Learning Description Logic Concepts: When can Positive and Negative Examples be Separated?

Maurice Funk[1] , **Jean Christoph Jung**[1] , **Carsten Lutz**[1] , **Hadrien Pulcini**[2] , **Frank Wolter**[2]

[1]University of Bremen

[2]University of Liverpool

Abstract

Learning description logic (DL) concepts from positive and negative examples given in the form of labeled data items in a KB has received significant attention in the literature. We study the fundamental question of when a separating DL concept exists and provide useful model-theoretic characterizations as well as complexity results for the associated decision problem. For expressive DLs such as $\mathcal{ALC}$ and $\mathcal{ALCQI}$, our characterizations show a surprising link to the evaluation of ontology-mediated conjunctive queries. We exploit this to determine the combined complexity (between EXPTIME and NEXPTIME) and data complexity (Σ_2^p) of separability. For the Horn DL $\mathcal{EL}$, separability is EXPTIME-complete both in combined and in data complexity while for its modest extension $\mathcal{ELI}$ it is even undecidable. Separability is also undecidable when the KB is formulated in $\mathcal{ALC}$ and the separating concept is required to be in $\mathcal{EL}$ or $\mathcal{ELI}$.

1 Introduction

An important challenge for adopting ontologies in practical applications is the knowledge acquisition bottleneck, that is, the significant time and effort it takes to build the required ontologies. As a promising approach to help overcoming this difficulty, the varied field of *ontology learning* has received a lot of attention in the last two decades, see [Lehmann and Völker, 2014] for a recent overview. A prominent line of research within ontology learning is *concept learning*, also called *concept induction*, where the aim is to learn a structured class description (a *concept*) formulated in a relevant ontology language from positive and negative examples, given an already available ontology that contains background knowledge [Lehmann *et al.*, 2014]. Applications of concept learning include the support of ontology development and the construction of concept based classifiers [Bühmann *et al.*, 2016; Sarker *et al.*, 2017].

Since description logics (DLs) form the basis for the OWL family of ontology languages, DL concept learning has received particularly much attention [Lehmann and Hitzler, 2010; Lisi, 2012; Tran *et al.*, 2014; Fanizzi *et al.*, 2018]. The precise formulation is as follows. Given a knowledge base (KB) $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ where $\mathcal{T}$ is a TBox that is formulated in a DL $\mathcal{L}_T$ and serves as an ontology providing background knowledge and $\mathcal{A}$ is an ABox in which individuals from designated sets P and N serve as positive and negative examples, respectively, find a concept C formulated in a DL $\mathcal{L}_S$ that *separates* the positive from the negative examples, that is, $\mathcal{K} \models C(a)$ for all $a \in P$ and $\mathcal{K} \not\models C(a)$ for all $a \in N$. In addition to separation, one also wants to achieve that the learned concept C generalizes the positive examples in a meaningful way, classifying new examples accordingly.

As a prominent system for DL concept learning, we mention DL LEARNER. It encompasses several learning algorithms that support a range of DLs, including expressive ones such as $\mathcal{ALC}$ and $\mathcal{ALCQ}$, Horn DLs such as $\mathcal{EL}$, and even full OWL 2 [Bühmann *et al.*, 2018; Bühmann *et al.*, 2016]. Like competing systems such as DL-FOIL, YINYANG, and PFOIL-DL [Fanizzi *et al.*, 2018; Iannone *et al.*, 2007; Straccia and Mucci, 2015], DL LEARNER uses a carefully crafted *refinement operator* along with various heuristics to learn concepts that provide an as good as possible generalization of the given examples, avoiding overfitting. The study of such operators originated in [Badea and Nienhuys-Cheng, 2000], see also [Lehmann and Haase, 2009; Lehmann and Hitzler, 2010]. If possible, refinement operators are designed so that the resulting algorithm terminates on any input and is complete in the sense that whenever there is a concept that separates the positive and negative examples in the input, then such a concept is indeed learned.

