On the Computational Complexity of Decidable Fragments of First-Order Linear Temporal Logics

Ian Hodkinson,1 Roman Kontchakov,2 Agi Kurucz,2 Frank Wolter3 and Michael Zakharyaschev2

1 Department of Computing, Imperial College, London SW7 2AZ, U.K.
2 Department of Computer Science, King’s College, Strand, London WC2R 2LS, U.K.
3 Department of Computer Science, University of Liverpool, Liverpool L69 7ZF, U.K.
(e-mails: imh@doc.ic.ac.uk, {romanvk,kuag,mz}@dcs.kcl.ac.uk, frank@csc.liv.ac.uk)

Abstract

We study the complexity of some fragments of first-order temporal logic over natural numbers time. The one-variable fragment of linear first-order temporal logic even with sole temporal operator $\square$ is EXPSPACE-complete (this solves an open problem of [10]). So are the one-variable, two-variable and monadic monodic fragments with Until and Since. If we add the operators $\bigcirc^n$, with $n$ given in binary, the fragments become 2EXPSPACE-complete. The packed monodic fragment has the same complexity as its pure first-order part — 2EXPTIME-complete. Over any class of flows of time containing one with an infinite ascending sequence — e.g., rationals and real numbers time, and arbitrary strict linear orders — we obtain EXPSPACE lower bounds (which solves an open problem of [16]). Our results continue to hold if we restrict to models with finite first-order domains.

1. Introduction

What is known about the computational complexity of linear time temporal logics? Everything seems to be clear in the propositional case. The logics with only one temporal operator $\square$ (‘always in the future’) are known to be co-NP-complete for linear time, for the flows of time $\langle \mathbb{Q}, < \rangle$ and $\langle \mathbb{R}, < \rangle$ [15] as well as for $\langle \mathbb{N}, < \rangle$ [22]. The complexity remains the same if one adds the corresponding past operator $\bigcirc [15, 22, 25]$. The addition of the ‘next-time’ operator $\bigcirc$ and/or the ‘until’ operator $\mathcal{U}$ to this primitive language makes the logic PSPACE-complete over $\langle \mathbb{N}, < \rangle$ [22], and $\langle \mathbb{Q}, < \rangle$, $\langle \mathbb{R}, < \rangle$, and the class of arbitrary strict linear orders [17, 18]. The succinctness of the operators $\bigcirc^n$ (‘in $n$ moments of time’), where $n > 1$ is given in binary, increases the complexity to EXPSPACE (over $\langle \mathbb{N}, < \rangle$) [1], but, of course, does not change the expressive power of the language.

Compared to this ‘well cultivated garden’, the complexity of first-order temporal logics and their fragments is still terra incognita. There are well known ‘negative’ results: for example, $\Pi^1_1$-completeness of the two-variable monadic temporal logic of the flow of time $\langle \mathbb{N}, < \rangle$; see, e.g., [11] and references therein. But we could find only one ‘positive’ result: Halpern and Vardi [10] and, independently, Sistla and German [23] showed that the one-variable fragment of the logic with $\square$, $\bigcirc$, and/or $\mathcal{U}$ over $\langle \mathbb{N}, < \rangle$ is EXPSPACE-complete.

Halpern and Vardi considered this fragment as a propositional epistemic temporal logic with one agent modelled by the propositional modal system $\mathbf{S}5$. They conjectured that, as in the propositional case, “even with knowledge operators in the language, the complexity still becomes much simpler without $\bigcirc$ and $\mathcal{U}$” [10, page 231].

We take up this conjecture as a starting point of our investigation of the computational complexity of decidable fragments of first-order linear temporal logic. The main technical result of this paper is that over a wide range of flows of time, the one-variable fragment of linear temporal logic even with sole operator $\square$ is EXPSPACE-hard.

We also establish matching EXPSPACE upper bounds for the one-variable, two-variable and monadic monodic fragments of the first-order temporal logic based on the flow of time $\langle \mathbb{N}, < \rangle$ and having $\square$, $\bigcirc$, $\mathcal{U}$, and $\mathcal{S}$ (since) as their temporal operators. The fragments are EXPSPACE-complete even if we restrict to models with finite first-order domains. If we add the operators $\bigcirc^n$, with $n$ given in binary, the fragments become 2EXPSPACE-complete. Finally, the packed monodic fragment turns out to be as complex as its pure first-order part, i.e., 2EXPTIME-complete.

1[21] and [2] determined the complexity of certain temporalised description logics, which can be regarded as fragments of first-order temporal logics.
The 2EXPTIME upper bound for $\mathcal{QTL}_{1}$ over $\langle \mathbb{Q}, < \rangle$ can be obtained using the mosaic technique from [26] (for details see [6]).

Thus, in surprising contrast to the propositional case, the omission of $\circ$ and $\mathcal{U}$ does not alter the computational complexity of first-order temporal logics. The addition of $\circ^n$ ($n > 0$) increases the complexity by one exponential, as in the propositional case. We will not discuss here the pros and cons of adding or omitting these operators for various applications, but refer the reader to, e.g., [24, 12, 1].

