LTCS–Report

LTL over $\mathcal{EL}$ Axioms

Stefan Borgwardt   Veronika Thost

LTCS-Report 15-07

Contents

1 Introduction

Description Logics (DLs) [BCM$^+$07] are popular knowledge representation formalisms, mainly because they are the basis of the standardized OWL 2 Direct Semantics,[1] their expressiveness can be tailored to the application at hand, and many optimized reasoning systems are available.[2] DLs describe domain knowledge using *axioms* such as

$$\exists\mathsf{teaches}.\mathsf{Course} \sqsubseteq \exists\mathsf{holdsDegree}.\mathsf{DoctoralDegree},$$

which says that everyone who teaches a (university) course must necessarily hold a doctoral degree. The basic building blocks are *concept names* (Course, DoctoralDegree) that describe subsets the domain of discourse, and *role names* (teaches, holdsDegree) that allow to draw connections between domain elements.

However, pure DLs are not suited for representing temporal dependencies that occur in many real-world domains. For this purpose, diverse extensions of DLs with temporal formalisms have been developed [AF00, LWZ08]. In particular, combinations of DLs with the operators of propositional temporal logics have received much attention [WZ00, AKL$^+$07, AKRZ09, BGL12, GK12, BLT13, AKWZ13]. The approach we follow in this report is based on the idea to replace the propositional variables in formulae of Linear-Time Temporal Logic (LTL) [Pnu77] by description logic axioms to describe the possible evolution of a system. In this setting, concept and role names may be designated as *rigid* to express that their interpretation is not allowed to change over time.

The satisfiability problem of LTL over axioms of different DLs has been analyzed before. For the lightweight description logic $DL\text{-}Lite_{krom}$, this problem has been shown to be PSPACE-complete in [AKL$^+$07], matching the complexity of propositional LTL [SC85]. For the more expressive DL $\mathcal{ALC}$, the complexity increases, depending on what kind of rigid names one allows to occur in the formulae [BGL12]. Using similar techniques, the same results can be shown even for $\mathcal{SHOQ}$, if role axioms are restricted to be *global*, which means that they must be satisfied at every time point [Lip14]. It is therefore interesting to investigate whether $\mathcal{EL}$ shows a similarly nice behavior as $DL\text{-}Lite$ in this regard, or whether $\mathcal{EL}$-LTL is as complex as $\mathcal{ALC}$-LTL, as conjectured in [BGL12].

We show in this report that, while $\mathcal{EL}$-LTL is not as well-behaved as its $DL\text{-}Lite$ counterpart, the complexity of satisfiability is reduced when compared to $\mathcal{ALC}$-LTL. Only in the case that rigid concept names are allowed, but no rigid role names, does the complexity match that of satisfiability in $\mathcal{ALC}$-LTL. If we allow only *global GCIs*, then the satisfiability problem is PSPACE-complete in all cases. Table 1.1 gives an overview over all mentioned complexity results (all of them are

---

[1]http://www.w3.org/TR/owl2-overview/
[2]http://www.w3.org/2001/sw/wiki/OWL/Implementations

Table 1.1: The complexity of satisfiability in LTL over DL axioms

rigid symbols?	none	concepts	roles	Global GCIs none	concepts	roles
$DL\text{-}Lite_{krom}$	PSPACE	PSPACE	PSPACE	PSPACE	PSPACE	PSPACE
$\mathcal{EL}$	PSPACE	NEXPTIME	NEXPTIME	PSPACE	PSPACE	PSPACE
$\mathcal{ALC}/\mathcal{SHOQ}$	EXPTIME	NEXPTIME	2-EXPTIME	EXPTIME	EXPTIME	2-EXPTIME

tight). Rigid concept names can be simulated by rigid role names [BGL12], and thus there are only three cases to consider.

2 Preliminaries

We first introduce syntax and semantics of the description logic $\mathcal{EL}$, of the propositional linear-temporal logic LTL, and of their combination into $\mathcal{EL}$-LTL.

2.1 $\mathcal{EL}$ and Extensions

In the description logic $\mathcal{EL}$, *(complex) concepts* are constructed from a set $\mathsf{N_C}$ of *concept names* and a set $\mathsf{N_R}$ of *role names* inductively as follows: Every concept name and the special symbol $\top$ (*top*) are concepts, and whenever C and D are concepts, then so are $C \sqcap D$ (*conjunction*), and $\exists r.C$ (*existential restriction* for $r \in \mathsf{N_R}$).

Given a set $\mathsf{N_I}$ of *individual names*, an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a non-empty set $\Delta^{\mathcal{I}}$, the *domain* of $\mathcal{I}$, and an *interpretation function* $\cdot^{\mathcal{I}}$ assigning to every individual name a an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, to every concept name A a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and to every role name r a relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. This function is extended to complex concepts as follows:

- $\top^{\mathcal{I}} := \Delta^{\mathcal{I}}$,

- $(C \sqcap D)^{\mathcal{I}} := C^{\mathcal{I}} \cap D^{\mathcal{I}}$, and

- $(\exists r.C)^{\mathcal{I}} := \{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} \colon (x,y) \in r^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}}\}$.

In this report, we make the standard *unique name assumption (UNA)* for all interpretations, which requires that different individual names always be interpreted by different domain elements, i.e. we have $a^{\mathcal{I}} \neq b^{\mathcal{I}}$ for all $a, b \in \mathsf{N_I}$ with $a \neq b$.

An *assertion* is an expression of the form $C(a)$ (*concept assertion*) or $r(a,b)$ (*role assertion*) for $a, b \in \mathsf{N_I}$, $r \in \mathsf{N_R}$, and a concept C. A *general concept inclusion*

(GCI) is of the form $C \sqsubseteq D$ for concepts C and D. An *axiom* is either an assertion or a GCI, a *TBox* is a finite set of GCIs, and an *ABox* is a finite set of assertions. Together, a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$ form a *knowledge base (KB)* $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$. An interpretation $\mathcal{I}$ *satisfies* (or is a *model* of)

- an assertion $C(a)$ if $a^{\mathcal{I}} \in C^{\mathcal{I}}$ (written $\mathcal{I} \models C(a)$);

- an assertion $r(a, b)$ if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ (written $\mathcal{I} \models r(a, b)$);

- a GCI $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ (written $\mathcal{I} \models C \sqsubseteq D$);

- a KB if it satisfies all its axioms (written $\mathcal{I} \models \mathcal{K}$).

We further denote the fact that every model of a knowledge base $\mathcal{K}$ satisfies an axiom α by $\mathcal{K} \models \alpha$. A knowledge base is *consistent* if it has a model.

In the course of this report, we will need the two following extensions of $\mathcal{EL}$. First, $\mathcal{EL}_{\perp}$ extends $\mathcal{EL}$ by the concept constructor $\perp$ (*bottom*) that is interpreted as $\perp^{\mathcal{I}} := \emptyset$ in all interpretations $\mathcal{I}$. Second, $\mathcal{ELO}_{\perp}$ further extends $\mathcal{EL}_{\perp}$ with the constructor $\{a\}$ (*nominal* for $a \in \mathsf{N_I}$) with the semantics that $\{a\}^{\mathcal{I}} := \{a^{\mathcal{I}}\}$ in any interpretation $\mathcal{I}$.

2.2   Propositional LTL

(Propositional) LTL-formulae are built from a set of *propositional variables P* by applying the constructors $\phi \wedge \psi$ (*conjunction*), $\neg \phi$ (*negation*), $\bigcirc \phi$ (*next*) and $\phi \, \mathsf{U} \, \psi$ (*until*). An *LTL-structure* is a sequence $\mathfrak{I} = (w_i)_{i \geq 0}$ of *worlds $w_i \subseteq P$* that specify which propositional variables are true at the (linearly ordered) time points $i \geq 0$. *Validity* of an LTL-formula ϕ in an LTL-structure $\mathfrak{I}$ at time point $i \geq 0$, written $\mathfrak{I}, i \models \phi$, is inductively defined as follows:

$$
\begin{array}{lll}
\mathfrak{I}, i \models p \text{ for } p \in P & \text{iff} & p \in w_i \\
\mathfrak{I}, i \models \phi \wedge \psi & \text{iff} & \mathfrak{I}, i \models \phi \text{ and } \mathfrak{I}, i \models \psi \\
\mathfrak{I}, i \models \neg \phi & \text{iff} & \text{not } \mathfrak{I}, i \models \phi \\
\mathfrak{I}, i \models \bigcirc \phi & \text{iff} & \mathfrak{I}, i + 1 \models \phi \\
\mathfrak{I}, i \models \phi \, \mathsf{U} \, \psi & \text{iff} & \text{there is some } k \geq i \text{ such that } \mathfrak{I}, k \models \psi \\
& & \text{and } \mathfrak{I}, j \models \phi \text{ for all } j, \, i \leq j < k
\end{array}
$$

An LTL-formula is *satisfiable* if it is valid in some LTL-structure at time point 0.

As usual, we use the abbreviations $\mathsf{true} := p \vee \neg p$ for an arbitrary $p \in P$, $\psi \vee \phi := \neg(\neg \psi \wedge \neg \phi)$, $\phi \rightarrow \psi := \neg \phi \vee \psi$, $\phi \leftrightarrow \psi := (\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi)$, $\Diamond \phi := \mathsf{true} \, \mathsf{U} \, \phi$, and $\Box \phi := \neg \Diamond \neg \phi$.

2.3 $\mathcal{EL}$-LTL

As was done for *DL-Lite* [AKL$^+$07] and $\mathcal{ALC}$ [BGL12], we combine LTL with $\mathcal{EL}$ into the temporalized description logic $\mathcal{EL}$-LTL by applying the temporal operators of LTL to axioms of $\mathcal{EL}$. The resulting $\mathcal{EL}$-*LTL-formulae* are then analyzed for their satisfiability according to a semantics that is suitably lifted from propositional worlds to $\mathcal{EL}$-interpretations over a common domain. We additionally consider some concept and role names to be *rigid*, which means that they are not allowed to change over time. For this purpose, we designate the two sets $\mathsf{N_{RC}} \subseteq \mathsf{N_C}$ of *rigid concept names* and $\mathsf{N_{RR}} \subseteq \mathsf{N_R}$ of *rigid role names*. All other names are called *flexible*. All individual names are implicitly assumed to be rigid.

We now introduce the formal semantics of $\mathcal{EL}$-LTL. An $\mathcal{EL}$-*LTL-structure* is a sequence $\mathfrak{I} = (\mathcal{I}_i)_{i \geq 0}$ of $\mathcal{EL}$-interpretations (also called *worlds*) over a common domain Δ that respect the rigid names, i.e. we have $x^{\mathcal{I}_i} = x^{\mathcal{I}_j}$ for all $x \in \mathsf{N_I} \cup \mathsf{N_{RC}} \cup \mathsf{N_{RR}}$ and $i, j \geq 0$. *Validity* of an $\mathcal{EL}$-LTL-formula ϕ in an $\mathcal{EL}$-LTL-structure $\mathfrak{I} = (\mathcal{I}_i)_{i \geq 0}$ at time point $i \geq 0$, written $\mathfrak{I}, i \models \phi$, is inductively defined as for LTL, but we have $\mathfrak{I}, i \models \alpha$ for an axiom α iff α is satisfied by $\mathcal{I}_i$. An $\mathcal{EL}$-LTL-formula is *satisfiable* if it is valid in some $\mathcal{EL}$-LTL-structure at time point 0.

Before we present the main results of this report, we establish an auxiliary result about the satisfiability of certain atemporal combinations of $\mathcal{EL}$-axioms, which will be used in the proofs of the upper bounds. For this, we consider conjunctions of $\mathcal{EL}$-*literals*, which are axioms and negated axioms.[3] Since these formulae do not contain temporal operators, it suffices to consider a single interpretation to determine satisfiability, and thus rigid names are irrelevant.

Lemma 2.1. *Satisfiability of conjunctions of $\mathcal{EL}$-literals can be decided in* P.

