, i.e., we can handle $\mathsf{FO}[<, +\mathbb{Q}]$ formulas of the following form for any rational number r, $0 \leq r < 1$:

$$\begin{aligned}\exists z_0 \ldots \exists z_{n-1}\, (x = z_0 < \cdots < z_{n-1}) \;&\wedge d(x, z_1) > r \wedge d(x, z_{n-1}) < 1 \\ &\wedge \bigwedge\{\Phi_i(z_i) : 1 \leq i < n\} \\ &\wedge \forall u \left(x < u \wedge u < z_1 \wedge d(x, u) > r \implies \Psi_0(u)\right) \\ &\wedge \bigwedge\{\forall u \left(z_i < u < z_{i+1} \implies \Psi_i(u)\right) : 1 \leq i < n-1\} \\ &\wedge \forall u \left(z_{n-1} < u \wedge d(x, u) < 1 \implies \Psi_{n-1}(u)\right).\end{aligned}$$

The proof is by induction on the number of existential quantifiers in $\overline{\delta}(x)$. Before we proceed with the proof, we define a function f that maps a Boolean combination Ω of atomic formulas over $\overline{\mathbf{P}} \cup \overline{\mathbf{Q}}$ and i, $-N \leq i < N$ to an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula $f(\Omega, i)$ over $\mathbf{P}$:

- $f(P_j, i) = \begin{cases} \diamondminus_{=(i-j)} P & \text{if } i > j \\ P & \text{if } i = j \\ \Diamond_{=(j-i)} P & \text{if } i < j \end{cases}$
- $f(Q_j, i) = \begin{cases} \diamondminus_{=(i-j)} \mathbf{true} & \text{if } i > j \\ \mathbf{true} & \text{if } i = j \\ \Diamond_{=(j-i)} \mathbf{true} & \text{if } i < j \end{cases}$
- $f(\mathbf{true}, i) = \mathbf{true}$
- $f(\Omega_1 \wedge \Omega_2, i) = f(\Omega_1, i) \wedge f(\Omega_2, i)$
- $f(\neg\Omega, i) = \neg f(\Omega, i)$.

Now first consider the base step. We have

$$\overline{\delta}(x) = \forall u \left(x < u \wedge d(x, u) > r \wedge d(x, u) < 1 \implies \Psi(u)\right)$$

where Ψ is a Boolean combination of atomic formulas. It is clear that

$$\varphi = \bigwedge_{0 \leq i < N} \left(\Box_{(i+r, i+1)}\, f(\Psi, i)\right) \wedge \bigwedge_{-N \leq i < 0} \left(\boxminus_{(|i+1|, |i+r|)}\, f(\Psi, i)\right).$$

For the induction step we need to consider how $z_1, \ldots, z_{n-1}$ are scattered in $(r, 1)$. Let us split $(r, 1)$ into an open interval $(r, r + \frac{1-r}{2n})$ and $2n - 1$ half-open intervals $[r + \frac{1-r}{2n}, r + \frac{2(1-r)}{2n})$, $[r + \frac{2(1-r)}{2n}, r + \frac{3(1-r)}{2n})$, …, $[r + \frac{(2n-1)(1-r)}{2n}, 1)$. Consider the following cases:

(i). $\{z_1, \ldots, z_{n-1}\} \subseteq (r, r + \frac{1-r}{2n})$ or $\{z_1, \ldots, z_{n-1}\} \subseteq [r + \frac{k(1-r)}{2n}, r + \frac{(k+1)(1-r)}{2n})$ for some k, $1 \leq k < n$.

(ii). $\{z_1, \ldots, z_{n-1}\} \subseteq [r + \frac{k(1-r)}{2n}, r + \frac{(k+1)(1-r)}{2n})$ for some k, $n \leq k < 2n$.

(iii). There exists k, $1 \leq k < 2n$ and l, $1 \leq l < n-1$ such that $z_l < r + \frac{k(1-r)}{2n} \leq z_{l+1}$ (i.e., $z_1, \ldots, z_{n-1}$ are not in a single interval).

We detail the construction of a formula ψ in each case; the desired formula φ is the disjunction of these ψ. The correctness proofs are omitted as they are similar to the proof of Proposition 3.13.

- Case (i): Consider the subcase $z_1 > r + \frac{k(1-r)}{2n}$. Let

$$\overrightarrow{\varphi}^{i}_{n-1} = \bigwedge_{0 \leq j < N-i} \big(\Box_{(j, j+\frac{1-r}{2n})} f(\Psi_{n-1}, i+j) \big) \wedge \bigwedge_{-N-i \leq j < 0} \big(\boxminus_{(|j+\frac{1-r}{2n}|, |j|)} f(\Psi_{n-1}, i+j) \big)$$

for all i, $-N \leq i < N$ and recursively define

$$\overrightarrow{\varphi}^{i}_{m} = \bigvee_{-N-i \leq j < N-i} \Bigg(\bigwedge_{-N-i \leq h < N-i} \Big(\big(f(\Psi_m, i+h) \big) \, \mathfrak{U}^{h}_{(j, j+\frac{1-r}{2n})} \, \big(f(\Phi_{m+1}, i+j) \wedge \overrightarrow{\varphi}^{i+j}_{m+1} \big) \Big) \Bigg)$$

for all i, $-N \leq i < N$ and m, $1 \leq m < n-1$. Let α_k be the conjunction of

$$\bigwedge_{0 \leq i < N} \big(\Box_{(i+r, i+r+\frac{k(1-r)}{2n}]} f(\Psi_0, i) \big) \wedge \bigwedge_{-N \leq i < 0} \big(\boxminus_{[|i+r+\frac{k(1-r)}{2n}|, |i+r|)} f(\Psi_0, i) \big)$$

and

$$\bigvee_{-N \leq j < N} \Bigg(\bigwedge_{-N \leq h < N} \Big(\big(f(\Psi_0, h) \big) \, \mathfrak{U}^{h+r+\frac{k(1-r)}{2n}}_{(j+r+\frac{k(1-r)}{2n}, j+r+\frac{(k+1)(1-r)}{2n})} \, \big(f(\Phi_1, j) \wedge \overrightarrow{\varphi}^{j}_{1} \big) \Big) \Bigg)$$

and

$$\bigwedge_{0 \leq i < N} \big(\Box_{[i+r+\frac{(k+1)(1-r)}{2n}, i+1)} f(\Psi_{n-1}, i) \big) \wedge \bigwedge_{-N \leq i < 0} \big(\boxminus_{(|i+1|, |i+r+\frac{(k+1)(1-r)}{2n}|]} f(\Psi_{n-1}, i) \big) .$$

Similarly, we construct α'_k to handle the subcase $z_1 = r + \frac{k(1-r)}{2n}$. The formula ψ is the disjunction of formulas $\{\alpha_k \mid 0 \leq k < n\}$ and $\{\alpha'_k \mid 0 < k < n\}$.

- Case (ii): Let

$$\overleftarrow{\varphi}^{i}_{1} = \bigwedge_{0 < j < N-i} \big(\Box_{(j-\frac{1-r}{2n}, j)} f(\Psi_0, i+j) \big) \wedge \bigwedge_{-N-i \leq j \leq 0} \big(\boxminus_{(|j|, |j-\frac{1-r}{2n}|)} f(\Psi_0, i+j) \big)$$

for all i, $-N \leq i < N$ and recursively define

$$\overleftarrow{\varphi}^{i}_{m} = \bigvee_{-N-i \leq j < N-i} \Bigg(\bigwedge_{-N-i \leq h < N-i} \Big(\big(f(\Psi_{m-1}, i+h) \big) \, \mathfrak{S}^{h}_{(j, j+\frac{1-r}{2n})} \, \big(f(\Phi_{m-1}, i+j) \wedge \overleftarrow{\varphi}^{i+j}_{m-1} \big) \Big) \Bigg)$$

for all i, $-N \leq i < N$ and m, $1 < m \leq n-1$. Let β_k be the conjunction of

$$\bigwedge_{0 \leq i < N} \big(\Box_{[i+r+\frac{(k+1)(1-r)}{2n}, i+1)} f(\Psi_{n-1}, i) \big) \wedge \bigwedge_{-N \leq i < 0} \big(\boxminus_{(|i+1|, |i+r+\frac{(k+1)(1-r)}{2n}|]} f(\Psi_{n-1}, i) \big)$$

and

$$\bigvee_{-N \leq j < N} \Bigg(\bigwedge_{-N \leq h < N} \Big(\big(f(\Psi_{n-1}, h) \big) \, \mathfrak{S}^{-(h+r+\frac{(k+1)(1-r)}{2n})}_{(-(j+r+\frac{(k+1)(1-r)}{2n}), -(j+r+\frac{k(1-r)}{2n})]} \, \big(f(\Phi_{n-1}, j) \wedge \overleftarrow{\varphi}^{j}_{n-1} \big) \Big) \Bigg)$$

and

$$\bigwedge_{0 \leq i < N} \big(\Box_{(i+r, i+r+\frac{k(1-r)}{2n})} f(\Psi_0, i) \big) \wedge \bigwedge_{-N \leq i < 0} \big(\boxminus_{(|i+r+\frac{k(1-r)}{2n}|, |i+r|)} f(\Psi_0, i) \big) .$$

The formula ψ is the disjunction of β_k, $n \leq k < 2n$.

- Case (iii): Suppose that $z_l < r + \frac{k(1-r)}{2n} \leq z_{l+1}$ for some k, $1 \leq k < 2n$ and l, $1 \leq l < n-1$. Consider the following subcases:
  - $r + \frac{k(1-r)}{2n} < z_{l+1}$: This can be handled by the conjunction of the formulas below:
    * $\{z_1, \ldots, z_l\} \subseteq (r, r + \frac{k(1-r)}{2n})$: We can scale the corresponding $\mathsf{FO}[<, +\mathbb{Q}]$ formula by $\frac{1}{r+\frac{k(1-r)}{2n}}$, apply the induction hypothesis (with $r' = \frac{r}{r+\frac{k(1-r)}{2n}}$) and scale the resulting $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula by $r + \frac{k(1-r)}{2n}$.
    * $\{z_{l+1}, \ldots, z_{n-1}\} \subseteq (r + \frac{k(1-r)}{2n}, 1)$: We can set $r' = r + \frac{k(1-r)}{2n}$ and apply the induction hypothesis.
  - $r + \frac{k(1-r)}{2n} = z_{l+1}$: Exactly similar except that we also use the following formula as a conjunct:

$$\bigvee_{0 \leq i < N} \left(\Diamond_{=i+r+\frac{k(1-r)}{2n}} f(\Phi_{l+1}, i) \right) \vee \bigvee_{-N \leq i < 0} \left(\diamondminus_{=|i+r+\frac{k(1-r)}{2n}|} f(\Phi_{l+1}, i) \right).$$

The formula ψ is the disjunction of these formulas for all k, $1 \leq k < 2n$ and l, $1 \leq l < n-1$.

