



On the Complexity of Model Checking for Temporal Equilibrium Logic

Oscar Barreca¹ , Thomas Eiter² , and Davide Soldà² 

¹ Center for Scalable Data Analytics and AI (ScaDS.AI) Dresden/Leipzig,
Technische Universität Dresden, Dresden, Germany
oscar.barreca@tu-dresden.de

² Technische Universität Wien, Vienna, Austria
{thomas.eiter, davide.solda}@tuwien.ac.at

Abstract. Given an interpretation I and a formula φ written in a logic L , the problem of *model checking* (MC) for L is the problem of determining whether I is a model of φ . In this work, the computational complexity of MC for *temporal equilibrium logic* (TEL), a temporal extension of ASP, is investigated. The complexity of the *universal* and the *existential variant* of MC for TEL over Kripke structures, and of MC over linear traces, is determined. The complexity of these problems when the set of temporal operators, the *temporal height* or the *implication height* of the input formula is restricted is also determined.

Keywords: Model checking · Temporal logics · Nonmonotonic logics · Complexity analysis

1 Introduction

Given a formula φ written in a logic L and an interpretation I for φ , the problem of *model checking* (MC) for L is the problem of deciding whether I is a model of φ . The MC problem is studied in the context of temporal logics, like *linear temporal logic* (LTL) [27] or *computation tree logic* (CTL) [14], where the interpretation I is taken to be a Kripke structure, but also for other logics and formalisms, like *circumscription* [13], *answer set programming* (ASP) [16], *default logic* [23] and SAT [7].

Motivated by the growing interest in temporal reasoning and planning, and by the emergence of practical tools [9, 11], the present work is dedicated to the study of the computational complexity of the MC problem for *temporal equilibrium logic* (TEL) [12], a temporal and nonmonotonic logic extending the nonmonotonic, but atemporal, *equilibrium logic* (EL) [24, 25] to the temporal setting. Given that EL extends the syntax of ASP to allow the free intermixing of logical operators while preserving the *stable model semantics* [18], TEL can ultimately be seen as a temporal extension of ASP.

Contributions. This work studies the complexity of deciding the TELKS problem, the MC problem for TEL over Kripke structures. The obtained complexity results justify the division of this problem into two variants, TELKS $\forall$, the universal variant of TELKS,

and $\text{TELKS}\exists$, the existential one. A restriction of TELKS over traces, TELT , is also studied.

Fragments of the $\text{TELKS}\exists$, $\text{TELKS}\forall$ and TELT problems are also studied by restricting the set of allowed temporal operators, the *temporal height* (the number of nested temporal operators) and the *implication height* of a formula (the number of nested $\rightarrow$ operators). For $P \in \{\text{TELKS}\exists, \text{TELKS}\forall, \text{TELT}\}$, $P_m^k(O_1, \dots, O_r)$ denotes the MC problem P where the set of allowed temporal operators is $\{O_1, \dots, O_r\}$ and where the formulas have a temporal height of value at most m and an implication height of value at most k .

A summary of the results is given by Table 1. Similarly to LTL [27], deciding $\text{TELKS}\forall$ is PSPACE-complete. However, unlike LTL its restriction TELT over linear traces remains PSPACE-complete, instead of becoming polynomial-time tractable [15], and it remains so even under the $\text{TELT}_2^1(\circ, \mathbb{R})$ fragment. On the other hand, $\text{TELKS}\exists$ displays a strong similarity to the satisfiability problem for TEL [4], both on a complexity level, given that both problems are EXPSpace-complete, and on a conceptual one, given that both problems ask about the existence of a model under the TEL semantics.

A detailed exposition of the same results under a slightly different notation can be found in [2].

Table 1. Contributions of the work.

Problem	Complexity
$\text{TELKS}\forall$	PSPACE-complete (Theorem 1)
$\text{TELKS}\exists$	EXPSpace-complete (Theorem 4)
$\text{TELKS}\forall(\mathbb{R}), \text{TELKS}\forall(\circ, \mathbb{R}), \text{TELKS}\forall(\mathbb{U}, \mathbb{R})$	PSPACE-complete (Theorem 7)
$\text{TELKS}\exists(\mathbb{R}), \text{TELKS}\exists(\circ, \mathbb{R}), \text{TELKS}\exists(\mathbb{U}, \mathbb{R})$	EXPSpace-complete (Theorem 9)
$\text{TELT}, \text{TELT}_2^1$	PSPACE-complete (Theorem 11, Corollary 2)
$\text{TELT}(\circ, \mathbb{R}), \text{TELT}_2^1(\circ, \mathbb{R})$	PSPACE-complete (Theorem 13, Corollary 3)

Related Work. The complexity of the satisfiability problem for TEL has been studied by [4], while expressivity results by [5]; in particular, these two results provide a connection between TEL and $\exists\forall\text{LTL}$ [28]. Closely related to the present work, [10] discusses the complexity of satisfiability for a variant of TEL producing shorter traces by adopting a stronger notion of minimization, whereas [22] investigates MC for *quantified LTL* (QLTL). A different temporal extension of ASP, based on a notion of *reduct* rather than on EL, along with an analysis of its computational properties is given by [19]. Different *reduct*-based metric temporal extensions of ASP, including LARS [3], its weighted variant [17], and DatalogMTL [6] have been also studied.

2 Background

In this section, general notions of complexity theory and TEL are introduced. Knowledge of the *here-and-there* (HT) logic [20], of EL [24, 25] and of the Kripke semantics

for intuitionistic logic [21,29] is essential to understand TEL in detail, but for reasons of space no explanation of these topics is included here.

Complexity Theory. Given a language L , understood as a set of words of finite length, a *decision problem* is a subset P of L . Elements of P are called *positive instances*, whereas elements of its complement $\bar{P} \stackrel{\text{def}}{=} L \setminus P$ *negative instances*. If not specified beforehand, the *host language* L of P , consisting of both the positive and negative instances, is denoted by Σ_P .

Given two decision problems P_1 and P_2 , a *Karp reduction* is a function $f : \Sigma_{P_1} \rightarrow \Sigma_{P_2}$ such that $x \in P_1$ iff $f(x) \in P_2$ and such that $f(x)$ can be computed in polynomial time with respect to the size $|x|$ of x .

TEL. On an abstract level, TEL can be seen as the union of EL and LTL. On a more concrete level, however, TEL is the union of *temporal here-and-there* (THT), a temporal extension of HT, with EL, a nonmonotonic extension of HT.

TEL shares the same syntax of THT, which in turn shares the same syntax of LTL:

Definition 1 (THT syntax). *Given a set A of atomic symbols, a formula φ is a THT formula if described by the $\varphi \stackrel{\text{def}}{=} \perp \mid p \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \varphi \rightarrow \varphi \mid \bigcirc \varphi \mid \varphi \cup \varphi \mid \varphi \mathbb{R} \varphi$ grammar, where $p \in A$.*

In Definition 1, $\bigcirc$ is referred to as the *next* operator, $\cup$ as the *until* and $\mathbb{R}$ as the *release* one. On the basis of these basic operators, a few derived ones can be also introduced: $\neg \varphi \stackrel{\text{def}}{=} \varphi \rightarrow \perp$, $\top \stackrel{\text{def}}{=} \neg \perp$, $\varphi_1 \leftrightarrow \varphi_2 \stackrel{\text{def}}{=} (\varphi_1 \rightarrow \varphi_2) \wedge (\varphi_2 \rightarrow \varphi_1)$, $\Box \varphi \stackrel{\text{def}}{=} \perp \mathbb{R} \varphi$, and $\Diamond \varphi \stackrel{\text{def}}{=} \top \cup \varphi$. $\neg$ is the *default negation* operator, whereas $\Box$ and $\Diamond$ are the *always* and the *eventually* operators.

An interpretation for the LTL logic is a *trace* consisting of a finite or infinite sequence of atomic propositions from a countable set A . The length, or number of states, of an LTL trace T is denoted by $|T|$. If T is infinite, the length of T is denoted by ω . Given two traces T and U of equal length, $T \leq U$ if, for all i , $T_i \subseteq U_i$, and $T < U$ if $T \leq U$ and some i exists for which $T_i \subset U_i$.