Proof. We reduce this problem to the consistency problem of $\mathcal{ELO}_\perp$. Given a conjunction ϕ of $\mathcal{EL}$-literals, we construct a knowledge base that is consistent iff ϕ is satisfiable. We convert the literals of ϕ into a KB $\langle \mathcal{T}, \mathcal{A} \rangle$ by replacing all

- negated concept assertions $\neg C(a)$ by the axioms $C'(a)$ and $C \sqcap C' \sqsubseteq \perp$, where C' is a fresh concept name;

- negated role assertions $\neg r(a, b)$ by the axiom $\{a\} \sqcap \exists r.\{b\} \sqsubseteq \perp$;

- negated GCIs $\neg(C \sqsubseteq D)$ by the axioms $C(a)$, $D'(a)$, and $D \sqcap D' \sqsubseteq \perp$, where a is a fresh individual name and D' is a fresh concept name.

---

[3]Such conjunctions are also called *Boolean knowledge bases* in the literature [BGL12].

It is easy to check with the help of the introduced definitions that $\langle \mathcal{T}, \mathcal{A} \rangle$ is consistent iff ϕ is satisfiable. Since consistency in $\mathcal{ELO}_\perp$ is decidable in polynomial time [BBL05], this implies the claim. $\qquad \square$

3 A Lower Bound

We first show the negative result of this report, namely that satisfiability in $\mathcal{EL}$-LTL w.r.t. rigid concept names is already NExpTime-hard, as in $\mathcal{ALC}$-LTL. However, allowing also rigid role names does not further increase the complexity, and we show a matching upper bound in Section 4. Furthermore, satisfiability without rigid names is in PSpace—as for propositional LTL—which we also show in Section 4.

NExpTime-hardness is shown in two steps: we first reduce the 2^{n+1}-bounded domino problem [Lew78, BGG97] to the satisfiability problem in $\mathcal{EL}_\perp$-LTL, and afterwards get rid of the unwanted constructor $\perp$ in the axioms. The basic idea of the first reduction is the same as for $\mathcal{ALC}$-LTL in [BGL12], with some added difficulties due to the lower expressivity of $\mathcal{EL}_\perp$. We will describe the differences to the proof in [BGL12] in detail during the construction.

We start introducing the bounded version of the domino problem used in our reduction. A *domino system* is a triple $\mathcal{D} = (D, H, V)$, where D is a finite set of *domino types* and $H, V \subseteq D \times D$ are the horizontal and vertical *matching conditions*. Let $\mathcal{D}$ be a domino system and $I = d_0, \dots, d_{n-1} \in D^n$ an *initial condition*, which is a sequence of domino types of length $n > 0$. A mapping $\tau \colon \{0, \dots, 2^{n+1} - 1\} \times \{0, \dots, 2^{n+1} - 1\} \to D$ is a 2^{n+1}-*bounded solution* of $\mathcal{D}$ respecting the initial condition I iff, for all $x, y < 2^{n+1}$, the following holds:

- If $\tau(x, y) = d$ and $\tau(x \oplus_{2^{n+1}} 1, y) = d'$, then $(d, d') \in H$;

- If $\tau(x, y) = d$ and $\tau(x, y \oplus_{2^{n+1}} 1) = d'$, then $(d, d') \in V$;

- $\tau(i, 0) = d_i$ for $i < n$;

where $\oplus_{2^{n+1}}$ denotes addition modulo 2^{n+1}. It is shown in [BGG97, Theorem 6.1.2] that there is a domino system $\mathcal{D} = (D, H, V)$ such that, given an initial condition $I = d_0, \dots, d_{n-1} \in D^n$, the problem of deciding if $\mathcal{D}$ has a 2^{n+1}-bounded solution respecting I is NExpTime-hard. In what follows, we show that this problem can be reduced in polynomial time to satisfiability in $\mathcal{EL}_\perp$-LTL w.r.t. rigid concept names.

In our reduction, we discern *global*[4] concept names that are flexible and are satisfied either by all individuals of the domain or by none; in contrast, *local*

---

[4]Not to be confused with *rigid* or *always* (in time).

concept names are rigid and used to identify specific domain elements. We need the following concept and individual names:

- an individual name a;

- flexible (global) concept names G_d, G_d^h, G_d^v, and a rigid (local) concept name L_d for all $d \in D$;

- rigid (local) concept names $X_0, \ldots, X_n$ and $Y_0, \ldots, Y_n$ that are used to realize two binary counters modulo 2^{n+1}, where the X-counter describes the horizontal and the Y-counter the vertical position of a domino;

- flexible (global) concept names $Z_0, \ldots, Z_{2n+1}$, $Z_0^h \ldots, Z_{2n+1}^h$, $Z_0^v, \ldots, Z_{2n+1}^v$ that are used to realize three binary counters modulo 2^{2n+2}, whose function is explained below;

- concept names $\overline{X}_0, \ldots, \overline{X}_n$, $\overline{Y}_0, \ldots, \overline{Y}_n$, $\overline{Z}_0, \ldots, \overline{Z}_{2n+1}$, $\overline{Z}_0^h, \ldots, \overline{Z}_{2n+1}^h$, and $\overline{Z}_0^v, \ldots, \overline{Z}_{2n+1}^v$ representing the complements of above counters;

- auxiliary flexible concept names N, $E_0^h, \ldots, E_{2n+1}^h$, $E_0^v, \ldots, E_{2n+1}^v$.

The first $n + 1$ bits of the Z-, Z^h- and Z^v-counters are used to represent 2^{n+1} horizontal components $0 \le x < 2^{n+1}$, and the second $n + 1$ bits of these counters are used to represent 2^{n+1} vertical components $0 \le y < 2^{n+1}$. By counting with the Z-counter up to 2^{2n+2} in the temporal dimension, we ensure that every position $(x, y) \in \{0, \ldots, 2^{n+1} - 1\} \times \{0, \ldots, 2^{n+1} - 1\}$ is represented at some time point. To count, we enforce that, for every possible value of the Z-counter, there is a world where a belongs to the concepts from the corresponding subset of $\{Z_0, \ldots, Z_{2n+1}\}$. We will restrict the concept names Z_i to be global, and thus the value of the Z-counter is transferred to all other elements of the domain. For every position given by the Z-counter, the Z^h- and Z^v-counters represent the top and right neighbor position, respectively.

The rigid concept names $X_0, \ldots, X_n$ and $Y_0, \ldots, Y_n$ are then used to ensure that, in every world, there is one individual whose X- and Y-values match the value of the global Z-counter. Since they are rigid, this enforces that every position $(x, y) \in \{0, \ldots, 2^{n+1} - 1\} \times \{0, \ldots, 2^{n+1} - 1\}$ is represented by at least one individual in every world. Thus, for every position, we have a world representing it with the help of the global Z-counter, but we also have an individual representing it in every world with the help of the local X- and Y-counters.

Furthermore, appropriate GCIs are used to ensure that (i) every global/local position has exactly one domino type (given by G_d/L_d), and two global domino types for two neighbors (G_d^h, G_d^v); (ii) the domino types of G_d and L_d are the same, and G_d^h/G_d^v represent the same types as the value of L_d at the individuals corresponding to the correct neighbors (ii) the horizontal and vertical matching conditions are respected; and (iii) the initial condition is satisfied.

One of the main differences to the proof for $\mathcal{ALC}$-LTL [BGL12] lies in the presence of three global domino types. In $\mathcal{ALC}$-LTL, it was enough to have one local and one global type in order to enforce the matching conditions. Here, we enforce the matching conditions globally and then ensure that the local types of certain individuals are the same. Another difference is the presence of the concept names of the form $\overline{X_i}$ representing the complements of the various counters. In $\mathcal{ALC}$, these can be directly expressed as $\neg X_i$.

We now construct the $\mathcal{EL}$-LTL-formula $\phi_{\mathcal{D},I}$ as the conjunction of the following formulae:

- For every possible value of the Z-counter, there is a world where a belongs to the concepts from the corresponding subset of $\{Z_0, \dots, Z_{2n+1}\}$:

$$\Box \bigwedge_{0 \leq i \leq 2n+1} \left(\left(\bigwedge_{0 \leq j < i} Z_j(a) \right) \leftrightarrow \left(Z_i(a) \leftrightarrow \bigcirc \neg Z_i(a) \right) \right)$$

  This formula expresses that the i-th bit of the Z-counter is flipped from one world to the next iff all preceding bits were true. Thus, the value of the Z-counter at the next world is equal to the value at the current world incremented by one.

- In every world, the counters Z^h and Z^v are synchronized to the Z-counter, meaning that a belongs to the concepts from the subsets of $\{Z_0^h, \dots, Z_{2n+1}^h\}$ and $\{Z_0^v, \dots, Z_{2n+1}^v\}$ that point to the right and top neighbor position, respectively, of the position distinguished by the Z-counter. This is enforced using formulae similar to the ones for the Z-counter above. First, the horizontal component of the Z^h-counter is equal to the horizontal component of the Z-counter plus 1:

$$\Box \bigwedge_{0 \leq i \leq n} \left(\left(\bigwedge_{0 \leq j < i} Z_j(a) \right) \leftrightarrow \left(Z_i(a) \leftrightarrow \neg Z_i^h(a) \right) \right)$$

  The vertical component of the Z^h-counter is equal to that of the Z-counter:

$$\Box \bigwedge_{n+1 \leq i \leq 2n+1} \left(Z_i(a) \leftrightarrow Z_i^h(a) \right)$$

  And similarly for the Z^v-counter:

$$\Box \bigwedge_{n+1 \leq i \leq 2n+1} \left(\left(\bigwedge_{n+1 \leq j < i} Z_j(a) \right) \leftrightarrow \left(Z_i(a) \leftrightarrow \neg Z_i^v(a) \right) \right)$$

$$\Box \bigwedge_{0 \leq i \leq n} \left(Z_i(a) \leftrightarrow Z_i^v(a) \right)$$

- The values of the three global counters Z, Z^h, and Z^v are shared by all individuals in each world:

$$\Box \bigwedge_{0 \le i \le 2n+1} \Bigg(\Big((\top \sqsubseteq Z_i) \vee (Z_i \sqsubseteq \bot) \Big) \wedge$$
$$\Big((\top \sqsubseteq Z_i^h) \vee (Z_i^h \sqsubseteq \bot) \Big) \wedge \Big((\top \sqsubseteq Z_i^v) \vee (Z_i^v \sqsubseteq \bot) \Big) \Bigg)$$

- In every world, there is at least one individual for which the combined values of the X- and the Y-counter correspond to the value of the global Z-counter in this world:

$$\Box \Bigg(\neg(N \sqsubseteq \bot) \wedge \bigwedge_{0 \le i \le n} (N \sqcap Z_i \sqsubseteq X_i) \wedge \bigwedge_{n+1 \le i \le 2n+1} (N \sqcap Z_i \sqsubseteq Y_{i-(n+1)}) \wedge$$
$$\bigwedge_{0 \le i \le n} (N \sqcap X_i \sqsubseteq Z_i) \wedge \bigwedge_{n+1 \le i \le 2n+1} (N \sqcap Y_{i-(n+1)} \sqsubseteq Z_i) \Bigg)$$

Since the concept names X_i, Y_i are rigid, this ensures that in every world every possible combination of values of the X- and Y-counters is realized by some individual. For a given such combination, the corresponding individual represents the same value combination in every world.