The aim of this paper is to investigate the fundamental question of when a separating concept exists for a *learning instance* $(\mathcal{K}, P, N)$, considering the most popular choices of DLs for the TBox language $\mathcal{L}_T$ and the separation language $\mathcal{L}_S$. Our main contributions are model-theoretic characterizations that give important insight into when this is the case and a precise analysis of the computational complexity of separability viewed as a decision problem, which we refer to as $(\mathcal{L}_T, \mathcal{L}_S)$ *concept separability* and as $\mathcal{L}$ *concept separability* when $\mathcal{L}_T = \mathcal{L}_S = \mathcal{L}$. We also consider *concept definability*, the special case of concept separability in which $P \cup N$ comprises all individuals from $\mathcal{A}$. In fact, all our complexity results hold for both separability and definability.

We believe that these problems are relevant both from a practical and from a theoretical perspective. In fact, complexity lower bounds for concept separability point to an inherent complexity that no practical system that aims for completeness can avoid. Undecidability results even mean that there can be no practical learning system that is both terminating and complete. From the viewpoint of machine learning theory, a separating concept corresponds to a consistent hypothesis and understanding the existence of such a hypothesis is considered the most fundamental problem for exploring the version space which in our setup is the space of all separating concepts [Hirsh *et al.*, 2004]. The associated decision problem is often called the *consistency problem*, and it is known to be closely related to PAC learnability [Pitt and Valiant, 1988; Kietz, 1993].

We cover the expressive DLs $\mathcal{ALC}$, $\mathcal{ALCI}$, $\mathcal{ALCQ}$, and $\mathcal{ALCQI}$ as well as the Horn DLs $\mathcal{EL}$ and $\mathcal{ELI}$. For the former, overfitting is a risk because the disjunction operator available in such DLs enables the construction of separating concepts that do not provide the desired generalization of the positive examples. Nevertheless, most practical systems such as DL LEARNER work with expressive DLs and avoid overfitting by using appropriate refinement operators. Horn DLs do not admit disjunction and therefore are not prone to overfitting. On the other hand, they provide less separating power and, as we show, tend to incur higher computational (worst-case) cost for learning.

For expressive DLs, we start with initial characterizations in terms of (some form of) bisimulations and then improve them to more refined characterizations based on homomorphisms. Interestingly and unexpectedly to us, these establish a tight link between concept separability and the evaluation of ontology-mediated queries (OMQs) based on unions of rooted conjunctive queries [Calvanese *et al.*, 2013; Bienvenu *et al.*, 2014]. Here, 'rooted' means that queries have at least one answer variable and are connected. We use the OMQ connection to obtain complexity upper and lower bounds. In fact, $\mathcal{L}$ concept separability is NEXPTIME-complete for $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALCQ}\}$ while $\mathcal{ALCQI}$ concept separability is only EXPTIME-complete. This refers to combined complexity where all components of the learning instance are part of the input. We also study data complexity where the ABox is the only input while the TBox is fixed and thus of constant size. In all expressive DLs mentioned above, concept separability is Σ_2^p-complete in data complexity.

The connection to OMQ evaluation does not extend to Horn DLs, for which we use characterizations based on products of universal models and simulations. Based on these, we show that $(\mathcal{L}_T, \mathcal{EL})$ concept separability is EXPTIME-complete for $\mathcal{L}_T \in \{\mathcal{EL}, \mathcal{ELI}\}$, both in combined complexity and in data complexity. Rather surpsiringly, we also prove that $\mathcal{ELI}$ concept separability is undecidable, thus ruling out terminating and complete learning systems. The proof is by a subtle reduction of a tiling problem. We finally consider $(\mathcal{L}_T, \mathcal{L}_S)$ concept separability where $\mathcal{L}_T$ is any of the expressive DLs mentioned above and $\mathcal{L}_S$ is $\mathcal{EL}$ or $\mathcal{ELI}$. These problems also turn out to be undecidable.

A stronger version of concept separability that is also considered in the literature requires that $\mathcal{K} \models \neg C(a)$ for all $a \in N$, rather than only $\mathcal{K} \not\models C(a)$. Clearly, this is a meaningful notion only for DLs with negation. We present some first results also for this version of concept separability, giving model-theoretic characterizations for expressive DLs based on bisimulations and proving that for $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}\}$, $\mathcal{L}$ concept separability is EXPTIME-complete in combined complexity and CONP-complete in data complexity.