Differently than for LTL, an interpretation for the THT logic is not just a single LTL trace, but a pair $M \stackrel{\text{def}}{=} (H, T)$ of LTL traces of finite or infinite length subject to the condition that $|H| = |T|$ and $H \leq T$. M is said to be a *total* THT interpretation if $H = T$, in which case M can be expressed as a single LTL trace T . The length of a THT trace M is defined as that of its constituting traces, in symbols $|M| \stackrel{\text{def}}{=} |H| = |T|$. By M_i , the pair (H_i, T_i) is meant.

Definition 2 (THT semantics). *A THT interpretation $M = (H, T)$ satisfies a THT formula φ at step i , written $M, i \models \varphi$, if it satisfies the following conditions:*

1. $M, i \not\models \perp$;
2. $M, i \models a$ if $a \in H_i$;
3. $M, i \models \varphi_1 \vee \varphi_2$ if $M, i \models \varphi_1$ or $M, i \models \varphi_2$;
4. $M, i \models \varphi_1 \wedge \varphi_2$ if $M, i \models \varphi_1$ and $M, i \models \varphi_2$;
5. $M, i \models \varphi_1 \rightarrow \varphi_2$ if:

- (a) $(H, T), i \models \varphi_1$ implies $(H, T), i \models \varphi_2$, and
- (b) $(T, T), i \models \varphi_1$ implies $(T, T), i \models \varphi_2$;
- 6. $M, i \models \bigcirc \varphi$ if $i+1 < |M|$ and $M, i+1 \models \varphi$;
- 7. $M, i \models \varphi_1 \cup \varphi_2$ if $\exists j \in [i..|M|)$ such that $M, j \models \varphi_2$, and $\forall j' \in [i..j), M, j' \models \varphi_1$;
- 8. $M, i \models \varphi_1 \mathbb{R} \varphi_2$ if $\forall j \geq i, M, j \models \varphi_2$, or $\exists k \in [i..|M|)$ such that $M, k \models \varphi_1$ and $\forall k' \in [i..k], M, k' \models \varphi_2$.

For an LTL trace T and a THT formula φ , $T \models \varphi$ is written in place of $T, 0 \models \varphi$. (Similarly for a THT trace (H, T) .) Additionally, $T, i \models_{\text{LTL}} \varphi$ is written in place of $T, i \models \varphi$ to make the LTL semantics explicit. (Similarly for THT.)

The semantics of THT from Definition 2 coincides with the one of LTL, but differs from it in the interpretation of the $\rightarrow$ operator, requiring a formula of the form $\varphi_1 \rightarrow \varphi_2$ to follow the same behavior of the HT logic. Under total traces the semantics of THT and LTL do however coincide:

Proposition 1 ([1]). *Given an LTL trace T of length λ , a THT formula φ and a time point $k \in [0..\lambda)$, $T, k \models_{\text{LTL}} \varphi$ iff $(T, T), k \models_{\text{THT}} \varphi$.*

The semantics of TEL is defined on top of the one of THT by adding a minimization process, analogous to the one that EL adds to HT but extended to the temporal case:

Definition 3. *A THT interpretation M is a temporal equilibrium model (TEM) of a THT formula φ if $M = (T, T)$ is total, $M \models_{\text{THT}} \varphi$ and $\nexists H < T$ for which $(H, T) \models_{\text{THT}} \varphi$.*

Since TEMs are, by definition, required to be total traces, they are typically expressed as a single LTL trace.

TEL and MC. Although typically interpreted over linear traces, in a MC context it is appropriate to interpret TEL over Kripke structures in about the same way as it is done for LTL [27].

Definition 4. *A Kripke structure is a tuple $K = (Q, q_0, \rightarrow, A, L)$, where Q is a finite set of states, $q_0 \in Q$ is the initial state, $\rightarrow \subseteq Q \times Q$ is a total transition relation between states, A is a set of propositional atoms, and $L : Q \rightarrow 2^A$ is a labeling function between states and sets of propositions.*

A relation $R \subseteq S \times S$ is said to be *total* if, for every $s \in S$, $\exists s' \in S$ such that $(s, s') \in R$. As an alternative notation, $q \rightarrow q'$ is written in place of $(q, q') \in \rightarrow$ to mean that state q has state q' as one of its successors in the relation $\rightarrow$.

Given a Kripke structure K , a *path* π within K is either a finite sequence $\pi \stackrel{\text{def}}{=} q_0 q_1 \cdots q_n$ of states from Q such that for all $i \in [0..n)$, $q_i \rightarrow q_{i+1}$; or an infinite sequence $\pi \stackrel{\text{def}}{=} q_0 q_1 q_2 \cdots$ of states from Q such that for all $i \geq 0$, $q_i \rightarrow q_{i+1}$. A *trace* within K is a sequence T of sets of atomic propositions obtained by a path from K . In other words, if $\pi = q_0 q_1 q_2 \cdots$ is an infinite path from K , then the infinite trace T obtained from π is $T \stackrel{\text{def}}{=} L(q_0) \cdot L(q_1) \cdot L(q_2) \cdot \cdots$. (Similarly for the case of a finite path.)

The complexity results from the upcoming sections justify the introduction of two different, albeit related, semantics for the interpretation of a TEL formula over a Kripke

structure. A Kripke structure K is said to be a TEM of a THT formula φ under the *universal semantics* if for all traces T in K , T is a TEM of φ . Similarly, K is said to be a TEM of φ under the *existential semantics* if there exists a trace T in K which is a TEM of φ .

3 The TELKS Problem

Given a Kripke structure K and a THT formula φ , the problem of MC for TEL, denoted TELKS, is the problem of determining whether K is a TEM of φ . As already anticipated, two distinct semantics exist for how to interpret whether K is a TEM of φ , giving rise to two distinct variants of the TELKS problem:

Definition 5 (TELKS $\forall$). *The universal variant of the MC problem for TEL over Kripke structures is the language $\text{TELKS}\forall \stackrel{\text{def}}{=} \{(K, \varphi)\}$, where K is a Kripke structure and φ a THT formula such that all traces in K are a TEM of φ .*

Definition 6 (TELKS $\exists$). *The existential variant of the MC problem for TEL over Kripke structures is the language $\text{TELKS}\exists \stackrel{\text{def}}{=} \{(K, \varphi)\}$, where K is a Kripke structure and φ a THT formula such that a trace in K exists which is a TEM of φ .*

No TELKS problem without the specification of the concrete variant properly exists, and the use of this term is meant only as an abbreviation or a generalization in contexts where it is not relevant to specify the concrete variant.

The TELKS $\forall$ Problem. TELKS $\forall$ is the first variant of the TELKS problem to be studied. For this problem, the following holds:

Theorem 1. *Deciding TELKS $\forall$ is PSPACE-complete.*

The way the PSPACE-hardness of the TELKS $\forall$ problem can be shown is by a Karp reduction from the corresponding LTL problem, asking whether, for a given Kripke structure K and an LTL formula φ , φ holds on all traces obtained from K . More formally, the LTLKS $\forall$ and the LTLKS $\exists$ problems are introduced in about the same style as the TELKS $\forall$ and the TELKS $\exists$ ones from Definition 5 and Definition 6, adapting the previously given definitions to the LTL semantics.

The PSPACE-hardness of the LTLKS $\forall$ problem [27] and the fact that this problem can be reduced to the TELKS $\forall$ one allow deducing the PSPACE-hardness of the TELKS $\forall$ problem. The way the reduction from the LTLKS $\forall$ problem to TELKS $\forall$ can be obtained is by means of the following formula:

Definition 7 (Excluded middle axiom for TEL). *Given a finite alphabet A , the excluded middle axiom for TEL is the formula $\varphi_{\text{EM}}(A) \stackrel{\text{def}}{=} \bigwedge_{p \in A} \Box(p \vee \neg p)$.*

Proposition 2 ([8, Proposition 1]). *Given a THT formula φ defined over the alphabet A and a THT interpretation (H, T) , $(H, T) \models_{\text{THT}} \varphi \wedge \varphi_{\text{EM}}(A)$ iff $H = T$ and $T \models_{\text{LTL}} \varphi$.*

The idea of the reduction from LTLKS $\forall$ to TELKS $\forall$ consists of using the $\varphi_{\text{EM}}(A)$ formula to disable the minimization process in the semantics of TEL. In particular, given an instance (K, φ) of the LTLKS $\forall$ problem, an instance (K, φ') of the TELKS $\forall$ problem is obtained such that $(K, \varphi) \in \text{LTLKS}\forall$ iff $(K, \varphi') \in \text{TELKS}\forall$.