- The interpretation of the concept names $\overline{Z}_i$, $\overline{Z}_i^h$, $\overline{Z}_i^v$, $\overline{X}_i$, $\overline{Y}_i$ as the complements of Z_i, Z_i^h, Z_i^v, X_i, Y_i is enforced by the following formulae. First, we must restrict $\overline{Z}_i$, $\overline{Z}_i^h$, $\overline{Z}_i^v$ to be global concept names:

$$\Box \bigwedge_{0 \le i \le 2n+1} \Bigg(\Big((\top \sqsubseteq \overline{Z}_i) \vee (\overline{Z}_i \sqsubseteq \bot) \Big) \wedge$$
$$\Big((\top \sqsubseteq \overline{Z}_i^h) \vee (\overline{Z}_i^h \sqsubseteq \bot) \Big) \wedge \Big((\top \sqsubseteq \overline{Z}_i^v) \vee (\overline{Z}_i^v \sqsubseteq \bot) \Big) \Bigg)$$

The complements of the global counters are easy to express:

$$\Box \bigwedge_{0 \le i \le 2n+1} \Bigg(\Big(\overline{Z}_i(a) \leftrightarrow \neg Z_i(a) \Big) \wedge \Big(\overline{Z}_i^h(a) \leftrightarrow \neg Z_i^h(a) \Big) \wedge \Big(\overline{Z}_i^v(a) \leftrightarrow \neg Z_i^v(a) \Big) \Bigg)$$

For the complements of the local counters, we again use the concept name N that marks the individual whose X- and Y-counter values correspond to the current value of the Z-counter:

$$\Box \Bigg(\bigwedge_{0 \le i \le n} (N \sqcap \overline{Z}_i \sqsubseteq \overline{X}_i) \wedge \bigwedge_{n+1 \le i \le 2n+1} (N \sqcap \overline{Z}_i \sqsubseteq \overline{Y}_{i-(n+1)}) \wedge$$
$$\bigwedge_{0 \le i \le n} (N \sqcap \overline{X}_i \sqsubseteq \overline{Z}_i) \wedge \bigwedge_{n+1 \le i \le 2n+1} (N \sqcap \overline{Y}_{i-(n+1)} \sqsubseteq \overline{Z}_i) \Bigg)$$

- Every world gets exactly one (global) domino type that belongs to the position given by the global Z-counter:

$$\Box\left(\bigvee_{d\in D}\left((\top\sqsubseteq G_d)\wedge\bigwedge_{d'\in D\setminus\{d\}}(G_{d'}\sqsubseteq\bot)\right)\right)$$

Furthermore, every world has exactly one global domino type G_d^h and G_d^v for the right and top neighbor positions, respectively (corresponding to the positions given by Z^h and Z^v):

$$\Box\left(\bigvee_{d\in D}\left((\top\sqsubseteq G_d^h)\wedge\bigwedge_{d'\in D\setminus\{d\}}(G_{d'}^h\sqsubseteq\bot)\right)\right)$$

$$\Box\left(\bigvee_{d\in D}\left((\top\sqsubseteq G_d^v)\wedge\bigwedge_{d'\in D\setminus\{d\}}(G_{d'}^v\sqsubseteq\bot)\right)\right)$$

- Given the global types of the neighbor positions, the horizontal and vertical matching condition can be enforced easily:

$$\Box\left(\bigvee_{(d,d')\in H}\left((\top\sqsubseteq G_d)\wedge(\top\sqsubseteq G_{d'}^h)\right)\wedge\bigvee_{(d,d')\in V}\left((\top\sqsubseteq G_d)\wedge(\top\sqsubseteq G_{d'}^v)\right)\right)$$

- To synchronize the domino types G_d, G_d^h, and G_d^v among the different worlds (otherwise G_d^h need not be equal to the value of G_d at the world whose Z-counter is equal to the current Z^h-counter), we use the local (rigid) domino types L_d. First, we ensure that the local type of the individual representing the same position as the current world is the same as the current global type:

$$\Box\bigwedge_{d\in D}\left((N\sqcap G_d\sqsubseteq L_d)\wedge(N\sqcap L_d\sqsubseteq G_d)\right)$$

Since the concept names L_d are rigid, this type is then associated with the individual in every world. And because every world has exactly one global domino type G_d (which is shared by all its individuals), every individual also has exactly one local domino type: the one of the world representing the same position.

To synchronize the domino types of the neighbors given by G_d^h and G_d^v, we employ the auxiliary concept names E_i^h, E_i^v:

$$\Box\bigwedge_{0\leq i\leq n}\left((Z_i^h\sqcap X_i\sqsubseteq E_i^h)\wedge(\overline{Z}_i^h\sqcap\overline{X}_i\sqsubseteq E_i^h)\wedge\right.$$
$$\left.(Z_i^v\sqcap X_i\sqsubseteq E_i^v)\wedge(\overline{Z}_i^v\sqcap\overline{X}_i\sqsubseteq E_i^v)\right)$$

$$\Box\bigwedge_{n+1\leq i\leq 2n+1}\left((Z_i^h\sqcap Y_{i-(n+1)}\sqsubseteq E_i^h)\wedge(\overline{Z}_i^h\sqcap\overline{Y}_{i-(n+1)}\sqsubseteq E_i^h)\wedge\right.$$
$$\left.(Z_i^v\sqcap Y_{i-(n+1)}\sqsubseteq E_i^v)\wedge(\overline{Z}_i^v\sqcap\overline{Y}_{i-(n+1)}\sqsubseteq E_i^v)\right)$$

In this way, the interpretation of $E_1^h \sqcap \cdots \sqcap E_{2n+1}^h$ must include all those domain elements whose X- and Y-counters match the current Z^h-counter. This includes in particular the one individual that was created in the corresponding world using the literal $\neg(N \sqsubseteq \bot)$—at which the local domino type equals the current global domino type. Thus, all that remains to do is to ensure that the global domino type G_d^h matches the local domino type L_d at all domain elements satisfying $E_1^h \sqcap \cdots \sqcap E_{2n+1}^h$. Of course, similar arguments apply for the vertical direction.

$$\Box\Big((E_0^h \sqcap \ldots \sqcap E_{2n+1}^h \sqcap G_d^h \sqsubseteq L_d) \wedge (E_0^h \sqcap \ldots \sqcap E_{2n+1}^h \sqcap L_d \sqsubseteq G_d^h) \wedge$$
$$(E_0^v \sqcap \ldots \sqcap E_{2n+1}^v \sqcap G_d^v \sqsubseteq L_d) \wedge (E_0^v \sqcap \ldots \sqcap E_{2n+1}^v \sqcap L_d \sqsubseteq G_d^v)\Big)$$

- It remains to represent the initial condition $I = d_0, \ldots, d_{n-1}$. For this, we use the following formula for all $i = 0, \ldots, n-1$:

$$\Box\Big((C_Z^x = i) \sqcap \overline{Z}_{n+1} \sqcap \cdots \sqcap \overline{Z}_{2n+1} \sqsubseteq G_{d_i}\Big),$$

where, for any $b_j \in \{0, 1\}$, $0 \le j \le n$,

$$\Big(C_Z^x = \sum_{0 \le j \le n} 2^j * b_j\Big) := \prod_{\substack{0 \le j \le n \\ b_j = 0}} \overline{Z}_j \sqcap \prod_{\substack{0 \le j \le n \\ b_j = 1}} Z_j.$$

This conjunction identifies a particular x-position in the Z-counter. When additionally the y-component of the Z-counter is 0, then the corresponding type of the initial condition is enforced.

This finishes the definition of the $\mathcal{EL}_\bot$-LTL-formula $\phi_{\mathcal{D},I}$, which is the conjunction of all the $\Box$-formulae introduced above. It is easy to see that the size of $\phi_{\mathcal{D},I}$ is polynomial in n. Moreover, $\phi_{\mathcal{D},I}$ is satisfiable iff $\mathcal{D}$ has a 2^{n+1}-bounded solution respecting I.

In the last step, we describe how to eliminate the use of the bottom constructor from this reduction. We follow the idea of [BBL05] and introduce a new (rigid) concept name L and a new role name r for which the following formula ϕ_L must be satisfied:
$$\neg L(a) \wedge \Box(\exists r.L \sqsubseteq L)$$
By replacing the negated GCI $\neg(N \sqsubseteq \bot)$ in $\phi_{\mathcal{D},I}$ with $\top \sqsubseteq \exists r.N$, we ensure that

- all individuals representing the doubly exponentially many positions are connected to a via the role r (at the time point that represents the same position via the Z-counter), and

- the individual represented by a as well as those mentioned above do not satisfy L at any time point (since L is rigid).

This means that we can now use L instead of $\bot$ at all other places in the formula $\phi_{\mathcal{D},I}$ without changing the semantics. The reason for this is that it suffices to enforce the GCIs of the form $Z_i \sqsubseteq \bot$ at the individuals representing the 2^{2n+2} relevant positions. We denote by $\phi'_{\mathcal{D},I}$ the formula resulting from $\phi_{\mathcal{D},I}$ by doing the described replacements. We now have that $\phi'_{\mathcal{D},I} \wedge \phi_L$ is satisfiable iff $\mathcal{D}$ has a 2^{n+1}-bounded solution respecting I. NExpTime-hardness of the latter problem [BGG97] yields the following result.

Theorem 3.1. *If* $\mathsf{N_{RC}} \neq \emptyset$, *then satisfiability in* $\mathcal{EL}$-*LTL is* NExpTime-*hard.*

4   Two Upper Bounds

We now show that we can match the lower bound from the previous section even if rigid roles are allowed in addition. On the other hand, if no rigid names are used, then satisfiability in $\mathcal{EL}$-LTL becomes PSpace-complete. Our proofs of both upper bounds follow the basic approach from [BGL12], but additionally utilize the characteristics of $\mathcal{EL}$. In a nutshell, the problem of checking if an $\mathcal{EL}$-LTL-formula is satisfiable is split into two separate satisfiability tests—one for an LTL-formula and one for a conjunction of $\mathcal{EL}$-literals.

In the following, let ϕ be an $\mathcal{EL}$-LTL-formula to be tested for satisfiability. The *propositional abstraction* ϕ^{p} of ϕ is created by replacing each axiom by a propositional variable such that there is a 1–1 relationship between the axioms $\alpha_1, \ldots, \alpha_n$ occurring in ϕ and the propositional variables $p_1, \ldots, p_n$ used for the abstraction. In what follows, we assume that p_i was used to replace α_i for all i, $1 \leq i \leq n$. For a subset $X \subseteq \{p_1, \ldots, p_n\}$, we denote by $\overline{X}$ its complement $\{p_1, \ldots, p_n\} \setminus X$.

We now consider sets of the form $\mathcal{S} \subseteq 2^{\{p_1, \ldots, p_n\}}$ that constrain the types of interpretations allowed to occur in the model of ϕ. Every such set induces the LTL-formula

$$\phi^{\mathsf{p}}_{\mathcal{S}} := \phi^{\mathsf{p}} \wedge \square \left(\bigvee_{X \in \mathcal{S}} \left(\bigwedge_{p \in X} p \wedge \bigwedge_{p \in \overline{X}} \neg p \right) \right)$$

that expresses satisfiability of ϕ^{p} in an LTL-structure that is restricted to only use the worlds contained in $\mathcal{S}$.

The satisfiability of ϕ implies the satisfiability of $\phi^{\mathsf{p}}_{\mathcal{S}}$ for some $\mathcal{S}$. However, guessing such a set $\mathcal{S}$ and then testing whether the induced formula $\phi^{\mathsf{p}}_{\mathcal{S}}$ is satisfiable is not sufficient for checking satisfiability of ϕ. It must also be checked whether $\mathcal{S}$ can indeed be induced by some $\mathcal{EL}$-LTL-structure that respects the rigid concept and role names.

Assume for now that a set $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1, \ldots, p_n\}}$ is given. For every i, $1 \leq i \leq k$, and every flexible (concept or role) name x occurring in ϕ, we introduce a copy $x^{(i)}$, the *i-th copy of* x. The axiom $\alpha_j^{(i)}$ is obtained from α_j by

replacing every occurrence of a flexible name by its i-th copy. In this way, the set $\mathcal{S}$ induces the following conjunction of $\mathcal{EL}$-literals:

$$\chi_{\mathcal{S}} := \bigwedge_{i=1}^{k} \left(\bigwedge_{p_j \in X_i} \alpha_j^{(i)} \wedge \bigwedge_{p_j \in \overline{X_i}} \neg \alpha_j^{(i)} \right)$$

The following fact has been shown in [BGL12] for $\mathcal{ALC}$-LTL, but also applies to our setting since every $\mathcal{EL}$-LTL-formula is also an $\mathcal{ALC}$-LTL-formula.

