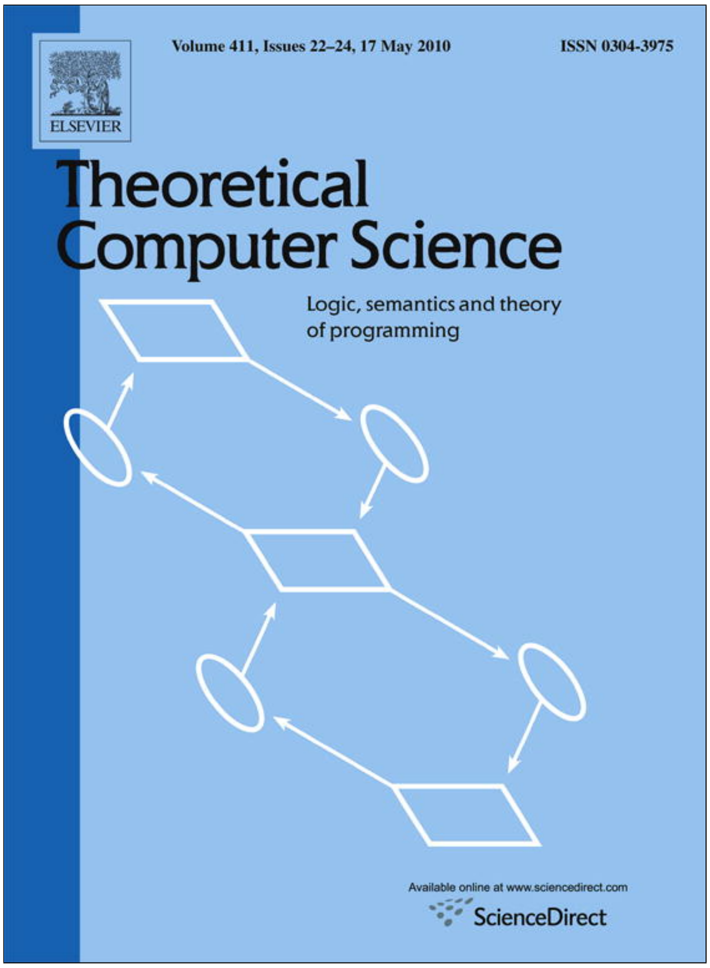
Volume 411, Issues 22–24, 17 May 2010
ISSN 0304-3975
ELSEVIER
Theoretical Computer Science
Logic, semantics and theory of programming
Available online at www.sciencedirect.com
ScienceDirect

Theoretical Computer Science 411 (2010) 2331–2342

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Complexity of metric temporal logics with counting and the Pnueli modalities

Alexander Rabinovich *

The Blavatnik School of Computer Science, Tel Aviv University, 69978, Israel

ARTICLE INFO

Article history:
Received 27 November 2008
Received in revised form 7 July 2009
Accepted 2 March 2010
Communicated by J. Tiuryn

Keywords:
Real time temporal logics
Complexity
Expressive power

ABSTRACT

The common metric temporal logics for continuous time were shown to be insufficient, when it was proved in Hirshfeld and Rabinovich (1999, 2007) [7,12] that they cannot express a modality suggested by Pnueli. Moreover, no temporal logic with a finite set of modalities can express all the natural generalizations of this modality. The temporal logic with counting modalities (*TLC*) is the extension of until–since temporal logic $TL(\mathbf{U}, \mathbf{S})$ by "counting modalities" $C_n(X)$ and $\overleftarrow{C}_n$ ($n \in \mathbb{N}$); for each n the modality $C_n(X)$ says that X will be true at least at n points in the next unit of time, and its dual $\overleftarrow{C}_n(X)$ says that X has happened n times in the last unit of time. In Hirshfeld and Rabinovich (2006) [11] it was proved that this temporal logic is expressively complete for a natural decidable metric predicate logic. In particular the Pnueli modalities $Pn_k(X_1, \ldots, X_k)$, "there is an increasing sequence $t_1, \ldots, t_k$ of points in the unit interval ahead such that X_i holds at t_i", are definable in *TLC*.

In this paper we investigate the complexity of the satisfiability problem for *TLC* and show that the problem is PSPACE complete when the index of C_n is coded in unary, and EXPSPACE complete when the index is coded in binary. We also show that the satisfiability problem for the until–since temporal logic extended by the Pnueli modalities is PSPACE complete.

1. Introduction

The temporal logic that is based on the two modalities "Since" and "Until" is popular among computer scientists as a framework for reasoning about a system evolving in time. By Kamp's theorem [4,13] this logic has the same expressive power as the first-order monadic logic of order, whether the system evolves in discrete steps or in continuous time. We will denote this logic by $TL(\mathbf{U}, \mathbf{S})$.

For systems evolving in discrete steps, this logic seems to supply all the expressive power needed. This is not the case for systems evolving in continuous time, as the logic cannot express metric properties like: "X will happen within one unit of time". The most straightforward extension which allows one to express metric properties is to add modality which says that "X will happen exactly after one unit of time". Unfortunately, this logic is undecidable. Over the years different decidable extensions of $TL(\mathbf{U}, \mathbf{S})$ were suggested. Most extensively researched was *MITL* [1,2,5]. Other logics are described in [3,6,14,18]. Hirshfeld and Rabinovich introduced the language *QTL* (quantitative temporal logic) [8–10], which extends the until–since temporal logic by two modalities: $\Diamond_1 X$ and $\overleftarrow{\Diamond}_1 X$. The formula $\Diamond_1 X$ (respectively $\overleftarrow{\Diamond}_1 X$) expresses that "$X$ will be true at some point during the next unit of time" (respectively, "X was true at some point during the previous unit of time").

* Tel.: +972 36405388.
E-mail address: rabinoa@post.tau.ac.il.

0304-3975/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2010.03.017

QTL and *MITL* have the same expressive power, which indicates that they capture a natural fragment of what can be said about the system which evolve in time. These "first generation" metric extensions of $TL(\mathbf{U}, \mathbf{S})$ can be called *simple metric temporal logics*.

A. Pnueli was probably the first person to question if these simple logics are expressive enough for our needs. The conjecture that they cannot express the property "X and then Y will both happen in the coming unit of time" is usually referred to as "Pnueli's conjecture" [2,18].

Hirshfeld and Rabinovich [7,12] proved Pnueli's conjecture, and strengthened it significantly. To do this, we defined for every natural k the "*Pnueli modality*" $Pn_k(X_1, \ldots, X_k)$, which states that there is an increasing sequence $t_1, \ldots, t_k$ of points in the unit interval ahead such that X_i holds at t_i. We also defined the weaker "*Counting modalities*" $C_k(X)$ which state that X is true at least at k points in the unit interval ahead (so that $C_k(X) = Pn_k(X, \ldots, X)$). To deal with the past we define also the dual past modality, $\overleftarrow{Pn}_k(X_1, \ldots, X_k)$: there is a decreasing sequence $t_1, \ldots, t_k$ of points in the previous unit interval such that X_i holds at t_i, and $\overleftarrow{C}_k(X)$ which state that X was true at least at k points in the previous unit interval.

This yields a sequence of temporal logics TLP_n ($n \in \mathbb{N}$), where TLP_n is the standard temporal logic, with "Until" and "Since", and with the addition of the k-place modalities Pn_k and $\overleftarrow{Pn}_k$ for $k \leq n$. Similarly, TLC_n is the extension of $TL(\mathbf{U}, \mathbf{S})$ with the addition of modalities C_k and $\overleftarrow{C}_k$ for $k \leq n$. We note also that TLP_1 is just the logic *QTL* and it represents the simple metric logics.

Let *TLP* be the union of TLP_n and *TLC* be the union of TLC_n.

We proved in [7,12] that:

1. The sequence of temporal logics TLP_n is strictly increasing in expressive power. In particular, $C_{n+1}(X)$ is not expressible in TLP_n.
2. *TLP* and *TLC* are decidable and have the same expressive power. Moreover, they are expressively equivalent to a natural decidable fragment of first-order logic.

In this paper we investigate the complexity of the satisfiability problem for *TLP* and *TLC*. In [17] it was shown that $TL(\mathbf{U}, \mathbf{S})$ is PSPACE complete. In [7,10] we provided a polynomial satisfiability preserving translation from *QTL* to $TL(\mathbf{U}, \mathbf{S})$ and derived PSPACE completeness of *QTL*.

In this paper we first prove that the satisfiability problem for *TLP* is PSPACE complete.

When one write a *TLC* formula there are two natural possibilities: to write index n of C_n in unary or in binary. We show that the satisfiability problem for *TLC* is PSPACE complete when the index of C_n is coded in unary, and EXPSPACE complete when the index is coded in binary.