The known and new results are summarised in terms of satisfiability in Table 1. The languages in the table are explained at the end of §2 and at the start of §4. For a complexity class $\mathcal{C}$, an entry $'\leq \mathcal{C}'$ in the table indicates that the complexity is in $\mathcal{C}$; an entry $'\geq \mathcal{C}'$ indicates $\mathcal{C}$-hardness; and an entry $'\mathcal{C}'$ indicates $\mathcal{C}$-completeness.

We note that finding upper bounds over $\langle \mathbb{Q}, < \rangle$ and $\langle \mathbb{R}, < \rangle$ for most of the first-order logics in the table remains an open problem.

2. Preliminaries

We begin by introducing the logics dealt with in this paper.

The alphabet of the first-order (or quantified) temporal language $\mathcal{QTL}$ without equality consists of a countably infinite set of individual variables, a non-empty list of predicate symbols $P_0, P_1, \ldots$, each of which is equipped with some fixed arity $\geq 0$, the Booleans $\neg, \land, \lor$, and $\mathcal{U}$, the existential quantifiers $\exists x$ for every variable $x$, and the temporal operators $\circ$ (‘next-time’), $\mathcal{S}$ (‘since’) and $\mathcal{U}$ (‘until’). The set of $\mathcal{QTL}$-formulas $\varphi$ is defined as usual:

$$\varphi ::= \ P(x_1, \ldots, x_m) \mid \neg \varphi \mid \varphi_1 \land \varphi_2 \mid \mathcal{T} \mid \mathcal{U} \mid \exists x \ \varphi \mid \circ \varphi \mid \varphi_1 \mathcal{S} \varphi_2 \mid \varphi_1 \mathcal{U} \varphi_2,$$

where $P$ is an $m$-ary predicate symbol and $x_1, \ldots, x_m$ are variables. We also use the standard abbreviations $\lor, \rightarrow,$ and $\forall x \varphi := \neg \exists x \neg \varphi, \circ \varphi := \mathcal{T} \mathcal{U} \varphi, \square \varphi := \neg \circ \neg \varphi, \boxdot \varphi := \varphi \land \square \varphi$.

Given a formula $\varphi$, we write $\varphi(x_1, \ldots, x_m)$ to indicate that all free variables of $\varphi$ are in the set $\{x_1, \ldots, x_m\}$; in particular, $\varphi(x)$ has at most one free variable $x$. $\mathcal{QTL}$ is interpreted in first-order temporal models of the form $\mathfrak{M} = \langle \mathfrak{g}, D, I \rangle$, where $\mathfrak{g} = \langle W, < \rangle$ is a strict linear order representing the flow of time, $D$ is a non- set, the domain of $\mathfrak{M}$, and $I$ is a function associating with every moment of time $w \in W$ a first-order structure

$$I(w) = \langle D, P_0^{I(w)}, P_1^{I(w)}, \ldots \rangle,$$

where each $P_i^{I(w)}$ is a relation on $D$ of the same arity as $P_i$. An assignment $a$ in $\mathfrak{M}$ is a function from the set of individual variables to $D$. Given a $\mathcal{QTL}$-formula $\varphi$, the truth relation $\langle \mathfrak{M}, w \rangle \models a \varphi$ (‘$\varphi$ is true at moment $w$ in model $\mathfrak{M}$ under assignment $a$’) is defined inductively on the construction of $\varphi$:

- $\langle \mathfrak{M}, w \rangle \models a P(x_1, \ldots, x_m)$ iff $\langle a(x_1), \ldots, a(x_m) \rangle \in P^{I(w)}$;
- $\langle \mathfrak{M}, w \rangle \models a \neg \psi$ iff $\langle \mathfrak{M}, w \rangle \not\models a \psi$;
- $\langle \mathfrak{M}, w \rangle \models a \varphi_1 \land \varphi_2$ iff $\langle \mathfrak{M}, w \rangle \models a \varphi_1$ and $\langle \mathfrak{M}, w \rangle \models a \varphi_2$;
- $\langle \mathfrak{M}, w \rangle \models a \mathcal{T}$ and $\langle \mathfrak{M}, w \rangle \not\models a \mathcal{U}$;
- $\langle \mathfrak{M}, w \rangle \models a \exists x \psi$ iff $\langle \mathfrak{M}, w \rangle \models b \psi$ for some assignment $b$ that may differ from $a$ only on $x$;
- $\langle \mathfrak{M}, w \rangle \models a \varphi_1 \mathcal{U} \varphi_2$ iff there is $v > w$ such that $\langle \mathfrak{M}, v \rangle \models a \varphi_2$ and $\langle \mathfrak{M}, u \rangle \models a \varphi_1$ for all $u \in (v, w)$, where $(w, v) = \{u \in W \mid w < u < v\}$;
- $\langle \mathfrak{M}, w \rangle \models a \varphi_1 \mathcal{S} \varphi_2$ iff there is $v < w$ such that $\langle \mathfrak{M}, v \rangle \models a \varphi_2$ and $\langle \mathfrak{M}, u \rangle \models a \varphi_1$ for all $u \in (v, w)$;
- $\langle \mathfrak{M}, w \rangle \models a \circ \psi$ iff there is an immediate successor $v$ of $w$ in $W$ with $\langle \mathfrak{M}, v \rangle \models a \psi$.