Proposition 4.1 ([BGL12]). *The $\mathcal{EL}$-LTL-formula ϕ is satisfiable iff there is a set $\mathcal{S} \subseteq 2^{\{p_1,\dots,p_n\}}$ such that $\phi_{\mathcal{S}}^{\mathsf{p}}$ and $\chi_{\mathcal{S}}$ are both satisfiable.*

The first upper bound now follows from this proposition, Lemma 2.1, and the observation that satisfiability of $\phi_{\mathcal{S}}^{\mathsf{p}}$ can be checked in exponential time.

Theorem 4.2. *Satisfiability in $\mathcal{EL}$-LTL is decidable in* NExpTime.

Proof. To check the $\mathcal{EL}$-LTL-formula ϕ for satisfiability, we first guess a set $\mathcal{S} = \{X_1, \dots, X_k\} \subseteq 2^{\{p_1,\dots,p_n\}}$ in exponential time, and then construct $\phi_{\mathcal{S}}^{\mathsf{p}}$ and $\chi_{\mathcal{S}}$ as above. It was shown in [BGL12] that satisfiability of $\phi_{\mathcal{S}}^{\mathsf{p}}$ can be tested in exponential time by appropriately modifying a Büchi automaton accepting all LTL-structures satisfying ϕ^{p} and testing this automaton for emptiness. Furthermore, $\chi_{\mathcal{S}}$ is of exponential size in the size of ϕ, and thus can be tested for satisfiability in exponential time using Lemma 2.1. In conclusion, Proposition 4.1 yields the desired upper bound. □

For the case that no rigid names are used, PSpace-hardness of satisfiability in $\mathcal{EL}$-LTL directly follows from PSpace-completeness of the satisfiability problem in propositional LTL [SC85]. Obtaining inclusion in PSpace is a little more involved.

We consider again an $\mathcal{EL}$-LTL-formula ϕ (without rigid names). By Proposition 4.1, ϕ is satisfiable iff there is a set $\mathcal{S} = \{X_1, \dots, X_k\} \subseteq 2^{\{p_1,\dots,p_n\}}$ such that both $\phi_{\mathcal{S}}^{\mathsf{p}}$ and $\chi_{\mathcal{S}}$ are satisfiable. There are three problems with applying this reduction directly. First, the set $\mathcal{S}$ is of exponential size, and thus guessing (and storing) it is not possible in PSpace. Second, the reduction to the emptiness of a Büchi automaton employed in [BGL12] to test satisfiability of $\phi_{\mathcal{S}}^{\mathsf{p}}$ yields a complexity of ExpTime. Finally, checking $\chi_{\mathcal{S}}$ for satisfiability using Lemma 2.1 also requires exponential time.

The latter problem was solved in [BGL12] by observing that the conjunctions

$$\chi_{X_i} := \bigwedge_{p_j \in X_i} \alpha_j^{(i)} \wedge \bigwedge_{p_j \in \overline{X_i}} \neg \alpha_j^{(i)}$$

do not share any concept or role names, and thus can be independently tested for satisfiability. By Lemma 2.1, each of these (exponentially many) tests requires only polynomial time.

To solve the first two problems, we propose a procedure based on the original polynomial-space-bounded Turing machines for LTL-satisfiability constructed in [SC85]. Given a propositional LTL-formula ϕ^{p}, the machine $\mathcal{M}_{\phi^{\mathsf{p}}}$ iteratively guesses complete sets of (negated) subformulae of ϕ^{p} specifying which subformulae are satisfied at each point in time. Every such set induces a unique world $X_i \subseteq \{p_1, \ldots, p_n\}$ containing the propositional variables that are true.

In [SC85, Theorem 4.7], it is shown that if ϕ^{p} is satisfiable, there must be a *periodic* model of ϕ^{p} with a period that is exponential in the size of ϕ^{p}. Hence, $\mathcal{M}_{\phi^{\mathsf{p}}}$ first guesses two polynomial-sized indices specifying the beginning and end of the first period. Then it continuously increments a (polynomial-sized) counter and in each step guesses a complete set of (negated) subformulae of ϕ^{p}. It then checks Boolean consistency of this set and consistency with the set of the previous time point according to the temporal operators. For example, if the previous set contains the formula $p_1 \cup p_2$, then either it also contains p_2 or it must contain p_1 and the current set must contain $p_1 \cup p_2$. In this way, the satisfaction of the $\cup$-formula is deferred to the next time point.

In each step, the oldest set is discarded and replaced by the next one. When the counter reaches the beginning of the period, it stores the current set and continues until it reaches the end of the period. At that point, instead of guessing the next set of subformulae, the set stored at the beginning of the period is used and checked for consistency with the previous set as described above. $\mathcal{M}_{\phi^{\mathsf{p}}}$ additionally has to ensure that all $\cup$-subformulae are satisfied within the period. Thus, the Turing machine never has to remember more than three sets of polynomial size.

We now modify this procedure to obtain the claimed upper bound.

Theorem 4.3. *If $\mathsf{N}_{\mathsf{RC}} = \mathsf{N}_{\mathsf{RR}} = \emptyset$, then satisfiability in $\mathcal{EL}$-LTL is decidable in* PSpace.

Proof. By Proposition 4.1, the satisfiability of an $\mathcal{EL}$-LTL-formula ϕ is equivalent to the existence of a set $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1, \ldots, p_n\}}$ such that $\phi_{\mathcal{S}}^{\mathsf{p}}$ and all χ_{X_i}, $1 \leq i \leq k$, are satisfiable. Note that the only difference between ϕ^{p} and $\phi_{\mathcal{S}}^{\mathsf{p}}$ is the requirement that all worlds in an LTL-structure satisfying $\phi_{\mathcal{S}}^{\mathsf{p}}$ should be included in $\mathcal{S}$. It is thus not necessary to actually construct the whole set $\mathcal{S}$—it is enough to show that each world X we encounter when checking ϕ^{p} (*not* $\phi_{\mathcal{S}}^{\mathsf{p}}$) for satisfiability induces a satisfiable conjunction χ_X.

To check ϕ for satisfiability, we can thus run a modified version of the Turing machine $\mathcal{M}_{\phi^{\mathsf{p}}}$ that tests for each guessed complete set of subformulae whether the induced world satisfies the additional requirement that χ_X is satisfiable. The latter tests can be done in polynomial time. The set $\mathcal{S}$ from Proposition 4.1 corresponds to the set of all worlds X encountered during a run of $\mathcal{M}_{\phi^{\mathsf{p}}}$. As described before, this set does not have to be stored explicitly.

14

Since all this can be done with a nondeterministic Turing machine using only polynomial space (in the size of ϕ), according to [Sav70], satisfiability in $\mathcal{EL}$-LTL can be decided in PSpace. $\qquad\square$

5 Global GCIs

In this section, we propose an approach that makes it possible to consider rigid names while the complexity is still in PSpace, and hence overcome the rather negative result of NExpTime-hardness for this case. Specifically, we restrict the $\mathcal{EL}$-LTL-formulae to be of the form $(\square \bigwedge \mathcal{T}) \wedge \psi$, where $\mathcal{T}$ is a TBox whose GCIs should hold *globally*, i.e., at every point in time, and ψ is an $\mathcal{EL}$-LTL-formula that contains only assertions [BGL12].

5.1 Canonical Models

To facilitate the proofs, we recall the construction of *canonical models* for deciding subsumption w.r.t. $\mathcal{EL}$-knowledge bases $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ [BBL05]. In the following, we denote by $\mathsf{N_C}(\mathcal{T})$ the set of all concept names occurring in the TBox $\mathcal{T}$, and similarly define $\mathsf{N_I}(\mathcal{A})$, $\mathsf{N_R}(\mathcal{T})$, and so on. We denote by $\mathsf{Sub}(\mathcal{T})$ the set of all concepts occurring as subconcepts in axioms of $\mathcal{T}$. Further, a *concept over $\mathcal{T}$* is a concept that is constructed from the concept and role names occurring in $\mathcal{T}$. If it contains only rigid names, then it is called a *rigid concept over $\mathcal{T}$*. An *atom over $\mathcal{T}$* is a concept over $\mathcal{T}$ of the form A or $\exists r.A$, where $A \in \mathsf{N_C}(\mathcal{T}) \cup \{\top\}$ and $r \in \mathsf{N_R}(\mathcal{T})$.

We assume the KB to be *normalized*, i.e., the TBox may only contain GCIs of the following forms:

$$A_1 \sqcap A_2 \sqsubseteq B, \ \exists r.A \sqsubseteq B, \ \text{and} \ A \sqsubseteq \exists r.B,$$

and the ABox may only contain assertions of the forms

$$A(a), \ r(a,b), \ \text{and} \ \exists r.A(a)$$

where $A_1, A_2, A, B \in \mathsf{N_C} \cup \{\top\}$, $r \in \mathsf{N_R}$, and $a, b \in \mathsf{N_I}$. We further assume that all concept and role names occurring in $\mathcal{K}$ also occur in $\mathcal{T}$. These assumptions are clearly without loss of generality [BBL05].

Definition 5.1. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a normalized $\mathcal{EL}$-knowledge base. We first define the set*
$$\Delta_{\mathsf{u}}^{\mathcal{I}_\mathcal{K}} := \{c_A \mid A \in \mathsf{N_C}(\mathcal{T}) \cup \{\top\}\}.$$

The canonical interpretation $\mathcal{I}_\mathcal{K}$ for $\mathcal{K}$ is defined as follows, for all $a \in \mathsf{N_I}$, $A \in \mathsf{N_C}$, and $r \in \mathsf{N_R}$:

$$\Delta^{\mathcal{I}_\mathcal{K}} := \mathsf{N_I}(\mathcal{A}) \cup \Delta_\mathsf{u}^{\mathcal{I}_\mathcal{K}},$$

$$a^{\mathcal{I}_\mathcal{K}} := a,$$

$$A^{\mathcal{I}_\mathcal{K}} := \{a \in \mathsf{N_I}(\mathcal{A}) \mid \mathcal{K} \models A(a)\} \cup$$
$$\{c_B \in \Delta_\mathsf{u}^{\mathcal{I}_\mathcal{K}} \mid \mathcal{T} \models B \sqsubseteq A\}, \ and$$

$$r^{\mathcal{I}_\mathcal{K}} := \{(a,b) \mid r(a,b) \in \mathcal{A}\} \cup$$
$$\{(a,c_B) \in \mathsf{N_I}(\mathcal{A}) \times \Delta_\mathsf{u}^{\mathcal{I}_\mathcal{K}} \mid \mathcal{K} \models \exists r.B(a)\} \cup$$
$$\{(c_A,c_B) \in \Delta_\mathsf{u}^{\mathcal{I}_\mathcal{K}} \times \Delta_\mathsf{u}^{\mathcal{I}_\mathcal{K}} \mid \mathcal{T} \models A \sqsubseteq \exists r.B\}.$$

Based on $\mathcal{I}_\mathcal{K}$, we now define the rigid canonical interpretation $\mathcal{I}'_\mathcal{K}$ for $\mathcal{K}$. Similar as above, we first define the set

$$\Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}} := \{c'_A \mid A \in \mathsf{N_C}(\mathcal{T}) \cup \{\top\}\}.$$

For all $a \in \mathsf{N_I}$ and $A \in \mathsf{N_{RC}}$:

$$\Delta^{\mathcal{I}'_\mathcal{K}} := \Delta^{\mathcal{I}_\mathcal{K}} \cup \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}},$$

$$a^{\mathcal{I}'_\mathcal{K}} := a, \ and$$

$$A^{\mathcal{I}'_\mathcal{K}} := \{e \in \Delta^{\mathcal{I}_\mathcal{K}} \mid e \in A^{\mathcal{I}_\mathcal{K}}\} \cup \{c'_B \in \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}} \mid \mathcal{T} \models B \sqsubseteq A\}.$$