Theorem 2. *Deciding TELKS $\forall$ is PSPACE-hard.*

Before continuing to discuss the complexity of TELKS $\forall$ and show its PSPACE membership, a number of techniques is introduced. The first of these techniques is the *star translation* [8], a transformation that allows a THT interpretation to be encoded into an LTL one. THT interpretations are represented by a pair (H, T) of LTL traces, and this translation allows to merge (H, T) into a single LTL trace M :

Definition 8. *Given a THT trace (H, T) with atoms drawn from a set $A = \{a_1, \dots, a_n\}$, the star translation $(H, T)^*$ of (H, T) is the LTL trace M such that for all i , $M_i \stackrel{\text{def}}{=} \{a' \mid a \in H_i\} \cup \{a \mid a \in T_i\}$.*

Setting $X' \stackrel{\text{def}}{=} \{x' \mid x \in X\}$, M can therefore be said to be defined over the alphabet $A^* \stackrel{\text{def}}{=} A \cup A'$, in such a way that if $a' \in H_i$, then $a \in T_i$. Equivalently, M can also be said to satisfy the formula $\varphi_{\leq}(A) \stackrel{\text{def}}{=} \bigwedge_{a \in A} \Box(a' \rightarrow a)$, corresponding to the requirement that $H_i \subseteq T_i$ for a THT interpretation (H, T) . Conversely, for any LTL trace M defined over a set $A \cup A'$ of symbols such that $M \models_{\text{LTL}} \varphi_{\leq}(A)$, a THT trace (H, T) can be found such that $(H, T)^* = M$. More precisely, the following holds:

Lemma 1 ([8, Lemma 1]). *Given an LTL trace M constructed over a set of atoms $A \cup A'$ such that $M \models_{\text{LTL}} \varphi_{\leq}(A)$, a unique THT trace (H, T) over A exists such that $H \leq T$ and $(H, T)^* = M$.*

Other than for THT traces, the star translation is also defined for THT formulas:

Definition 9. *The star translation for THT formulas is a function $(\cdot)^*$ from THT formulas to THT formulas, defined by structural recursion as:*

$$\begin{aligned} \perp^* &\stackrel{\text{def}}{=} \perp, & p^* &\stackrel{\text{def}}{=} p', & (\varphi_1 \vee \varphi_2)^* &\stackrel{\text{def}}{=} \varphi_1^* \vee \varphi_2^*, & (\varphi_1 \wedge \varphi_2)^* &\stackrel{\text{def}}{=} \varphi_1^* \wedge \varphi_2^*, \\ (\varphi_1 \rightarrow \varphi_2)^* &\stackrel{\text{def}}{=} (\varphi_1^* \rightarrow \varphi_2^*) \wedge (\varphi_1 \rightarrow \varphi_2), & (\bigcirc \varphi)^* &\stackrel{\text{def}}{=} \bigcirc \varphi^*, \\ (\varphi_1 \cup \varphi_2)^* &\stackrel{\text{def}}{=} \varphi_1^* \cup \varphi_2^*, & (\varphi_1 \mathbb{R} \varphi_2)^* &\stackrel{\text{def}}{=} \varphi_1^* \mathbb{R} \varphi_2^*. \end{aligned}$$

As an example, $(\Box(a \rightarrow b) \wedge \Diamond(\neg b))^* = \Box(a' \rightarrow b' \wedge a \rightarrow b) \wedge \Diamond(\neg b' \wedge \neg b)$.

Defining the star translation over THT traces and THT formulas allows turning THT satisfiability into LTL satisfiability:

Lemma 2 ([1, Theorem 4]). *Let φ be a THT formula and (H, T) a THT interpretation. Then $(H, T), k \models_{\text{THT}} \varphi$ iff $(H, T)^*, k \models_{\text{LTL}} \varphi^* \wedge \varphi_{\leq}(A)$.*

This equivalence between THT and LTL traces can be brought one step further by introducing the $\varphi_{<}(A) \stackrel{\text{def}}{=} \bigvee_{a \in A} \Diamond(\neg a' \wedge a)$ formula, requiring a THT interpretation to be non total. For this formula, the following holds:

Lemma 3 ([8, Lemma 4]). *For any LTL trace M constructed over a set of atoms $A \cup A'$ such that $M \models_{\text{LTL}} \varphi_{\leq}(A) \wedge \varphi_{<}(A)$, there exists a unique THT trace (H, T) over A such that $H < T$ and $(H, T)^* = M$.*

As a consequence, a restriction of Lemma 2 to non-total traces also holds:

Lemma 4. *Let (H, T) be a THT interpretation for which $H < T$. Then $(H, T), k \models_{THT} \varphi$ iff $(H, T)^*, k \models_{LTL} \varphi^* \wedge \varphi_{\leq}(A) \wedge \varphi_{<}(A)$.*

Before proving the PSPACE-membership of TELKS $\forall$, another technique is introduced which, given a Kripke structure K , obtains an LTL formula encoding the behavior of K . More specifically, this formula encodes the states of K , constrains the way each state can be reached by the others, and enforces the set of literals holding at a state of K (and only that set of literals) to be satisfied. This technique, already described by [26, p. 13] under a different formulation, is termed *K-embedding* and is defined as follows:

Definition 10 (K-embedding). *Given a Kripke structure $K = (Q, q_0, \rightarrow, A, L)$, the K-embedding of K is the THT formula*

$$\begin{aligned} \varphi_K \stackrel{\text{def}}{=} & q_0 \wedge \bigwedge_{q \in Q} \Box \left(q \rightarrow \bigvee_{q \rightarrow q'} q' \right) \wedge \bigwedge_{q \in Q} \Box \left(q \rightarrow \bigwedge_{a \in A \cap L \nabla q \partial} a \wedge \bigwedge_{a \in A \setminus L \nabla q \partial} \neg a \right) \\ & \wedge \bigwedge_{q \in Q} \Box \left(q \rightarrow \bigwedge_{q' \in Q \setminus \{q\}} \neg q' \right). \end{aligned}$$

Proposition 3 (Characterization of K-embeddings). *Given an LTL formula φ constructed over an alphabet A and a Kripke structure $K = (Q, q_0, \rightarrow, A, L)$, an LTL trace T within K exists such that $T \models_{LTL} \varphi$ iff $\varphi \wedge \varphi_K$ is LTL-satisfiable.*

Lemma 5. *Let φ be an LTL formula constructed over an alphabet A and $K = (Q, q_0, \rightarrow, A, L)$ a Kripke structure. Then an LTL trace T within K exists such that $(H, T) \models_{THT} \varphi$ for some $H < T$ iff $\varphi^* \wedge \varphi_{\leq}(A) \wedge \varphi_{<}(A) \wedge \varphi_K$ is LTL-satisfiable.*

Proof. If an LTL trace $T = T_0 T_1 T_2 \dots$ within K exists such that $(H, T) \models_{THT} \varphi$ for some $H < T$, then by Lemma 4 the star-translation $M = (H, T)^*$ of (H, T) LTL-satisfies $\varphi^* \wedge \varphi_{\leq}(A) \wedge \varphi_{<}(A)$. Given that T is from K , each state of T draws its atoms from a single state of K . This makes it possible to label each state of T with an atom $r_i \in Q$, indicating which state from K a single state of T_i obtains its atoms from; however, rather than perform this labeling to T , it is performed to M , so that a new trace $M' \stackrel{\text{def}}{=} (M_0 \cup \{q_0\}) \cdot (M_1 \cup \{r_1\}) \cdot (M_2 \cup \{r_2\}) \cdot \dots$ is obtained. M' continues to satisfy $\varphi^* \wedge \varphi_{\leq}(A) \wedge \varphi_{<}(A)$ and, additionally, also satisfies φ_K , because all of the subformulas of φ_K are satisfied. Hence $\varphi^* \wedge \varphi_{\leq}(A) \wedge \varphi_{<}(A) \wedge \varphi_K$ is LTL-satisfiable.