Note that, according to the given semantics, $\circ \varphi$ is equivalent to $\mathcal{U} \varphi$. Instead of $\langle \mathfrak{M}, w \rangle \models a \varphi(x)$, we may write $\langle \mathfrak{M}, w \rangle \models a \varphi[a]$, where $a(x) = a$. For a sentence $\varphi$ (i.e., with no free variables), we just write $\langle \mathfrak{M}, w \rangle \models a \varphi$.

A $\mathcal{QTL}$-formula $\varphi$ is said to be satisfiable if $\langle \mathfrak{M}, w \rangle \models a \varphi$ holds for some model $\mathfrak{M}$, some moment $w$ and some assignment $a$.

Fragments. For $n \in \mathbb{N}$, the $n$-variable fragment of $\mathcal{QTL}$, consisting of $\mathcal{QTL}$-formulas with at most $n$ variables, is denoted by $\mathcal{QTL}^n$ (remember that the one-variable fragment of classical first-order logic is a notational variant of propositional modal logic $\mathcal{S}5$). $\mathcal{QTL}^1_{\mathcal{C}}$ is the subfragment of $\mathcal{QTL}^1$ with sole temporal operator $\mathcal{C}$. $\mathcal{QTL}^n_{\mathcal{bin}}$ extends $\mathcal{QTL}^1$ with the temporal operators $\circ^n$ ($n \in \mathbb{N}$), where $n$ is given in binary. The propositional fragments $\mathcal{PTL}_{\mathcal{C}}, \mathcal{PTL}_{\mathcal{bin}}$ are defined analogously to fragments of $\mathcal{QTL}^1 = \mathcal{PTL}$. Some further ‘monodic’ fragments (e.g., the $\mathcal{QTL}^n_{\mathcal{mon}}, \mathcal{QTL}^2_{\mathcal{mon}}$, and $\mathcal{TPF}_{\mathcal{mon}}$ of Table 1) will be defined in §4.

3. Lower bounds

The main result in this section is the following:

Theorem 3.1. Let $\mathcal{C}$ be any class of strict linear orders, at least one of which contains an infinite ascending chain. Then the satisfiability problem for $\mathcal{QTL}^n_{\mathcal{C}}$-formulas in models based on flows of time from $\mathcal{C}$ is EXPSPACE-hard. The
same holds if we restrict to models $\langle \mathcal{F}, D, I \rangle$ with finite domain $D$.

**Proof.** The proof uses some ideas from [14, 19]. First we treat the case of arbitrary infinite domains. The proof is by reduction of the following infinite version of the 2$^n$-corridor tiling problem, which is known to be EXPSPACE-complete (cf. results in [5]): given an instance $\mathcal{T} = \langle T, t_0, n \rangle$, where $T$ is a finite set of tile types, $t_0 \in T$ is a tile type, and $n \in \mathbb{N}$ is given in binary, decide whether $T$ tiles the $\mathbb{N} \times 2^m$-corridor $\{(x, y) \mid x \in \mathbb{N}, 0 \leq y < 2^n\}$ in such a way that $t_0$ is placed at $(0, 0)$ and the top and bottom sides of the corridor are of some fixed colour, say, white.

Suppose that $\mathcal{T} = \langle T, t_0, n \rangle$ is given. Our aim is to write a $\mathcal{L}_m^{0,1}$-formula $\varphi_\mathcal{T}$ such that (i) $\varphi_\mathcal{T}$ is constructible from $\mathcal{T}$ in polynomial time, and (ii) $\varphi_\mathcal{T}$ is satisfiable in a first-order temporal model $\mathfrak{M} = \langle \mathcal{F}, D, I \rangle$ based on some $\mathcal{F} = \langle W, < \rangle$ from $C$ iff $T$ tiles the $\mathbb{N} \times 2^n$-corridor so that the top and bottom sides are white and $t_0$ is placed at $(0, 0)$.

We will write down nine numbered conjuncts of $\varphi_\mathcal{T}$. To aid our explanation, we will assume that they hold in a model $\mathfrak{M} = \langle \langle W, < \rangle, D, I \rangle$ at a time $x_0 \in W$ and a point $y_0 \in D$, and show how they force a tiling.