The interpretation of all $A \in \mathsf{N_C} \setminus \mathsf{N_{RC}}$, $r \in \mathsf{N_{RR}}$, and $s \in \mathsf{N_R} \setminus \mathsf{N_{RR}}$, is now specified iteratively, where we assume that all interpretations $\mathcal{I}'_{\mathcal{K},i}$ are defined as $\mathcal{I}'_\mathcal{K}$ up to this point and further as below:

$$A^{\mathcal{I}'_{\mathcal{K},0}} := \{c'_B \in \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}} \mid \mathcal{T} \models B \sqsubseteq A\},$$

$$r^{\mathcal{I}'_{\mathcal{K},0}} := \{(e_1, e_2) \in \Delta^{\mathcal{I}_\mathcal{K}} \times \Delta^{\mathcal{I}_\mathcal{K}} \mid (e_1, e_2) \in r^{\mathcal{I}_\mathcal{K}}\} \cup$$
$$\{(c'_A, c'_B) \in \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}} \times \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}} \mid \mathcal{T} \models A \sqsubseteq \exists r.B\},$$

$$s^{\mathcal{I}'_{\mathcal{K},0}} := \{(c'_A, c'_B) \in \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}} \times \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}} \mid \mathcal{T} \models A \sqsubseteq \exists s.B\},$$

$$A^{\mathcal{I}'_{\mathcal{K},i+1}} := \{e \in C^{\mathcal{I}'_{\mathcal{K},i}} \mid \mathcal{T} \models C \sqsubseteq A, \ C \text{ an atom over } \mathcal{T}\},$$

$$r^{\mathcal{I}'_{\mathcal{K},i+1}} := r^{\mathcal{I}'_{\mathcal{K},i}} \cup \{(e, c'_B) \in C^{\mathcal{I}'_{\mathcal{K},i}} \times \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}} \mid \mathcal{T} \models C \sqsubseteq \exists r.B, \ C \text{ an atom over } \mathcal{T}\},$$

$$s^{\mathcal{I}'_{\mathcal{K},i+1}} := s^{\mathcal{I}'_{\mathcal{K},i}} \cup \{(e, c'_B) \in C^{\mathcal{I}'_{\mathcal{K},i}} \times \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}} \mid \mathcal{T} \models C \sqsubseteq \exists s.B, \ C \text{ an atom over } \mathcal{T}\},$$

$$A^{\mathcal{I}'_\mathcal{K}} := \bigcup_{i \geq 0} A^{\mathcal{I}'_{\mathcal{K},i}},$$

$$r^{\mathcal{I}'_\mathcal{K}} := \bigcup_{i \geq 0} r^{\mathcal{I}'_{\mathcal{K},i}}, \ and$$

$$s^{\mathcal{I}'_\mathcal{K}} := \bigcup_{i \geq 0} s^{\mathcal{I}'_{\mathcal{K},i}}.$$

Since the domain of $\mathcal{I}'_\mathcal{K}$ is of polynomial size, it is easy to see that the above construction is finished after polynomially many iterations, and thus $\mathcal{I}'_\mathcal{K}$ can be constructed in polynomial time.

For future reference, we next state the known result that the canonical interpretation of a KB always is a model of that KB.

Proposition 5.2 ([BBL05]). *For a normalized $\mathcal{EL}$-KB $\mathcal{K}$, we have $\mathcal{I}_\mathcal{K} \models \mathcal{K}$.* $\square$

We now refer to so-called *simulations*, which in [Baa03] are described as binary relations between nodes of two so-called $\mathcal{EL}$-description graphs that respect the labels and edges of those graphs. Such an $\mathcal{EL}$-description graph is obtained for an interpretation $\mathcal{I}$ by regarding $\mathcal{I}$ as a graph such that the domain elements are the nodes, labeled by the concept names the elements satisfy; and the (labeled) edges are given by the roles connecting the elements in $\mathcal{I}$. We define the notion of simulation directly w.r.t. two interpretations.

Definition 5.3. *A simulation $\sigma : \mathcal{I} \to \mathcal{J}$ (of $\mathcal{I}$ by $\mathcal{J}$), is a relation $\sigma \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{J}$ iff the following hold, for all $(x, y) \in \sigma$:*

- *$x \in A^\mathcal{I}$ implies $y \in A^\mathcal{J}$, for all $A \in \mathsf{N_C}$; and*

- *$(x, x') \in r^\mathcal{I}$ implies that there is an element $y' \in \Delta^\mathcal{J}$, such that $(x', y') \in \sigma$ and $(y, y') \in r^\mathcal{J}$, for all $r \in \mathsf{N_R}$.*

It is easy to inductively construct a simulation of the canonical model of a KB $\mathcal{K}$ by any other model of $\mathcal{K}$.

Proposition 5.4. *Let $\mathcal{J}$ be a model of a knowledge base $\mathcal{K}$. Then there is a simulation σ of $\mathcal{I}_\mathcal{K}$ by $\mathcal{J}$ such that $(a, a^\mathcal{J}) \in \sigma$, for all $a \in \mathsf{N_I}$.* $\square$

We now prove some auxiliary lemmas concerning $\mathcal{I}'_\mathcal{K}$. The first is a characterization of the behavior of $\mathcal{I}'_{\mathcal{K},i}$, $i \geq 0$, on the newly introduced elements from $\Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}}$. Since it is independent of i, it in particular shows that the interpretation of these elements is never changed.

Lemma 5.5. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an $\mathcal{EL}$-knowledge base. For all $c'_B \in \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}}$, atoms C over $\mathcal{T}$, and $i \geq 0$, we have $c'_B \in C^{\mathcal{I}'_{\mathcal{K},i}}$ iff $\mathcal{T} \models B \sqsubseteq C$.*

Proof. We show the claim by induction on i and start with $i = 0$. If $C \in \mathsf{N_C}(\mathcal{T})$, then it holds by the definition of $C^{\mathcal{I}'_{\mathcal{K},0}}$. For the case that $C = \exists r.A$ with $r \in \mathsf{N_R}(\mathcal{T})$ and $A \in \mathsf{N_C}(\mathcal{T})$, assume first that $c'_B \in (\exists r.A)^{\mathcal{I}'_{\mathcal{K},0}}$. Then, there must be an element $c'_D \in \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}}$ such that $(c'_B, c'_D) \in r^{\mathcal{I}'_{\mathcal{K},0}}$ and $c'_D \in A^{\mathcal{I}'_{\mathcal{K},0}}$, and hence $\mathcal{T} \models B \sqsubseteq \exists r.D$ and $\mathcal{T} \models D \sqsubseteq A$, which implies that $\mathcal{T} \models B \sqsubseteq \exists r.A$. Conversely, if $\mathcal{T} \models B \sqsubseteq \exists r.A$, then we have $(c'_B, c'_A) \in r^{\mathcal{I}'_{\mathcal{K},0}}$ and $c'_A \in A^{\mathcal{I}'_{\mathcal{K},0}}$, and hence $c'_B \in (\exists r.A)^{\mathcal{I}'_{\mathcal{K},0}}$.

17

Consider now the case that $i > 0$. If $C \in \mathsf{N}_\mathsf{C}(\mathcal{T})$, then there is an atom D over $\mathcal{T}$ such that $c'_B \in D^{\mathcal{I}'_{\mathcal{K},i-1}}$ and $\mathcal{T} \models D \sqsubseteq C$. By the induction hypothesis, we know that $\mathcal{T} \models B \sqsubseteq D$, and hence $\mathcal{T} \models B \sqsubseteq C$, as claimed. If $C = \exists r.A$ for $r \in \mathsf{N}_\mathsf{R}(\mathcal{T})$ and $A \in \mathsf{N}_\mathsf{C}(\mathcal{T})$, then there is a $c'_E \in \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}}$ such that $(c'_B, c'_E) \in r^{\mathcal{I}'_{\mathcal{K},i}}$ and $c'_E \in A^{\mathcal{I}'_{\mathcal{K},i}}$. From the former and the definition of $r^{\mathcal{I}'_{\mathcal{K},0}}, \ldots, r^{\mathcal{I}'_{\mathcal{K},i}}$, we know that there is an atom D over $\mathcal{T}$ such that $c'_B \in D^{\mathcal{I}'_{\mathcal{K},j}}$ for some $j < i$ and $\mathcal{T} \models D \sqsubseteq \exists r.E$. From the latter, we find an atom F over $\mathcal{T}$ such that $c'_E \in F^{\mathcal{I}'_{\mathcal{K},i-1}}$ and $\mathcal{T} \models F \sqsubseteq A$. By the induction hypothesis, we get $\mathcal{T} \models B \sqsubseteq D$ and $\mathcal{T} \models E \sqsubseteq F$. Putting all this together, we conclude that $\mathcal{T} \models B \sqsubseteq \exists r.A$. $\qquad\square$

The second auxiliary lemma describes the behavior of $\mathcal{I}'_\mathcal{K}$ on the original domain elements of $\mathcal{I}_\mathcal{K}$.

Lemma 5.6. *Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a consistent KB, $e \in \Delta^{\mathcal{I}_\mathcal{K}}$, and an atom C over $\mathcal{T}$. If $e \in C^{\mathcal{I}'_\mathcal{K}}$, then there is a rigid concept C' over $\mathcal{T}$ such that $e \in C'^{\mathcal{I}_\mathcal{K}}$ and $\mathcal{T} \models C' \sqsubseteq C$.*

Proof. For $C \in \mathsf{N}_\mathsf{RC}$, we have $e \in C^{\mathcal{I}_\mathcal{K}}$ by Definition 5.1, and trivially $\mathcal{T} \models C \sqsubseteq C$. For the remaining cases, we prove the claim by induction over the construction of $\mathcal{I}'_\mathcal{K}$, i.e., we show that for all $i \geq 0$, $e \in C^{\mathcal{I}'_{\mathcal{K},i}}$ implies the existence of a concept C' as above. Consider first $i = 0$:

- For $C \in \mathsf{N}_\mathsf{C} \setminus \mathsf{N}_\mathsf{RC}$, the claim is vacuously true since $C^{\mathcal{I}'_{\mathcal{K},0}} \cap \Delta^{\mathcal{I}_\mathcal{K}}$ is empty.

- For $C = \exists r.A$, there must be an element $e' \in \Delta^{\mathcal{I}'_\mathcal{K}}$ such that $(e, e') \in r^{\mathcal{I}'_{\mathcal{K},0}}$ and $e' \in A^{\mathcal{I}'_{\mathcal{K},0}}$. Since $e \in \Delta^{\mathcal{I}_\mathcal{K}}$, we must have $r \in \mathsf{N}_\mathsf{RR}$, $e' \in \Delta^{\mathcal{I}_\mathcal{K}}$, and $(e, e') \in r^{\mathcal{I}_\mathcal{K}}$. But then also $A \in \mathsf{N}_\mathsf{RC}$ and $e' \in A^{\mathcal{I}_\mathcal{K}}$, and thus we can choose $C' := \exists r.A$.

We come to the induction step and assume that $e \in C^{\mathcal{I}'_{\mathcal{K},i+1}} \setminus C^{\mathcal{I}'_{\mathcal{K},i}}$.

- In case $C \in \mathsf{N}_\mathsf{C} \setminus \mathsf{N}_\mathsf{RC}$, we have an atom D over $\mathcal{T}$ such that $e \in D^{\mathcal{I}'_{\mathcal{K},i}}$ and $\mathcal{T} \models D \sqsubseteq C$. By the induction hypothesis, there is a rigid concept C' over $\mathcal{T}$ such that $e \in C'^{\mathcal{I}_\mathcal{K}}$ and $\mathcal{T} \models C' \sqsubseteq D$. But then we also have $\mathcal{T} \models C' \sqsubseteq C$.