Our results holds both when the interpretation of temporal variable is arbitrary and when we assume that they satisfy the finite variability assumption (FVA) which states that no variable changes its truth-value infinitely many times in any bounded interval.

In [12] we proved that there is no temporal logic L with finitely many modalities definable in the monadic second-order logic expanded by $+1$ function such that over the reals L is at least as expressive as *TLC*. Our conjecture was that this result can be extended to the non-negative reals. Our proofs refute this conjecture.

The paper is organized as follows: In Section 2, we recall definitions and previous results. In Section 3, we prove PSPACE completeness for *TLP* and as a consequence obtain PSPACE completeness for *TLC* under the unary coding of indexes. In Section 4, EXPSPACE completeness for *TLC* under the binary coding of indexes is proved. Section 5 contains complexity results for related more succinct temporal logics. Section 6 discusses the expressive power of *TLC* and shows that our results provide a negative answer to a conjecture from [12].

An extended abstract of this paper was published in [16].

2. Preliminaries

First, we recall the syntax and semantics of temporal logics and how temporal modalities are defined using truth tables, with notations adopted from [4,9].

Temporal logics use logical constructs called "modalities" to create a language that is free from quantifiers.

The syntax of a *Temporal Logic* has in its vocabulary a countably infinite set of *variables* $\{X_1, X_2, \ldots\}$ and a possibly infinite set $B = \{O_1^{l_1}, O_2^{l_2}, \ldots\}$ of *modality names* (sometimes called "temporal connectives" or "temporal operators") with prescribed arity indicated as superscript (we usually omit the arity notation). $TL(B)$ denotes the *temporal logic based on modality set B* (and B is called the *basis* of $TL(B)$). Temporal formulas are built by combining atoms (the variables X_i) and other formulas using Boolean connectives and modalities (with prescribed arity). Formally, the syntax of $TL(B)$ is given by the following grammar:

$$\phi ::= X_i | \phi_1 \wedge \phi_2 | \phi_1 \vee \phi_2 | \phi_1 \leftrightarrow \phi_2 | \neg \phi_1 | O_i(\phi_1, \phi_2, \ldots, \phi_{l_i}).$$

We will use (in our metalanguage) S, X, Y, Z to range over variables.

A structure for Temporal Logic, in this work, is the non-negative real line with monadic predicates $\mathcal{M} = \langle \mathbb{R}^+, <, S_1, S_2, \ldots \rangle$, where the predicate S_i are the interpretation in $\mathcal{M}$ of the variable S_i. (All our complexity results can be easily adopted to the models over the whole real line $\mathbb{R}$.) Every modality O^k is interpreted in the structure $\mathcal{M}$ as an operator

$O^k_{\mathcal{M}} : [\mathbb{P}(\mathbb{R}^+)]^k \to \mathbb{P}(\mathbb{R}^+)$ which assigns "the set of points where $O^k[A_1, \ldots, A_k]$ holds" to the k-tuple $\langle A_1, \ldots, A_k\rangle \in \mathbb{P}(\mathbb{R}^+)^k$. ($\mathbb{P}(\mathbb{R}^+)$ denotes the set of all subsets of $\mathbb{R}^+$.) Once every modality corresponds to an operator the semantics is defined by structural induction:

- for atomic formulas: $\mathcal{M}, t \models S$ iff $t \in S$
- for Boolean combinations the definition is the usual one
- for $O^k(\varphi_1, \ldots, \varphi_k)$

$$\mathcal{M}, t \models O^k(\varphi_1, \ldots, \varphi_k) \quad \text{iff} \quad t \in O^k_{\mathcal{M}}(A_{\varphi_1}, \ldots, A_{\varphi_k})$$

where $A_\varphi = \{\tau : \mathcal{M}, \tau \models \varphi\}$.

For the modality to be of interest the operator O should reflect some intended connection between the sets A_{φ_i} of points satisfying φ_i and the set of points $O[A_{\varphi_1}, \ldots, A_{\varphi_k}]$. The intended meaning is usually given by a formula in an appropriate predicate logic:

Truth Tables: A formula $\overline{O}(t, X_1, \ldots X_k)$ in the predicate logic L is a *Truth Table* for the modality O^k if for every structure $\mathcal{M}$

$$O_{\mathcal{M}}(A_1, \ldots, A_k) = \{\tau : \mathcal{M} \models \overline{O}[\tau, A_1, \ldots, A_k]\}.$$

2.1. Since–until temporal logic

The modalities *until* and *since* are most commonly used in temporal logic for computer science. They are defined through the following truth tables:

- The modality $X\mathbf{U}\,Y$, "X *until* Y", is defined by

$$\psi(t_0, X, Y) \equiv \exists t_1(t_0 < t_1 \wedge Y(t_1) \wedge \forall t(t_0 < t < t_1 \to X(t))).$$

- The modality $X\mathbf{S}\,Y$, "X *since* Y", is defined by

$$\psi(t_0, X, Y) \equiv \exists t_1(t_0 > t_1 \wedge Y(t_1) \wedge \forall t(t_1 < t < t_0 \to X(t))).$$

Reynolds [17] proved.

Theorem 2.1. *The satisfiability problem for TL*($\mathbf{U}$, $\mathbf{S}$) *over the reals is PSPACE complete.*

We will use standard abbreviations. E.g., $\Diamond X$ – sometimes in the future X holds – abbreviates $True\mathbf{U}X$; $\Box X$ – always in the future X holds – abbreviates $\neg(True\mathbf{U}\neg X)$; the past modalities $\overleftarrow{\Diamond} X$ – "X happened in the past", and $\overleftarrow{\Box} X$ – "X have been always true", are defined similarly. The modality *always* acts like the universal quantifier and is defined as

$$always(X): \qquad \overleftarrow{\Box} X \wedge X \wedge \Box X.$$

$Llim(X)$ and $Rlim(X)$ abbreviate the formulas

$$Llim(X): \qquad \neg(\neg X\mathbf{S}True)$$
$$Rlim(X): \qquad \neg(\neg X\mathbf{U}True).$$

$Llim(X)$ holds at t if t is a left limit of X, i.e., for every $t_1 < t$ there is an X in the interval (t_1, t). $Rlim(X)$ holds at t if t is a right limit of X.

2.2. Three metric temporal logics

We recall the definitions of three temporal logics: Quantitative Temporal Logic—*QTL*, Temporal Logic with Counting—*TLC* and Temporal Logic with the Pnueli modalities—*TLP*.

The logic *QTL* in addition to modalities $\mathbf{U}$ and $\mathbf{S}$ has two modalities $\Diamond_1 X$ and $\overleftarrow{\Diamond}_1 X$. These modalities are defined by the tables with free variable t_0:

$$\Diamond_1 X: \qquad \exists t((t_0 < t < t_0 + 1) \wedge X(t))$$
$$\overleftarrow{\Diamond}_1 X: \qquad \exists t((t_0 - 1 < t < t_0) \wedge X(t))\,).$$

In [7] it was proved.

Theorem 2.2. *The satisfiability problem for QTL is PSPACE complete.*

The logic *TLP* is the extension of $TL(\mathbf{U},\mathbf{S})$ by an infinite set of modalities $Pn_k(X_1,\ldots,X_k)$ and $\overleftarrow{Pn}_k(X_1,\ldots,X_k)$. These modalities are defined by the tables with free variable t_0:

$$Pn_k(X_1,\ldots,X_k): \qquad \exists t_1\ldots\exists t_k\left(t_0<t_1<\cdots<t_k<t_0+1\wedge\bigwedge_{i=1}^{k}X_i(t_i)\right)$$

$$\overleftarrow{Pn}_k(X_1,\ldots,X_k): \qquad \exists t_1\ldots\exists t_k\left(t_0-1<t_1<\cdots<t_k<t_0\wedge\bigwedge_{i=1}^{k}X_i(t_i)\right).$$

Finally, the logic *TLC* – the temporal logic with counting modalities – is the extension of $TL(\mathbf{U},\mathbf{S})$ by an infinite set of modalities $C_k(X)$ and $\overleftarrow{C}_k(X)$. These modalities are defined by the tables with free variable t_0:

$$C_k(X): \qquad \exists t_1\ldots\exists t_k\left(t_0<t_1<\cdots<t_k<t_0+1\wedge\bigwedge_{i=1}^{k}X(t_i)\right)$$

$$\overleftarrow{C}_k(X): \qquad \exists t_1\ldots\exists t_k\left(t_0-1<t_1<\cdots<t_k<t_0\wedge\bigwedge_{i=1}^{k}X(t_i)\right).$$

We recall the terminology that is used when comparing the expressive power of languages.

Let $\mathcal{C}$ be a class of structures and let L and L' be temporal logics.