Finally, observe that the original claim can be achieved by setting $r = 0$ and using the conjunct $f(\Phi_0, 0)$. We can now state the following theorem.

Theorem 4.9. *For every N-bounded $\mathsf{FO}[<, +1]$ formula $\vartheta(x)$, there exists an equivalent $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ (with rational constants).*

4.3. Expressive completeness of $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$.

In this section, we show that any $\mathsf{FO}[<, +\mathbb{Q}]$ formula with one free variable can be expressed as an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula. The crucial idea here is that we can separate formulas 'far enough' that all references to a certain variable become vacuous. To this end, we define $fr(\varphi)$ and $pr(\varphi)$ (*future-reach* and *past-reach*) for an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ as follows:

- $fr(\mathbf{true}) = pr(\mathbf{true}) = fr(P) = pr(P) = 0$ for all $P \in \mathbf{P}$
- $fr(\varphi_1 \, \mathcal{U}_I \, \varphi_2) = \sup(I) + \max\big(fr(\varphi_1), fr(\varphi_2)\big)$
- $pr(\varphi_1 \, \mathcal{S}_I \, \varphi_2) = \sup(I) + \max\big(pr(\varphi_1), pr(\varphi_2)\big)$
- $fr(\varphi_1 \, \mathcal{S}_I \, \varphi_2) = \max\big(fr(\varphi_1), fr(\varphi_2) - \inf(I)\big)$
- $pr(\varphi_1 \, \mathcal{U}_I \, \varphi_2) = \max\big(pr(\varphi_1), pr(\varphi_2) - \inf(I)\big)$
- $fr(\varphi_1 \, \mathfrak{U}^c_I \, \varphi_2) = \max\big(c + |I| + fr(\varphi_1), \sup(I) + fr(\varphi_2)\big)$
- $pr(\varphi_1 \, \mathfrak{S}^c_I \, \varphi_2) = \max\big(c + |I| + pr(\varphi_1), \sup(I) + pr(\varphi_2)\big)$
- $fr(\varphi_1 \, \mathfrak{S}^c_I \, \varphi_2) = \max\big(fr(\varphi_1) - c, fr(\varphi_2) - \inf(I)\big)$
- $pr(\varphi_1 \, \mathfrak{U}^c_I \, \varphi_2) = \max\big(pr(\varphi_1) - c, pr(\varphi_2) - \inf(I)\big)$.

The cases for Boolean connectives are defined in the expected way. Intuitively, these are meant as over-approximations of the lengths of the time horizons needed to determine the truth value of φ.

Theorem 4.10. *For every $\mathsf{FO}[<, +1]$ formula $\vartheta(x)$, there exists an equivalent $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ (with rational constants).*

Proof. The proof is by induction on the quantifier depth of $\vartheta(x)$. In what follows, let the set of monadic predicates be $\mathbf{P}$. As before, we assume that each quantifier in $\vartheta(x)$ uses a fresh new variable.

- *Base step.* $\vartheta(x)$ is a Boolean combination of atomic formulas $P(x)$, $x < x$, $d(x, x) \sim c$, **true**. We can replace them by P, **false**, $0 \sim c$ and **true** respectively to obtain φ.

- *Induction step.* Without loss of generality assume that $\vartheta(x) = \exists y\, \theta(x, y)$. Our goal here is to remove x from $\theta(x, y)$. For this purpose, we can remove $x < x$ and $d(x, x) \sim c$ as before and use a mapping $\gamma : \mathbf{P} \mapsto \{\mathbf{true}, \mathbf{false}\}$ to determine the truth value of $P(x)$ for each $P \in \mathbf{P}$. Thus we can rewrite $\exists y\, \theta(x, y)$ as

$$\bigvee_{\gamma} \big(\eta_\gamma(x) \wedge \exists y\, \theta_\gamma(x, y)\big) \tag{4.1}$$

  where

$$\eta_\gamma = \bigwedge_{P \in \mathbf{P}} \big(P(x) \iff \gamma(P)\big)$$

  and $\theta_\gamma(x, y)$ is obtained by replacing $P(x)$ with $\gamma(P)$ for all $P \in \mathbf{P}$ in $\theta(x, y)$. Observe that in each $\theta_\gamma(x, y)$, x only appears in atomic formulas of the form $x < z$, $z < x$, $d(x, z) \sim c$ and $d(z, x) \sim c$ where $\sim\, \in \{<, >\}$. We now introduce new monadic predicates $P_<$, $P_>$, and $P_{\sim c}$ for each $d(x, z) \sim c$ or $d(z, x) \sim c$ that correspond to these atomic formulas. Namely, we write $x < z$ as $P_<(z)$, $z < x$ as $P_>(z)$, and $d(x, z) \sim c$ or $d(z, x) \sim c$ as $P_{\sim c}(z)$. Apply these substitutions to (4.1) yields

$$\bigvee_{\gamma} \big(\eta_\gamma(x) \wedge \exists y\, \theta'_\gamma(y)\big) \tag{4.2}$$

  where x does not occur in each $\theta'_\gamma(y)$. In particular, (4.1) and (4.2) have the same truth value at any given point if $P_<$, $P_>$ and all $P_{\sim c}$ are interpreted correctly with respect to that point. Each $\eta_\gamma(x)$ is clearly equivalent to an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula ψ_γ. By the induction hypothesis, each $\theta'_\gamma(y)$ is also equivalent to an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ_γ. It follows that (4.2) is equivalent to the following $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula:

$$\varphi' = \bigvee_{\gamma} \big(\psi_\gamma \wedge (\diamondminus\, \varphi_\gamma \vee \varphi_\gamma \vee \Diamond\, \varphi_\gamma)\big)\,.$$

  By Theorem 4.7 and noting that $M \in \mathbb{N}$ can be chosen arbitrarily, φ' is equivalent to a Boolean combination φ'' of
  – bounded formulas
  – formulas of the form $\mathbf{false}\, \mathfrak{U}^{M}_{\geq M}\, \psi$ such that $M > c_{max} + pr(\psi)$
  – formulas of the form $\mathbf{false}\, \mathfrak{S}^{M}_{\geq M}\, \psi$ such that $M > c_{max} + fr(\psi)$.
  where c_{max} are the largest constants appearing in monadic predicates of the form $P_{\sim c}$ in respective formulas ψ. Now suppose that φ'' is evaluated at t_1. For the formulas of the second type, since all references to $P_<$, $P_>$ and all $P_{\sim c}$ must happen at time strictly greater than $t_1 + c_{max}$, we can simply replace them with $\mathbf{true}$, $\mathbf{false}$ and $c_{max} + 1 \sim c$ to obtain equivalent $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas over $\mathbf{P}$. The formulas of the third type can be dealt with similarly. Finally, for the formulas of the first type, we replace $P_<$, $P_>$ and all $P_{\sim c}$ with $x < z$, $z < x$ and $d(x, z) \sim c$. The resulting formulas are clearly bounded $\mathsf{FO}[<, +\mathbb{Q}]$ formulas. We can scale them to bounded $\mathsf{FO}[<, +1]$ formulas, apply Theorem 4.9 and then scale back to obtain equivalent $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas over $\mathbf{P}$. □

The main result of this chapter now follows from a simple scaling argument.

Theorem 4.11. $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ *with rational constants is expressively complete for* $\mathsf{FO}[<, +\mathbb{Q}]$ *over infinite timed words.*

5. Monitoring of $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ properties

While the expressive completeness result in the last section may be interesting from a theoretical point of view, it is unclear how it can benefit practical verification tasks as the model-checking problem for $\mathsf{MTL}_{\mathsf{fut}}$ is already undecidable [OW06]. Nonetheless, we show that those results can be very useful in *monitoring*, a core element of *runtime verification*. We first define some basic notions used throughout this section. Then we give a bi-linear offline trace-checking algorithm for $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$, which is later modified to work in an online fashion (under a bounded-variability assumption) and used as the basis of a monitoring procedure for an 'easy-to-monitor' fragment of $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$.[13] The main advantage of the proposed procedure is that it is *trace-length independent*, i.e., the space usage is independent of the length of the (growing) trace. Finally, we show that our approach extends to arbitrary $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas via the syntactic rewriting rules in Section 4.1.

5.1. **Satisfaction over finite prefixes.**

Truncated semantics. As one is naturally concerned with truncated traces in monitoring, it is useful to define satisfaction relations of $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas over finite timed words. To this end, we adopt a timed version of the *truncated semantics* [EFHL03] which offers *strong* and *weak* views on satisfaction over finite timed words. Intuitively, these views indicate whether the satisfaction of the formula on the whole (infinite) trace is 'clearly' confirmed/refuted by the finite prefix read so far. In the strong view, one is *pessimistic* on satisfaction—for example, $\Box P$ can never be strongly satisfied by any finite timed word, as any such finite timed word can be extended into an infinite timed word that violates the formula. Conversely, in the weak view one is *optimistic* on satisfaction—for example, $\Diamond_{<5} P$ is weakly satisfied by any finite timed word whose timestamps are all strictly less than 5, since one can always extend such into an infinite timed word that satisfies the formula. We also consider the *neutral* view, which extends the traditional LTL semantics over finite words [MP95] to MTL. In what follows, we denote the strong, neutral and weak satisfaction relations by $\models_f^+$, $\models_f$ and $\models_f^-$ respectively. We write $\rho \models_f^+ \varphi$ ($\rho \models_f \varphi$, $\rho \models_f^- \varphi$) if $\rho, 0 \models_f^+ \varphi$ ($\rho, 0 \models_f \varphi$, $\rho, 0 \models_f^- \varphi$).