- If $C = \exists r.A$, there is an $e' \in \Delta^{\mathcal{I}'_\mathcal{K}}$ such that $(e, e') \in r^{\mathcal{I}'_{\mathcal{K},i+1}}$ and $e' \in A^{\mathcal{I}'_{\mathcal{K},i+1}}$.

  If $e' \in \Delta^{\mathcal{I}_\mathcal{K}}$, then we must have $r \in \mathsf{N}_\mathsf{RR}$, $(e, e') \in r^{\mathcal{I}'_{\mathcal{K},0}}$, and thus $(e, e') \in r^{\mathcal{I}_\mathcal{K}}$. By the assumption that $e \in C^{\mathcal{I}'_{\mathcal{K},i+1}} \setminus C^{\mathcal{I}'_{\mathcal{K},i}}$, we thus obtain $e' \in A^{\mathcal{I}'_{\mathcal{K},i+1}} \setminus A^{\mathcal{I}'_{\mathcal{K},i}}$. By our analysis above, we know that there is a rigid concept C' over $\mathcal{T}$ such that $e' \in C'^{\mathcal{I}_\mathcal{K}}$ and $\mathcal{T} \models C' \sqsubseteq A$. But then $\exists r.C'$ is as required since $e \in (\exists r.C')^{\mathcal{I}_\mathcal{K}}$ and $\mathcal{T} \models \exists r.C' \sqsubseteq \exists r.A$.

  If $e' = c'_B \in \Delta_\mathsf{u}^{\mathcal{I}'_\mathcal{K}}$, then Lemma 5.5 yields that $\mathcal{T} \models B \sqsubseteq A$ and $c'_B \in A^{\mathcal{I}_{\mathcal{K},0}}$. Since $e \in C^{\mathcal{I}'_{\mathcal{K},i+1}} \setminus C^{\mathcal{I}'_{\mathcal{K},i}}$, we know that there is an atom D over $\mathcal{T}$ such that

$e \in D^{\mathcal{I}'_{\mathcal{K},i}}$ and $\mathcal{T} \models D \sqsubseteq \exists r.B$. By the induction hypothesis, there is a rigid concept C' over $\mathcal{T}$ such that $e \in C'^{\mathcal{I}_{\mathcal{K}}}$ and $\mathcal{T} \models C' \sqsubseteq D$. Thus, we conclude that $\mathcal{T} \models C' \sqsubseteq \exists r.A$, as required. $\qquad\qquad\qquad\qquad\qquad\square$

5.2 r-satisfiability and r-completeness

In what follows, let $\phi = (\square \wedge \mathcal{T}) \wedge \psi$ be an $\mathcal{EL}$-LTL-formula with global GCIs to be tested for satisfiability, where $\mathcal{T}$ is a normalized TBox and ψ contains only normalized assertions of the forms $A(a)$ and $r(a,b)$. We again consider a set $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1, \ldots, p_m\}}$, where $\alpha_1, \ldots, \alpha_m$ are the assertions in ψ, and the propositional LTL-formula $\psi_{\mathcal{S}}^{\mathsf{p}}$ from Section 4. We define the ABoxes $\mathcal{A}_i := \{\alpha_j \mid p_j \in X_i\} \cup \{\neg\alpha_j \mid p_j \in \overline{X_i}\}$, for $1 \leq i \leq k$. Observe that we also consider negated assertions $\neg A(a)$ and $\neg r(a,b)$ here, with the obvious semantics that an interpretation $\mathcal{I}$ satisfies them if $a^{\mathcal{I}} \notin A^{\mathcal{I}}$ and $(a^{\mathcal{I}}, b^{\mathcal{I}}) \notin r^{\mathcal{I}}$, respectively.

Similar to before, we now show that the test for satisfiability of ϕ can be again split into two parts. We now restate the property described by $\chi_{\mathcal{S}}$ in terms of the knowledge bases $\langle \mathcal{T}, \mathcal{A}_i \rangle$.

Definition 5.7. *A set $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1, \ldots, p_n\}}$ is* r-satisfiable *if there are interpretations $\mathcal{J}_1, \ldots, \mathcal{J}_k$ such that*

- *each $\mathcal{J}_i$, $1 \leq i \leq k$, is a model of $\langle \mathcal{T}, \mathcal{A}_i \rangle$; and*

- *they share the same domain and respect rigid names (cf. Section 2.3).*

The following result is the same as Proposition 4.1, restated using the above definition.

Proposition 5.8 ([BBL15]). *The $\mathcal{EL}$-LTL-formula ϕ is satisfiable iff there is an r-satisfiable set $\mathcal{S} = \{X_1, \ldots, X_k\} \subseteq 2^{\{p_1, \ldots, p_n\}}$ such that $\phi_{\mathcal{S}}^{\mathsf{p}}$ is satisfiable.* $\qquad\square$

Contrary to the PSPACE-result of Theorem 4.3, we cannot simply split the r-satisfiability test into individual consistency tests for $\langle \mathcal{T}, \mathcal{A}_i \rangle$, due to the presence of rigid names. We can nevertheless combine this test with the satisfiability test for $\phi_{\mathcal{S}}^{\mathsf{p}}$ using the Turing machine $\mathcal{M}_{\phi^{\mathsf{p}}}$ described before, by guessing polynomially many additional assertions that allow us to separate the r-satisfiability test for $\mathcal{S}$ into independent consistency tests.

Definition 5.9. *An ABox type for ϕ is a set*

$$\mathcal{A}_{\mathsf{R}} \subseteq \{A(a), \neg A(a), r(a,b), \neg r(a,b) \mid a, b \in \mathsf{N}_{\mathsf{I}}(\phi), \; A \in \mathsf{N}_{\mathsf{RC}}(\mathcal{T}), r \in \mathsf{N}_{\mathsf{RR}}(\mathcal{T})\}$$

such that $A(a) \in \mathcal{A}_{\mathsf{R}}$ iff $\neg A(a) \notin \mathcal{A}_{\mathsf{R}}$, and $r(a,b) \in \mathcal{A}_{\mathsf{R}}$ iff $\neg r(a,b) \notin \mathcal{A}_{\mathsf{R}}$.

We now consider tuples of the form $(\mathcal{A}_\mathsf{R}, \mathcal{A}'_\mathsf{R})$, where $\mathcal{A}_\mathsf{R}$ is an ABox type for ϕ and

$$\mathcal{A}'_\mathsf{R} \subseteq \{\exists r.A(a) \mid a \in \mathsf{N}_\mathsf{I}(\phi),\ r \in \mathsf{N}_\mathsf{RR}(\mathcal{T}),\ A \in \mathsf{N}_\mathsf{C}(\mathcal{T}) \cup \{\top\}\}.$$

For all i, $1 \le i \le k$, we denote by $\mathcal{K}_\mathsf{R}^i$ the knowledge base $\langle \mathcal{T}, \mathcal{A}_\mathsf{R} \cup \mathcal{A}'_\mathsf{R} \cup \mathcal{A}_i \rangle$ and consider the rigid canonical interpretation $\mathcal{I}'_i := \mathcal{I}'_{[\mathcal{K}_\mathsf{R}^i]^+}$, where $[\mathcal{K}_\mathsf{R}^i]^+$ is equal to $\mathcal{K}_\mathsf{R}^i$ without the negated assertions in $\mathcal{A}_\mathsf{R} \cup \mathcal{A}_i$. A tuple $(\mathcal{A}_\mathsf{R}, \mathcal{A}'_\mathsf{R})$ as above is called r-complete for ϕ (w.r.t. $\mathcal{S}$) if, for all i, $1 \le i \le k$,

- $\mathcal{K}_\mathsf{R}^i$ is consistent, and

- for all $a \in \mathsf{N}_\mathsf{I}(\phi)$, $r \in \mathsf{N}_\mathsf{RR}(\mathcal{T})$, and $A \in \mathsf{N}_\mathsf{C}(\mathcal{T}) \cup \{\top\}$, $a \in (\exists r.A)^{\mathcal{I}'_i}$ implies $\exists r.A(a) \in \mathcal{A}'_\mathsf{R}$.

The idea is that $\mathcal{A}_\mathsf{R}$ fixes the interpretation of the rigid names on the named individuals, and $\mathcal{A}'_\mathsf{R}$ specifies what kind of r-successors (for a rigid r) need to be present at every time point. Note that each $\mathcal{K}_\mathsf{R}^i$ can be seen as a conjunction of $\mathcal{EL}$-literals (whose size is polynomial in the size of ϕ), and thus its consistency can be decided in deterministic polynomial time, by Lemma 2.1. Likewise, since the size of $\mathcal{I}'_i$ is polynomial in the size of ϕ (see Definition 5.1), we can check the second condition in polynomial time (for a single index i).

As described above, we show in Lemmas 5.10 and 5.15 that the existence of an r-complete tuple fully characterizes the r-satisfiability of $\mathcal{S}$.

Lemma 5.10. *If $\mathcal{S}$ is r-satisfiable, then there is an r-complete tuple for ϕ w.r.t. $\mathcal{S}$.*

Proof. Let $\mathcal{J}_1, \ldots, \mathcal{J}_k$ be the interpretations that exist by the r-satisfiability of $\mathcal{S}$. We construct $\mathcal{A}_\mathsf{R}$ as follows:

$$\begin{aligned}
\mathcal{A}_\mathsf{R} := \ & \{A(a) \mid A \in \mathsf{N}_\mathsf{RC}(\mathcal{T}),\ a \in \mathsf{N}_\mathsf{I}(\phi),\ \mathcal{J}_1 \models A(a)\} \cup \\
& \{\neg A(a) \mid A \in \mathsf{N}_\mathsf{RC}(\mathcal{T}),\ a \in \mathsf{N}_\mathsf{I}(\phi),\ \mathcal{J}_1 \not\models A(a)\} \cup \\
& \{r(a,b) \mid r \in \mathsf{N}_\mathsf{RR}(\mathcal{T}),\ a,b \in \mathsf{N}_\mathsf{I}(\phi),\ \mathcal{J}_1 \models r(a,b)\} \cup \\
& \{\neg r(a,b) \mid r \in \mathsf{N}_\mathsf{RR}(\mathcal{T}),\ a,b \in \mathsf{N}_\mathsf{I}(\phi),\ \mathcal{J}_1 \not\models r(a,b)\}.
\end{aligned}$$

This set is obviously an ABox type. Since $\mathcal{J}_1, \ldots, \mathcal{J}_k$ respect the rigid names, each $\mathcal{J}_i$ is also a model of $\mathcal{A}_\mathsf{R}$. We further define

$$\begin{aligned}
\mathcal{A}'_\mathsf{R} := \ & \{\exists r.A(a) \mid r \in \mathsf{N}_\mathsf{RR}(\mathcal{T}),\ A \in \mathsf{N}_\mathsf{C}(\mathcal{T}) \cup \{\top\},\ a \in \mathsf{N}_\mathsf{I}(\phi), \\
& \qquad C \text{ a rigid concept over } \mathcal{T},\ a \in C^{\mathcal{J}_1},\ \mathcal{T} \models C \sqsubseteq \exists r.A\}.
\end{aligned}$$

Since all $\mathcal{J}_i$ agree on all rigid names and satisfy $\mathcal{T}$, they also satisfy $\mathcal{A}'_\mathsf{R}$. This already shows that each $\mathcal{K}_\mathsf{R}^i = \langle \mathcal{T}, \mathcal{A}_\mathsf{R} \cup \mathcal{A}'_\mathsf{R} \cup \mathcal{A}_i \rangle$ is consistent.