1. L is *at least as expressive as* L' over a class $\mathcal{C}$ if for every formula φ of L' there is a formula ψ in L such that for every structure $\mathcal{M}$ in $\mathcal{C}$ and for every $\tau\in\mathcal{M}$: $\mathcal{M},\tau\models\varphi$ iff $\mathcal{M},\tau\models\psi$.
2. L and L' are *expressively equivalent* over $\mathcal{C}$ if L is at least as expressive as L' over $\mathcal{C}$ and L' is at least as expressive as L over $\mathcal{C}$.

We deal here with the temporal logics over the class of non-negative real numbers. We will say "L is *at least as expressive as* (respectively, is expressively equivalent to) L' if L is *at least as expressive as* (respectively, is expressively equivalent to) L' over this class.

The following theorem from [12] compares the expressive power of *TLP*, *TLC* and *QTL*.

Theorem 2.3 (*Comparing the Expressive Power*). *TLP and TLC are expressively equivalent. TLP and TLC are strictly more expressive than QTL.*

2.3. Size of formulas

Usually the *size* of a formula is defined as its length (string representation) or the size of its directed acyclic graph representation (DAG). The logics *TLC* and *TLP* have infinite sets of modalities and therefore we have to agree how to code the names of modalities. There are two natural possibilities: to write index k of C_k and Pn_k in unary or in binary. For *TLP* formulas this decision affects the size of the formulas up to a constant factor, and, therefore, it is not important. For *TLC* formulas the binary coding might be exponentially shorter than the unary coding. Our main results show that the satisfiability problem for *TLC* is PSPACE complete when the index of C_k is coded in unary, and EXPSPACE complete when the index is coded in binary.

Note that there might be an exponential gap in the size of a DAG representation of a formula and its length. Our proofs of upper bounds will be given for DAG representation (and hence the bounds are valid for string representations). Our proofs of lower bounds will be given for string representations (and hence the bounds are valid for DAG representation).

3. TLP is PSPACE complete

Theorem 3.1. *The satisfiability problem for TLP is PSPACE complete.*

The PSPACE hardness immediately follows from PSPACE hardness for the satisfiability problem for $TL(\mathbf{U},\mathbf{S})$ which is a subset of *TLP*. Below we prove that the satisfiability problem is in PSPACE.

A structure $\mathcal{M}$ is called *proper* if it is an expansion of $\langle\mathbb{R}^+,<,\mathbb{N},Even,Odd\rangle$ by unary predicates. Here $\mathbb{N}$, $Even$, and Odd are the sets of natural, even and odd numbers; these sets will be denoted by predicate names N, E, O.

In contrast to the fact that *TLP* is much more expressive than *QTL* over the class of all real structures and over the class of finite variability structures [7,9], we are going to show that they are expressively equivalent over the class of proper structures. Moreover, there is a polynomial meaning preserving (over the proper structures) translation from *TLP* to *QTL*.

Lemma 3.2. 1. *For every k there is a QTL formula $\Psi_k(X_1,\ldots,X_k,N,E,O)$ which is equivalent over the proper structures to $Pn_k(X_1,\ldots,X_k)$. Furthermore, the size of Ψ_k is less than $100k^2$.*

2. *For every k there is a QTL formula $\overleftarrow{\Psi}_k(X_1,\ldots,X_k,N,E,O)$ which is equivalent over the proper structures to $\overleftarrow{Pn}_k(X_1,\ldots,X_k)$. Furthermore, the size of Ψ_k is less than $100k^2$.*

Proof. (1) For $i \le j \le k$ define formulas $\phi_{i,j}$ as follows:

$$\phi_{i,i} := (\neg N)\mathbf{U}X_i$$
$$\phi_{i,i+l+1} := (\neg N)\mathbf{U}(X_i \wedge \neg N \wedge \phi_{i+1,i+l+1}).$$

It is clear that the size of $\phi_{i,j}$ is less than $10(j-i+1)$ and that $\phi_{i,j}$ holds at t iff there are $t < t_i < t_{i+1} < \cdots < t_j \le n$, where n is the smallest integer greater than t, such that $\wedge_{l=i}^{j} X_l(t_l)$. Similarly, there are formulas $\overleftarrow{\phi}_{i,j}$ such that $\overleftarrow{\phi}_{i,j}$ holds at t iff there are $t > t_j > \cdots > t_i > n$, where n is the largest integer less than t, and $\wedge_{l=i}^{j} X_l(t_l)$ holds.

The formula Ψ_k which is equivalent to Pn_k over the proper structures can be defined as the disjunction of the following formulas:

1. $\phi_{1,k}$ – "$\phi_{1,k}$ holds at t if there are $t < t_1 < t_2 < \cdots < t_k \le n$, where n is the smallest integer greater than t, such that $\wedge_{l=1}^{k} X_l(t_l)$".
2. $\bigvee_{n=1}^{k-1}(\neg N)\mathbf{U}E \wedge \phi_{1,n} \wedge \Diamond_1((\neg N)\mathbf{S}E \wedge \overleftarrow{\phi}_{n+1,k})$ – this covers the case when t is in an interval $[2m-1, 2m]$ for some integer m. The nth disjunct says that $\wedge_{l=1}^{n} X_l(t_l)$ holds for $t < t_1 < \cdots < t_n \le 2m$ and in the interval $(2m, 2m+1)$ there are $t_{n+1} < \cdots < t_k < t+1$ such that $\wedge_{l=n+1}^{k} X_l(t_l)$ holds.
3. $\bigvee_{n=1}^{k-1}(\neg N)\mathbf{U}O \wedge \phi_{1,n} \wedge \Diamond_1((\neg N)\mathbf{S}O \wedge \overleftarrow{\phi}_{n+1,k})$ – this is similar to the previous disjunct, but deals with t in the intervals $[2m, 2m+1]$, where m is an integer.

This proves (1). The proof of (2) is similar. □

Corollary 3.3. *TLP and QTL are expressively equivalent over the class of proper structures. Furthermore, for every TLP formula φ there is a QTL formula ψ which is equivalent to φ over the proper structures and $|\psi|$ is $O(|\varphi|^2)$.*

Proof. We define a meaning preserving translation Tr from *TLP* to *QTL*.

1. For variables $Tr(X) := X$.
2. If *op* is a Boolean connective $Tr(\varphi_1 op \varphi_2) := Tr(\varphi_1) op Tr(\varphi_2)$.
3. For until and since modalities:
$$Tr(\varphi_1\mathbf{U}\varphi_2) := (Tr(\varphi_1))\mathbf{U}(Tr(\varphi_2)) \quad \text{and} \quad Tr(\varphi_1\mathbf{S}\varphi_2) := (Tr(\varphi_1))\mathbf{S}(Tr(\varphi_2)).$$
4. $Tr(Pn_k(\varphi_1,\ldots,\varphi_k))$ is obtained by substitution of $Tr(\varphi_i)$ instead of X_i in Ψ_k. Similarly, $Tr(\overleftarrow{Pn}_k(\varphi_1,\ldots,\varphi_k))$ is obtained by substitution of $Tr(\varphi_i)$ instead of X_i in $\overleftarrow{\Psi}_k$.

It is clear that φ is equivalent to $Tr(\varphi)$ over the proper structures. In Ψ_k and in $\overleftarrow{\Psi}_k$ every variable appears at most k times, therefore the size (of the DAG representation) of $Tr(\varphi)$ is $O(|\varphi|^2)$. □

The next lemma shows that the set of proper structures is definable by a *QTL* formula.

Lemma 3.4. *There is a QTL formula $PROPER(Y, Z, U)$ such that $\mathbb{R}^+, t \models PROPER(N, E, O)$ iff N is the set of natural numbers, and E and O are the sets of even and odd numbers.*

Proof. (1) Let $\mathrm{Nat}(Y)$ be the conjunction of the following formulas:

1. $\overleftarrow{\Box} False \rightarrow Y$ – "Y holds at zero".
2. $always(Y \rightarrow \Box_1 \neg Y)$ – "If Y holds at t then $\neg Y$ holds at all points in $(t, t+1)$".
3. $always(\neg Y \rightarrow \Diamond_1 Y)$ – "If Y does not hold at t then Y holds at some point in $(t, t+1)$".

It is clear that the set of naturals is unique set that satisfies $\mathrm{Nat}(Y)$.

(2) Let $\mathrm{EVEN}(Y, Z)$ be the conjunction of

1. $\mathrm{Nat}(Y)$ – "Y is the set of the natural numbers".
2. $always(Z \rightarrow Y)$ – "Z is a subset of the natural numbers".
3. $\overleftarrow{\Box} False \rightarrow Z$ – "Z holds at zero".
4. $always(Z \rightarrow (\neg Y)U(Y \wedge \neg Z))$ – " if Z holds at a natural number n then it does not hold at the next natural number".
5. $always(\neg Z \wedge Y \rightarrow (\neg Y)U(Y \wedge Z))$ – " if Z does not hold at a natural number n then it holds at the next natural number".