Conversely, if $\varphi^* \wedge \varphi_{\leq}(A) \wedge \varphi_{<}(A) \wedge \varphi_K$ is LTL-satisfiable, then an LTL trace M over the alphabet $A \cup A' \cup Q$ exists which satisfies it. Since $\varphi^* \wedge \varphi_{\leq}(A) \wedge \varphi_{<}(A)$ is defined only over the $A \cup A'$ alphabet, it continues to be satisfied by the projection M' of M over the $A \cup A'$ alphabet, defined as $M'_i \stackrel{\text{def}}{=} M_i \cap (A \cup A')$. In particular, given that M' satisfies $\varphi_{\leq}(A) \wedge \varphi_{<}(A)$, by Lemma 3 a unique THT trace (H, T) over the A alphabet exists such that $H < T$ and $(H, T)^* = M'$. Additionally, since $M', 0 \models_{LTL} \varphi^* \wedge \varphi_{\leq}(A) \wedge \varphi_{<}(A)$, by Lemma 4 $(H, T), 0 \models_{THT} \varphi$. Finally, since M satisfies φ_K , the projection M'' of M' over the A alphabet, $M''_i \stackrel{\text{def}}{=} M'_i \cap A$, is a trace from K . But M'' is T itself. Hence an LTL trace T within K such that $(H, T) \models_{THT} \varphi$ for some $H < T$ exists. $\square$

Thanks to Lemma 5, the last major result of this section can be derived:

Theorem 3. $\text{TELKS}\forall \in \text{PSPACE}$.

Proof. Given that $\text{PSPACE} = \text{coPSPACE}$, to prove $\text{TELKS}\forall \in \text{PSPACE}$, $\overline{\text{TELKS}\forall} \in \text{PSPACE}$ is shown.

To prove whether a given $(K, \varphi) \in \Sigma_{\overline{\text{TELKS}\forall}}$ belongs to $\overline{\text{TELKS}\forall}$, an LTL trace T within K needs to be found which is not a TEM of φ . By Definition 3, an LTL trace T within K is not a TEM of φ if either (a) (T, T) is not a total THT trace; or (b) $(T, T) \not\models_{\text{THT}} \varphi$; or (c) $\exists H < T$ for which $(H, T) \models_{\text{THT}} \varphi$.

As condition (a) never holds, only condition (b) and condition (c) need to be verified.

To verify whether a trace T exists such that $(T, T) \not\models_{\text{THT}} \varphi$, a trace T such that $T \not\models_{\text{LTL}} \varphi$ is looked for. The problem of looking for such a trace can be reduced to the validity problem of a formula over a Kripke structure, that is to the problem of verifying whether $T \models_{\text{LTL}} \varphi$ for all traces T in K . If $T \not\models_{\text{LTL}} \varphi$ for some trace T in K , then (K, φ) is declared to belong to $\overline{\text{TELKS}\forall}$; conversely, if $T \models_{\text{LTL}} \varphi$ for all traces T in K , the next condition is examined. Since checking the validity of an LTL formula over a Kripke structure is in PSPACE, this type of check is also in PSPACE.

To verify whether an LTL trace T within K such that $(H, T) \models_{\text{THT}} \varphi$ for some $H < T$ exists, the formula $\psi \stackrel{\text{def}}{=} \varphi^* \wedge \varphi_{\leq}(A) \wedge \varphi_{<}(A) \wedge \varphi_K$ is constructed and checked for satisfiability. By Lemma 5, if ψ is satisfiable, then an LTL trace T in K such that $(H, T) \models_{\text{THT}} \varphi$ for some $H < T$ exists, in which case (K, φ) is declared to belong to $\overline{\text{TELKS}\forall}$. If, instead, ψ is unsatisfiable, then by contraposition of Lemma 5 no LTL trace T from K such that $(H, T) \models_{\text{THT}} \varphi$ for some $H < T$ exists, in which case (K, φ) is declared not to belong to $\overline{\text{TELKS}\forall}$. Since by [27] deciding whether an LTL formula is satisfiable is in PSPACE, this third check is also in PSPACE. $\square$

The TELKS $\exists$ Problem. The TELKS $\exists$ problem is the second variant of the TELKS problem to be considered. Contrasting with TELKS $\forall$, the following holds for TELKS $\exists$:

Theorem 4. *Deciding TELKS $\exists$ is EXPSpace-complete.*

The way the EXPSpace-hardness of TELKS $\exists$ can be shown is by a reduction from the satisfiability problem TEL, already shown to be EXPSpace-hard [4]. Specifically, given a formula $\varphi \in \Sigma_{\text{TEL}}$, a pair $(K(\varphi), \varphi')$, composed of a Kripke structure $K(\varphi)$ and a THT formula φ' is associated to it, such that φ admits a TEM iff a trace T exists in $K(\varphi)$ which is a TEM of φ' . The Kripke structure associated to φ is defined as follows:

Definition 11 (Associated Kripke structure). *Given a THT formula φ defined over an ordered alphabet $A \stackrel{\text{def}}{=} \{p_1, \dots, p_n\}$, the associated Kripke structure of φ ¹ is the Kripke structure $K(\varphi) \stackrel{\text{def}}{=} (Q, \{\alpha\}, \rightarrow, A \cup \{\alpha, \beta\}, L)$, where $Q \stackrel{\text{def}}{=} \{\alpha, \beta\} \cup \{p_1, \overline{p_1}, \dots, p_n, \overline{p_n}\}$ and $\rightarrow$ and L are obtained by generalizing the example of Fig. 1.*

The formula φ' associated to the initial THT formula φ is, instead, based on the following transformation:

¹ Although the associated Kripke structure $K(\varphi)$ of a THT formula φ is said to depend on φ , it actually only depends on the alphabet A on which φ is defined.

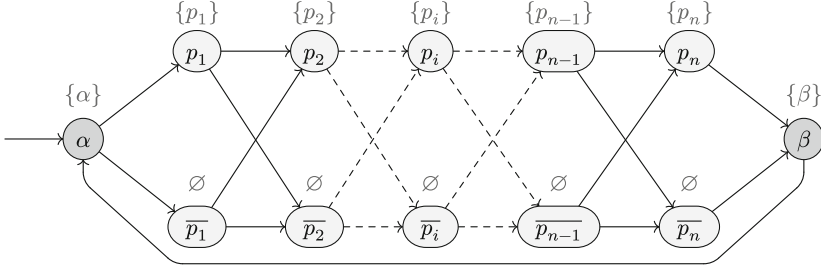


Fig. 1. The associated Kripke structure K_φ for a THT formula φ defined over an alphabet $A = \{p_1, \dots, p_n\}$.

Definition 12 (L-transformation). The L-transformation is a function from THT formulas to THT formulas, defined by structural recursion as:

$$\begin{aligned} \perp^L &\stackrel{\text{def}}{=} \perp; & p_i^L &\stackrel{\text{def}}{=} \circ^i p_i; & (\varphi_1 \odot \varphi_2)^L &\stackrel{\text{def}}{=} \varphi_1^L \odot \varphi_2^L, \odot \in \{\vee, \wedge, \rightarrow\}; \\ (\odot \varphi)^L &\stackrel{\text{def}}{=} \odot^{n+2} \varphi^L; & (\varphi_1 \sqcup \varphi_2)^L &\stackrel{\text{def}}{=} \neg \alpha \vee \varphi_1^L \sqcup \alpha \wedge \varphi_2^L; & (\varphi_1 \mathbb{R} \varphi_2)^L &\stackrel{\text{def}}{=} \alpha \wedge \varphi_1^L \mathbb{R} \neg \alpha \vee \varphi_2^L. \end{aligned}$$

Given a THT formula φ over an alphabet $A \stackrel{\text{def}}{=} \{p_1, \dots, p_n\}$, its associated Kripke structure $K(\varphi)$ and an infinite LTL trace T , an LTL trace $T_{K(\varphi)}$, referred to as the *unfolding* of T over $K(\varphi)$, can be associated to T where the set of literals of each state T_i of T is spread over a sequence of $n+2$ states. For instance, if $T \stackrel{\text{def}}{=} \{a, b, c\} \cdot \{a, c, d\} \cdot \{a\}^\omega$, then $T_{K(\varphi)} = \underbrace{\{\alpha\} \cdot \{a\} \cdot \{b\} \cdot \{c\} \cdot \emptyset \cdot \{\beta\}}_{\{a,b,c\}} \cdot \underbrace{\{\alpha\} \cdot \{a\} \cdot \emptyset \cdot \{c\} \cdot \{d\} \cdot \{\beta\}}_{\{a,c,d\}} \cdot \underbrace{\{\alpha\} \cdot \{a\} \cdot \emptyset \cdot \emptyset \cdot \emptyset \cdot \{\beta\}}_{\{a\}^\omega}^\omega$.