$\varphi_\mathcal{T}$ will contain, among many others, unary predicates $t(x)$ for all $x \in T$. Our first step in the construction of $\varphi_\mathcal{T}$ is to write down formulas forcing not only an infinite sequence $y_0, y_1, \ldots$ of distinct elements from $D$, but at the same time an infinite sequence $x_0 < x_1 < x_2 < \ldots$ of points from $W$, such that for each $i \in \mathbb{N}$, $\mathfrak{M}, x_i \models t[y_i]$ for a unique tile type $t_i$. If $i = k \cdot 2^n + j$ for some $j < 2^n$ then we will use $\langle x_i, y_i \rangle$ to encode the pair $(k, j)$ of the $\mathbb{N} \times 2^n$-grid. Thus, the upper neighbour $(k, j + 1)$ of $(k, j)$ (if $j + 1 < 2^n$) will be coded by the element $y_{i+1}$ at time $x_{i+1}$, and its right neighbour $(k + 1, j)$ by $y_{i+2^n}$ at the moment $x_{i+2^n}$.

Let $q_0, \ldots, q_{n-1}$ be pairwise distinct propositional variables, and $P_0, \ldots, P_{n-1}$ be distinct unary predicates. We will require that the truth values of the $P_i$ do not change over time. This requirement can be ensured by the sentence

$$\forall x \left( \Diamond^+ P_i(x) \lor \square^+ \neg P_i(x) \right). \quad (1)$$

For any atomic formula $\alpha$, write $\alpha^1$ for $\alpha$ and $\alpha^C$ for $\neg \alpha$.

For each $j < 2^n$, define formulas

$$\tau_j = q_0^{d_0} \land \cdots \land q_{n-1}^{d_{n-1}},$$

$$\delta_j(x) = P_0^{d_0}(x) \land \cdots \land P_{n-1}^{d_{n-1}}(x),$$

where $d_0 \ldots d_n$ is the binary representation of $j$. We say that the moment $u \in W$ is of type $j$ if $\mathfrak{M}, u \models \tau_j$. Assuming that (1) holds at $x_0$, we also say that the element $y \in D$ is of type $j$ if $\mathfrak{M}, u \models \delta_j[y]$ for all $u \in W$ with $u \geq x_0$.

Now define the formula

$$\text{equ}(x) = \bigwedge_{i < n} (P_i(x) \leftrightarrow q_i).$$

It should be clear that for all moments $u \geq x_0$ and all elements $y \in D$, if $\mathfrak{M}, u \models \text{equ}[y]$ then $u$ and $y$ are of the same type ($j$, for some $j < 2^n$).

We can now define ‘counting’ formulas of length polynomial in $n$. Suppose that $\text{succ}(x)$ is a unary predicate and

| Table 1. Complexity of the satisfiability problem for various linear temporal logics |
|-----------------------------------------------|------------------|------------------|
| language | flow of time | flow of time | flow of time |
| $\mathcal{L}_m^{0,1}$ | EXPSPACE | $\geq$ EXPSPACE | $\geq$ EXPSPACE |
| $\mathcal{L}_m^{1}$ | EXPSPACE | $\geq$ EXPSPACE | $\geq$ EXPSPACE |
| $\mathcal{L}^{1}_{bin}$ | 2EXPSPACE | $\geq$ EXPSPACE | $\geq$ EXPSPACE |
| $\mathcal{L}^{\text{mo}}_m$ | EXPSPACE | $\geq$ EXPSPACE | $\geq$ EXPSPACE |
| $\mathcal{L}^{\text{lo}}_m$ | EXPSPACE | $\geq$ EXPSPACE | $\geq$ EXPSPACE |
| $\mathcal{T}^\text{F}$ | 2EXPTIME | $\geq$ 2EXPTIME | $\geq$ 2EXPTIME |

<table>
<thead>
<tr>
<th>language</th>
<th>flow of time</th>
<th>flow of time</th>
<th>flow of time</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{L}_m^{0,1}$</td>
<td>EXPSPACE</td>
<td>$\geq$ EXPSPACE</td>
<td>$\geq$ EXPSPACE</td>
</tr>
<tr>
<td>$\mathcal{L}_m^{1}$</td>
<td>EXPSPACE</td>
<td>$\geq$ EXPSPACE</td>
<td>$\geq$ EXPSPACE</td>
</tr>
<tr>
<td>$\mathcal{L}^{1}_{bin}$</td>
<td>2EXPSPACE</td>
<td>$\geq$ EXPSPACE</td>
<td>$\geq$ EXPSPACE</td>
</tr>
<tr>
<td>$\mathcal{L}^{\text{mo}}_m$</td>
<td>EXPSPACE</td>
<td>$\geq$ EXPSPACE</td>
<td>$\geq$ EXPSPACE</td>
</tr>
<tr>
<td>$\mathcal{L}^{\text{lo}}_m$</td>
<td>EXPSPACE</td>
<td>$\geq$ EXPSPACE</td>
<td>$\geq$ EXPSPACE</td>
</tr>
<tr>
<td>$\mathcal{T}^\text{F}$</td>
<td>2EXPTIME</td>
<td>$\geq$ 2EXPTIME</td>
<td>$\geq$ 2EXPTIME</td>
</tr>
</tbody>
</table>
that (1) and the two sentences

\[ \square^+ \bigwedge_{k < n} \left( \left( \bigwedge_{i < k} q_i \land \neg q_k \right) \rightarrow \forall x [\text{succ}(x) \leftrightarrow \bigwedge_{i < k} \neg P_i(x)] \right) \land \neg \bigwedge_{k < j < n} \left( P_j(x) \leftrightarrow q_j \right) \right) \]