It is clear that $\mathrm{EVEN}(N, E)$ holds iff N is the set of naturals and E is the set even numbers.

$PROPER(Y, Z, U)$ can be defined as $\mathrm{EVEN}(Y, Z) \wedge always(U \leftrightarrow (Y \wedge \neg Z))$. □

Finally, to complete the proof of Theorem 3.1, observe that a *TLP* formula φ is satisfiable iff φ is satisfiable over a proper structure iff $PROPER(N, E, O) \wedge \varphi$ is satisfiable iff the *QTL* formula $PROPER(N, E, O) \wedge \psi$ is satisfiable, where ψ is constructed as in Corollary 3.3. Since, the satisfiability problem for *QTL* is in PSPACE we obtain that the satisfiability problem for *TLP* is in PSPACE and this completes the proof of Theorem 3.1.

As a consequence we obtain the following corollary.

Corollary 3.5. *The satisfiability problem for TLC is PSPACE complete under the unary coding.*

Proof. Note that $C_k(X)$ is equivalent to $Pn_k(X, X, \ldots, X)$. The translation from *TLC* to *TLP* based on this equivalence is linear in the size of DAG representation. Hence, by Theorem 3.1, *TLC* is in PSPACE.

The PSPACE hardness immediately follows from PSPACE hardness for the satisfiability problem for $TL(\mathbf{U}, \mathbf{S})$ which is a subset of *TLC*. □

4. EXPSPACE completeness for *TLC*

Theorem 4.1. *The satisfiability problem for TLC is EXPSPACE complete under the binary coding.*

The upper bound immediately follows from Corollary 3.5. Below we prove that the satisfiability problem is EXPSPACE hard. For every Turing Machine M which works in space 2^n and every input x of length n we construct a *TLC* formula $Acc_{M,x}$ which is satisfiable iff M accepts x. Moreover, $Acc_{M,x}$ is computable from M and x in polynomial time. This proves EXPSPACE hardness with respect to the polynomial reductions.

A one-tape deterministic Turing machine M is $(Q, q_0, q_{acc}, q_{rej}, \Gamma, b, \nu)$, where Q is the set of states, $q_0, q_{acc}, q_{rej} \in Q$ are initial, accepting and rejecting states, Γ is the alphabet, $b \in \Gamma$ is the blank symbol and $\nu : ((Q \setminus \{q_{acc}, q_{rej}\}) \times \Gamma) \rightarrow (Q \times \Gamma \times \{-1, 0, 1\})$ is the transition function. If the head is over a symbol σ and M is in a state q and $\nu(q, \sigma) = \langle q'\sigma', d\rangle$, then M replaces σ by σ' changes its state to q' and moves d cells to the right (if $d = -1$ then it moves one cell left). There is no transition from the accepting and rejecting states.

A configuration (or an instantaneous description) is a member of $\Gamma^* Q \Gamma^+$ and represents a complete state of the Turing machine.

Let $\alpha = xq\sigma y$ be a configuration, where $\sigma \in \Gamma,\ x, y \in \Gamma^*$ and $q \in Q$. We define $tape(\alpha) = x\sigma y$, and $state(\alpha) = q$. It describes that for $i \leq |tape(\alpha)|$, the ith cell of the tape contains the ith symbol of $tape(\alpha)$ and all other cells contain blank; the control state is q and the head is over the symbol σ at the position $|x| + 1$.

We deal with Turing machines which use at most 2^n tape cells on inputs of length n. A configuration α is an n-configuration if $tape(\alpha)$ has 2^n symbols. Hence, a computation of M on an input $x = x_1 \cdots x_n$ of length n can be described by a sequence $\alpha_1\alpha_2 \cdots$ of n-configurations, where $\alpha_1 = q_0x_1x_2 \cdots x_nb^{2^n-n}$ is the initial n-configuration for the input x.

For n-configurations α and β we write $\alpha \rightarrow_M \beta$ if β is obtained from α according to the transition function of M. Whenever M is clear from the context we will write $\alpha \rightarrow \beta$. Note that if $\alpha \rightarrow \beta$ then $tape(\alpha)$ and $tape(\beta)$ have the same length.

A *computation sequence* is a sequence of configurations $\alpha_1 \cdots \alpha_k$ for which $\alpha_i \rightarrow \alpha_{i+1}$, $1 \leq i < k$. A configuration β is *reachable* from a configuration α if there exists a computation sequence $\alpha_1 \cdots \alpha_k$ with $\alpha = \alpha_1$ and $\beta = \alpha_n$.

Acceptance conditions. A configuration α is an accepting (respectively, rejecting) configuration if $state(\alpha)$ is accepting (respectively, rejecting) state. A computation sequence $\alpha_1 \cdots \alpha_m$ is accepting (respectively, rejecting) if α_m is accepting (respectively, rejecting).

We are going to encode computations of M over proper structures, i.e., over expansions of $\langle \mathbb{R}^+, <, \mathbb{N}, Even, Odd\rangle$ by monadic predicates. All these predicates will have finite variability and the EXPSPACE lower bound holds both under the finite variability and arbitrary interpretations. We will denote by $\mathcal{M}$ an expansion of $\langle \mathbb{R}^+, <, \mathbb{N}\rangle$ by unary predicates.

From now on we fix a Turing machine M with the alphabet $\{0, 1, b\}$ of space complexity $\leq 2^n$. W.l.o.g. we assume that M never moves to the left of the first input cell. All definitions and constructions below will be for this M.

Let $\alpha_1, \ldots, \alpha_k$ be a sequence of n-configurations (not necessary a computation sequence). The ith configuration α_i will be encoded on the interval $(i - 1, i)$ with integer end-points as follows: The interval will contain 2^n points $\tau_{i,j}$ such that $i - 1 < \tau_{i,1} < \tau_{i,2} < \cdots < \tau_{i,2^n} < i$ and the predicate T will hold exactly at these points in the interval. All other predicates described below will be subsets of T. Predicates A_0, A_1 and A_b will partition T; $\tau_{i,j}$ will be in A_0 (respectively, in A_1 or in A_b) if the jth tape symbol of α_i is 0 (respectively 1, or blank). Predicates S_q for $q \in Q$ are interpreted in $(i-1, i)$ as follows: $\tau_{i,j} \in S_q$ if q is the state of α_i and the head is over the jth tape symbol.

Definition 4.2. Let $\mathcal{M}$ be an expansion of $\langle \mathbb{R}^+, <, \mathbb{N}\rangle$ by predicates T, A_0, A_1, A_b, S_q for $q \in Q$. For $i \in \mathbb{N}$, we say that the interval $[i, i + 1]$ of $\mathcal{M}$ represents a legal n-configuration if

1. it contains 2^n points in T and all these points are inside $(i, i + 1)$.
2. A_0, A_1 and A_b partition T.
3. $\cup_{q\in Q} S_q \subseteq T$ and there is exactly one $q \in Q$ such that $S_q \cap [i, i + 1]$ is a singleton and for all $q' \neq q$, the set $S_{q'} \cap [i, i + 1]$ is empty.

The following lemma is easy. We use there $\overrightarrow{S}$ for the tuple of predicate names $\langle S_q : q \in Q\rangle$.

Lemma 4.3. 1. *There is a TLC formula $\varphi_0(N, T, A_0, A_1, A_b, \overrightarrow{S})$ which holds in a structure $\mathcal{M}$ iff there is $l \in \mathbb{N}$ such that for every $i < l$ the interval $[i, i+1]$ represents a legal n-configuration, the configuration represented in the interval $[l-1, l]$ is accepting or rejecting, and no $\tau \geq l$ is in $T \cup A_0 \cup A_1 \cup A_b \cup \cup_{q\in Q} S_q$. Furthermore, the size of φ_0 is $O(n)$.*

2. *For every $x = x_1 \cdots x_n \in \{0, 1\}^n$, there is a formula $INIT_x$ which holds in a structure $\mathcal{M}$ iff the interval $[0, 1]$ represents the initial n-configuration σ_0 with input x. Furthermore, the size of $INIT_x$ is $O(n)$.*

Our next task is to specify that the configuration represented in an interval $[i, i + 1]$ is obtained from the configuration represented in $[i - 1, i]$ according to the transition function of M. We have to express (1) the head is moved properly and update the symbols under the head correctly and (2) all other symbols are unchanged.

The next lemma shows that the cells numbered from 1 to 2^n can be succinctly described by their binary representations.