Definition 5.1. The satisfaction relation $\rho, i \models_f^+ \varphi$ for an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ, a finite timed word $\rho = (\sigma, \tau)$ and a position i, $0 \le i < |\rho|$ is defined as follows:

- $\rho, i \models_f^+ P$ iff $P \in \sigma_i$
- $\rho, i \models_f^+ \mathbf{true}$
- $\rho, i \models_f^+ \varphi_1 \wedge \varphi_2$ iff $\rho, i \models_f^+ \varphi_1$ and $\rho, i \models_f^+ \varphi_2$
- $\rho, i \models_f^+ \neg\varphi$ iff $\rho, i \not\models_f^- \varphi$
- $\rho, i \models_f^+ \varphi_1 \mathcal{U}_I \varphi_2$ iff there exists j, $i < j < |\rho|$ such that $\rho, j \models_f^+ \varphi_2$, $\tau_j - \tau_i \in I$, and $\rho, k \models_f^+ \varphi_1$ for all k with $i < k < j$
- $\rho, i \models_f^+ \varphi_1 \mathcal{S}_I \varphi_2$ iff there exists j, $0 \le j < i$ such that $\rho, j \models_f^+ \varphi_2$, $\tau_i - \tau_j \in I$, and $\rho, k \models_f^+ \varphi_1$ for all k with $j < k < i$
- $\rho, i \models_f^+ \varphi_1 \mathfrak{U}_I^c \varphi_2$ iff there exists j, $i < j < |\rho|$ such that $\rho, j \models_f^+ \varphi_2$, $\tau_j - \tau_i \in I$, and $\rho, k \models_f^+ \varphi_1$ for all k, $i < k < j$ such that $\tau_k - \tau_i > c$ and $\tau_j - \tau_k > a - c$ where $a = \inf(I)$

[13] In this section we assume that all timestamps are rational, can be finitely represented (e.g., with a built-in data type), and additions and subtractions on them can be done in constant time.

- $\rho, i \models_f^+ \varphi_1 \, \mathfrak{S}_I^c \, \varphi_2$ iff there exists j, $0 \leq j < i$ such that $\rho, j \models_f^+ \varphi_2$, $\tau_i - \tau_j \in I$, and $\rho, k \models_f^+ \varphi_1$ for all k, $j < k < i$ such that $\tau_i - \tau_k > c$ and $\tau_k - \tau_j > a - c$ where $a = \inf(I)$.

Definition 5.2. The satisfaction relation $\rho, i \models_f \varphi$ for an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ, a finite timed word $\rho = (\sigma, \tau)$ and a position i, $0 \leq i < |\rho|$ is defined as follows:

- $\rho, i \models_f P$ iff $P \in \sigma_i$
- $\rho, i \models_f$ **true**
- $\rho, i \models_f \varphi_1 \wedge \varphi_2$ iff $\rho, i \models_f \varphi_1$ and $\rho, i \models_f \varphi_2$
- $\rho, i \models_f \neg\varphi$ iff $\rho, i \not\models_f \varphi$
- $\rho, i \models_f \varphi_1 \, \mathcal{U}_I \, \varphi_2$ iff there exists j, $i < j < |\rho|$ such that $\rho, j \models_f \varphi_2$, $\tau_j - \tau_i \in I$, and $\rho, k \models_f \varphi_1$ for all k with $i < k < j$
- $\rho, i \models_f \varphi_1 \, \mathcal{S}_I \, \varphi_2$ iff there exists j, $0 \leq j < i$ such that $\rho, j \models_f \varphi_2$, $\tau_i - \tau_j \in I$, and $\rho, k \models_f \varphi_1$ for all k with $j < k < i$
- $\rho, i \models_f \varphi_1 \, \mathfrak{U}_I^c \, \varphi_2$ iff there exists j, $i < j < |\rho|$ such that $\rho, j \models_f \varphi_2$, $\tau_j - \tau_i \in I$, and $\rho, k \models_f \varphi_1$ for all k, $i < k < j$ such that $\tau_k - \tau_i > c$ and $\tau_j - \tau_k > a - c$ where $a = \inf(I)$
- $\rho, i \models_f \varphi_1 \, \mathfrak{S}_I^c \, \varphi_2$ iff there exists j, $0 \leq j < i$ such that $\rho, j \models_f \varphi_2$, $\tau_i - \tau_j \in I$, and $\rho, k \models_f \varphi_1$ for all k, $j < k < i$ such that $\tau_i - \tau_k > c$ and $\tau_k - \tau_j > a - c$ where $a = \inf(I)$.

Definition 5.3. The satisfaction relation $\rho, i \models_f^- \varphi$ for an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ, a finite timed word $\rho = (\sigma, \tau)$ and a position i, $0 \leq i < |\rho|$ is defined as follows:

- $\rho, i \models_f^- P$ iff $P \in \sigma_i$
- $\rho, i \models_f^-$ **true**
- $\rho, i \models_f^- \varphi_1 \wedge \varphi_2$ iff $\rho, i \models_f^- \varphi_1$ and $\rho, i \models_f^- \varphi_2$
- $\rho, i \models_f^- \neg\varphi$ iff $\rho, i \not\models_f^+ \varphi$
- $\rho, i \models_f^- \varphi_1 \, \mathcal{U}_I \, \varphi_2$ iff either of the following holds:
  - there exists j, $i < j < |\rho|$ such that $\rho, j \models_f^- \varphi_2$, $\tau_j - \tau_i \in I$, and $\rho, k \models_f^- \varphi_1$ for all k with $i < k < j$
  - $\tau_{|\rho|-1} - \tau_i < \sup(I)$ and $\rho, k \models_f^- \varphi_1$ for all k with $i < k < |\rho|$
- $\rho, i \models_f^- \varphi_1 \, \mathcal{S}_I \, \varphi_2$ iff there exists j, $0 \leq j < i$ such that $\rho, j \models_f^- \varphi_2$, $\tau_i - \tau_j \in I$, and $\rho, k \models_f^- \varphi_1$ for all k with $j < k < i$
- $\rho, i \models_f^- \varphi_1 \, \mathfrak{U}_I^c \, \varphi_2$ iff either of the following holds:
  - there exists j, $i < j < |\rho|$ such that $\rho, j \models_f^- \varphi_2, \tau_j - \tau_i \in I$, and $\rho, k \models_f^- \varphi_1$ for all k, $i < k < j$ such that $\tau_k - \tau_i > c$ and $\tau_j - \tau_k > a - c$ where $a = \inf(I)$
  - $\tau_{|\rho|-1} - \tau_i < \sup(I)$ and $\rho, k \models_f^- \varphi_1$ for all k, $i < k < |\rho|$ such that $\tau_k - \tau_i > c$ and $\tau_{|\rho|-1} - \tau_k \geq a - c$ where $a = \inf(I)$
- $\rho, i \models_f^- \varphi_1 \, \mathfrak{S}_I^c \, \varphi_2$ iff there exists j, $0 \leq j < i$ such that $\rho, j \models_f^- \varphi_2$, $\tau_i - \tau_j \in I$, and $\rho, k \models_f^- \varphi_1$ for all k, $j < k < i$ such that $\tau_i - \tau_k > c$ and $\tau_k - \tau_j > (a - c)$ where $a = \inf(I)$.

Proposition 5.4. *For a finite timed word ρ, a position i in ρ and an $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ,*

$$\rho, i \models_f^+ \varphi \implies \rho, i \models_f \varphi \textit{ and } \rho, i \models_f \varphi \implies \rho, i \models_f^- \varphi \, .$$

Informative prefixes. We say that ρ is *informative* for φ if either of the following holds:

- ρ strongly satisfies φ, i.e., $\rho \models_f^+ \varphi$. In this case we say that ρ is an *informative good prefix* for φ; or

- ρ fails to weakly satisfy φ, i.e., $\rho \not\models_f^- \varphi$. In this case we say that ρ is an *informative bad prefix* for φ.[14]

The following proposition follows immediately from the definition of informative prefixes. In words, negating (syntactically) a formula swaps its set of informative good prefixes and informative bad prefixes.

Proposition 5.5. *For an* $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ *formula, a finite timed word* ρ *is an informative good prefix for* φ *if and only if* ρ *is an informative bad prefix for* $\neg\varphi$*.*

Example 5.6. Consider the following formula over $\{P\}$:

$$\varphi = \Diamond\Box(\neg P) \land \Box(P \implies \Diamond_{<3} P)\,.$$

We say that the finite timed word $\rho = (\{P\},0)(\{P\},2)(\emptyset,5.5)$ is an informative bad prefix for φ as the second conjunct has been 'clearly' violated, i.e., there is a P-event with no P-event in the following three time units ($\rho \models_f^+ \neg\varphi$, or equivalently $\rho \not\models_f^- \varphi$). On the other hand, while $\rho' = (\{P\},0)(\{P\},2)(\{P\},4)$ is indeed a bad prefix for φ, it is not informative as both the first and second conjuncts are not yet 'clearly' violated ($\rho' \not\models_f^+ \neg\varphi$, or equivalently $\rho' \models_f^- \varphi$).

Example 5.7. Consider the following formula over $\{P\}$:

$$\varphi' = \Box(\neg P) \land \Box(P \implies \Diamond_{<3} P)\,.$$

This formula is equivalent to the formula φ in the previous example. However, all the bad prefixes ρ for φ' are informative ($\rho \models_f^+ \neg\varphi$, or equivalently $\rho \not\models_f^- \varphi$).

5.2. **Offline trace-checking algorithm.** *Trace checking* can be seen as a much more restricted case of model checking where one is only concerned with a single finite trace. Formally, the trace-checking problem for $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ asks the following: given a finite trace ρ and an $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ formula φ, is $\rho \models_f \varphi$? An offline algorithm for the problem is shown as Algorithms 1 and 2. For given ρ and φ, the algorithm maintains a two-dimensional Boolean array `table` of $|\psi|$ rows and $|\rho|$ columns. Each row is used to store the truth values of a subformula at all positions. The algorithm proceeds by filling up the array `table` in a bottom-up manner, starting from minimal subformulas. We only detail the cases for $\varphi_1 \,\mathcal{U}_I\, \varphi_2$ and $\varphi_1 \,\mathfrak{U}^c_I\, \varphi_2$ as other cases are either symmetric or trivial. In what follows, we write $x \leq I$ for $x < \sup(I)$ if I is right-open and for $x \leq \sup(I)$ otherwise. To ease the presentation we omit the array-bounds checks, e.g., the algorithm should stop when *ptr* (*ptr1*) reaches -1.

Proposition 5.8. *After executing* FillTable(`table`, $\varphi_1 \,\mathcal{U}_I\, \varphi_2$)*, we have*

$$\texttt{table}[\varphi_1 \,\mathcal{U}_I\, \varphi_2][i] \iff \rho, i \models_f \varphi_1 \,\mathcal{U}_I\, \varphi_2$$

for all $0 \leq i < |\rho|$ *if* `table`$[\varphi_1]$ *and* `table`$[\varphi_2]$ *were both correct.*

Proof. Suppose that `table`$[\varphi_1 \,\mathcal{U}_I\, \varphi_2][i] = \top$. Since each entry in `table`$[\varphi_1 \,\mathcal{U}_I\, \varphi_2]$ is filled exactly once, it must be filled at either line 8 or line 12. In the former case it is clear that $\rho, i \models_f \varphi_1 \,\mathcal{U}_I\, \varphi_2$. In the latter case we must have $ptr \leq j-2$. If $ptr = j-2$ then we are done, so we assume $ptr < j-2$. If there is a maximal position ptr', $ptr+1 < ptr' < j$ such that `table`$[\varphi_1][ptr'] = \bot$, we must have $ptr + 1 = ptr'$, which is a contradiction. We therefore conclude that $\rho, i \models_f \varphi_1 \,\mathcal{U}_I\, \varphi_2$.