The long version with appendix is available at http://www.informatik.uni-bremen.de/tdki/research/papers.html.

2 Preliminaries

We introduce the basics of DLs as required for this paper, for full details see [**?**]. Let $\mathsf{N_C}$ be a set of *concept names* and $\mathsf{N_R}$ a set of *role names*, both countably infinite. A *role* is either a role name or an *inverse role* r^-, r a role name. For uniformity, we identify $(r^-)^-$ with r. An $\mathcal{ALCQI}$ *concept* is formed according to the syntax rule

$$C, D ::= \top \mid A \mid \neg C \mid C \sqcap D \mid ({\geqslant}\, n\ r\ C)$$

where A ranges over concept names, r over roles, and n over $\mathbb{N}$. We use $\exists r.C$ as an abbreviation for $({\geqslant}\, 1\ r\ C)$, $C \sqcup D$ for $\neg(\neg C \sqcap \neg D)$, $\forall r.C$ for $\neg\exists r.\neg C$, and $({\leqslant}\, n\ r\ C)$ for $\neg({\geqslant}\, n+1\ r\ C)$.

There are several fragments of $\mathcal{ALCQI}$ that are relevant for this paper. An $\mathcal{EL}$ *concept* admits as constructors only the top concept '$\top$', conjunction '$\sqcap$', and existential restriction '$\exists r.C$' with r a role name (but not an inverse role). $\mathcal{ALC}$ *concepts* additionally admit negation '$\neg$'. Further constructors are indicated by concatenation of a corresponding letter where $\mathcal{Q}$ stands for at least restrictions '$({\geqslant}\, n\ r\ C)$' and $\mathcal{I}$ for inverse roles. This explains the name $\mathcal{ALCQI}$ and allows us to refer to fragments such as $\mathcal{ELI}$ and $\mathcal{ALCI}$. We refer to $\mathcal{ALC}$ and extensions thereof as *expressive DLs* and to $\mathcal{EL}$ and $\mathcal{ELI}$ as *Horn DLs*.

For any of the DLs $\mathcal{L}$ introduced above, an $\mathcal{L}$ *TBox* is a finite set of *concept inclusions* (CIs) $C \sqsubseteq D$, where C and D are $\mathcal{L}$ concepts. Let $\mathsf{N_I}$ be a countably infinite set of *individual names*. An *ABox* $\mathcal{A}$ is a finite set of *concept assertions* $A(a)$ and *role assertions* $r(a, b)$ where $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, and $a, b \in \mathsf{N_I}$. We use $\mathsf{ind}(\mathcal{A})$ to denote the set of all individual names that occur in $\mathcal{A}$. An ABox is associated with a directed graph $G_\mathcal{A} = (\mathsf{ind}(\mathcal{A}), \{(a, b) \mid r(a, b) \in \mathcal{A}\})$. We use graph theoretic terminology when speaking about ABoxes, implicitly referring to the graph $G_\mathcal{A}$, speaking for example about a being reachable from b for $a, b \in \mathsf{ind}(\mathcal{A})$. In technical constructions, we will sometimes use *extended ABoxes* in which concept assertions take the more general form $C(a)$, C a concept of the DL under consideration. An $\mathcal{L}$ *knowledge base* (KB) $(\mathcal{T}, \mathcal{A})$ consists of an $\mathcal{L}$ TBox $\mathcal{T}$ and an ABox $\mathcal{A}$, and *extended* KBs admit extended ABoxes. We use $\mathsf{sub}(\mathcal{K})$ to denote the set of subconcepts of concepts that occur in $\mathcal{K}$.