(2)

\[ \square^+ \left( \bigwedge_{i < n} q_i \rightarrow \forall x [\text{succ}(x) \leftrightarrow \bigwedge_{i < n} \neg P_i(x)] \right) \]

(3)

hold in \( \mathcal{M} \) at \( x_0 \). For \( u \geq x_0 \) and \( y \in D \) with \( (\mathcal{M}, u) \models \text{succ}(y) \), if \( u \) is of type \( j \) \((j < 2^n)\) then \( y \) is of type \( j + 1 \) \((\text{mod } 2^n)\).

Write

\[ \text{tile}(x) = \bigvee_{t \in T} t(x), \quad \text{and} \quad \chi = \exists x \text{ tile}(x). \]

Now we can generate the required infinite sequences of points using the formula

\[ \tau_0 \land \text{equ}(x) \land \text{tile}(x) \land \square \neg \text{tile}(x) \]

\[ \land \square^+ \left( \chi \rightarrow \exists x [\text{succ}(x) \land \diamond (\text{equ}(x) \land \text{tile}(x))] \right) \land \square (\diamond (\text{tile}(x) \rightarrow \neg \chi)) \]

(4)

Indeed, suppose that the conjunction of (1)–(4) holds at \( x_0 \) on some element \( y_0 \in D \). Then \( x_0, y_0 \) are of type 0. Since \( (\mathcal{M}, x_0) \models \exists x [\text{succ}(x) \land \diamond (\text{equ}(x) \land \text{tile}(x))] \land \square (\text{tile}(x) \rightarrow \neg \chi) \), there are \( y_1 \in D \) and \( x_1 > x_0 \) in \( W \) such that

- \( (\mathcal{M}, x_0) \models \text{succ}(y_1) \) (so \( y_1 \) is of type 1),
- \( (\mathcal{M}, x_1) \models \text{equ}(y_1) \) (so the moment \( x_1 \) is of type 1),
- \( (\mathcal{M}, x_1) \models \text{tile}(y_1) \) (note that since \( (\mathcal{M}, x_0) \models \square \neg \text{tile}(y_0) \), we have \( y_1 \neq y_0 \)),
- no moment \( u > x_1 \) makes \( \text{tile}(y_1) \) true,
- no moment \( u \) with \( x_0 < u < x_1 \) makes \( \chi \) true.

Repeating this argument with \( x_1 \) in place of \( x_0 \), we find \( y_0 \notin \{y_0, y_1\} \) and \( x_2 > x_1 \) of type 2, etc., and so forth, until we get to a moment \( x_{2^n-1} \) which is of type \( 2^n - 1 \), and then to \( x_{2^n} \) of type 0 again. See Fig. 1.

Our next aim is to write down formulas to locate the upper and right neighbours of a given tile in the corridor. Let

\[ \text{up}(x) = \diamond \text{tile}(x) \land \square (\diamond \text{tile}(x) \rightarrow \neg \chi), \]

\[ \text{right}(x) = \text{equ}(x) \land \diamond \text{tile}(x) \land \square (\chi \land \diamond \text{tile}(x) \rightarrow \neg \text{equ}(x)). \]

It is easy to see that for all \( i, j \in \mathbb{N} \),

- \( (\mathcal{M}, x_i) \models \text{up}(y_j) \iff j = i + 1 \),
- \( (\mathcal{M}, x_i) \models \text{right}(y_{j+2^n}) \iff j = i + 2^n \).

Now, the formulas below enforce that \( (0, 0) \) is covered by \( t_0 \), every point of the \( \mathbb{N} \times 2^n \)-corridor is covered by at most one tile, the top and bottom sides of the corridor are white, and the colours on adjacent edges of adjacent tiles match:

\[ t_0(x) \land \square^+ \forall x \left( \bigwedge_{t, t' \in T, t \neq t'} \neg (t(x) \land t'(x)) \right), \]

(5)

\[ \square^+ \forall x \left( t_0 \land \text{tile}(x) \rightarrow \bigvee_{t \in T, \downarrow \text{up}(t) = \text{white}} t(x) \right), \]

(6)

\[ \square^+ \forall x \left( \tau_{2n-1} \land \text{tile}(x) \rightarrow \bigvee_{t \in T, \uparrow \text{down}(t) = \text{white}} t(x) \right), \]

(7)

\[ \square^+ \left( \neg \tau_{2n-1} \rightarrow \right. \]

\[ \forall x \left( \bigwedge_{t, t' \in T, \text{up}(t) \neq \text{down}(t')} (t(x) \rightarrow \forall x \left( \text{up}(x) \rightarrow \square \neg t'(x) \right)) \right), \]