[14] Note that informative good/bad prefixes are under-approximations of good/bad prefixes; see Section 6 for a discussion.

Algorithm 1 FillTable($\texttt{table}, \varphi_1 \, \mathcal{U}_I \, \varphi_2$)

```
$ptr \leftarrow |\rho| - 1$
for $j = |\rho| - 1$ to $0$ do
    if $ptr = j$ then
        $\texttt{table}[\varphi_1 \, \mathcal{U}_I \, \varphi_2][ptr] \leftarrow \bot$
        $ptr \leftarrow ptr - 1$
    if $\texttt{table}[\varphi_2][j]$ then
        if $ptr = j - 1$ then
            $\texttt{table}[\varphi_1 \, \mathcal{U}_I \, \varphi_2][ptr] \leftarrow (\tau_j - \tau_{ptr} \in I)$
            $ptr \leftarrow ptr - 1$
        while $\texttt{table}[\varphi_1][ptr + 1] \wedge \tau_j - \tau_{ptr} \leq I$ do
            if $\tau_j - \tau_{ptr} \in I$ then
                $\texttt{table}[\varphi_1 \, \mathcal{U}_I \, \varphi_2][ptr] \leftarrow \top$
            else
                $\texttt{table}[\varphi_1 \, \mathcal{U}_I \, \varphi_2][ptr] \leftarrow \bot$
            $ptr \leftarrow ptr - 1$
```

For the other direction, assume $\rho, i \models_f \varphi_1 \, \mathcal{U}_I \, \varphi_2$ and let $j' > i$ be the witness position, i.e., $\tau_{j'} - \tau_i \in I$, $\texttt{table}[\varphi_2][j'] = \top$ and $\texttt{table}[\varphi_1][j''] = \top$ for all j'', $i < j'' < j'$. Now consider $j = j'$. If $ptr \geq i$ then we are done. So we assume $ptr < i$. If we already have $\texttt{table}[\varphi_1 \, \mathcal{U}_I \, \varphi_2][i] = \bot$, then it must be the case that $\tau_{j'} - \tau_i \notin I$, which is a contradiction. Therefore we must have $\texttt{table}[\varphi_1 \, \mathcal{U}_I \, \varphi_2][i] = \top$. □

Algorithm 2 FillTable($\texttt{table}, \varphi_1 \, \mathfrak{U}^{c}_{I} \, \varphi_2$)

```
$ptr1, ptr2 \leftarrow |\rho| - 1$
for $j = |\rho| - 1$ to $0$ do
    while $\tau_j - \tau_{ptr2} \leq \inf(I) - c \vee \texttt{table}[\varphi_1][ptr2]$ do
        $ptr2 \leftarrow ptr2 - 1$
    if $ptr1 = j$ then
        $\texttt{table}[\varphi_1 \, \mathfrak{U}^{c}_{I} \, \varphi_2][ptr1] \leftarrow \bot$
        $ptr1 \leftarrow ptr1 - 1$
    if $\texttt{table}[\varphi_2][j]$ then
        if $ptr1 = j - 1$ then
            $\texttt{table}[\varphi_1 \, \mathfrak{U}^{c}_{I} \, \varphi_2][ptr1] \leftarrow (\tau_j - \tau_{ptr1} \in I)$
            $ptr1 \leftarrow ptr1 - 1$
        while $\tau_j - \tau_{ptr1} \leq I \wedge \tau_{ptr2} - \tau_{ptr1} \leq c$ do
            if $\tau_j - \tau_{ptr1} \in I$ then
                $\texttt{table}[\varphi_1 \, \mathfrak{U}^{c}_{I} \, \varphi_2][ptr1] \leftarrow \top$
            else
                $\texttt{table}[\varphi_1 \, \mathfrak{U}^{c}_{I} \, \varphi_2][ptr1] \leftarrow \bot$
            $ptr1 \leftarrow ptr1 - 1$
```

Proposition 5.9. *After executing* FillTable($\texttt{table}, \varphi_1 \,\mathfrak{U}^c_I\, \varphi_2$)*, we have*

$$\texttt{table}[\varphi_1 \,\mathfrak{U}^c_I\, \varphi_2][i] \iff \rho, i \models_f \varphi_1 \,\mathfrak{U}^c_I\, \varphi_2$$

for all $0 \leq i < |\rho|$ *if* $\texttt{table}[\varphi_1]$ *and* $\texttt{table}[\varphi_2]$ *were both correct.*

Proof. Observe that after line 5, *ptr2* is equal to the maximal position such that $\tau_j - \tau_{ptr2} > \inf(I) - c$ and $\texttt{table}[\varphi_1][ptr2] = \bot$. The proof is very similar to the case of $\varphi_1 \,\mathcal{U}_I\, \varphi_2$. □

5.3. **Monitoring procedure.** Conceptually, we can regard a monitor as a *deterministic* automaton over finite traces. The monitoring process, then, can be carried out by simply moving a token as directed by the prefix. However, it is well-known that in a dense real-time setting, such a monitor (say, which accepts all the bad prefixes for φ) needs an unbounded number of clocks and therefore cannot be realised in practice [AH92, MNP05, Rey14]. For this reason, we shall from now on assume that all input traces have variability at most k_{var}, i.e., there are at most k_{var} events in any (open) unit time interval. Based on this assumption, we give a monitoring procedure for $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas of the form

$$\hat{\varphi} = \Phi(\psi_1, \dots, \psi_m)$$

where ψ_1, …, ψ_m are *bounded* $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas and Φ is an LTL formula. The main idea is similar to the one used in the previous section: we introduce new propositions $\mathbf{Q} = \{Q_1, \dots, Q_m\}$ that correspond to ψ_1, …, ψ_m. In this way, we can monitor Φ as an LTL property over $\mathbf{Q}$.[15] Since these propositions correspond to bounded formulas, their truth values can be obtained by running the trace-checking algorithm on subtraces: as the input trace has variability at most k_{var}, we only need to store a 'sliding window' of a certain size.

The untimed LTL *part.* We recall briefly the standard methodology to construct finite automata that accept exactly the informative good/bad prefixes for a given $\mathsf{LTL}_{\mathsf{fut}}$ formula [KV01]. Given such a formula Ψ, first use a standard construction [Var96] to obtain a language-equivalent alternating Büchi automaton $\mathcal{A}_\Psi$. Then redefine its accepting set to be the empty set and treat it as an automaton over finite words; the resulting automaton $\mathcal{A}^{true}_\Psi$ accepts exactly all informative good prefixes for Ψ. In particular, one can determinise $\mathcal{A}^{true}_\Psi$ with the usual subset construction. The same can be done for $\neg\Psi$ to obtain a deterministic automaton that accepts exactly the informative bad prefixes for Ψ.

In our case, we first translate the LTL formulas Φ and $\neg\Phi$ into a pair of *two-way* alternating Büchi automata [GO03]. With the same modifications, we obtain two automata that accept informative good prefixes and informative bad prefixes for Φ. We then apply existing procedures that translate two-way alternating automata over finite words into deterministic automata (e.g., [Bir93]) and obtain $\mathcal{D}_{good}$ and $\mathcal{D}_{bad}$, respectively. To detect both types of prefixes simultaneously, we will execute $\mathcal{D}_{good}$ and $\mathcal{D}_{bad}$ in parallel.

Proposition 5.10. *For an* $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ *formula* $\hat{\varphi}$ *of the form described above, the automata* $\mathcal{D}_{good}$ *and* $\mathcal{D}_{bad}$ *are of size* $2^{2^{O(|\Phi|)}}$ *where* Φ *is the 'backbone'* LTL *formula.*

[15] A similar idea is used in [FK09] to synthesise smaller monitor circuits for $\mathsf{LTL}_{\mathsf{fut}}$ formulas.

Naïve evaluation of the bounded metric parts. In what follows, let $l_{fr}(\psi) = k_{var} \cdot \lceil fr(\psi) \rceil$ and $l_{pr}(\psi) = k_{var} \cdot \lceil pr(\psi) \rceil$ (the functions *fr* and *pr* are defined in Section 4.3). Suppose that we want to obtain the truth value of ψ_i at position j in the input trace $\rho = (\sigma, \tau)$. Since ψ_i is bounded, only the events occurring between $\tau_j - pr(\psi_i)$ and $\tau_j + fr(\psi_i)$ can affect the truth value of ψ_i at j. This implies that $\rho, j \models \psi_i \iff \rho', j \models_f \psi_i$ where ρ' is a prefix of ρ that contains all the events between $\tau_j - pr(\psi_i)$ and $\tau_j + fr(\psi_i)$ in ρ. Since ρ is of bounded variability k_{var}, there can be at most $l_{pr}(\psi_i) + 1 + l_{fr}(\psi_i)$ events between $\tau_j - pr(\psi_i)$ and $\tau_j + fr(\psi_i)$. It follows that we can simply 'record' all events in this interval with a two-dimensional array of $l_{pr}(\psi_i) + 1 + l_{fr}(\psi_i)$ columns and $1 + |\psi_i|$ rows: the first row is used to store the timestamps whereas the other rows are used to store the truth values. Intuitively, the two-dimensional array acts as a sliding window around position j in ρ. Now consider all the propositions in $\mathbf{Q}$: their truth values at position j can be evaluated using a two-dimensional array of $l_{pr}^{\mathbf{Q}} + 1 + l_{fr}^{\mathbf{Q}}$ columns and $1 + |\psi_1| + \cdots + |\psi_m|$ rows where $l_{pr}^{\mathbf{Q}} = \max_{1 \le i \le m} l_{pr}(\psi_i)$ and $l_{fr}^{\mathbf{Q}} = \max_{1 \le i \le m} l_{fr}(\psi_i)$. Each row can be filled in time $O(l_{pr}^{\mathbf{Q}} + 1 + l_{fr}^{\mathbf{Q}})$ with the trace-checking algorithm. Overall, we need a two-dimensional array of size $O(k_{var} \cdot c_{sum} \cdot |\hat{\varphi}|)$ where c_{sum} is the sum of the constants in $\hat{\varphi}$; for each position j, we need time $O(k_{var} \cdot c_{sum} \cdot |\hat{\varphi}|)$ to obtain the truth values of all propositions in $\mathbf{Q}$, which are then used as input to $\mathcal{D}_{good}$ and $\mathcal{D}_{bad}$.