As usual, the semantics of DLs is defined in terms of *interpretations* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where $\Delta^\mathcal{I}$ is a non-empty set and $\cdot^\mathcal{I}$ maps each concept name $A \in \mathsf{N_C}$ to a subset $A^\mathcal{I}$ of $\Delta^\mathcal{I}$ and each role name $r \in \mathsf{N_R}$ to a binary relation $r^\mathcal{I}$ on $\Delta^\mathcal{I}$. We refer to [**?**] for details on how to extend $\cdot^\mathcal{I}$ to compound concepts. An interpretation $\mathcal{I}$ *satisfies* a CI $C \sqsubseteq D$ if $C^\mathcal{I} \subseteq D^\mathcal{I}$ and is a *model* of a TBox $\mathcal{T}$ if it satisfies all inclusions in $\mathcal{T}$. An interpretation is a *model* of an ABox $\mathcal{A}$ if it *satisfies* all assertions in $\mathcal{A}$, that is, $a \in A^\mathcal{I}$ if $A(a) \in \mathcal{A}$ and $(a, b) \in r^\mathcal{I}$ if $r(a, b) \in \mathcal{A}$. We thus make the standard names assumption. An interpretation is a *model* of a KB $\mathcal{K}$ if it is a common

[Atom]	for all $(d,e) \in S$: $d \in A^{\mathcal{I}}$ iff $e \in A^{\mathcal{J}}$
[AtomR]	if $(d,e) \in S$ and $d \in A^{\mathcal{I}}$, then $e \in A^{\mathcal{J}}$
[Forth]	if $(d,e) \in S$ and $d' \in \text{succ}_r^{\mathcal{I}}(d)$, then there is a $e' \in \text{succ}_r^{\mathcal{J}}(e)$ with $(d',e') \in S$.
[Back]	dual of [Forth]
[QForth]	if $(d,e) \in S$ and $D \subseteq \text{succ}_r^{\mathcal{I}}(d)$ finite, then there is a $E \subseteq \text{succ}_r^{\mathcal{J}}(e)$ such that S contains a bijection between D and E.
[QBack]	dual of [QForth]

Figure 1: Conditions on $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$.

model of its TBox and ABox and $\mathcal{K}$ is *satisfiable* if it has a model. The assertion $C(a)$ is a *consequence of the KB* $\mathcal{K}$, in symbols $\mathcal{K} \models C(a)$, if $a \in C^{\mathcal{I}}$ for all models $\mathcal{I}$ if $\mathcal{K}$.

We next recall model-theoretic characterizations of when elements in interpretations are indistinguishable by concepts formulated in one of the DLs $\mathcal{L}$ introduced above. A *pointed interpretation* is a pair $\mathcal{I}, d$ with $\mathcal{I}$ an interpretation and $d \in \Delta^{\mathcal{I}}$. For pointed interpretations $\mathcal{I}, d$ and $\mathcal{J}, e$, we write $\mathcal{I}, d \equiv_{\mathcal{L}} \mathcal{J}, e$ and say that $\mathcal{I}, d$ and $\mathcal{J}, e$ are *$\mathcal{L}$-equivalent* if $d \in C^{\mathcal{I}}$ iff $e \in C^{\mathcal{J}}$ for all $\mathcal{L}$ concepts C.

As for the model-theoretic characterizations, we start with $\mathcal{ALC}$. A relation $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$ is a *bisimulation* if conditions [Atom], [Forth] and [Back] from Figure 1 hold, where A ranges over all concept names, r over all role names, $\mathsf{succ}_r^{\mathcal{I}}(d) = \{d' \in \Delta^{\mathcal{I}} \mid (d,d') \in r^{\mathcal{I}}\}$, and 'dual' refers to swapping $\mathcal{I}, d, d'$ and $\mathcal{J}, e, e'$. We write $\mathcal{I}, d \sim_{\mathcal{ALC}} \mathcal{J}, e$ and call $\mathcal{I}, d$ and $\mathcal{J}, e$ *bisimilar* if there exists a bisimulation S such that $(d,e) \in S$. For $\mathcal{ALCQ}$, we define $\sim_{\mathcal{ALCQ}}$ by replacing bisimulations with *counting bisimulations*, defined as bisimulations, but with [Forth] and [Back] replaced by [QForth] and [QBack]. For $\mathcal{ALCI}$ and $\mathcal{ALCQI}$, we define $\sim_{\mathcal{ALCI}}$ and $\sim_{\mathcal{ALCQI}}$ analogously, but now demanding that in all conditions in Figure 1, r additionally ranges over inverse roles.