Lemma 4.4. *There is a formula $\varphi_1(N, T, B_1, \ldots, B_n)$ such that if for every $i \in \mathbb{N}$ the interval $(i, i + 1)$ contains at most 2^n points from T then $\mathcal{M}, 0 \models \varphi_1$ iff* *for every $i \in \mathbb{N}$ and $\tau \in (i, i+1)$: if τ is the jth occurrence of T in this interval then $\tau \in B_l$ iff the lth bit of the binary representation of $j-1$ is one.*

Furthermore, the size of φ_1 is $O(n^2)$.

Proof. φ_1 is $always(\psi_1)$, where ψ_1 is the conjunction of

1. $\vee B_l \rightarrow (T \wedge \neg N)$ — B_l are subsets of $T \setminus \mathbb{N}$.
2. $N \wedge (\neg N)\mathbf{U}(T \wedge \neg N) \rightarrow ((\neg N)\mathbf{U}(T \wedge \bigwedge_{l=1}^{n} \neg B_l))$ — the first occurrence of T in $(i, i+1)$ has binary representation $00\cdots 0$, i.e., is not in $\cup B_l$.
3. $[T \wedge \neg N \wedge (\neg N)\mathbf{U}(T \wedge \neg N)] \rightarrow \bigwedge_k \gamma_k$, where γ_k is
$$\left(\neg B_k \wedge \bigwedge_{m=k+1}^{n} B_m\right) \rightarrow \left((\neg T)\mathbf{U}(T \wedge B_k \wedge \bigwedge_{m=k+1}^{n} \neg B_m) \wedge \bigwedge_{m=1}^{k-1}(B_m \rightarrow (\neg T)\mathbf{U}(T \wedge B_m))\right).$$
The formula γ_k expresses that if τ is not the last occurrence of T in $(i, i+1)$ and its binary code has 0 at kth place and 1 at places $k+1, \ldots, n$ then the code of the next occurrence of T has 1 at kth place and zero at places $k+1, \ldots, m$ and both occurrences have the same bit in the binary code at places $1, \ldots, k-1$. □

Now we can express that the head moves properly, state is updated correctly and the type symbol under the head is updated correctly.

Lemma 4.5. *There is a formula φ_2 such that if $\mathcal{M}$ represents a terminating sequence of configurations $\alpha_1, \ldots, \alpha_l$ and $\mathcal{M}, 0 \models \varphi_1$, then*[1] *$\mathcal{M}, 0 \models \varphi_2$ iff*

for every $i < l$ if in α_i the head is over symbol σ at position j and the state is q and $\nu(q, \sigma) = \langle q'.\sigma', d\rangle$ then the state in the α_{i+1} is q' the head is at the position $j+d$, the symbol at position j is σ'.

Furthermore, the size of φ_2 is $O(n^2)$.

Proof. Let $\nu(q, \sigma) = \langle q'.\sigma', 1\rangle$ and let $S := \vee_{q_1 \in Q} S_{q_1}$
Let $\psi_{q,\sigma}$ be the conjunction of

1. The head moved one position to the right: $\left(S_q \wedge A_\sigma\right) \rightarrow \bigwedge_{k=1}^{n} \gamma'_k$ where γ'_k is obtained from γ_k after substitution of S instead T (see proof of Lemma 4.4).
2. The state and the symbols under the head were updated correctly:
$$(S_q \wedge A_\sigma) \rightarrow (\neg S)\mathbf{U}(S_{q'} \wedge (\neg T)\mathbf{S}A_{\sigma'}).$$

When $\nu(q, \sigma) = \langle q'.\sigma', 0\rangle$ and $\nu(q, \sigma) = \langle q'.\sigma', -1\rangle$ the formula $\psi_{q,\sigma}$ is defined similarly.
The desirable formula φ_2 can be defined as $always(\bigwedge_{q \notin \{q_{acc}, q_{rej}\}} \bigwedge_\sigma \psi_{q,\sigma})$. □

The creative part of our proof is to show how to express succinctly that the symbols not under the head are unchanged. In order to do this we introduce the following notion.

Assume that $\mathcal{M}$ represents a terminating sequence of configurations $\alpha_1, \ldots, \alpha_l$. Recall that $\tau_{i,j} \in \mathbb{R}^+$ is the jth occurrence of T in the interval $(i-1, i)$. We denote by $tape(\alpha_i)[j]$ the jth symbol of $tape(\alpha_i)$. We say that $\mathcal{M}$ is *well-timed* if for all $i < l$ and $j \leq 2^n$ and some positive $\epsilon_{i,j}$, $\delta_{i,j}$:

$$\tau_{i+1,j} = \begin{cases} 1 + \tau_{i,j} + \epsilon_{i,j} & \text{if } tape(\alpha_i)[j] \text{ is } 0 \\ 1 + \tau_{i,j} - \delta_{i,j} & \text{if } tape(\alpha_i)[j] \text{ is } 1 \\ 1 + \tau_{i,j} & \text{if } tape(\alpha_i)[j] \text{ is blank.} \end{cases} \tag{WT}$$

First observe

Lemma 4.6. *If $\alpha_1 \cdots \alpha_l$, is a terminating sequence of n-configuration, then there is a well-timed $\mathcal{M}$ which represents this sequence.*

Proof. Just choose $\tau_{1,j}$ as $\frac{j}{2^n+1}$ (for $j = 1, \ldots 2^n$) and choose $\epsilon_{i,j} = \delta_{i,j}$ as $\frac{1}{3l \times (2^n+1)}$. Define $\tau_{i+1,j}$ as in Eq. (WT). Our choice of $\epsilon_{i,j}, \delta_{i,j}$ ensures that $i - 1 < \tau_{i,1} < \tau_{i,2} < \cdots < \tau_{i,2^n} < i$ for all $i \leq l$. □

Lemma 4.7. *There is a formula φ_3 such that $\mathcal{M} \models \varphi_3$ iff $\mathcal{M}$ is a well-timed sequence of n-configurations. Furthermore, the size of φ_3 is $O(n)$.*

Proof. Let ψ be the conjunction of the following formulas

1. $A_b \rightarrow (C_{2^n-1}(T) \wedge Llim(C_{2^n}(T)) \wedge Rlim(C_{2^n}(T)))$

[1] Until the end of this section φ_1 is the formula from Lemma 4.3. The scope of the definition of φ_2 from this lemma and formulas φ_3 and φ_4 from the following lemmas extends to the end of this section.

2. $A_1 \to (C_{2^n}(T) \wedge Llim(C_{2^n+1}(T)) \wedge Rlim(C_{2^n}(T)))$
3. $A_0 \to (C_{2^n-1}(T) \wedge Llim(C_{2^n}(T)) \wedge Rlim(C_{2^n-1}(T)))$.

(Recall that $Llim(X)$ (respectively, $Rlim(X)$) holds at t iff t is a left limit (respectively, a right limit of X), see Section 2.1.)

Let $\mathcal{M}'$ represents an n-configuration α_i in $[i.i+1]$ and has 2^n occurrences of T in $[i+1, i+2]$ all the occurrences inside $(i+1, i+2)$. The crucial observation is that Eq. (WT) holds iff $\mathcal{M}', \tau \models \psi$ for every $\tau \in [i, i+1]$.

From ψ it is easy to construct φ_3. Just express that φ_0 holds, and ψ holds at all points except the points of the interval where the last configuration is represented. $\square$

We are now ready to specify that if a symbols is not under the head then in the next configuration it will be unchanged.

Lemma 4.8. *There is a formula φ_4 such that if $\mathcal{M}$ represents a well-timed terminating sequence of n-configurations $\alpha_1, \ldots, \alpha_l$ and $\mathcal{M}, 0 \models \varphi_1$, then $\mathcal{M}, 0 \models \varphi_4$ iff*

for every $i < l$ if in α_i the head is at position j, then $tape(\alpha_i)[m] = tape(\alpha_{i+1})[m]$ for every $m \neq j$.

Furthermore, the size of φ_4 is $O(n)$.

Proof. Let ψ be the conjunction of the following formulas

1. $A_b \to (\overleftarrow{C}_{2^n-1}(T) \wedge Llim(\overleftarrow{C}_{2^n}(T)) \wedge Rlim(\overleftarrow{C}_{2^n}(T)))$
2. $A_1 \to (\overleftarrow{C}_{2^n}(T) \wedge Rlim(\overleftarrow{C}_{2^n+1}(T)) \wedge Llim(\overleftarrow{C}_{2^n}(T)))$
3. $A_0 \to (\overleftarrow{C}_{2^n-1}(T) \wedge Rlim(\overleftarrow{C}_{2^n}(T)) \wedge Llim(\overleftarrow{C}_{2^n-1}(T)))$.

Assume that $\mathcal{M}$ is well-timed. Hence, Eq. (WT) holds. Then ψ holds at $\tau_{i+1,m}$ iff $tape(\alpha_i)[m] = tape(\alpha_{i+1})[m]$.