Incremental evaluation of the bounded metric parts. While the procedure above uses only bounded space, it is clearly inefficient as for each j we have to fill the whole two-dimensional array from scratch. This is because some of the filled entries (other than those for position j) may depend on the events outside of the sliding window, and thus can be incorrect. We now describe an optimisation which enables the reuse of previously filled entries.

We first deal with the simpler case of past subformulas. Observe that as the trace-checking algorithm is filling a row for $\varphi_1 \, \mathcal{S}_I \, \varphi_2$ or $\varphi_1 \, \mathfrak{S}_I^c \, \varphi_2$, the variables *ptr*, *ptr1* and *ptr2* all increases *monotonically.* This implies that for past subformulas, the trace-checking algorithm can be used in an online manner: simply suspend the algorithm when we have filled all entries using the truth values of φ_1 and φ_2 (at various positions) that are currently known, and resume the algorithm when the truth values of φ_1 and φ_2 (at some other positions) that are previously unknown become available.

The case of future subformulas is more involved. Suppose that we want to evaluate the truth value of a subformula $P_1 \, \mathcal{U}_{(a,b)} \, P_2$ at position j in the input trace $\rho = (\sigma, \tau)$. It is clear that the value may depend on future events if $\tau_j + b$ is greater than the timestamp of the last acquired event. However, observe that even when this is the case, we may still do the evaluation if any of the following holds:

- P_1 fails to hold at some position j' such that $\tau_{j'}$ is less or equal than the timestamp of the last acquired event. In this case, we know that all the truth values of $P_1 \, \mathcal{U}_{(a,b)} \, P_2$ at positions $< j'$ cannot depend on the events at positions $> j'$.
- P_2 holds at some position $j' > j$ and P_1 holds at all positions j'', $j < j'' < j'$. In this case, the truth values of $P_1 \, \mathcal{U}_{(a,b)} \, P_2$ at positions $k < j'$ where $\tau_{j'} - \tau_k \in (a, b)$ are $\top$ and do not depend on the events at positions $> j'$.

We generalise this observation to handle the general case of updating the row for $\varphi_1 \, \mathcal{U}_{(a,b)} \, \varphi_2$. First of all, we maintain indices j_{φ_1}, j_{φ_2}, $j_{\varphi_1 \mathcal{U}_I \varphi_2}$ which point to the first unknown entries in the rows for φ_1, φ_2 and $\varphi_1 \, \mathcal{U}_I \, \varphi_2$. Let $t_{max} = \min\{\tau_{(j_{\varphi_1} - 1)}, \tau_{(j_{\varphi_2} - 1)}\}$ and update its value when either j_{φ_1} or j_{φ_2} changes. Whenever t_{max} is updated to a new value, we also update the following indices:

- j_1 is the maximal position such that $\tau_{j_1} + b \le t_{max}$
- j_2 is the maximal position such that $\tau_{j_2} \le t_{max}$ and φ_2 holds at j_2
- j_3 is the maximal position such that $\tau_{j_3} + a < \tau_{j_2}$
- j_4 is the maximal position such that $\tau_{j_4} \le t_{max}$ and φ_1 does not hold at j_4.

Now, after both the rows for φ_1 and φ_2 have been updated, if any of j_1, j_3, $j_4 - 1$ is greater or equal than $j_{\varphi_1 \mathcal{U}_I \varphi_2}$, we let $j_5 = \max\{j_1, j_3, j_4 - 1\}$ and start Algorithm 1 from line 3 with $ptr = j_5$ and $j = j_2$. We run the algorithm till all the entries at positions $\le j_5$ in the row for $\varphi_1 \,\mathcal{U}_I\, \varphi_2$ have been filled. The crucial observation here is that j_1, j_2, j_3, j_4 all increase monotonically, and therefore can be maintained in amortised linear time. Also, the truth value of any subformula at any position will be filled only once. The case of $\varphi_1 \,\mathfrak{U}^c_{(a,b)}\, \varphi_2$ is similar (but slightly more involved). These observations imply that each entry in the two-dimensional array can be filled in amortised constant time. Assuming that moving a token on a deterministic finite automaton takes constant time, we can state the following theorem.

Theorem 5.11. *For an* $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ *formula* $\hat{\varphi}$ *of the form described earlier and an infinite trace of variability* k_{var}*, our monitoring procedure uses two DFAs of size* $2^{2^{O(|\Phi|)}}$*, a two-dimensional array of size* $O(k_{var} \cdot c_{sum} \cdot |\hat{\varphi}|)$ *where* c_{sum} *is the sum of the constants in* $\hat{\varphi}$*, and amortised time* $O(|\hat{\varphi}|)$ *per event.*

Correctness. We now show that our procedure is sound and complete for detecting informative prefixes.

Proposition 5.12. *For a bounded* $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ *formula* ψ*, a finite trace* $\rho = (\sigma, \tau)$ *and a position* $0 \le i < |\rho|$ *such that* $\tau_i + fr(\psi) \le \tau_{|\rho|-1}$*, we have*

$$\rho, i \models^+_f \psi \iff \rho, i \models_f \psi \iff \rho, i \models^-_f \psi\,.$$

Proposition 5.13. *For an* $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ *formula* φ*, a finite trace* ρ *and a position* i *in* ρ*, if* ρ *is a prefix of a longer finite trace* ρ'*, then*

$$\rho, i \models^+_f \varphi \implies \rho', i \models^+_f \varphi \text{ and } \rho, i \not\models^-_f \varphi \implies \rho', i \not\models^-_f \varphi\,.$$

Theorem 5.14 (Soundness)**.** *In our procedure, if we ever reach an accepting state of* $\mathcal{D}_{good}$ *(*$\mathcal{D}_{bad}$*) via a finite word* $u \in \Sigma_Q^*$*, then we must eventually read an informative good (bad) prefix for* $\hat{\varphi}$*.*

Proof. For such u and a corresponding $\rho = (\sigma, \tau)$ such that $\tau_{|u|-1} + l^Q_{fr} \le \tau_{|\rho|-1}$, we have

$$\forall i \in [0, |u|)\, \big((u, i \models^+_f \Psi \implies \rho, i \models^+_f \psi) \wedge (u, i \not\models^-_f \Psi \implies \rho, i \not\models^-_f \psi)\big)$$

where Ψ is a subformula of Φ and $\psi = \Psi(\psi_1, \ldots, \psi_m)$. This can easily be proved by structural induction and Proposition 5.12. If u is accepted by $\mathcal{D}_{good}$, we have $u, 0 \models^+_f \Phi$ by construction. By the above we have $\rho, 0 \models^+_f \Phi(\psi_1, \ldots, \psi_m)$, as desired. The case of $\mathcal{D}_{bad}$ is symmetric. □

Theorem 5.15 (Completeness)**.** *Whenever we read an informative good (bad) prefix* $\rho = (\sigma, \tau)$ *for* $\hat{\varphi}$*,* $\mathcal{D}_{good}$ *(*$\mathcal{D}_{bad}$*) must eventually reach an accepting state.*

Proof. For the finite word $u' \in \Sigma_Q^*$ obtained a bit later with $|u'| = |\rho|$,

$$\forall i \in [0, |u'|) \big((\rho, i \models_f^+ \psi \implies u', i \models_f^+ \Psi) \land (\rho, i \not\models_f^- \psi \implies u', i \not\models_f^- \Psi)\big)$$

where Ψ is a subformula of Φ and $\psi = \Psi(\psi_1, \ldots, \psi_m)$. This can be proved by structural induction and Proposition 5.13. The theorem follows. □

5.4. **Preservation of informative prefixes.** As we have seen earlier in Example 5.6 and 5.7, it is possible for two equivalent $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas to have different sets of informative good/bad prefixes. In this section, we show that this is cannot be the case when the two formulas are related by one of the rewriting rules in Section 4.1. In other words, the rewriting rules in Section 4.1 preserves the 'informativeness' of formulas.

Lemma 5.16. *For an* $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ *formula* φ*, let* φ' *be the formula obtained from* φ *by applying one of the rules in Section 4.1 on some of its subformula. We have*

$$\rho \models_f^+ \varphi \iff \rho \models_f^+ \varphi' \text{ and } \rho \models_f^- \varphi \iff \rho \models_f^- \varphi' .$$

Given the lemma above, we can state the following theorem on any MTL formula φ and the equivalent formula $\hat{\varphi}$ (of our desired form) obtained from φ by applying the rewriting rules in Section 4.1.

Theorem 5.17. *The set of informative good prefixes of* φ *coincides with the set of informative good prefixes of* $\hat{\varphi}$*. The same holds for informative bad prefixes.*

We now have a way to detect the informative good/bad prefixes for an arbitrary $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formula φ: use the rewriting rules to obtain $\hat{\varphi}$, and apply the monitoring procedure we described in the last subsection. The monitor only needs a bounded amount of memory, even for complicated $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$ formulas with arbitrary nestings of (bounded and unbounded) past and future operators.

Proof of Lemma 5.16. Since the satisfaction relations are defined inductively, we can work directly on the relevant subformulas. We would like to prove that for a finite timed word ρ and a position i in ρ,

$$\rho, i \models_f^+ \phi \iff \rho, i \models_f^+ \phi' \text{ and } \rho, i \models_f^- \phi \iff \rho, i \models_f^- \phi'$$

where $\phi \iff \phi'$ matches one of the rules in Section 4.1. For a group of similar rules we will only prove a representative one, as the proof for others follow similarly. In the following let the LHS be ϕ and RHS be ϕ'.