With these notions at hand, the following can be stated:

Lemma 6. Let φ be a THT formula defined over the finite alphabet A and $K(\varphi)$ its associated Kripke structure. Then for any THT trace (H, T) over 2^A and any $k \geq 0$, $(H, T), k \models_{\text{THT}} \varphi$ iff $(H_{K(\varphi)}, T_{K(\varphi)}), (n+2)k \models_{\text{THT}} \varphi^L$.

One final lemma useful to show the EXPSPACE-hardness of TELKS $\exists$ is the following, showing a correspondence between the TEMs of a given formula φ and the TEMs found within its associated Kripke structure:

Lemma 7. Let φ be a THT formula and T an LTL trace. Then T is a TEM of φ iff $T_{K(\varphi)}$ is a TEM of $\varphi' \stackrel{\text{def}}{=} \varphi^L \wedge \varphi_\alpha \wedge \varphi_\beta$, where $\varphi_\alpha \stackrel{\text{def}}{=} \Box(\alpha \vee \neg \alpha)$ and $\varphi_\beta \stackrel{\text{def}}{=} \Box(\beta \vee \neg \beta)$.

Theorem 5. Deciding TELKS $\exists$ is EXPSPACE-hard.

Proof. To show the EXPSPACE-hardness of deciding TELKS $\exists$, a Karp reduction $f: \Sigma_{\text{TEL}} \rightarrow \Sigma_{\text{TELKS}\exists}$ is defined such that $\varphi \in \text{TEL}$ iff $f(\varphi) \in \text{TELKS}\exists$. Since by [4] deciding TEL is EXPSPACE-hard, deciding TELKS $\exists$ turns out to be also EXPSPACE-hard.

The reduction f is defined as $f(\varphi) \stackrel{\text{def}}{=} (K(\varphi), \varphi')$, where $K(\varphi)$ is the associated Kripke structure of φ and $\varphi' \stackrel{\text{def}}{=} \varphi^L \wedge \varphi_\alpha \wedge \varphi_\beta$. What needs to be shown is that f is a function from Σ_{TEL} to $\Sigma_{\text{TELKS}\exists}$, that $\varphi \in \text{TEL}$ iff $f(\varphi) \in \text{TELKS}\exists$, and that f is computable in polynomial time.

The first and third points can be easily argued. To show the second point, $\varphi \in \text{TEL}$ is supposed. If $\varphi \in \text{TEL}$, an LTL trace T exists which is a TEM of φ , and by Lemma 7 the unfolding $T_{K(\varphi)}$ of T over $K(\varphi)$ is a TEM of φ' . Therefore a trace from $K(\varphi)$ which is a TEM of φ' exists, and $(K(\varphi), \varphi') \in \text{TELKS}\exists$. Conversely, if $\varphi \in \Sigma_{\text{TEL}}$ is such that $(K(\varphi), \varphi') = f(\varphi)$ belongs to $\text{TELKS}\exists$, then an LTL trace $T_{K(\varphi)}$ in $K(\varphi)$ exists which is a TEM of φ' . Further, an LTL trace T such that $T_{K(\varphi)}$ is the unfolding of T over $K(\varphi)$ also exists, and by Lemma 7 T is a TEM of φ . Therefore $\varphi \in \text{TEL}$. $\square$

What remains to be done for the $\text{TELKS}\exists$ problem is show its membership to EXPSpace . To that end, the K-embedding technique, previously introduced to show the PSPACE -membership of $\text{TELKS}\forall$, is modified to obtain what could be thought of as a weakened version of it:

Definition 13 (K-quasiembedding). Given a Kripke structure $K = (Q, q_0, \rightarrow, A, L)$, the K-quasiembedding of K is the THT formula

$$\begin{aligned} \psi_K \stackrel{\text{def}}{=} & q_0 \wedge \bigwedge_{q \in Q} \square \left(q \rightarrow \bigvee_{q \rightarrow q'} q' \right) \wedge \bigwedge_{q \in Q} \square \left(q \rightarrow \bigwedge_{a \in A \cap L \nabla q \partial} \neg \neg a \wedge \bigwedge_{a \in A \setminus L \nabla q \partial} \neg a \right) \\ & \wedge \bigwedge_{q \in Q} \square \left(q \rightarrow \bigwedge_{q' \in Q \setminus \{q\}} \neg q' \right). \end{aligned}$$

K-quasiembeddings are defined almost exactly as K-embeddings, with the only difference that each atom a in the state of a Kripke structure K is required to be *not false*, expressed as $\neg \neg a$, rather than true, expressed as a . As HT (and, by extension, THT) is a three-valued logic, $\neg \neg a$ does not automatically imply a , as $\neg \neg a$ is satisfied by the $(\{a\}, \{a\})$ interpretation, corresponding to assigning the “true” truth value to a , or by the $(\emptyset, \{a\})$ one, corresponding to assigning the “unknown” or “assumed” truth value to a .

The purpose of K-quasiembeddings remains however equivalent to those of K-embeddings: K-quasiembeddings allow turning a MC problem into a satisfiability one:

Lemma 8. Let φ be a THT formula constructed over the alphabet A and $K = (Q, q_0, \rightarrow, A, L)$ a Kripke structure. Then an LTL trace T in K exists which is a TEM of φ iff $\varphi \wedge \psi_K$ admits a TEM.

Proof (abridged). If an LTL trace $T = T_0 T_1 T_2 \dots$ in K exists which is a TEM of φ , then the LTL trace $T' \stackrel{\text{def}}{=} (T_0 \cup \{\rho_0\}) \cdot (T_1 \cup \{\rho_1\}) \cdot (T_2 \cup \{\rho_2\}) \cdot \dots$ can be also defined which marks every state T_i of T with an atom $\rho_i \in Q$, indicating which state of K T_i obtains its atoms from. Since $(T, T) \models_{\text{THT}} \varphi$, $(T', T') \models_{\text{THT}} \varphi$ continues to hold. Additionally, it can be shown that $(T', T') \models_{\text{THT}} \psi_K$ also holds, meaning that $(T', T') \models_{\text{THT}} \varphi \wedge \psi_K$. What then needs to be shown is that $\nexists H' < T'$ such that $(H', T') \models_{\text{THT}} \varphi \wedge \psi_K$, meaning that (T', T') is a TEM of $\varphi \wedge \psi_K$.

Conversely, if $\varphi \wedge \psi_K$ admits a TEM, then an LTL trace constructed over the alphabet $A \cup Q$ exists which is a TEM of $\varphi \wedge \psi_K$. From T , its projection T' over the A alphabet, $T'_i \stackrel{\text{def}}{=} T_i \cap A$, can be obtained. By virtue of ψ_K , T' is a trace from K . Further, since φ is constructed over the A alphabet, $(T', T') \models_{\text{THT}} \varphi$ continues to hold. Finally, it can be argued that $\nexists H' < T'$ for which $(H', T') \models_{\text{THT}} \varphi$ holds. Hence an LTL trace T in K exists which is a TEM of φ , and that trace is T' itself. $\square$

Using Lemma 8, the following can be concluded:

Theorem 6. $\text{TELKS}\exists \in \text{EXPSpace}$.

Proof. To show $\text{TELKS}\exists$ to belong to EXPSpace , a Karp reduction $f : \Sigma_{\text{TELKS}\exists} \rightarrow \Sigma_{\text{TEL}}$ is constructed. Given that by [8] TEL belongs to EXPSpace , $\text{TELKS}\exists$ also results to belong to EXPSpace .

Given an instance $(K, \varphi) \in \Sigma_{\text{TELKS}\exists}$, $f((K, \varphi)) \stackrel{\text{def}}{=} \varphi \wedge \psi_K$, where ψ_K is the K-quasiembedding of K . What needs to be shown is that f is a function from $\Sigma_{\text{TELKS}\exists}$ to Σ_{TEL} , that $(K, \varphi) \in \text{TELKS}\exists$ iff $\varphi \wedge \psi_K \in \text{TEL}$, and that f is computable in polynomial time with respect to the size of the instance (K, φ) .