The head is at position m in σ_i iff at $\tau_{i+1,m}$ the following formula γ holds:

$$\gamma := \bigwedge_k (B_k \leftrightarrow ((\neg N)\mathbf{S}(N \wedge ((\neg N)\mathbf{S}(\vee_{q\in Q} S_q \wedge B_k))).$$

Indeed, this formula says that B_k holds at τ iff in the previous interval B_k holds at the (unique) position where $\vee_{q\in Q} S_q$ holds (this is the position of the head in the configuration σ_i). Hence, $T \to ((\neg\gamma) \to \psi)$ holds in every point of the interval $[i+1, i+2]$ iff $tape(\alpha_i)[m] = tape(\alpha_{i+1})[m]$ for every m different from the head position in σ_i.

Finally, φ_4 should express that $T \to ((\neg\gamma) \to \psi)$ holds at all points except the points of the interval $[0, 1]$. Note that $t \in [0, 1]$ iff $\overleftarrow{\Diamond}(N \wedge \overleftarrow{\Diamond} N)$ does not hold at t. Hence, φ_4 can be defined as follows: $\varphi_4 := (\overleftarrow{\Diamond}(N \wedge \overleftarrow{\Diamond} N)) \to (T \to ((\neg\gamma) \to \psi))$. $\square$

From Lemmas 4.3–4.5, 4.7 and 4.8 we obtain:

Lemma 4.9. *For every $x \in \{0, 1\}^n$ let $Acc_{M,x}$ be $INIT_x \wedge \varphi_0 \wedge \varphi_1 \wedge \varphi_2 \wedge \varphi_3 \wedge \varphi_4 \wedge \Diamond q_{acc}$. Then $\mathcal{M}, 0 \models Acc_{M,x}$ iff $\mathcal{M}$ represents a well-timed accepting computation sequence of M on x.*

The size $Acc_{M,x}$ is polynomial in the size of x, therefore this lemma together with Lemma 4.6 implies EXPSPACE hardness of the satisfiability problem for *TLC*.

Remark 4.10. 1. Our proof of the EXPSPACE lower bound require an unbounded set of variables. We do not know what happens if the set of variables is assumed to be bounded?

2. In the proof of Theorem 4.1, we assumed that the Turing machine M works over the binary alphabet and in space 2^n. Formally speaking, for the EXPSPACE-completeness one needs to consider space $2^{f(n)}$, for any polynomial f. The proof clearly works pretty well for any such f.

5. Further results

5.1. QTLI and MITL logics and their complexity

In this subsection we recall the syntax and semantics of temporal logics *QTLI* and *MITL* and results on the complexity of their satisfiability problem.

Often in the literature the temporal logics with modalities $\Diamond_{(m,n)}(X)$ for integers $m < n$ are considered. These modalities are defined by the truth tables:

$$\Diamond_{(m,n)}(X): \quad \exists t((t_0 + m < t < t_0 + n) \wedge X(t)).$$

The logic *QTLI* in addition to modalities **U** and **S** has infinity many modalities $\Diamond_{(m,n)}(X)$ for all integers $m < n$. The logic $QTLI_0$ is a fragment of *QTLI*; it has in addition to modalities **U** and **S** the modalities $\Diamond_{(0,n)}(X)$, $\Diamond_{(-n,0)}(X)$ for all natural n.

The logics *QTL*, $QTLI_0$ and *QTLI* have the same expressive power (under arbitrary interpretations). However, there is an exponential succinctness gap (under the binary coding) between *QTL* and $QTLI_0$ and between $QTLI_0$ and *QTLI*.

The logic *MITL* was introduced in [1]; it also has the same expressive power as *QTL*. Syntactically, *MITL* modalities are obtained by decorating **U** and **S** modalities by the non-singular interval with integer end-points. For $m < n$ the modality $\mathbf{U}_{(m,n)}$ is defined by the truth tables:

$$\exists t(t_0 + m < t < t_0 + n \wedge X_2(t) \wedge \forall t_2(t_0 < t_2 < t \rightarrow X_1(t_2)).$$

The truth table of the modality $\mathbf{U}_{[m,n)}$ (respectively, of $\mathbf{U}_{(m,n]}$) is obtained from the above truth table by replacing the first (respectively, the second) occurrence of $<$ by $\leq$; the truth table of the modality $\mathbf{U}_{[m,n]}$ is obtained from the above truth table by replacing the first two occurrences of $<$ by $\leq$. The decorated "since" modalities $\mathbf{S}_{(m,n)}$, $\mathbf{S}_{(m,n]}$, $\mathbf{S}_{[m,n)}$ and $\mathbf{S}_{[m,n]}$ are defined similarly.

The logic *MITL* in addition to modalities **U** and **S** has infinity many modalities $\mathbf{U}_{(m,n)}$, $\mathbf{U}_{(m,n]}$, $\mathbf{U}_{[m,n)}$ and $\mathbf{U}_{[m,n]}$, $\mathbf{S}_{(m,n)}$, $\mathbf{S}_{(m,n]}$, $\mathbf{S}_{[m,n)}$ and $\mathbf{S}_{[m,n]}$. for all integers $m < n$.

The logic *MITL₀* is the fragment of *MITL* which uses only the modalities where one of the end-points of the intervals is zero.

Observe that for $0 \leq m < n$ the following equivalences hold:

$$\Diamond_{(m,n)}(X) \leftrightarrow True\mathbf{U}_{(m,n)}X$$
$$\Diamond_{(-m,-n)}(X) \leftrightarrow X\mathbf{S}_{(-m,-n)}True.$$

The meaning preserving polynomial translations from *QTLI* to *MITL* and from $QTLI_0$ to $MITL_0$ can be easily defined from these equivalences.

Observe that for $0 \leq m < n$

$$X\mathbf{U}_{(m,n)}Y \leftrightarrow \Diamond_{(m,n)}(Y) \wedge \Box_{(0,n]}X\mathbf{U}Y.$$

The meaning preserving polynomial translations form *MITL* to *QTLI* and from $MITL_0$ to $QTLI_0$ can be easily defined from the above equivalence and similar equivalences for the half closed and closed intervals and for the "since" modal operators [9].

The next theorem characterizes the complexity of these logics [1].

Theorem 5.1. 1. *The satisfiability problem for* $QTLI_0$ *is PSPACE complete under the binary coding.*
2. *The satisfiability problem for QTLI is EXPSPACE complete under the binary coding.*

Theorem 5.1 was proved for the finite variability interpretation in [1] and for arbitrary interpretation in [8,15].

5.2. Logics TLPI and TLCI and their complexity

In this section we consider temporal logics with the modalities $C_k^{(n,m)}(X)$ and $Pn_k^{(m,n)}(X_1,\ldots,X_k)$ for the integers $m < n$. These modalities are defined by the truth tables with free variable t_0:

$$Pn_k^{(m,n)}(X_1,\ldots,X_k): \quad \exists t_1\ldots\exists t_k\left(t_0+m<t_1<\cdots<t_k<t_0+n\wedge\bigwedge_{i=1}^{k}X_i(t_i)\right)$$

$$C_k^{(m,n)}(X): \quad \exists t_1\ldots\exists t_k\left(t_0+m<t_1<\cdots<t_k<t_0+n\wedge\bigwedge_{i=1}^{k}X(t_i)\right).$$

Note that Pn_k is equivalent to $Pn_k^{(0,1)}$ and C_k is equivalent to $C_k^{(0,1)}$

We consider the following temporal logics:

$$TLPI := TL(\mathbf{U},\mathbf{S},\{Pn_k^{(m,n)} : m<n\})$$
$$TLPI_0 := TL(\mathbf{U},\mathbf{S},\{Pn_k^{(0,n)}, Pn_k^{(-n,0)} : 0<n\})$$
$$TLCI := TL(\mathbf{U},\mathbf{S},\{C_k^{(m,n)} : m<n\})$$
$$TLCI_0 := TL(\mathbf{U},\mathbf{S},\{C_k^{(0,n)}, C_k^{(-n,0)} : 0<n\}).$$

All these logics are expressively equivalent to *TLC* [11]. We investigate the complexity of the satisfiability problems for these logics under the unary and binary codings. Under the unary (respectively, binary) coding all the numbers which occur in the superscripts and subscripts of these modalities are coded in unary (respectively, in binary). This section contains proofs of the results summarized in Table 1.

5.3. The complexity of the logics under the unary coding

The PSPACE hardness under the unary coding immediately follows from the PSPACE hardness of the satisfiability problem for $TL(\mathbf{U},\mathbf{S})$.

The PSPACE upper bound for the unary coding follows from simple linear translation of all these logic into *TLPI* and the following lemma:

Table 1
The complexity of the satisfiability problem.