For the second condition of Definition 5.9, assume now that $a \in (\exists r.A)^{\mathcal{I}'_i}$ holds for some $a \in \mathsf{N}_\mathsf{I}(\phi)$, $r \in \mathsf{N}_\mathsf{RR}(\mathcal{T})$, and $A \in \mathsf{N}_\mathsf{C}(\mathcal{T}) \cup \{\top\}$. We need to show that we

then have $\exists r.A \in \mathcal{A}'_{\mathsf{R}}$. By Lemma 5.6, there is a rigid concept C over $\mathcal{T}$ such that $a \in C^{\mathcal{I}_i}$ and $\mathcal{T} \models C \sqsubseteq \exists r.A$, where $\mathcal{I}_i := \mathcal{I}_{[\mathcal{K}^i_{\mathsf{R}}]^+}$ is the canonical interpretation for $[\mathcal{K}^i_{\mathsf{R}}]^+$. By Proposition 5.4, we obtain that $a \in C^{\mathcal{J}_i}$, and thus $a \in C^{\mathcal{J}_1}$ since $\mathcal{J}_i$ and $\mathcal{J}_1$ agree on all rigid names. $\qquad\square$

In the remainder of this section, we prove the converse of this lemma. We thus consider an r-complete tuple $(\mathcal{A}_{\mathsf{R}}, \mathcal{A}'_{\mathsf{R}})$, and denote by $\mathcal{I}_i$ the canonical interpretation $\mathcal{I}_i := \mathcal{I}_{[\mathcal{K}^i_{\mathsf{R}}]^+}$. We first show that $\mathcal{I}_i$ is also a model of $\mathcal{K}^i_{\mathsf{R}}$, for each i, $1 \le i \le k$. Since it satisfies $[\mathcal{K}^i_{\mathsf{R}}]^+$ by Proposition 5.2, we need to consider only the negated assertions in $\mathcal{A}_{\mathsf{R}}$ and $\mathcal{A}_i$. Since $\mathcal{K}^i_{\mathsf{R}}$ is consistent by assumption, we know that $[\mathcal{K}^i_{\mathsf{R}}]^+ \not\models A(a)$ holds for every negated concept assertion $\neg A(a) \in \mathcal{A}_{\mathsf{R}} \cup \mathcal{A}_i$, and thus $\mathcal{I}_i \models \neg A(a)$ by Definition 5.1. Similarly, for any negated role assertion $\neg r(a,b) \in \mathcal{A}_{\mathsf{R}} \cup \mathcal{A}_i$, we cannot have $r(a,b) \in \mathcal{A}_{\mathsf{R}} \cup \mathcal{A}_i$, and thus $\mathcal{I}_i \models \neg r(a,b)$ by Definition 5.1.

To distinguish the elements of $\Delta^{\mathcal{I}_i}_{\mathsf{u}}$, we write $c_{A,i}$ for the element $c_A \in \Delta^{\mathcal{I}_i}_{\mathsf{u}}$ in the domain of $\mathcal{I}_i$. Thus, the domain of each $\mathcal{I}_i$ is composed of the pairwise disjoint components $\mathsf{N}_{\mathsf{I}}(\phi)$ and $\Delta^{\mathcal{I}_i}_{\mathsf{u}}$. We state this fact for future reference.

Fact 5.11. *The set $\mathsf{N}_{\mathsf{I}}(\phi)$ and all sets $\Delta^{\mathcal{I}_i}_{\mathsf{u}}$, $1 \le i \le k$, are pairwise disjoint.*

We now construct the interpretations $\mathcal{J}_1, \ldots, \mathcal{J}_k$ as required for the r-satisfiability of $\mathcal{S}$ by joining the domains of the interpretations $\mathcal{I}_i$ and ensuring that they interpret all rigid names in the same way. We use the common domain

$$\Delta := \mathsf{N}_{\mathsf{I}}(\phi) \cup \bigcup_{i=1}^{k} \Delta^{\mathcal{I}_i}_{\mathsf{u}}$$

and, for all i, $1 \le i \le k$, define the interpretations $\mathcal{J}_i$ as follows:

- For all $a \in \mathsf{N}_{\mathsf{I}}(\phi)$, we set $a^{\mathcal{J}_i} := a$.

- For all rigid concept names A, we define $A^{\mathcal{J}_i} := \bigcup_{j=1}^{k} A^{\mathcal{I}_j}$.

- For all flexible concept names A, we define

$$A^{\mathcal{J}_i} := A^{\mathcal{I}_i} \cup \{e \mid 1 \le j \le k,\ e \in \Delta^{\mathcal{I}_j}_{\mathsf{u}},\ e \in A^{\mathcal{I}'_j}\}.$$

- For all rigid role names r, we define

$$r^{\mathcal{J}_i} := \bigcup_{j=1}^{k} r^{\mathcal{I}_j} \cup \bigcup_{\ell=1}^{k} \{(e, c_{A,\ell}) \mid 1 \le j \le k,\ e \in \Delta^{\mathcal{I}_j}_{\mathsf{u}},\ e \in (\exists r.A)^{\mathcal{I}'_j}\}.$$

- For all flexible role names r, we define

$$r^{\mathcal{J}_i} := r^{\mathcal{I}_i} \cup \{(e, c_{A,i}) \mid 1 \le j \le k,\ e \in \Delta^{\mathcal{I}_j}_{\mathsf{u}},\ e \in (\exists r.A)^{\mathcal{I}'_j}\}.$$

21

We have thus constructed interpretations $\mathcal{J}_1, \ldots, \mathcal{J}_k$ that have the same domain, respect rigid names, and satisfy the UNA, for all relevant individual names. It remains to show that each $\mathcal{J}_i$ is still a model of $\langle \mathcal{T}, \mathcal{A}_i \rangle$. To facilitate this, we first provide an auxiliary lemma.

Lemma 5.12. *For all $i, j \in \{1, \ldots, k\}$ and all concepts $C \in \mathsf{Sub}(\mathcal{T})$, the following hold:*

a) *For all $a \in \mathsf{N}_\mathsf{I}(\phi)$, we have $a \in C^{\mathcal{J}_i}$ iff $a \in C^{\mathcal{I}_i}$.*

b) *For all $e \in \Delta_\mathsf{u}^{\mathcal{I}_j}$, we have $e \in C^{\mathcal{J}_i}$ iff*

- *$i = j$ and $e \in C^{\mathcal{I}_i}$, or*
- *$e \in C^{\mathcal{I}_j'}$.*

Proof. Observe first that $e \in C^{\mathcal{I}_i'}$ implies that $e \in C^{\mathcal{I}_i}$ by Lemma 5.6 and the fact that $\mathcal{I}_i$ is a model of $\mathcal{T}$. This means that, if $i = j$, then the two items in b) are actually equivalent to the first item. On the other hand, if $i \neq j$, then only the second item has to be considered.

We now prove a) and b) simultaneously by induction on the structure of C. The claims obviously hold for $C = \top$. For a flexible concept name C, they follow directly from the definition of $\mathcal{J}_i$ and Fact 5.11. Consider now any $C \in \mathsf{N}_{\mathsf{RC}}(\mathcal{T})$.

- We begin with a). We have $a \in C^{\mathcal{J}_i}$ iff there is some j, $1 \leq j \leq k$, such that $a \in C^{\mathcal{I}_j}$, by the definition of $\mathcal{J}_i$. Since both $\mathcal{I}_i$ and $\mathcal{I}_j$ are models of the ABox type $\mathcal{A}_\mathsf{R}$, this is equivalent to $a \in C^{\mathcal{I}_i}$.

- We consider b). For $i = j$, Fact 5.11 and the definition of $\mathcal{J}_i$ yield the claim. For $i \neq j$, we additionally observe that $C^{\mathcal{I}_j'} \cap \Delta^{\mathcal{I}_j} = C^{\mathcal{I}_j}$ by Definition 5.1.

We now come to the induction steps. Since it can be easily treated based on the semantics, we skip the case for $C = A_1 \sqcap A_2$. Let thus $C = \exists r.A$. The direction ($\Leftarrow$) of both claims easily follows from the observations that $r^{\mathcal{I}_i} \subseteq r^{\mathcal{J}_i}$, $A^{\mathcal{I}_i} \subseteq A^{\mathcal{J}_i}$, and that $e \in (\exists r.A)^{\mathcal{I}_j'}$ implies $(e, c_{A,i}) \in r^{\mathcal{J}_i}$ and $c_{A,i} \in A^{\mathcal{J}_i}$ by the definition of $\mathcal{J}_i$ and Definition 5.1. We now consider the direction ($\Rightarrow$).

- We again begin with the proof of a). If r is rigid, then the definition of $\mathcal{J}_i$ implies that there is an element $e \in \Delta^{\mathcal{I}_j}$, $1 \leq j \leq k$, such that $(a, e) \in r^{\mathcal{I}_j}$ and $e \in A^{\mathcal{J}_i}$.

    - If $e \in \mathsf{N}_\mathsf{I}(\phi)$, then we have $r(a, e) \in \mathcal{A}_j \cup \mathcal{A}_\mathsf{R}$ by Definition 5.1. Since $\mathcal{K}_\mathsf{R}^j$ is consistent and $\mathcal{A}_\mathsf{R}$ is an ABox type, this yields that $r(a, e) \in \mathcal{A}_\mathsf{R}$, and leads to $(a, e) \in r^{\mathcal{I}_i}$, by Proposition 5.2. Now $a \in (\exists r.A)^{\mathcal{I}_i}$ can be obtained by the induction hypothesis for e.

– If $i = j$ and $e \in \Delta_{\mathsf{u}}^{\mathcal{I}_i}$, the induction hypothesis yields that $e \in A^{\mathcal{I}_i}$, and thus we again obtain $a \in (\exists r.A)^{\mathcal{I}_i}$.

– It remains to consider the case that $i \neq j$ and $e \in \Delta_{\mathsf{u}}^{\mathcal{I}_j}$. Then, we have $(a, e) \in r^{\mathcal{I}_j'}$ by Definition 5.1. By the induction hypothesis, we obtain that $e \in A^{\mathcal{I}_j'}$. But this implies that $a \in (\exists r.A)^{\mathcal{I}_j'}$. Thus, we have $\exists r.A(a) \in \mathcal{A}_{\mathsf{R}}'$, and hence $a \in (\exists r.A)^{\mathcal{I}_i}$ since $\mathcal{I}_i$ is a model of $\mathcal{A}_{\mathsf{R}}'$.

If r is flexible, then there is an element $e \in \Delta^{\mathcal{I}_i}$ such that $(a, e) \in r^{\mathcal{I}_i}$ and $e \in A^{\mathcal{J}_i}$, by the definition of $\mathcal{J}_i$ and Fact 5.11. By applying the induction hypothesis to e (and $i = j$), we obtain that $e \in A^{\mathcal{I}_i}$ and thus $a \in (\exists r.A)^{\mathcal{I}_i}$.

• We finally consider b) and begin with $r \in \mathsf{N}_{\mathsf{RR}}$. By the definition of $\mathcal{J}_i$, either (i) there is an element d such that $(e, d) \in r^{\mathcal{I}_j}$ and $d \in A^{\mathcal{J}_i} \cap \Delta_{\mathsf{u}}^{\mathcal{I}_j}$ (see Definition 5.1 and Fact 5.11), or (ii) $e \in (\exists r.A)^{\mathcal{I}_j'}$, $(e, c_{A,\ell}) \in r^{\mathcal{J}_i}$, and $c_{A,\ell} \in A^{\mathcal{J}_i}$, for some ℓ, $1 \leq \ell \leq k$ (again by Fact 5.11). In case (ii) we are immediately done. We consider (i). By the induction hypothesis, we have either (i') $i = j$ and $d \in A^{\mathcal{I}_i}$, or (ii') $d \in A^{\mathcal{I}_j'}$. In the first case, we can infer that $e \in (\exists r.A)^{\mathcal{I}_i}$, while in the second case, we have $(e, d) \in r^{\mathcal{I}_j'}$ since r is rigid, and thus $e \in (\exists r.A)^{\mathcal{I}_j'}$.