The first and third points can be easily argued. For the second one, $(K, \varphi) \in \text{TELKS}\exists$ is first assumed. If $(K, \varphi) \in \text{TELKS}\exists$, then an LTL trace T within K such that (T, T) is a TEM of φ exists, and by Lemma 8 a TEM for $\varphi \wedge \psi_K$ exists, showing $f((K, \varphi))$ to belong to TEL . Vice versa, if $(K, \varphi) \in \Sigma_{\text{TELKS}\exists}$ is an instance of $\text{TELKS}\exists$ such that $f((K, \varphi)) = \varphi \wedge \psi_K \in \text{TEL}$, then a TEM of $\varphi \wedge \psi_K$ exists and by Lemma 8 an LTL trace T from K exists which is a TEM of φ , meaning that $(K, \varphi) \in \text{TELKS}\exists$. $\square$

4 Fragments of the TELKS Problem

The complexity of fragments of the $\text{TELKS}\forall$ and $\text{TELKS}\exists$ problems can be analyzed by restricting the set of allowed temporal operators, or the *temporal* and *implication height* of a formula.

Given a THT formula φ , φ has a *temporal height* of value m if the number of nested temporal operators is m . For instance, $a \cup \circ b$ has a temporal height of value 2, whereas $\Box a \rightarrow \Diamond a$ has a temporal height of value 1. Similarly, φ has an *implication height* of value k if the number of nested implication operators $\rightarrow$ is k . For instance, $a \rightarrow (b \rightarrow c)$ has an implication height equal to 2. However, given that $\neg\psi$ is a syntactic sugar for $\psi \rightarrow \perp$, $a \rightarrow (b \rightarrow \neg c)$ has an implication height of value 3.

For $P \in \{\text{TELKS}\forall, \text{TELKS}\exists\}$, $P_m^k(O_1, \dots, O_n)$ denotes the fragment of P where only the temporal operators $O_1, \dots, O_n$ are allowed, and where formulas with a temporal height of value *at most* m and an implication height of value *at most* k are considered.

TELKS $\forall$ ($\mathbb{R}$). $\text{TELKS}\forall(\mathbb{R})$ is the fragment of the $\text{TELKS}\forall$ problem where only the temporal operator $\mathbb{R}$ is allowed, and like $\text{TELKS}\forall$, deciding it is still PSPACE -complete. The PSPACE -hardness of this problem results from the fact that $\text{LTLKS}\forall(\mathbb{R})$ is still PSPACE -hard, and that since the $\mathbb{R}$ operator, and in particular the $\Box$ operator, is allowed, a reduction from $\text{LTLKS}\forall(\mathbb{R})$ to $\text{TELKS}\forall(\mathbb{R})$ in the style of Theorem 2 using the $\varphi_{\text{EM}}(A)$ formula is still possible. The membership of $\text{TELKS}\forall(\mathbb{R})$ to PSPACE continues to hold because $\text{TELKS}\forall(\mathbb{R})$ is a subset of $\text{TELKS}\forall$:

Theorem 7. *Deciding $\text{TELKS}\forall(\mathbb{R})$ is PSPACE -complete.*

TELKS $\exists$ ($\mathbb{R}$). $\text{TELKS}\exists(\mathbb{R})$ is the fragment of the $\text{TELKS}\exists$ problem where only the $\mathbb{R}$ operator is allowed, and like $\text{TELKS}\exists$, deciding it is also EXPSpace -complete. Showing the EXPSpace -hardness of the $\text{TELKS}\exists(\mathbb{R})$ problem can be done by readapting the

proof of the EXPSPACE-hardness of the unconstrained TELKS $\exists$ problem, by revising the definition of the L-transformation in a way that no $\bigcirc$ and $\bigcup$ operators are used. For lack of a better name, this modified transformation is termed *M-transformation*:

Definition 14 (M-transformation). *The M-transformation is a function from THT formulas to THT formulas, defined by structural recursion as:*

$$\begin{aligned} \perp^M &\stackrel{\text{def}}{=} \perp; & p_i^M &\stackrel{\text{def}}{=} p_i \mathbb{R} \neg \beta; & (\varphi_1 \odot \varphi_2)^M &\stackrel{\text{def}}{=} \varphi_1^M \odot \varphi_2^M, \odot \in \{\vee, \wedge, \rightarrow\}; \\ (\varphi_1 \mathbb{R} \varphi_2)^M &\stackrel{\text{def}}{=} \alpha \wedge \varphi_1^M \mathbb{R} \neg \alpha \vee \varphi_2^M. \end{aligned}$$

By adapting Lemma 6 and Lemma 7, already used for the L-transformation, to the case of the M-transformation, the following can be concluded:

Theorem 8. *Deciding TELKS $\exists(\mathbb{R})$ is EXPSPACE-hard.*

Proof. Similarly to the proof of Theorem 5, a Karp reduction $f(\varphi) \stackrel{\text{def}}{=} (K(\varphi), \varphi')$ from TEL($\mathbb{R}$) to TELKS $\exists(\mathbb{R})$ is defined, where $\varphi' \stackrel{\text{def}}{=} \varphi^M \wedge \varphi_\alpha \wedge \varphi_\beta$. Since by [4, Theorem III.1] deciding TEL $_2^2(\square)$, and in particular deciding TEL($\mathbb{R}$), is EXPSPACE-hard, deciding TELKS $\exists(\mathbb{R})$ also turns out to be EXPSPACE-hard. $\square$

Since TELKS $\exists(\mathbb{R}) \subset \text{TELKS}\exists$ and $\text{TELKS}\exists \in \text{EXPSPACE}$:

Corollary 1. $\text{TELKS}\exists(\mathbb{R}) \in \text{EXPSPACE}$.

Theorem 9. *Deciding TELKS $\exists(\mathbb{R})$ is EXPSPACE-complete.*

5 The TELT Problem

If the requirement that the transition relation $\rightarrow$ of K only form a single loop, like in Fig. 2, is imposed, a special instance of the TELKS problem results where only one trace is obtained by traversing K . Kripke structures satisfying this condition are termed *linear*. Deciding whether an input formula φ holds against a linear Kripke structure essentially corresponds to deciding whether φ holds over a single *periodically infinite* LTL trace T , a trace $T \stackrel{\text{def}}{=} T_1 T_2^\omega$ where T_1 and T_2 are LTL traces of *finite* length and T_2^ω represents the infinite concatenation of T_2 . This restriction leads to the *TELT problem*:

Definition 15 (TELT). *The MC problem for TEL over traces is the language $\text{TELT} \stackrel{\text{def}}{=} \{(T, \varphi)\}$, where $T \stackrel{\text{def}}{=} T_1 T_2^\omega$ is a periodically infinite LTL trace and φ a THT formula such that T is a TEM of φ .*

As only one trace can be obtained by traversing a linear Kripke structure K , asking whether a THT formula φ is a TEM for *all* of the traces in K is equivalent to asking whether φ is a TEM for a *single* trace in K . Therefore, in the case of TELT the universal and the existential semantics of the TELKS problem from Sect. 3 collapse into one and TELT can be seen to correspond to a subset of TELKS $\forall$. In particular, this means that deciding TELT cannot be harder than deciding TELKS $\forall$:

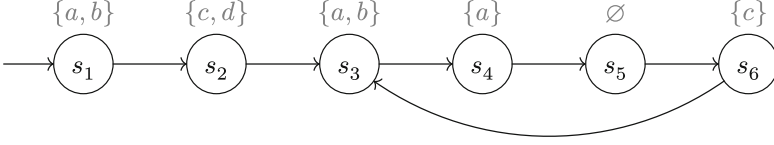


Fig. 2. A linear Kripke structure.

Proposition 4. $\text{TELT} \in \text{PSPACE}$.

Deciding TELT is in fact not only at most as hard as $\text{TELKS}\forall$ —it is *just as* hard as deciding $\text{TELKS}\forall$. For this problem, the following also holds:

Theorem 10. *Deciding TELT is PSPACE-hard.*

Due to the constraints imposed to the TELT problem, a different proof technique than the one used for the PSPACE-hardness proof of the $\text{TELKS}\forall$ problem has to be used:

Proof (abridged). To show the PSPACE-hardness of deciding TELT, it is shown how any language L in PSPACE can be reduced to $\overline{\text{TELT}}$ by proving the existence of a Karp reduction $f : \Sigma_L \rightarrow \Sigma_{\overline{\text{TELT}}}$ assigning to each word $x \in \Sigma_L$ a pair $(T, \varphi) \in \Sigma_{\overline{\text{TELT}}}$, where T is a periodically infinite LTL trace and φ a THT formula. As the complement of any PSPACE-hard problem is also PSPACE-hard, if deciding $\overline{\text{TELT}}$ is shown PSPACE-hard, deciding TELT also results to be PSPACE-hard.