- $\varphi_1 \mathcal{U}_{(a,\infty)} \varphi_2 \iff \varphi_1 \mathcal{U} \varphi_2 \land \Box_{(0,a]}(\varphi_1 \land \varphi_1 \mathcal{U} \varphi_2)$:
  - $\rho, i \models_f^+ \phi \iff \rho, i \models_f^+ \phi'$:
    Assume $\rho, i \models_f^+ \phi$. By definition we have $\rho, i \models_f^+ \varphi_1 \mathcal{U} \varphi_2$. If there is no event in $(\tau_i, \tau_i + a]$, since there must be an event in $(\tau_i + a, \tau_{|\rho|-1}]$, we are done. If there are events in $(\tau_i, \tau_i + a]$, then for all j such that $\tau_j - \tau_i \in (0, a]$ we have $\rho, j \not\models_f^- \neg\varphi_1$. Also for all such j we have $\rho, j \not\models_f^- \neg\varphi_1 \mathcal{U} \varphi_2$ since it is obvious that $\rho, j \models_f^+ \varphi_1 \mathcal{U} \varphi_2$. For the other direction, if the witness (for $\rho, i \models_f^+ \varphi_1 \mathcal{U} \varphi_2$) is in $(\tau_i + a, \tau_{|\rho|-1})$ then we are done. If this is not the case, since $\rho, i \not\models_f^- \Diamond_{(0,a]} \big(\neg\varphi_1 \lor \neg(\varphi_1 \mathcal{U} \varphi_2)\big)$, we must have $\tau_{|\rho|-1} \geq a$. Now for all j such that $\tau_j - \tau_i \in (0, a]$ we have $\rho, j \models_f^+ \varphi_1$ and $\rho, j \models_f^+ \varphi_1 \mathcal{U} \varphi_2$, which imply $\rho, i \models_f^+ \phi$.

- $\rho, i \models^{-}_{f} \phi \iff \rho, i \models^{-}_{f} \phi'$:
  Assume $\rho, i \models^{-}_{f} \phi$. This holds if there is a witness in (a, ∞) or $\rho, i \models^{-}_{f} \Box \varphi_1$. In both cases we have $\rho, i \models^{-}_{f} \varphi_1 \mathcal{U} \varphi_2$. If there is no event in $(\tau_i, \tau_i + a]$ then we are done. If there is a witness, then for all such j that $\tau_j - \tau_i \in (0, a]$ we have $\rho, j \models^{-}_{f} \varphi_1$ and $\rho, j \models^{-}_{f} \varphi_1 \mathcal{U} \varphi_2$. If there is no witness then for all such j we again have $\rho, j \models^{-}_{f} \varphi_1$ and $\rho, j \models^{-}_{f} \varphi_1 \mathcal{U} \varphi_2$. For the other direction, if there is no event in $(\tau_i, \tau_i + a]$ we are done. If there are events in $(\tau_i, \tau_i + a]$, all j such that $\tau_j - \tau_i \in (0, a]$ will satisfy $\rho, j \models^{-}_{f} \varphi_1$ and $\rho, j \models^{-}_{f} \varphi_1 \mathcal{U} \varphi_2$. This clearly gives $\rho, i \models^{-}_{f} \phi$.

- $\varphi_1 \mathfrak{U}^{c}_{(a,\infty)} \varphi_2 \iff \varphi_1 \mathfrak{U}^{c}_{(a,2a]} \varphi_2 \vee \Big(\Diamond^{w}_{[0,c]} \big(\varphi_1 \mathcal{U}_{(a,\infty)} (\varphi_2 \vee \Diamond_{\leq a-c} \varphi_2) \big) \Big)$:
  The proof is very similar to the proof of Proposition 4.1.

- $\neg(\varphi_1 \mathcal{U} \varphi_2) \iff \Box \neg\varphi_2 \vee \big(\neg\varphi_2 \mathcal{U} (\neg\varphi_2 \wedge \neg\varphi_1)\big)$:
  - $\rho, i \models^{+}_{f} \phi \iff \rho, i \models^{+}_{f} \phi'$:
    Assume $\rho, i \models^{+}_{f} \phi \iff \rho, i \not\models^{-}_{f} \varphi_1 \mathcal{U} \varphi_2$. This implies that φ_1 fails to hold before φ_2 holds, and we have $\rho, i \models^{+}_{f} \neg\varphi_2 \mathcal{U} (\neg\varphi_2 \wedge \neg\varphi_1)$. For the other direction note that $\rho, i \not\models^{+}_{f} \Box \neg\varphi_2$, the second disjunct must be satisfied, and it is easy to see that $\rho, i \models^{+}_{f} \phi$.

  - $\rho, i \models^{-}_{f} \phi \iff \rho, i \models^{-}_{f} \phi'$:
    Assume $\rho, i \models^{-}_{f} \neg(\varphi_1 \mathcal{U} \varphi_2) \iff \rho, i \not\models^{+}_{f} \varphi_1 \mathcal{U} \varphi_2$. This implies either $\rho, j \not\models^{+}_{f} \varphi_2 \iff \rho, j \models^{-}_{f} \neg\varphi_2$ for all $j > i$ in ρ (this gives $\rho, i \models^{-}_{f} \Box \neg\varphi_2$) or φ_1 fails to hold before φ_2 holds—$\rho, i \models^{-}_{f} \neg\varphi_2 \mathcal{U} (\neg\varphi_2 \wedge \neg\varphi_1)$. For the other direction, if $\rho, i \models^{-}_{f} \Box \neg\varphi_2 \iff \rho, i \not\models^{+}_{f} \Diamond \varphi_2$ then $\rho, i \models^{+}_{f} \varphi_1 \mathcal{U} \varphi_2$ cannot hold. If $\rho, i \models^{-}_{f} \neg\varphi_2 \mathcal{U} (\neg\varphi_2 \wedge \neg\varphi_1)$ then either $\rho, i \models^{-}_{f} \Box \neg\varphi_2$ or there is a witness, and it is easy to see that $\rho, i \models^{+}_{f} \varphi_1 \mathcal{U} \varphi_2$ cannot hold.

- $\theta \mathcal{U}_{(a,b)} \big((\varphi_1 \mathcal{U} \varphi_2) \wedge \chi \big) \iff \theta \mathcal{U}_{(a,b)} \big((\varphi_1 \mathcal{U}_{(0,2b)} \varphi_2) \wedge \chi \big)$
  $\vee \Big(\big(\theta \mathcal{U}_{(a,b)} (\Box_{(0,2b)} \varphi_1 \wedge \chi)\big) \wedge \varphi_{ugb} \Big)$:
  The proof is very similar to the proof of Proposition 4.2.

- $\big((\varphi_1 \mathcal{U} \varphi_2) \vee \chi \big) \mathcal{U}_{(a,b)} \theta \iff \big((\varphi_1 \mathcal{U}_{(0,2b)} \varphi_2) \vee \chi \big) \mathcal{U}_{(a,b)} \theta$
  $\vee \left(\Big(\big((\varphi_1 \mathcal{U}_{(0,2b)} \varphi_2) \vee \chi \big) \mathcal{U}_{(0,b)} (\Box_{(0,2b)} \varphi_1) \Big) \wedge \Diamond_{(a,b)} \theta \wedge \varphi_{ugb} \right)$
  - $\rho, i \models^{+}_{f} \phi \iff \rho, i \models^{+}_{f} \phi'$:
    Assume $\rho, i \models^{+}_{f} \phi$. It is obvious that $\rho, i \models^{+}_{f} \Diamond_{(a,b)} \theta$ holds. If the first disjunct of ϕ' does not hold, then $\rho, i \models^{+}_{f} \big((\varphi_1 \mathcal{U}_{(0,2b)} \varphi_2) \vee \chi \big) \mathcal{U}_{(0,b)} (\Box_{(0,2b)} \varphi_1)$ must hold. The last conjunct holds by an argument similar to the proof of Proposition 4.2. For the other direction, if the first disjunct of ϕ' holds then we are done. If it does not hold, then there must be a witness (at which φ_2 holds) in $[\tau_i + 2b, \tau_{|\rho|-1}]$, and it is easy to see that $\rho, i \models^{+}_{f} \phi$.

  - $\rho, i \models^{-}_{f} \phi \iff \rho, i \models^{-}_{f} \phi'$:
    Assume $\rho, i \models^{-}_{f} \phi$. If the first disjunct of ϕ' does not hold then there must be events in $[\tau_i + 2b, \tau_{|\rho|-1}]$. It follows that $\rho, i \models^{-}_{f} \big((\varphi_1 \mathcal{U}_{(0,2b)} \varphi_2) \vee \chi \big) \mathcal{U}_{(0,b)} (\Box_{(0,2b)} \varphi_1)$ and $\rho, i \models^{-}_{f} \Diamond_{(a,b)} \theta$ must hold. The rest is similar to the proof to Proposition 4.2. For the

other direction, if the first disjunct of ϕ' holds then we are done. Otherwise if $\tau_{|\rho|-1} < b$, it is easy to see that $\rho, i \models^{-}_{f} \phi$. If this is not the case then the proof again follows Proposition 4.2. □

6. Conclusion and future work

Expressive completeness over bounded timed words. We showed that MTL extended with our new modalities '*generalised Until*' and '*generalised Since*' (MTL[$\mathfrak{U}, \mathfrak{S}$]) is expressively complete for FO[$<, +1$] over bounded timed words. Moreover, the time-bounded satisfiability and model-checking problems for MTL[$\mathfrak{U}, \mathfrak{S}$] remain EXPSPACE-complete, same as that of MTL. The situation here is similar to LTL over general, possibly non-Dedekind complete, linear orders (e.g., the rationals): in this case, LTL can be made expressively complete (for FO[$<$]) by adding the Stavi modalities [GHR94], yet the complexity of the satisfiability problem remains PSPACE-complete [Rab10]. Along the way, we also obtained a strict hierarchy of metric temporal logics based on their expressiveness over bounded timed words.

One drawback of the modalities $\mathfrak{U}^{c}_{I}$ and $\mathfrak{S}^{c}_{I}$ is that they are not very intuitive. However, as we proved that simpler versions of these modalities ($\mathcal{B}^{\rightarrow}_{I}$ and $\mathcal{B}^{\leftarrow}_{I}$) are strictly less expressive, we believe it is unlikely that any other expressively complete extension of MTL could be much simpler than ours.

The satisfiability and model-checking procedures for MTL over time-bounded signals in [ORW09] are based on the satisfiability procedure for LTL over signals in [Rey10]. While the satisfiability problem for LTL remains PSPACE-complete when interpreted over signals, very few implementations are currently available [FMDR13]. This is in sharp contrast with the discrete case where a number of mature, industrial-strength tools (e.g., SPIN [Hol97]) are readily available. Our results enable the direct application of these tools to time-bounded verification. Whether this yields efficiency gains in practice, however, can only be evaluated empirically, which we leave as future work.

Expressive completeness over unbounded timed words. Building upon a previous work of Hunter, Ouaknine and Worrell [HOW13], we showed that the *rational* version of MTL[$\mathfrak{U}, \mathfrak{S}$] is expressively complete for FO[$<, +\mathbb{Q}$] over infinite timed words. The result answers an implicit open question in a long line of research started in [AH90] and further developed in [BCM05, PD06, DP06, PS11].