Logic	Unary coding	Binary coding
$TLPI_0$	PSPACE complete	PSPACE complete
$TLPI$	PSPACE complete	EXPSPACE complete
$TLCI_0$	PSPACE complete	EXPSPACE complete
$TLCI$	PSPACE complete	EXPSPACE complete

Lemma 5.2. *The satisfiability problem for TLPI is in PSPACE under the unary coding.*

Proof. For $i \le j \le k$ let $\phi_{i,j}(X_i, \dots, X_j, N)$ be defined as in the proof of Lemma 3.2. Recall that the size of $\phi_{i,j}$ is less than $10(j-i+1)$ and that $\phi_{i,j}$ holds at t iff there are $t < t_i < t_{i+1} < \dots < t_j \le l$, where l is the smallest integer (element of N) greater than t, such that $\wedge_{s=i}^{j} X_s(t_s)$.

For $d > 0$ a structure $\mathcal{M}$ is called *d-proper* if it is an expansion of $\langle \mathbb{R}^+, <, \mathbb{N}, Even, Odd, Z_0^d, \dots, Z_{d-1}^d\rangle$ by unary predicates. Here $\mathbb{N}$, $Even$, and Odd are the sets of natural, even and odd numbers and for $r \in \{0, \dots, d-1\}$ the predicate Z_r^d is interpreted as the set of natural numbers equal r modulo d.

Set

$$\phi_{i,j,\ r}^d := (\neg N)\mathbf{U}Z^d_{(r+1) mod\ d} \wedge \phi_{i,j}\{Z_r^d/N\},$$

where $\phi_{i,j}\{Z_r^d/N\}$ is obtained from $\phi_{i,j}$ when Z_r^d is substituted instead of N.

Set

$$\phi_{i,j}^d := \bigvee_{r=0}^{d-1} \phi_{i,j,\ r}^d.$$

Note that the size of $\phi_{i,j}^d$ is polynomial in (unary representation of) k and d. Observe that $\phi_{i,j}^d$ holds at t in a d-proper structure iff there are $t < t_i < t_{i+1} < \dots < t_j \le l+d$, where l is the smallest integer greater than t, such that $\wedge_{s=i}^{j} X_s(t_s)$.

Similarly, for $i \le j \le k$ and $d \in \mathbb{N}$ there are formulas $\overleftarrow{\phi_{i,j}^d}(X_i, \dots, X_j, N, Z_0^d, \dots, Z_{d-1}^d)$ such that $\overleftarrow{\phi_{i,j}^d}$ holds at t if there are $t > t_j > \dots > t_i > l-d$, where l is the largest integer less than t, and $\wedge_{s=i}^{j} X_s(t_s)$ holds. Furthermore, the size of $\overleftarrow{\phi_{i,j}^d}$ is polynomial in (unary representation of) k and d.

We are going to construct a *QTL* formula $\Psi_k^{(m,n)}(X_1, \dots, X_k, N, E, O, Z_0^d, \dots, Z_{d-1}^d)$ which is equivalent over the d-proper structures to $Pn_k^{(m,n)}(X_1, \dots, X_k)$, where $d = n - m$. Furthermore, the size of $\Psi_k^{(m,n)}$ is polynomial in $m,\ n,\ k$.

Our construction of $\Psi_k^{(m,n)}(X_1, \dots, X_k)$ depends whether m is even or odd and whether n is even or odd. These four cases are similar and we describe below only the case when m and n are even.

Let Ev be defined as $(\neg N)\mathbf{U}E$. Ev holds at t iff t is in an interval $[2l-1, 2l)$ for a natural l.

Let A_i be defined as

$$A_i := Ev \wedge \Diamond_{(m,m+1)}(Ev \wedge \phi_{1,i}) \wedge \Diamond_{(n-1,n)}(\neg Ev \wedge \overleftarrow{\phi_{i+1,k}^d}), \quad \text{where } d = n - m.$$

Note that in the case when m and n are even, $Ev(t)$ iff $Ev(t+m)$ iff $Ev(t+n)$.

Assume that $Ev \wedge \neg N$ holds at t. Then A_i holds at t iff for $j = \lceil t+m \rceil$ the following conditions hold:

1. there are $t_1 < t_2 < \dots < t_i$ in the interval $(t+m, j]$ such that $X_1(t_1) \wedge X_2(t_2) \wedge \dots \wedge X_i(t_i)$ and
2. there are $t_{i+1}, t_{i+2} < \dots < t_k$ in the interval $(j, t+n)$ such that $X_{i+1}(t_{i+1}) \wedge X_{i+2}(t_{i+2}) \wedge \dots \wedge X_k(t_k)$.

Hence, $Ev \wedge \neg N$ implies that $\bigvee A_i$ is equivalent (over $(n-m)$-proper structures) to $Pn_k^{(m,n)}$.

Note that A_i are *QTLI* formulas of size polynomial in $n,\ m,\ k$. We can transform A_i into an equivalent *QTL* formula A_i' using the following equivalences from [8]. For $0 < s \in \mathbb{N}$:

$$\Diamond_{(s,s+1)} Y \leftrightarrow \underbrace{(\Diamond_1 \Box_1) \cdots (\Diamond_1 \Box_1)}_{s \text{ times}} \Diamond_1 Y$$

$$\Diamond_{(-s-1,-s)} Y \leftrightarrow \underbrace{(\overleftarrow{\Diamond}_1 \overleftarrow{\Box}_1) \cdots (\overleftarrow{\Diamond}_1 \overleftarrow{\Box}_1)}_{s \text{ times}} \overleftarrow{\Diamond}_1 Y.$$

(These equivalences can be proved by the induction on s.) Let $A := \bigvee A_i'$. Then $Ev \wedge \neg N \rightarrow (A \leftrightarrow Pn_k^{(m,n)})$ holds over the $(n-m)$-proper structures and the size of A is polynomial in $n,\ m,\ k$.

Similarly, one can construct *QTL* formulas B and C of size polynomial in $n,\ m,\ k$ such that $(\neg Ev \wedge \neg N) \rightarrow (B \leftrightarrow Pn_k^{(m,n)})$ and $N \rightarrow (C \leftrightarrow Pn_k^{(m,n)})$ hold over the $(n-m)$-proper structures.

Let $\Psi_k^{(m,n)}(X_1, \ldots, X_k)$ be defined as $(A \wedge B \wedge C)$. Then, $\Psi_k^{(m,n)} \leftrightarrow Pn_k^{(m,n)}$ hold over the $(n-m)$-proper structures and the size of $\Psi_k^{(m,n)}$ is polynomial in $n,\ m,\ k$.

As in Lemma 3.4, for every $d > 0$ there is a *QTL* formula $PROPER_d$ such that $\mathcal{M}, t \models PROPER_d$ iff $\mathcal{M}$ is a d-proper structure. Moreover, the size of $PROPER_d$ is polynomial in d.

Now, relying on the equivalence of $\Psi_k^{(m,n)}$ and $Pn_k^{(m,n)}$, we can easily translate every *TLPI* formula α into an equi-satisfiable *QTL* formula of the size polynomial in the unary coding of α. This together with Theorem 2.2 implies that the satisfiability problem *TLPI* is in PSPACE under the unary coding. □

Since $TLPI_0 \subset TLPI$, $TLCI_0 \subset TLCI$, and *TLCI* can be translated into *TLPI* applying equivalence $C_k(X) \leftrightarrow Pn_k(X, X, \ldots, X)$, we obtain that the satisfiability problems for $TLPI_0$, $TLCI_0$ and *TLCI* are also in PSPACE.

5.4. The complexity of the logics under the binary coding

Lemma 5.3. *The satisfiability problem for $TLPI_0$ is PSPACE complete under the binary coding.*

Proof. The PSPACE hardness immediately follows from the PSPACE hardness of the satisfiability problem for $TL(\mathbf{U}, \mathbf{S})$.

Let $l = \lceil \log_2 2m \rceil$. We say that a structure $\mathcal{M}$ is m-structure if it is an expansion of $\langle \mathbb{R}^+, <, \mathbb{N}, B_1^m, \ldots, B_l^m, P_m, P_{2m} \rangle$ by unary predicates, where $P_m, P_{2m}, B_i^m \subset \mathbb{N}$ and are defined as follows:

$i \in P_m$ iff i is a multiple of m

$i \in P_{2m}$ iff i is a multiple of $2m$

$i \in B_j^m$ iff jth bit of the binary representation of $i\ mod(2m)$ is one.

First observe that there is a *QTL* formula $Structure_m$ such that $Structure_m$ is satisfiable in $\mathcal{M}$ iff $\mathcal{M}$ is an m-structure. Moreover, the size of $Structure_m$ is polynomial in $\log m$. (The construction of these formulas is similar to the construction used in the proof of Lemma 4.4.)