If $L \in \text{PSPACE}$, then a single-tape Turing machine $M = (Q, \Gamma, \delta, q_0, F)$ deciding L in deterministic polynomial space exists, where Q is the non-empty, finite set of states, Γ the tape alphabet, $\delta : (Q \setminus F) \times \Gamma \rightarrow Q \times \Gamma \times \{-1, 0, 1\}$ the transition function, $q_0 \in Q$ the initial state and $F = \{q_{\text{accept}}, q_{\text{reject}}\} \subseteq Q$ the set of halting states. $\{-1, 0, 1\}$ denotes the possible directions the tape head of M can move to, corresponding to left, no movement, and right, represented as integers to facilitate the construction of f .

Since M decides L in polynomial space, a polynomial $p(n)$ exists such that for any $x \in \Sigma_L$, M decides in $p(|x|)$ space whether x belongs to L or not, where $|x|$ is the number of symbols in x . Each configuration of M at step i is encoded by storing the state q at step i , the position of the head of M at step i and at most $p(|x|)$ tape symbols. M is encoded into the trace T in the following way:

- To encode the state q of M , a set of symbols Q is used, in such a way that for a $q \in Q$, $q \in T_i$ iff the state of M at time step i is q ;
- To encode the head position of M at time step i , a set of variables h_k , $k \in [0..p(|x|))$ is used, so that $h_k \in T_i$ iff the head of M at time step i points to cell k ;
- The tape content of M at time step i can be encoded into state T_i of T by the use of $|\Gamma| \cdot p(|x|)$ many propositional atoms;
- For each tape symbol $\gamma \in \Gamma$ and $k \in [0..p(|x|))$, the propositional variable $\sigma_{k,\gamma}$ is added to the alphabet A of φ to signify that at the tape position k , the symbol is γ .

The alphabet A over which φ is defined is $A \stackrel{\text{def}}{=} A_Q \cup A_h \cup A_\sigma$, where $A_Q \stackrel{\text{def}}{=} Q$, $A_h \stackrel{\text{def}}{=} \{h_k \mid 0 \leq k < p(|x|)\}$, and $A_\sigma \stackrel{\text{def}}{=} \{\sigma_{k,\gamma} \mid 0 \leq k < p(|x|), \gamma \in \Gamma\}$, showing in particular that the cardinality of A is polynomial with respect to the size of M and $p(|x|)$.

Given an arbitrary word $x \in \Sigma_L$, $f(x) \stackrel{\text{def}}{=} (T, \varphi)$, where $T \stackrel{\text{def}}{=} (2^A)^\omega$ and $\varphi \stackrel{\text{def}}{=} \bigwedge_{1 \leq i \leq 7} \varphi_i$ is the formula encoding the behavior of M , split into the subformulas φ_i for easier understanding:

$$\begin{aligned} \varphi_1 &\stackrel{\text{def}}{=} q_0 \wedge h_0 \wedge \sigma_{0,x_0} \wedge \sigma_{1,x_1} \wedge \cdots \wedge \sigma_{|x|,x_{|x|}} \wedge \sigma_{|x|+1,\sqcup} \wedge \cdots \wedge \sigma_{p(|x|)-1,\sqcup}, \\ \varphi_2 &\stackrel{\text{def}}{=} \bigwedge_{q \in Q} \bigwedge_{0 \leq k < p(|x|)} \bigwedge_{\gamma \in \Gamma} \Box(q \wedge h_k \wedge \sigma_{k,\gamma} \wedge \delta(q,\gamma) = (q', \gamma', \Delta) \rightarrow \bigcirc(q' \wedge h_{k+\Delta} \wedge \sigma_{k,\gamma'})), \\ \varphi_3 &\stackrel{\text{def}}{=} \Box\left(\bigwedge_{0 \leq k < p(|x|)} \bigwedge_{\substack{0 \leq k' < p(|x|) \\ k' \neq k}} \bigwedge_{\gamma \in \Gamma} h_k \wedge \sigma_{k',\gamma} \rightarrow \bigcirc \sigma_{k',\gamma}\right), \\ \varphi_4 &\stackrel{\text{def}}{=} \Diamond\left(\bigvee_{q \in Q} \bigvee_{q' \in Q \setminus \{q\}} q \wedge q'\right) \rightarrow \varphi^\dagger, \quad \varphi_5 \stackrel{\text{def}}{=} \Diamond\left(\bigvee_{0 \leq k < p(|x|)} \bigvee_{\substack{0 \leq k' < p(|x|) \\ k' \neq k}} h_k \wedge h_{k'}\right) \rightarrow \varphi^\dagger, \\ \varphi_6 &\stackrel{\text{def}}{=} \Diamond\left(\bigvee_{0 \leq k < p(|x|)} \bigvee_{\gamma \in \Gamma} \bigvee_{\gamma' \in \Gamma \setminus \{\gamma\}} \sigma_{k,\gamma} \wedge \sigma_{k,\gamma'}\right) \rightarrow \varphi^\dagger, \quad \varphi_7 \stackrel{\text{def}}{=} \Diamond q_{\text{reject}} \rightarrow \varphi^\dagger. \end{aligned}$$

The φ_i formulas make use of the auxiliary formula $\varphi^\dagger$, defined as $\varphi^\dagger \stackrel{\text{def}}{=} \Box \bigwedge_{a \in A} a$ and used to induce a state of saturation over the TEMs of φ .

On the basis of this construction, it can be argued that f correctly maps x to a pair (T, φ) , with T a periodically infinite LTL trace and φ a THT formula; that $x \in L$ iff T is not a TEM of φ ; and that f is computable in polynomial time with respect to $|x|$. $\square$

Theorem 11. *Deciding TELT is PSPACE-complete.*

Given that the φ_i formulas from the proof of Theorem 10 have a temporal height no greater than 2 and an implication height no greater than 1:

Corollary 2. *Deciding TELT_2^1 is PSPACE-complete.*

If formulas making no use of the $\sqcup$ operator are considered, the $\text{TELT}(\bigcirc, \mathbb{R})$ fragment is obtained, for which the following holds:

Theorem 12. *Deciding $\text{TELT}(\bigcirc, \mathbb{R})$ is PSPACE-hard.*

Proof (abridged). To show the PSPACE-hardness of deciding $\text{TELT}(\bigcirc, \mathbb{R})$, a rewriting of the φ formula from the proof of Theorem 10 into a φ' formula using only the $\bigcirc$ and $\mathbb{R}$ operators is given. The way φ' can be defined is by adding a special atom $a^\dagger$, referred to as the *saturation atom*, to the $A \stackrel{\text{def}}{=} A_Q \cup A_h \cup A_\sigma$ alphabet over which φ is defined, obtaining an $A' \stackrel{\text{def}}{=} A \cup \{a^\dagger\}$ alphabet. Each φ_i subformula of φ can be subsequently rewritten into a subformula φ'_i of φ' in the following way:

$$\begin{aligned} \varphi'_1 &\stackrel{\text{def}}{=} \varphi_1, \quad \varphi'_2 \stackrel{\text{def}}{=} \varphi_2, \quad \varphi'_3 \stackrel{\text{def}}{=} \varphi_3, \quad \varphi'_4 \stackrel{\text{def}}{=} \Box\left(\bigvee_{q \in Q} \bigvee_{q' \in Q \setminus \{q\}} q \wedge q' \rightarrow a^\dagger\right), \\ \varphi'_5 &\stackrel{\text{def}}{=} \Box\left(\bigvee_{0 \leq k < p(|x|)} \bigvee_{\substack{0 \leq k' < p(|x|) \\ k' \neq k}} h_k \wedge h_{k'} \rightarrow a^\dagger\right), \\ \varphi'_6 &\stackrel{\text{def}}{=} \Box\left(\bigvee_{0 \leq k < p(|x|)} \bigvee_{\gamma \in \Gamma} \bigvee_{\gamma' \in \Gamma \setminus \{\gamma\}} \sigma_{k,\gamma} \wedge \sigma_{k,\gamma'} \rightarrow a^\dagger\right), \quad \varphi'_7 \stackrel{\text{def}}{=} \Box(q_{\text{reject}} \rightarrow a^\dagger). \end{aligned}$$