If r is flexible, then either (i) there is an element d such that $(e, d) \in r^{\mathcal{I}_i}$ and $d \in A^{\mathcal{J}_i} \cap \Delta_{\mathsf{u}}^{\mathcal{I}_i}$ (see Definition 5.1 and Fact 5.11), or (ii) $e \in (\exists r.A)^{\mathcal{I}_j'}$, $(e, c_{A,i}) \in r^{\mathcal{J}_i}$, and $c_{A,i} \in A^{\mathcal{J}_i}$. Again, case (ii) is trivial. In case (i), we have $i = j$, and thus by the induction hypothesis we get $d \in A^{\mathcal{I}_i}$. Thus, we conclude that $e \in (\exists r.A)^{\mathcal{I}_i}$. $\qquad\square$

We finally show that the interpretations $\mathcal{J}_i$ are in fact as intended, i.e., they satisfy the knowledge bases $\langle \mathcal{T}, \mathcal{A}_i \rangle$.

Lemma 5.13. *For all i, $1 \leq i \leq k$, $\mathcal{J}_i$ is a model of $\mathcal{A}_i$.*

Proof. We know that $\mathcal{I}_i$ satisfies $\mathcal{A}_i$, by Definition 5.9. Since we have $A^{\mathcal{I}_i} \subseteq A^{\mathcal{J}_i}$ and $r^{\mathcal{I}_i} \subseteq r^{\mathcal{J}_i}$ for all relevant concept names A and role names r, this means that the positive assertions in $\mathcal{A}_i$ are also satisfied by $\mathcal{J}_i$.

Consider now any negated concept assertion $\neg A(a) \in \mathcal{A}_i$. Since $\mathcal{I}_i$ is a model of $\mathcal{A}_i$, we have $a \notin A^{\mathcal{I}_i}$, and thus $a \notin A^{\mathcal{J}_i}$ by Lemma 5.12.

If $\neg r(a, b) \in \mathcal{A}_i$ for a flexible role name r, then we similarly get $(a, b) \notin r^{\mathcal{I}_i}$, and thus $(a, b) \notin r^{\mathcal{J}_i}$, by the definition of $\mathcal{J}_i$.

Finally, if $\neg r(a, b) \in \mathcal{A}_i$ and r is rigid, then we must have $\neg r(a, b) \in \mathcal{A}_{\mathsf{R}}$, by the consistency of $\mathcal{K}_{\mathsf{R}}^i$ and the fact that $\mathcal{A}_{\mathsf{R}}$ is an ABox type. Thus, we have $(a, b) \notin r^{\mathcal{I}_j}$ for all j, $1 \leq j \leq k$, and hence $(a, b) \notin r^{\mathcal{J}_i}$, again by the definition of $\mathcal{J}_i$. $\qquad\square$

It remains to show that all GCIs in $\mathcal{T}$ are satisfied by $\mathcal{J}_i$.

Lemma 5.14. *For all i, $1 \leq i \leq k$, $\mathcal{J}_i$ is a model $\mathcal{T}$.*

Proof. We consider an arbitrary GCI $C \sqsubseteq D \in \mathcal{T}$. Let first $e \in \Delta^{\mathcal{I}_i} \cap C^{\mathcal{J}_i}$. By Lemma 5.12, we have $e \in C^{\mathcal{I}_i}$, and hence $e \in D^{\mathcal{I}_i}$ since $\mathcal{I}_i$ satisfies $\mathcal{T}$. And, by applying Lemma 5.12 again, we obtain $e \in D^{\mathcal{J}_i}$.

For $i \neq j$ and $e \in \Delta_{\mathsf{u}}^{\mathcal{I}_j} \cap C^{\mathcal{J}_i}$, we get $e \in C^{\mathcal{I}_j'}$ by Lemma 5.12. We show below that then $e \in D^{\mathcal{I}_j'}$ holds, and hence $e \in D^{\mathcal{J}_i}$, by another application of Lemma 5.12.

Since $e \in C^{\mathcal{I}_j'}$, there is an $\ell \geq 0$ such that $e \in C^{\mathcal{I}_{j,\ell}'}$, where $\mathcal{I}_{j,\ell}' := \mathcal{I}_{[\mathcal{K}_{\mathsf{R}}^j]^+,\ell}'$ is as in Definition 5.1. If $D = \top$, then obviously $e \in D^{\mathcal{I}_j'}$, as desired. If D is a flexible concept name, then we get $e \in D^{\mathcal{I}_{j,\ell+1}'} \subseteq D^{\mathcal{I}_j'}$ by the definition of $\mathcal{I}_{j,\ell+1}'$. If D is a rigid concept name, then we have $e \in C^{\mathcal{I}_j} \subseteq D^{\mathcal{I}_j} \subseteq D^{\mathcal{I}_{j,0}'} \subseteq D^{\mathcal{I}_j'}$ by Lemma 5.6, Proposition 5.2, and the definition of $\mathcal{I}_{j,0}'$. Finally, if $D = \exists r.A$, then we get $(e, c_A') \in r^{\mathcal{I}_{j,\ell+1}'} \subseteq r^{\mathcal{I}_j'}$ by the definition of $\mathcal{I}_{j,\ell+1}'$. Since also $c_A' \in A^{\mathcal{I}_{j,0}'} \subseteq A^{\mathcal{I}_j'}$, we obtain $e \in (\exists r.A)^{\mathcal{I}_j'}$. $\qquad\square$

We hence have proven the following lemma.

Lemma 5.15. *If there is an r-complete tuple for ϕ w.r.t. $\mathcal{S}$, then $\mathcal{S}$ is r-satisfiable.*

This finishes the proof that the existence of an r-complete tuple characterizes the r-satisfiability of $\mathcal{S}$. Together with the PSPACE-hardness result for propositional LTL [SC85], we thus obtain the following.

Theorem 5.16. *Satisfiability in $\mathcal{EL}$-LTL with global GCIs is PSPACE-complete, even if $\mathsf{N_{RR}} \neq \emptyset$.*

Proof. We can argue as in the proof of Theorem 4.3. That is, to check ϕ for satisfiability, we can run a modified version of the Turing machine $\mathcal{M}_{\phi^{\mathsf{p}}}$. However, the difference now is that, before running $\mathcal{M}_{\phi^{\mathsf{p}}}$, we guess $\mathcal{A}_{\mathsf{R}}$ and $\mathcal{A}_{\mathsf{R}}'$, which can be done in PSPACE, and then include the two tests described in Definition 5.9, for each guessed complete set of subformulae (and world X_i and ABox $\mathcal{A}_i$ induced by it). Both of these tests can be done in polynomial time. The set $\mathcal{S}$ from Proposition 4.1 corresponds to the set of all worlds X encountered during a run of $\mathcal{M}_{\phi^{\mathsf{p}}}$. As described before, this set does not have to be stored explicitly.

Since all this can be done with a nondeterministic Turing machine using only polynomial space (in the size of ϕ), according to [Sav70], satisfiability in $\mathcal{EL}$-LTL with global GCIs can be decided in PSPACE. $\qquad\square$

6 Conclusions

We have shown that satisfiability in $\mathcal{EL}$-LTL is PSPACE-complete without rigid names and NEXPTIME-complete if any rigid names are used. This is lower than

for $\mathcal{ALC}$-LTL in some cases, but not as good as *DL-Lite*-LTL, where satisfiability is decidable in PSPACE even in the presence of rigid role names.

We plan to investigate the complexity of answering so-called temporal conjunctive queries [BBL15] over temporal $\mathcal{EL}$- and *DL-Lite*-knowledge bases, which is a closely related problem. For all DLs between $\mathcal{ALC}$ and $\mathcal{SHQ}$, the (combined) complexity of this problem is the same as that of the (complement of the) satisfiability problem in $\mathcal{ALC}$-LTL [BGL12, BBL15].

References

[AF00]   Alessandro Artale and Enrico Franconi. A survey of temporal extensions of description logics. *Annals of Mathematics and Artificial Intelligence*, 30(1/4):171–210, 2000.

[AKL⁺07]   Alessandro Artale, Roman Kontchakov, Carsten Lutz, Frank Wolter, and Michael Zakharyaschev. Temporalising tractable description logics. In Valentin Goranko and X. Sean Wang, editors, *Proc. of the 14th Int. Symp. on Temporal Representation and Reasoning (TIME'07)*, pages 11–22. IEEE Press, 2007.

[AKRZ09]   Alessandro Artale, Roman Kontchakov, Vladislav Ryzhikov, and Michael Zakharyaschev. *DL-Lite* with temporalised concepts, rigid axioms and roles. In Silvio Ghilardi and Roberto Sebastiani, editors, *Proc. of the 7th Int. Symp. on Frontiers of Combining Ssystem (FroCoS'09)*, volume 5749 of *Lecture Notes in Computer Science*, pages 133–148, 2009.

[AKWZ13]   Alessandro Artale, Roman Kontchakov, Frank Wolter, and Michael Zakharyaschev. Temporal description logic for ontology-based data access. In Francesca Rossi, editor, *Proc. of the 23rd Int. Joint Conf. on Artificial Intelligence (IJCAI'13)*, pages 711–717. AAAI Press, 2013.

[Baa03]   Franz Baader. Terminological cycles in a description logic with existential restrictions. In Georg Gottlob and Toby Walsh, editors, *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, pages 325–330. Morgan Kaufmann, 2003.

[BBL05]   Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the $\mathcal{EL}$ envelope. In Leslie Pack Kaelbling and Alessandro Saffiotti, editors, *Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI'05)*, pages 364–369. Professional Book Center, 2005.

[BBL15]   Franz Baader, Stefan Borgwardt, and Marcel Lippmann. Temporal query entailment in the description logic $\mathcal{SHQ}$. *Journal of Web Semantics*, 2015. In press.

[BCM+07] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications.* Cambridge University Press, 2 edition, 2007.

[BGG97]   Egon Börger, Erich Grädel, and Yuri Gurevich. *The Classical Decision Problem.* Perspectives in Mathematical Logic. Springer-Verlag, 1997.

[BGL12]   Franz Baader, Silvio Ghilardi, and Carsten Lutz. LTL over description logic axioms. *ACM Transactions on Computational Logic*, 13(3):21:1–21:32, 2012.

[BLT13]   Stefan Borgwardt, Marcel Lippmann, and Veronika Thost. Temporal query answering in the description logic *DL-Lite*. In Pascal Fontaine, Christophe Ringeissen, and Renate A. Schmidt, editors, *Proc. of the 9th Int. Symp. on Frontiers of Combining Systems (FroCoS'13)*, volume 8152 of *Lecture Notes in Artificial Intelligence*, pages 165–180. Springer-Verlag, 2013.

[GK12]   Víctor Gutiérrez-Basulto and Szymon Klarman. Towards a unifying approach to representing and querying temporal data in description logics. In Markus Krötzsch and Umberto Straccia, editors, *Proc. of the 6th Int. Conf. on Web Reasoning and Rule Systems (RR'12)*, volume 7497 of *Lecture Notes in Computer Science*, pages 90–105. Springer-Verlag, 2012.

[Lew78]   Harry R. Lewis. Complexity of solvable cases of the decision problem for the predicate calculus. In *Proc. of the 19th Annual Symp. on Foundations of Computer Science (SFCS'78)*, pages 35–47. IEEE Press, 1978.

[Lip14]   Marcel Lippmann. *Temporalised Description Logics for Monitoring Partially Observable Events.* PhD thesis, TU Dresden, Germany, 2014.

[LWZ08]   Carsten Lutz, Frank Wolter, and Michael Zakharyaschev. Temporal description logics: A survey. In Stéphane Demri and Christian S. Jensen, editors, *Proc. of the 15th Int. Symp. on Temporal Representation and Reasoning (TIME'08)*, pages 3–14. IEEE Press, 2008.

[Pnu77]   Amir Pnueli. The temporal logic of programs. In *Proc. of the 18th Annual Symp. on Foundations of Computer Science (SFCS'77)*, pages 46–57. IEEE Press, 1977.

[Sav70]   Walter J. Savitch. Relationships between nondeterministic and deterministic tape complexities. *Journal of Computer and System Sciences*, 4(2):177–192, 1970.

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

[WZ00]   Frank Wolter and Michael Zakharyaschev. Temporalizing description logics. In *FroCoS 2*, volume 7 of *Studies in Logic and Computation*, pages 379–402, 2000.