It is known that the *integer* version of MTL extended with counting modalities (and their past counterparts) is expressively complete for FO[$<, +1$] over the reals [Hun13].[16] We conjecture that the analogous result holds in the pointwise semantics, i.e., the integer version of MTL[$\mathfrak{U}, \mathfrak{S}$] becomes expressively complete for FO[$<, +1$] when we add counting modalities. Adapting the proof in [Hun13] to the pointwise case, however, is not a straight-forward task. In particular, the proof relies on the expressive completeness of MITL with counting modalities for Q2MLO [HR04], a result that itself requires a highly non-trivial proof [HR06] and seems to hold only in the continuous semantics.

Besides expressiveness, another major concern in the study of metric logics is *decidability*. We intend to investigate whether the expressiveness of MITL$_{\mathsf{fut}}$ or MITL can be enhanced

[16] This result is stronger than [HOW13] as counting modalities (and their past counterparts) can be expressed in MTL with rational endpoints.

with the new modalities while retaining decidability. Specifically, we would like to answer the following question: what is the complexity of the satisfiability problem for the logic obtained by adding $\mathcal{B}_I^{\rightarrow}$ (with non-singular I) into $\mathsf{MITL}_{\mathsf{fut}}$? Since $\mathcal{B}_I^{\rightarrow}$ can be expressed in one-clock alternating timed automata, it can possibly be handled in the framework of [BEG14]. More generally, we may consider MITL extended with $\mathcal{B}_I^{\rightarrow}$ and $\mathcal{B}_I^{\leftarrow}$ (with non-singular I); it is not clear whether allowing these modalities simultaneously leads to undecidability.

Monitoring. We identified an 'easy-to-monitor' fragment of $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$, for which we proposed an efficient trace-length independent monitoring procedure. This fragment is much more expressive than the fragments previously considered in the literature. Moreover, we showed that informative good/bad prefixes are preserved by the syntactic rewriting rules in Section 4.1. It follows that the informative good/bad prefixes for an arbitrary $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ formula can be monitored in a trace-length independent fashion, thus overcoming a long-standing barrier to the runtime verification of real-time properties.

For an arbitrary $\mathsf{MTL}[\mathfrak{U},\mathfrak{S}]$ formula, the syntactic rewriting process could potentially induce a non-elementary blow-up. In practice, however, the resulting formula is often of comparable size to the original one, which itself is typically small. For example, consider the following formula:

$$\Box\big(\texttt{ChangeGear} \implies \Diamond_{(0,30)}(\texttt{InjectFuel} \wedge \diamondminus\, \texttt{InjectAir})\big)\,.$$

The resulting formula after rewriting is

$$\begin{aligned}\Box\big(\texttt{ChangeGear} \implies\ & \Diamond_{(0,30)}(\texttt{InjectFuel} \wedge \diamondminus_{(0,30)}\, \texttt{InjectAir}) \\ & \vee (\Diamond_{(0,30)}\, \texttt{InjectFuel} \wedge \diamondminus\, \texttt{InjectAir})\big)\,.\end{aligned}$$

In fact, it can be argued that most common real-time specification patterns [KC05] belong syntactically to our 'easy-to-monitor' fragment and thus need no rewriting. Another way to alleviate the issue is to allow more liberal syntax (or more derived operators). For example, the procedure described in Section 5.3 can handle subformulas with unbounded past without modification.

To detect informative bad prefixes, our monitoring procedure uses a deterministic finite automaton doubly-exponential in the size of the input formula. While such a blow-up is rarely a problem in practice (see [BLS11, Section 2.5]), it would be better if it could be avoided altogether. In the untimed setting, it is known that if a safety property can be written as an LTL formula, then it is equivalent to a formula of the form $\Box\psi$ where ψ is a past-only LTL formula [LPZ85]. So, if we restrict our attention to safety properties, it suffices to consider formulas of this form, for which there is an efficient monitoring procedure that uses $O(|\psi|)$ time (per event) and $O(|\psi|)$ space [HR01]. Unfortunately, the question of whether a corresponding result holds for MTL (or similar metric temporal logics) is still open.

Our procedure detects only *informative* good/bad prefixes, which themselves can be regarded as easily-checkable certificates for the fulfilment/violation of the property. While we believe this limitation is in no way severe—in fact, the limitation is implicit in almost all current approaches to monitoring real-time properties—there are certain practical scenarios where detecting *all* good/bad prefixes is preferred. We could have used two deterministic finite automata that detect all good/bad prefixes for the backbone LTL formula, but still they cannot detect all good/bad prefixes for the whole formula (consider Example 5.6). We leave as future work a procedure that detects all good/bad prefixes.

Finally, we remark that the offline trace-checking problem is of independent theoretical interest [MS03]. It is known that the trace-checking problem for LTL [KF09] and MTL [BO14] are both in $\mathrm{AC}^1[\log \mathrm{DCFL}]$, yet their precise complexity is still open. It would be interesting to see whether the construction for MTL in [BO14] carries over to $\mathsf{MTL}[\mathfrak{U}, \mathfrak{S}]$.

References

[ABLS05] Oliver Arafat, Andreas Bauer, Martin Leucker, and Christian Schallhart. Runtime verification revisited. Technical Report TUM-I0518, Technische Universität München, 2005.

[AD94] Rajeev Alur and David Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–235, 1994.

[AFH96] Rajeev Alur, Tomás Feder, and Thomas A. Henzinger. The benefits of relaxing punctuality. *Journal of the ACM*, 43(1):116–146, 1996.

[AH90] Rajeev Alur and Thomas A. Henzinger. Real-time logics: Complexity and expressiveness. In *Proceedings of LICS 1990*, pages 390–401. IEEE Computer Society Press, 1990.

[AH92] Rajeev Alur and Thomas A. Henzinger. Back to the future: towards a theory of timed regular languages. In *Proceedings of FOCS 1992*, pages 177–186. IEEE Computer Society Press, 1992.

[AM04] Rajeev Alur and Parthasarathy Madhusudan. Decision problems for timed automata: A survey. In *Formal Methods for the Design of Real-Time Systems*, volume 3185 of *LNCS*, pages 1–24. Springer, 2004.

[AS87] Bowen Alpern and Fred B. Schneider. Recognizing safety and liveness. *Distributed Computing*, 2(3):117–126, 1987.

[BBBB09] Christel Baier, Nathalie Bertrand, Patricia Bouyer, and Thomas Brihaye. When are timed automata determinizable? In *Proceedings of ICALP 2009*, volume 5556 of *LNCS*, pages 43–54. Springer, 2009.

[BCE$^+$14] David A. Basin, Germano Caronni, Sarah Ereth, Matús Harvan, Felix Klaedtke, and Heiko Mantel. Scalable offline monitoring. In *Proceedings of RV 2014*, volume 8734 of *LNCS*, pages 31–47. Springer, 2014.

[BCM05] Patricia Bouyer, Fabrice Chevalier, and Nicolas Markey. On the expressiveness of TPTL and MTL. In *Proceedings of FSTTCS 2005*, volume 3821 of *LNCS*, pages 432–443. Springer, 2005.

[BEG14] Thomas Brihaye, Morgane Estiévenart, and Gilles Geeraerts. On MITL and alternating timed automata over infinite words. In *Proceedings of FORMATS 2014*, volume 8711 of *LNCS*, pages 69–84. Springer, 2014.

[BGHM17] Thomas Brihaye, Gilles Geeraerts, Hsi-Ming Ho, and Benjamin Monmege. MightyL: A compositional translation from MITL to timed automata. In *Proceedings of CAV 2017*, volume 10426 of *LNCS*, pages 421–440. Springer, 2017.

[Bir93] Jean-Camille Birget. State-complexity of finite-state devices, state compressibility and incompressibility. *Mathematical Systems Theory*, 26(3):237–269, 1993.

[BKV13] A Bauer, JC Küster, and Gil Vegliach. From propositional to first-order monitoring. In *Proceedings of RV 2013*, volume 8174 of *LNCS*, pages 59–75. Springer, 2013.

[BKZ11] David Basin, Felix Klaedtke, and Eugene Zălinescu. Algorithms for monitoring real-time properties. In *Proceedings of RV 2011*, volume 7186 of *LNCS*, pages 260–275. Springer, 2011.

[BLS11] Andreas Bauer, Martin Leucker, and Christian Schallhart. Runtime verification for LTL and TLTL. *ACM Transactions on Software Engineering and Methodology*, 20(4):14, 2011.

[BMOW07] Patricia Bouyer, Nicolas Markey, Joël Ouaknine, and James Worrell. The cost of punctuality. In *Proceedings of LICS 2007*, pages 109–120. IEEE Computer Society Press, 2007.

[BN12] Kevin Baldor and Jianwei Niu. Monitoring dense-time, continuous-semantics, metric temporal logic. In *Proceedings of RV 2012*, volume 7687 of *LNCS*, pages 245–259. Springer, 2012.

[BO14] Daniel Bundala and Joël Ouaknine. On the complexity of temporal-logic path checking. In *Proceedings of ICALP 2014*, volume 8573 of *LNCS*, pages 86–97. Springer, 2014.

[CE81] Edmund M. Clarke and E. Allen Emerson. Design and synthesis of synchronization skeletons using branching-time temporal logic. In *Proceedings of IBM Workshop on Logic of Programs*, volume 131 of *LNCS*, pages 52–71. Springer-Verlag, 1981.

[Dam94] Mads Dam. Temporal logic, automata, and classical theories - an introduction, 1994.

[DHV07] Deepak D'Souza, Raveendra Holla, and Deepak Vankadaru. On the expressiveness of TPTL in the pointwise and continuous semantics. Unpublished manuscript, 2007.

[DKL10] Christian Dax, Felix Klaedtke, and Martin Lange. On regular temporal logics with past. *Acta Informatica*, 47(4):251–277, 2010.

[DM13] Deepak D'Souza and Raj Mohan Matteplackel. A clock-optimal hierarchical monitoring automaton construction for MITL. Technical Report 2013-1, Department of Computer Science and Automation, Indian Institute of Science, 2013.

[DP06] Deepak D'Souza and Pavithra Prabhakar. On the expressiveness of MTL in the pointwise and continuous semantics. *International Journal on Software Tools for Technology Transfer*, 9(1):1–4, 2006.

[DT04] Deepak D'Souza and Nicolas Tabareau. On timed automata with input-determined guards. In *Proceedings of FORMATS/FTRTFT 2004*, volume 3253 of *LNCS*, pages 68–83. Springer, 2004.