We are going to construct a $QTLI_0$ formula Ψ_k^m of size polynomial in $k,\ \log m$ (i.e., polynomial in the size of $Pn_k^{(0,m)}(X_1, \ldots, X_k)$ under the binary coding) such that Ψ_k^m is equivalent to $Pn_k^{(0,m)}$ over the m-structures.

Note that there are $TL(\mathbf{U}, \mathbf{S})$ formulas $\chi_{i,j}$ for $i < j \le k$ such that

$\chi_{i,j}$ holds at t in an m-structure $\mathcal{M}$ iff there are $t < t_i < t_{i+1} < \cdots < t_j \le n$, where n is the smallest multiple of m greater than t, such that $\wedge_{s=i}^{j} X_s(t_s)$.

Similarly, there are formulas $\overleftarrow{\chi}_{i,j}$ such that

$\overleftarrow{\chi}_{i,j}$ holds at t if there are $t > t_j > \cdots > t_i > n$, where n is the largest multiple of m less than t, and $\wedge_{s=i}^{j} X_s(t_s)$ holds.

Let Ev_m be defined as $(\neg P_m)\mathbf{U}P_{2m}$. Ev_m holds at t iff t is in an interval $[2ml - m, 2ml)$ for a natural l.

Let $A_{k,m}$ be a $QTLI_0$ formula defined as follows:

$$A_{k,m} := \chi_{1,k} \vee \Diamond_{(0,m)}(\neg Ev_m \wedge \overleftarrow{\chi}_{1,k}) \vee \bigvee_{i=1}^{k} (\chi_{1,i} \wedge \Diamond_{(0,m)}(\neg Ev_m \wedge \overleftarrow{\chi}_{i+1,k})).$$

The size of $A_{k,m}$ is polynomial in $k,\ \log m$, and over the m-structures $Ev_m \to (A_{k,m} \leftrightarrow Pn_k^{(0,m)})$ holds.

Similarly, one can define $QTLI_0$ formulas $B_{k,m}$ of size polynomial in $k,\ \log m$, such that over the m-structures $\neg Ev_m \to (B_{k,m} \leftrightarrow Pn_k^{(0,m)})$ holds.

Now, Ψ_k^m can be defined as

$$\Psi_k^m := (Ev_m \to A_{k,m}) \wedge (\neg Ev_m \to B_{k,m}).$$

$\Psi_k^m \leftrightarrow Pn_k^{(0,m)}$ hold over the m-structures and the size of Ψ_k^m is polynomial in the size of $Pn_k^{(0,m)}$.

Relying on the equivalence of Ψ_k^m and $Pn_k^{(0,m)}$, we can easily transform every $TLPI_0$ formula α into an equivalent (over the m-structures) formula $\beta \in QTLI_0$ of the size polynomial in the binary coding of α.

Hence, α is satisfiable iff

$$\beta \wedge \bigwedge_{\{m \,:\, Pn_k^{(0,m)} \text{ occurs in } \alpha\}} Structure_m$$

is satisfiable. This together with Theorem 5.1(1) implies that the satisfiability problem $TLPI_0$ is in PSPACE under the binary coding. □

Lemma 5.4. *The satisfiability problem for TLPI, $TLCI_0$ and TLCI is EXPSPACE complete under the binary coding.*

Proof. The membership in EXPSPACE follows from the PSPACE upper bound under the unary coding for the satisfiability problem of these logics.

The EXPSPACE hardness for *TLPI* and *TLCI* follows from Theorem 5.1. The EXPSPACE hardness for $TLCI_0$ follows from Theorem 4.1. □

6. TLC and logics with finitely many modalities

Let $B = \{O_1^{l_1}, O_2^{l_2}, \ldots, O_k^{l_k}\}$ be a finite set of modality names, and assume that every modality in B has a truth table definable in the monadic second-order logic of order with $\lambda x.x + 1$ function (we denote this logic by MLO^{+1}). MLO^{+1} is a very expressive (and undecidable) logic, and most of the modalities considered in the literature can be easily formalized in it. We proved in [12] that there is n (which depends on B) such that C_n is not expressible over the reals by a $TL(B)$ formulas. Hence, there is no temporal logic L which is at least as expressive as TLC over the reals, which has a finite set of modalities with truth tables in MLO^{+1}.

Our conjecture was that this result can be extended to the non-negative real line. However, the results of Section 3 refute this conjecture.

Indeed, let L be the temporal logic with the modalities **U**, **S**, $\Diamond_1$ $\overleftarrow{\Diamond}_1$, **nat** and **even**, where **nat** and **even** are zero-arity modalities interpreted as the sets of natural and even numbers respectively Corollary 3.3 shows that TLP, TLC and QTL are expressively equivalent over the class of proper structures, i.e., over the expansions of $\langle \mathbb{R}^+, <, \mathbb{N}, Even, Odd\rangle$ by unary predicates.

Hence, L is at least as expressive (over the class of non-negative real structures) as TLC. Over the non-negative reals, the modalities **nat** and **even** are easily definable by truth tables in MLO^{+1} (see Lemma 3.4). This refutes the conjecture.

Similarly to Corollary 3.3 one can show that TLP, TLC and QTL are expressively equivalent over the class of the expansions of $\langle \mathbb{R}, <, \mathbb{Z}, Even\rangle$ by unary predicates, where $\mathbb{Z}$ and $Even$ are the sets of integers and even numbers. Hence, QTL with two additional zero-arity modalities for the set of integers and for the set of even numbers is at least as expressive as TLC. However, over the reals, these two modalities are not definable by truth tables in MLO^{+1}.

Acknowledgements

I am grateful to Yoram Hirshfeld for his insightful comments. I would like to thank the anonymous referees for their helpful suggestions.

References

[1] R. Alur, T. Feder, T.A. Henzinger, The benefits of relaxing punctuality, Journal of the ACM 43 (1996) 116–146.
[2] R. Alur, T.A. Henzinger, Logics and models of real time: a survey, in: de Bakker, et al. (Eds.), Real Time: Theory and Practice, in: LNCS, vol. 600, 1992, pp. 74–106.
[3] Baringer H. Barringer, R. Kuiper, A. Pnueli, A really abstract concurrent model and its temporal logic, in: Proceedings of the 13th POPL, 1986, pp. 173–183.
[4] D.M. Gabbay, I. Hodkinson, M. Reynolds, Temporal Logics, vol. 1, Clarendon Press, Oxford, 1994.
[5] T.A. Henzinger, It's about time: real-time logics reviewed, in: Concur 98, in: Lecture Notes in Computer Science, vol. 1466, 1998, pp. 439–454.
[6] T.H Henzinger, J.F Raskin, P.Y Schobbens, The regular real time languages, in: ICALP 1998, pp. 580–591.
[7] Y. Hirshfeld, A. Rabinovich, A framework for decidable metrical logics, in: Proc. 26th ICALP Colloquium, in: LNCS, vol. 1644, 1999, pp. 422–432.
[8] Y. Hirshfeld, A. Rabinovich, Quantitative temporal logic, in: Computer Science Logic 1999, in: LNCS, vol. 1683, Springer Verlag, 1999, pp. 172–187.
[9] Y. Hirshfeld, A. Rabinovich, Logics for real time: decidability and complexity, Fundamental Information 62 (1) (2004) 1–28.
[10] Y. Hirshfeld, A. Rabinovich, Timer formulas and decidable metric temporal logic, Information and Computation 198 (2) (2005) 148–178.
[11] Y. Hirshfeld, A. Rabinovich, An expressive temporal logic for real time, in: MFCS 2006, in: Springer LNCS, vol. 4162, 2006, pp. 492–504.
[12] Y. Hirshfeld, A. Rabinovich, Expressiveness of Metric modalities for continuous time, Logical Methods in Computer Science 3 (1) (2007).
[13] H. Kamp, Tense logic and the theory of linear order, Ph.D. Thesis, University of California L.A., 1968.
[14] Z. Manna, A. Pnueli, Models for reactivity, Acta Informatica 30 (1993) 609–678.
[15] C. Lutz, D. Walther, F. Wolter, Quantitative temporal logics over the reals: PSPACE and below, Information and Computation 205 (1) (2007) 99–123.
[16] A. Rabinovich, Complexity of metric temporal logics with counting and the pnueli modalities, in: FORMATS 2008, in: Springer LNCS, vol. 5215, 2008, pp. 93–108.
[17] M. Reynolds, The complexity of the temporal logic with until over general linear time, manuscript, 1999.
[18] T. Wilke, Specifying time state sequences in powerful decidable logics and time automata, in: Formal Techniques in Real Time and Fault Tolerance Systems, in: LNCS, vol. 863, 1994, pp. 694–715.