An additional set of formulas, denoted by ψ_i , also needs to be used to govern the behavior of the saturation atom $a^\dagger$:

$$\psi_1 \stackrel{\text{def}}{=} \Box(a^\dagger \rightarrow \bigcirc a^\dagger), \quad \psi_2 \stackrel{\text{def}}{=} \Box(\bigcirc a^\dagger \rightarrow a^\dagger), \quad \psi_3 \stackrel{\text{def}}{=} \Box a^\dagger \rightarrow \varphi^\dagger.$$

The $\varphi^\dagger$ formula is rewritten into $\varphi^\dagger \stackrel{\text{def}}{=} \Box \bigwedge_{a' \in A'} a'$ to take the $a^\dagger$ atom into account. Finally, the φ' formula is defined as $\varphi' \stackrel{\text{def}}{=} \bigwedge_{1 \leq i \leq 7} \varphi'_i \wedge \bigwedge_{1 \leq i \leq 3} \psi_i$. $\square$

Theorem 13. *Deciding $\text{TELT}(\bigcirc, \mathbb{R})$ is PSPACE-complete.*

Corollary 3. *Deciding $\text{TELT}_2^1(\bigcirc, \mathbb{R})$ is PSPACE-complete.*

6 Conclusion

This work studied the MC problem for TEL under Kripke structures, specifically the universal and existential variants of it, and the same problem under linear Kripke structures, corresponding to linear LTL traces.

For reasons of space, a number of already proven results, planned to be included in future work, has not been discussed. These are the coNP -completeness of TELT_1 ; the coNP -completeness of $\text{TELT}(\bigcirc, \bigcup)$, $\text{TELT}(\bigcirc)$ and $\text{TELT}(\bigcup)$; the PSPACE-completeness of $\text{TELT}(\bigcup, \mathbb{R})$ and $\text{TELT}_2^1(\bigcup, \mathbb{R})$; the coNP -completeness of TELT_f (the restriction of TELT where the input trace T is assumed to be finite) and the coNP -completeness of TELKS_0 (corresponding to MC over the atemporal EL). The coNP -completeness of TELT_f means, in particular, that MC for TEL in practical solvers like `telingo` is not computationally harder than for atemporal ASP.

Acknowledgments. This project received funding from the European Research Council (grant agreement no. 101034440), by the MOSAIC project (EU H2020-MSCA-RISE-2020 Project 101007627) and by the WWTF project ICT22-023.

References

1. Aguado, F., et al.: Linear-time temporal answer set programming. Theory Pract. Logic Program. (2023). <https://doi.org/10.1017/S1471068421000557>
2. Barreca, O.: On the Complexity of Model Checking for Temporal Equilibrium Logic. MSc thesis, Technische Universität Wien (2025). <https://doi.org/10.34726/hss.2025.125771>
3. Beck, H., Dao-Tran, M., Eiter, T.: LARS: a logic-based framework for analytic reasoning over streams. Artif. Intell. (2018). <https://doi.org/10.1016/j.artint.2018.04.003>
4. Bozzelli, L., Pearce, D.: On the complexity of temporal equilibrium logic. In: Proceedings of ACM/IEEE LICS-2015. IEEE (2015). <https://doi.org/10.1109/LICS.2015.65>
5. Bozzelli, L., Pearce, D.: On the expressiveness of temporal equilibrium logic. In: Proceedings of JELIA 2016. Springer (2016). https://doi.org/10.1007/978-3-319-48758-8_11
6. Brandt, S., Kalayci, E.G., Ryzhikov, V., Xiao, G., Zakharyashev, M.: Querying log data with metric temporal logic. J. Artif. Intell. Res. (2018). <https://doi.org/10.1613/jair.1.11229>
7. Buss, S.R.: The boolean formula value problem is in ALOGTIME. In: Proceedings of ACM STOC 1987. Association for Computing Machinery (1987). <https://doi.org/10.1145/28395.28409>

8. Cabalar, P., Demri, S.: Automata-based computation of temporal equilibrium models. In: *Logic-Based Program Synthesis and Transformation*. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32211-2_5
9. Cabalar, P., Diéguez, M., Hahn, S., Schaub, T.: Automata for dynamic answer set solving: preliminary report (2021)
10. Cabalar, P., Eiter, T., Soldà, D.: Contracted temporal equilibrium logic. In: *Proceedings of KR 2024* (2024). <https://doi.org/10.24963/kr.2024/21>
11. Cabalar, P., Kaminski, R., Morkisch, P., Schaub, T.: Telingo = ASP + time. In: *Proceedings of LPNMR 2019*. Springer (2019). https://doi.org/10.1007/978-3-030-20528-7_19
12. Cabalar, P., Pérez Vega, G.: Temporal equilibrium logic: a first approach. In: *Computer Aided Systems Theory – EUROCAST 2007* (2007). https://doi.org/10.1007/978-3-540-75867-9_31
13. Cadoli, M.: The complexity of model checking for circumscriptive formulae. *Inf. Process. Lett.* (1992). [https://doi.org/10.1016/0020-0190\(92\)90049-2](https://doi.org/10.1016/0020-0190(92)90049-2)
14. Clarke, E.M., Emerson, E.A.: Design and synthesis of synchronization skeletons using branching time temporal logic. In: *Logics of Programs*. Springer (1982). <https://doi.org/10.1007/BFb0025774>
15. Demri, S., Schnobelen, P.: The complexity of propositional linear temporal logics in simple cases. *Inf. Comput.* (2002). <https://doi.org/10.1006/inco.2001.3094>
16. Eiter, T., Faber, W., Fink, M., Pfeifer, G., Woltran, S.: Complexity of model checking and bounded predicate arities for non-ground answer set programming. In: *Proceedings of KR 2004*, pp. 377–387. AAAI Press (2004)
17. Eiter, T., Kiesel, R.: Weighted LARS for quantitative stream reasoning. In: *Proceedings of ECAI 2020*. SAGE Publications (2020). <https://doi.org/10.3233/FAIA200160>
18. Gelfond, M., Lifschitz, V.: The stable model semantics for logic programming. In: *Proceedings of ICLP/ILPS 1988*, pp. 1070–1080. MIT Press (1988)
19. Giordano, L., Martelli, A., Dupré, D.T.: Reasoning about actions with temporal answer sets. *Theory Pract. Log. Program.* (2013). <https://doi.org/10.1017/S1471068411000639>
20. Heyting, A.: Die formalen Regeln der intuitionistischen Logik. *Sitzungsberichte der Preussischen Akademie der Wissenschaften Berlin, phys.-math. Klasse*, pp. 42–56 (1930)
21. Kripke, S.A.: Semantical analysis of intuitionistic logic I. In: *Studies in Logic and the Foundations of Mathematics*. Elsevier (1965). [https://doi.org/10.1016/S0049-237X\(08\)71685-9](https://doi.org/10.1016/S0049-237X(08)71685-9)
22. Laroussinie, F.C., Leclercq, L., Sangnier, A.: QLTL model-checking. In: *Proceedings of CSL 2024. LIPICs*, vol. 288 (2024). <https://doi.org/10.4230/LIPICS.CSL.2024.35>
23. Liberatore, P., Schaerf, M.: The complexity of model checking for propositional default logics. *Data Knowl. Eng.* (2005). <https://doi.org/10.1016/j.datak.2005.03.002>
24. Pearce, D.: A new logical characterisation of stable models and answer sets. In: *Proceedings of NMELP 1996*. Springer, Heidelberg (1997). <https://doi.org/10.1007/BFb0023801>
25. Pearce, D.: Equilibrium logic. *Ann. Math. Artif. Intell.* (2006). <https://doi.org/10.1007/s10472-006-9028-z>
26. Schnobelen, P.: The complexity of temporal logic model checking. In: *Advances in Modal Logic 4*, pp. 393–436. King’s College Publications (2002). <http://www.aiml.net/volumes/volume4/Schnobelen.ps>
27. Sistla, A.P., Clarke, E.M.: The complexity of propositional linear temporal logics. *J. ACM* (1985). <https://doi.org/10.1145/3828.3837>
28. Sistla, A.P., Vardi, M.Y., Wolper, P.: The complementation problem for Büchi automata with applications to temporal logic. In: *Proceedings of ICALP 1985*. Springer (1985). <https://doi.org/10.1007/BFb0015772>
29. Van Dalen, D.: *Logic and Structure*. Springer (2013). <https://doi.org/10.1007/978-1-4471-4558-5>