(8)

\[ \square^+ \forall x \left( t(x) \rightarrow \forall x \left( \text{right}(x) \rightarrow \square \neg t'(x) \right) \right), \]

(9)

Let \( \varphi_T \) be the conjunction of (1)–(9). It is clear that \( \varphi_T \) is constructible from \( T \) in polynomial time. Suppose that \( (\mathcal{M}, x_0) \models \varphi_T[y_k] \). Then, after defining the points \( x_i, y_i \) \((i \in \mathbb{N})\) as above, we define a map \( f: \mathbb{N} \times 2^n \rightarrow T \) by taking

\[ f(k, j) = t \iff (\mathcal{M}, x_{k \cdot 2^n + j}) = \tau[y_{k \cdot 2^n + j}]. \]

We leave it to the reader to check that \( f \) is indeed a tiling of \( \mathbb{N} \times 2^n \) as required.

For the other direction, take a flow of time \( \delta \) from \( C \) having an infinite ascending chain of distinct points \( x_i \). Assuming that \( T = \{T, t_0, n\} \) tiles the \( \mathbb{N} \times 2^n \)-corridor, Fig. 1 shows that \( \varphi_T \) is satisfiable in a first-order temporal model based on \( \delta \) and with infinite domain.

Now we sketch how to deal with models with finite domains. By the pigeon-hole principle, any tiling of the \( \mathbb{N} \times 2^n \)-corridor by \( T \) \((T, t_0, n)\) has two identical columns \( X, Y \), so it can be converted into an eventually periodic tiling by iterating the part \( \{X, Y\} \) between the columns. Such a tiling can be specified by finitely many \( x_i, y_i \). So we modify \( \varphi_T \) by adding propositional variables \( X, Y \) to mark the end of the columns, relativising the main \( \square \) in (4) to times before \( Y \) by replacing the first \( \chi \) by \( \chi \land \diamond \chi' \), and including a statement that corresponding tiles in columns \( X \) and \( Y \) are the same. We leave the reader to write the required formulas. Since no pairs \( (x_i, y_i) \) are forced for \( x_i \) after \( Y \), the resulting formula has a model with finite domain iff \( T \) tiles the corridor.

It is a consequence of this theorem that the decision problem for the temporal epistemic logic \( C_{nf,sd,sync} \) of [10] for
synchronous systems with perfect recall and no forgetting for one agent with sole temporal operator $\Box$ is EXPSPACE-hard. To see this, it is enough to recall that $C_{nl,sync}$ is just a notational variant of that logic. Indeed, assume that $C_{nf,nl,sync}$ is based on the language with propositional variables $p_1, \ldots, p_n$, the knowledge operator $K$, the temporal operators $\Box$, and the Booleans, $\land$ and $\neg$. A translation $\tau$ from that language onto $\mathcal{LT}_\Box$ can be defined by taking

$p_i^1 = P_i(x)$

$(\psi_1 \land \psi_2)^1 = \psi_1^1 \land \psi_2^1$

$(\neg \psi)^1 = \neg \psi^1$

$(K \psi)^1 = \forall x \psi^1$

$(\Box \psi)^1 = \Box \psi^1$.

It is easy to prove that $\psi$ is satisfiable iff $\psi^1$ is satisfiable (see, e.g., [6]). Obviously, this equivalence neither depends on the flow of time nor on the temporal operators available — as long as we consider the same flows of time and the same temporal operators for both languages. So, every result formulated in the present paper for a variant of $\mathcal{LT}_\Box$ holds true for the corresponding variant of $C_{nf,nl,sync}$ as well.

Note also that in the literature on (products of) modal logic the corresponding systems are often denoted by $\mathbf{PTL} \times \mathbf{S5}$ [7]. Again, the results formulated here for $\mathcal{LT}_\Box$ hold true for the corresponding product logics with $\mathbf{S5}$.

Reynolds [16] proved the decidability of the product $\mathbf{Lin} \times \mathbf{S5}$, where $\mathbf{Lin}$ is the temporal logic of arbitrary strict linear orders with the operators ‘always in the future’ and ‘always in the past.’ He gave a $2\EXPSPACE$ decision procedure and conjectured that the lower bound should be $\EXPSPACE$. Theorem 3.1 shows this conjecture to be true.

Now consider the language $\mathcal{LT}_{bin}^1$ extending $\mathcal{LT}_\Box$ with the temporal operators $\circ^n$, $n$ given in binary.

**Theorem 3.2.** The satisfiability problem for $\mathcal{LT}_{bin}^1$, formulas over the flow of time $\langle \mathbb{N}, < \rangle$ is $2\EXPSPACE$-hard.

**Proof.** This result can be proved by an easy modification of the proof of Corollary 4.3(1) of [10]. Actually, it follows from this proof that it is sufficient to prove the following version of Lemma 4.1 (about ‘yardsticks’) of [10]:

Let, inductively, $\exp(0, n) = n$, and $\exp(k + 1, n) = \exp(k, n) \cdot 2^{\exp(k, n)}$. 