[EFHL03] Cindy Eisner, Dana Fisman, John Havlicek, and Yoad Lustig. Reasoning with temporal logic on truncated paths. In *Proceedings of CAV 2003*, volume 2725 of *LNCS*, pages 27–39. Springer, 2003.

[EL86] E. Allen Emerson and Chin-Laung Lei. Efficient model checking in fragments of the propositional mu-calculus. In *Proceedings of LICS 1986*, pages 267–278. IEEE Computer Society Press, 1986.

[EW96] Kousha Etessami and Thomas Wilke. An until hierarchy for temporal logic. In *Proceedings of LICS 1996*, pages 108–117. IEEE Computer Society Press, 1996.

[FK09] Bernd Finkbeiner and Lars Kuhtz. Monitor circuits for LTL with bounded and unbounded future. In *Proceedings of RV 2009*, volume 5779 of *LNCS*, pages 60–75. Springer, 2009.

[FMDR13] Tim French, John Christopher McCabe-Dansted, and Mark Reynolds. Verifying temporal properties in real models. In *Proceedings of LPAR 2013*, volume 8312 of *LNCS*, pages 309–323. Springer, 2013.

[GHR94] Dov M. Gabbay, Ian Hodkinson, and Mark Reynolds. *Temporal Logics: Mathematical Foundations and Computational Aspects, Volume 1.* Oxford University Press, 1994.

[GO03] Paul Gastin and Denis Oddoux. LTL with past and two-way very-weak alternating automata. In *Proceedings of MFCS 2003*, volume 2747 of *LNCS*, pages 439–448. Springer, 2003.

[GPSS80] Dov Gabbay, Amir Pnueli, Sharanon Shelah, and J. Stavi. On the temporal analysis of fairness. In *Proceedings of POPL 1980*, pages 163–173. ACM Press, 1980.

[HMP92] Thomas A. Henzinger, Zohar Manna, and Amir Pnueli. What good are digital clocks? In *Proceedings of ICALP 1992*, volume 623 of *LNCS*, pages 545–558. Springer, 1992.

[Hol97] Gerard J. Holzmann. The model checker SPIN. *IEEE Transactions on Software Engineering*, 23(5):279–295, 1997.

[HOW13] Paul Hunter, Joël Ouaknine, and James Worrell. Expressive completeness of metric temporal logic. In *Proceedings of LICS 2013*, pages 349–357. IEEE Computer Society Press, 2013.

[HR01] Klaus Havelund and Grigore Roşu. Testing linear temporal logic formulae on finite execution traces. Technical Report RIACS 01.08, Research Institute for Advanced Computer Science, 2001.

[HR04] Yoram Hirshfeld and Alexander Moshe Rabinovich. Logics for real time: Decidability and complexity. *Fundamenta Informaticae*, 62(1):1–28, 2004.

[HR06] Yoram Hirshfeld and Alexander Moshe Rabinovich. An expressive temporal logic for real time. In *Proceedings of MFCS 2006*, volume 4162 of *LNCS*, pages 492–504. Springer, 2006.

[HR07] Yoram Hirshfeld and Alexander Rabinovich. Expressiveness of metric modalities for continuous time. *Logical Methods in Computer Science*, 3(1), 2007.

[HRS98] Thomas A. Henzinger, Jean-François Raskin, and Pierre-Yves Schobbens. The regular real-time languages. In *Proceedings of ICALP 1998*, volume 1443 of *LNCS*, pages 580–591. Springer, 1998.

[Hun13] Paul Hunter. When is metric temporal logic expressively complete? In *Proceedings of CSL 2013*, volume 23 of *LIPIcs*, pages 380–394. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2013.

[Kam68] Johan A. Kamp. *Tense logic and the theory of linear order.* PhD thesis, University of California, Los Angeles, 1968.

[KC05] Sascha Konrad and Betty H. C. Cheng. Real-time specification patterns. In *Proceedings of ICSE 2005*, pages 372–381. ACM Press, 2005.

[KF09] Lars Kuhtz and Bernd Finkbeiner. LTL path checking is efficiently parallelizable. In *Proceedings of ICALP 2009*, volume 5556 of *LNCS*, pages 235–246. Springer, 2009.

[KKP11] Dileep Kini, Shankara N. Krishna, and Paritosh Pandya. On construction of safety signal automata for MITL[U,S] using temporal projections. In *Proceedings of FORMATS 2011*, volume 6919 of *LNCS*, pages 225–239. Springer, 2011.

[Koy90] Ron Koymans. Specifying real-time properties with metric temporal logic. *Real-Time Systems*, 2(4):255–299, 1990.

[KV01] Orna Kupferman and Moshe Y. Vardi. Model checking of safety properties. *Formal Methods in System Design*, 19(3):291–314, 2001.

[LPZ85] Orna Lichtenstein, Amir Pnueli, and Lenore D. Zuck. The glory of the past. In *Proceedings of Logics of Programs 1985*, volume 193 of *LNCS*, pages 196–218. Springer, 1985.

[LS09] Martin Leucker and Christian Schallhart. A brief account of runtime verification. *Journal of Logic and Algebraic Programming*, 78(5):293–303, 2009.

[LW08] Slawomir Lasota and Igor Walukiewicz. Alternating timed automata. *ACM Transactions on Computational Logic*, 9(2), 2008.

[MN04] Oded Maler and Dejan Nickovic. Monitoring temporal properties of continuous signals. In *Proceedings of FORMATS/FTRTFT 2004*, volume 3253 of *LNCS*, pages 152–166. Springer, 2004.

[MNP05] Oded Maler, Dejan Nickovic, and Amir Pnueli. Real time temporal logic: Past, present, future. In *Proceedings of FORMATS 2005*, volume 3829 of *LNCS*, pages 2–16. Springer, 2005.

[MNP06] Oded Maler, Dejan Nickovic, and Amir Pnueli. From MITL to timed automata. In *Proceedings of FORMATS 2006*, volume 4202 of *LNCS*, pages 274–289. Springer, 2006.

[MP95] Zohar Manna and Amir Pnueli. *Temporal verification of reactive systems: safety*, volume 2. Springer, 1995.

[MS03] Nicolas Markey and Philippe Schnoebelen. Model checking a path (preliminary report). In *Proceedings of CONCUR 2003*, volume 2761 of *LNCS*, pages 251–265. Springer, 2003.

[NP10] Dejan Nickovic and Nir Piterman. From MTL to deterministic timed automata. In *Proceedings of FORMATS 2010*, volume 6246 of *LNCS*, pages 152–167. Springer, 2010.

[ORW09] Joël Ouaknine, Alexander Rabinovich, and James Worrell. Time-bounded verification. In *Proceedings of CONCUR 2009*, volume 5710 of *LNCS*, pages 496–510. Springer, 2009.

[OW06] Joël Ouaknine and James Worrell. On metric temporal logic and faulty turing machines. In *Proceedings of FoSSaCS 2006*, volume 3921 of *LNCS*, pages 217–230. Springer, 2006.

[OW08] Joël Ouaknine and James Worrell. Some recent results in metric temporal logic. In *Proceedings of FORMATS 2008*, volume 5215 of *LNCS*, pages 1–13. Springer, 2008.

[OW10] Joël Ouaknine and James Worrell. Towards a theory of time-bounded verification. In *Proceedings of ICALP 2010*, volume 6199 of *LNCS*, pages 22–37. Springer, 2010.

[PD06] Pavithra Prabhakar and Deepak D'Souza. On the expressiveness of MTL with past operators. In *Proceedings of FORMATS 2006*, volume 4202 of *LNCS*, pages 322–336. Springer, 2006.

[PS11] Paritosh K. Pandya and Simoni S. Shah. On expressive powers of timed logics: Comparing boundedness, non-punctuality and deterministic freezing. In *Proceedings of CONCUR 2011*, volume 6901 of *LNCS*, pages 60–75. Springer, 2011.

[QS82] Jean-Pierre Queille and Joseph Sifakis. Specification and verification of concurrent systems in CESAR. In *Proceedings of Symposium on Programming 1982*, volume 137 of *LNCS*, pages 337–351. Springer, 1982.

[Rab10] Alexander Rabinovich. Temporal logics over linear time domains are in PSPACE. In *Proceedings of RP 2010*, volume 6227 of *LNCS*, pages 29–50. Springer, 2010.

[Rey10] Mark Reynolds. The complexity of temporal logic over the reals. *Annals of Pure and Applied Logic*, 161(8):1063–1096, 2010.

[Rey14] Mark Reynolds. Metric temporal logics and deterministic timed automata (long report version). Technical report, University of West Australia, 2014.

[Roş12] Grigore Roşu. On safety properties and their monitoring. *Scientific Annals of Computer Science*, 22(2):327–365, 2012.

[SC85] A. Prasad Sistla and Edmund M. Clarke. The complexity of propositional linear temporal logics. *Journal of the ACM*, 32(3):733–749, 1985.

[SHL11] Oleg Sokolsky, Klaus Havelund, and Insup Lee. Introduction to the special section on runtime verification. *International Journal on Software Tools for Technology Transfer*, 14(3):243–247, 2011.

[Sto74] Larry Stockmeyer. The complexity of decision problems in automata theory and logic. PhD thesis, TR 133, M.I.T., Cambridge, 1974.

[TR05] Prasanna Thati and Grigore Roşu. Monitoring algorithms for metric temporal logic specifications. *Electronic Notes in Theoretical Computer Science*, 113:145–162, 2005.

[Tri02] Stavros Tripakis. Fault diagnosis for timed automata. In *Proceedings of FTRTFT 2002*, volume 2469 of *LNCS*, pages 205–224. Springer, 2002.

[Var96] Moshe Y. Vardi. An automata-theoretic approach to linear temporal logic. In *Logics for Concurrency – Structure versus Automata (8th Banff Higher Order Workshop'95)*, volume 1043 of *LNCS*, pages 238–266. Springer, 1996.

[Wil94] Thomas Wilke. Specifying timed state sequences in powerful decidable logics and timed automata. In *Proceedings of FTRTFT 1994*, volume 863 of *LNCS*, pages 694–715. Springer, 1994.

[WVS83] Pierre Wolper, Moshe Y. Vardi, and A. Prasad Sistla. Reasoning about infinite computation paths. In *Proceedings of FOCS 1983*, pages 185–194. IEEE Computer Society Press, 1983.

This work is licensed under the Creative Commons Attribution License. To view a copy of this license, visit `https://creativecommons.org/licenses/by/4.0/` or send a letter to Creative Commons, 171 Second St, Suite 300, San Francisco, CA 94105, USA, or Eisenacher Strasse 2, 10777 Berlin, Germany