---

**Figure 1.** Satisfying $\varphi_T$ (with $n = 2$) in a first-order temporal model
Lemma 3.3. For every \( n \geq 0 \), there exists a satisfiable formula \( \varphi_n \) of temporal propositional logic extended by \( \circ^n \), \( n \) coded in binary, with \( |\varphi_n| \in O(n) \), such that if \( \varphi_n \) is true at moment 0, then there exists \( N \geq 0 \) such that the propositional variable \( p_N \) is true in a time point \( m \) iff \( m \) is of the form \( N + j \cdot \exp(2,n) \), for some \( j \geq 0 \).

This modified lemma can be proved by defining \( \varphi_{1,n} \) as in the proof of Lemma 4.1 of [10], and then using the operator \( \bigcirc \cdot \exp(1,n) \) in the construction of \( \varphi_n \) in the same way as the operators \( \bigcirc^n \) used in the definition of \( \varphi_{1,n} \). (Actually, the authors briefly discuss this construction on page 222 of [10]). We leave the details to the reader.

Obviously, given the computational complexity of the language with \( \bigcirc \) and \( [ \) we obtain an upper bound for the language with additional operators \( \bigcirc^n \), \( n > 1 \), by adding one exponential. The result above states that this upper bound is optimal. Note also that the proof above goes through for satisfiability in models with finite domains.

4. Upper bounds

Now we obtain the matching upper bounds for some decidable fragments of first-order temporal logics based on \( \langle \mathbb{N}, < \rangle \). The maximal ‘well-behaved’ sublanguage of \( \mathcal{VT} \) yet discovered [11] consists of so-called monodic formulas. A \( \mathcal{VT} \)-formula is said to be monodic if it has no subformula of the form \( \varphi \mathcal{S} \psi \) or \( \varphi \mathcal{U} \psi \) with more than one free variable. The set of all monodic formulas will be denoted by \( \mathcal{VT} \).

The result obtained in [11] states (roughly) that, if we take a fragment \( \mathcal{VT}' \) of \( \mathcal{VT} \) whose underlying first-order (non-temporal) part is decidable, then \( \mathcal{VT}' \) is decidable over \( \langle \mathbb{N}, < \rangle \), \( \langle \mathbb{Q}, < \rangle \), and some other flows of time. Examples of \( \mathcal{VT}' \) include:

- the monadic monodic fragment \( \mathcal{VT}_{\mathcal{M}} \),
- the one-variable fragment \( \mathcal{VT}_1 \),
- the two-variable monodic fragment \( \mathcal{VT}_{2\mathcal{M}} \),
- the packed monodic fragment \( \mathcal{VT}_{F\mathcal{M}} \).

Here, \( \mathcal{VT}_{F\mathcal{M}} \) is the fragment of \( \mathcal{VT} \) in which quantification is restricted to patterns \( \exists \tilde{y} (\gamma \wedge \varphi) \), where \( \tilde{y} \) is a tuple of variables, every free variable of \( \varphi \) is free in \( \gamma \) as well, and the ‘guard’ \( \gamma \) is a conjunction of atomic and existentially quantified atomic formulas such that for any two free variables \( x_1, x_2 \) of \( \gamma \), there is a conjunct of \( \gamma \) in which \( x_1, x_2 \) both occur free. This definition is based on the packed fragment of first-order logic, defined by Marx in [13].

\[ \text{real}_c = \forall x \bigcup_{t \in \psi} \bigwedge_{t \in c} \exists x \bigwedge_{t \in \psi} \psi(x) \]

is true in some (finite) first-order structure.

A quasimodel for a \( \mathcal{VT}_{F\mathcal{M}} \)-sentence \( \varphi \) (based on \( \mathcal{B} = \langle \mathbb{W}, < \rangle \)) is a triple \( \Omega = \langle \mathcal{B}, q, \mathcal{R} \rangle \), where \( q \), \( q \) a state function, is a map associating with each \( w \in \mathcal{W} \) a realisable state candidate \( q(w) \) for \( \varphi \), and \( \mathcal{R} \) is a set of runs — functions in \( \prod_{w \in \mathcal{W}} q(w) \) satisfying the following conditions:

- every \( r \in \mathcal{R} \) is coherent and saturated — that is,
for every $\psi_1 \cup \psi_2 \in \text{sub}_S \varphi$ and every $w \in W$, we have $\overline{\psi_1 \cup \psi_2} \in r(w)$ iff there is $v > w$ such that $\overline{\psi_2} \in r(v)$ and $\overline{\psi_1} \in r(u)$ for all $u \in (w, v)$, and

- for every $\psi_1 \cap \psi_2 \in \text{sub}_S \varphi$ and every $w \in W$, we have $\overline{\psi_1 \cap \psi_2} \in r(w)$ iff there is $v < w$ such that $\overline{\psi_2} \in r(v)$ and $\overline{\psi_1} \in r(u)$ for all $u \in (v, w)$;

- and for every $w \in W$ and every $t \in q(w)$, there exists a run $r \in \mathcal{R}$ such that $r(w) = t$.

The following general theorem provides upper bounds for the computational complexity of the satisfiability problem for decidable monodic fragments over the flow of time $\langle \mathbb{N}, < \rangle$.

**Theorem 4.1.** Let $\mathcal{QTL}'$ be a sublanguage of $\mathcal{QTL}_1$.

(i) Suppose that there is an algorithm which, given a state candidate $C$ for a $\mathcal{QTL}'$-sentence $\varphi$, can recognise whether $C$ is (finitely) realisable using exponential space in the length of $\varphi$. Then the satisfiability problem for $\mathcal{QTL}'$ in models over $\langle \mathbb{N}, < \rangle$ (with finite domains) is decidable in EXPSPACE.

(ii) Suppose that there is an algorithm which, given a state candidate $C$ for a $\mathcal{QTL}'$-sentence $\varphi$, can recognise whether $C$ is (finitely) realisable in deterministic double exponential time in the length of $\varphi$. Then the satisfiability problem for $\mathcal{QTL}'$ in models over $\langle \mathbb{N}, < \rangle$ (with finite domains) is decidable in 2EXPTIME.

**Proof.** Without loss of generality (see, e.g., [6]) we can consider only $S$-free formulas.

(i) We present a non-deterministic EXPSPACE satisfiability checking algorithm for $\mathcal{QTL}'$-sentences which is similar to that of [22]. Theorem 24 of [11] states that a $\mathcal{QTL}'$-sentence $\varphi$ is satisfiable over $\langle \mathbb{N}, < \rangle$ iff there is a ‘balloon-like’ quasimodel $\Omega = \langle \mathbb{N}, < \rangle, q, \mathcal{R} \rangle$, where $q(l_1 + n) = q(l_1 + l_2 + n)$ for some fixed $l_1, l_2$ ($l_2 > 0$) and every $n \in \mathbb{N}$, and both $l_1, l_2$ are double exponential in the length $\ell(\varphi)$ of $\varphi$.

Thus, given a $\mathcal{QTL}'$-sentence $\varphi$, the algorithm guesses the length of the prefix $l_1$ and the period $l_2$ of the quasimodel to be built. Then at every step $i$ it guesses a state candidate $q(i)$ and checks whether $q(i)$ is realisable and suitable for the quasimodel (in the sense that we have enough runs). Note that $q(i)$ can be represented using exponential space in $\ell(\varphi)$. The former test requires no more than exponential space in $\ell(\varphi)$, and the latter one can be done in deterministic polynomial time in the length of $q(i)$, so again using exponential space.

It is to be noted that this algorithm needs to store at most three state candidates at every step (previous $q(i - 1)$, current $q(i)$, and the beginning of the loop $q(l_1)$). It also needs to keep the list of unfilled eventualities (formulas of the form $\psi_1 \cup \psi_2$) for every type of the current state candidate. Therefore, the presented non-deterministic algorithm requires only an exponential amount of space. By [20], there is an equivalent deterministic algorithm that runs in exponential space.

(ii) The proof is similar to that of (i). The difference is in the algorithm for checking realisability of state candidates, which now uses alternation and runs in exponential space in the length of the formula. The existence of such an algorithm follows from the fact that 2EXPTIME coincides with AEXPSPACE [4].

The argument for finite domains is similar. □

**Theorem 4.2.** (i) The satisfiability problem for the languages $\mathcal{QTL}'_{\infty}$, $\mathcal{QTL}'_1$ and $\mathcal{QTL}'_2$ in models over the flow of time $\langle \mathbb{N}, < \rangle$ (with arbitrary or only finite domains) is EXPSPACE-complete.

(ii) The satisfiability problem for $\mathcal{TFA}$ in models over $\langle \mathbb{N}, < \rangle$ (with arbitrary or only finite domains) is 2EXPTIME-complete.

**Proof.** The lower bounds follow from Theorem 3.1 and [9].

To establish the upper bounds for the case of arbitrary domains, we apply Theorem 4.1 and use the formula $\text{real}_C$ stating that $C$ is realisable. Although the length of $\text{real}_C$ is exponential in the length $\ell(\varphi)$ of $\varphi$, using its specific structure one can show that for the monadic, one- and two-variable fragments the realisability test can be carried out by non-deterministic algorithms that run in exponential space in $\ell(\varphi)$ (see [3]), and by a deterministic algorithm in double exponential time in $\ell(\varphi)$ for the packed fragment [8].

The upper bounds for the case of finite domains follow in the same way from Theorem 4.1 and the fact that all the considered first-order fragments have the finite model property (and thus realisability coincides with finite realisability). □

**Acknowledgements**

The work of I. Hodkinson, R. Kontchakov and M. Zakharyaschev was partially supported by U.K. EPSRC grant no. GR/R45369/01; M. Zakharyaschev was also supported by EPSRC grant no. GR/R42474/01.

**References**