For $\mathcal{L} \in \{\mathcal{EL}, \mathcal{ELI}\}$, rather than characterizing $\equiv_{\mathcal{L}}$ we consider the non-symmetric relation $\mathcal{I}, d \leq_{\mathcal{L}} \mathcal{J}, e$ which holds if $d \in C^{\mathcal{I}}$ implies $e \in C^{\mathcal{J}}$ for all $\mathcal{L}$ concepts C. A relation $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$ is an $\mathcal{EL}$ *simulation* from $\mathcal{I}$ to $\mathcal{J}$ if it satisfies [AtomR] and [Forth] from Figure 1 where again A ranges over all concept names and r over all role names. $\mathcal{ELI}$ *simulations* are defined in the same way, but with r ranging also over inverse roles. We write $\mathcal{I}, d \preceq_{\mathcal{L}} \mathcal{J}, e$ if there exists an $\mathcal{L}$ simulation S from $\mathcal{I}$ to $\mathcal{J}$ with $(d,e) \in S$.

The next lemma summarizes the model-theoretic characterizations for all relevant DLs [Lutz *et al.*, 2011; Goranko and Otto, 2007]. An interpretation $\mathcal{I}$ has *finite outdegree* if the directed graph $G_{\mathcal{I}} = (\Delta^{\mathcal{I}}, \bigcup_{r \in \mathsf{N_R}} r^{\mathcal{I}})$ has.

Lemma 1 *Let $\mathcal{I}, d$ and $\mathcal{J}, e$ be pointed interpretations and let $\mathcal{J}$ have finite outdegree. Then*

1. *for $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALCQ}, \mathcal{ALCQI}\}$, $\mathcal{I}, d \equiv_{\mathcal{L}} \mathcal{J}, e$ iff $\mathcal{I}, d \sim_{\mathcal{L}} \mathcal{J}, e$;*
2. *for $\mathcal{L} \in \{\mathcal{EL}, \mathcal{ELI}\}$, $\mathcal{I}, d \leq_{\mathcal{L}} \mathcal{J}, e$ iff $\mathcal{I}, d \preceq_{\mathcal{L}} \mathcal{J}, e$.*

The 'if' directions also hold if $\mathcal{J}$ is of infinite outdegree.

3 Separability and Definability

We introduce concept separability and concept definability, the main notions studied in this paper. We also provide illustrating examples and give initial model-theoretic characterizations in all relevant DLs.

Definition 1 *Let $\mathcal{L}_T, \mathcal{L}_S$ be DLs. An $\mathcal{L}_T$* learning instance *is a triple $(\mathcal{K}, P, N)$ with $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ an $\mathcal{L}_T$ KB and $P, N \subseteq$ $\mathsf{ind}(\mathcal{A})$ non-empty sets of* positive *and* negative examples. *An $\mathcal{L}_S$* solution *to $(\mathcal{K}, P, N)$ is an $\mathcal{L}_S$ concept C such that*

1. *$\mathcal{K} \models C(a)$ for all $a \in P$ and*
2. *$\mathcal{K} \not\models C(a)$ for all $a \in N$.*

Any TBox language $\mathcal{L}_T$ and separation language $\mathcal{L}_S$ give rise to a decision problem of *concept separability*.

PROBLEM :	$(\mathcal{L}_T, \mathcal{L}_S)$ concept separability
INPUT :	$\mathcal{L}_T$ learning instance $(\mathcal{K}, P, N)$
QUESTION :	Does $(\mathcal{K}, P, N)$ have an $\mathcal{L}_S$ solution?

We speak of $\mathcal{L}$ *concept separability* when $\mathcal{L}_T = \mathcal{L}_S = \mathcal{L}$. We also study $(\mathcal{L}_T, \mathcal{L}_S)$ *concept definability*, the special case of $(\mathcal{L}_T, \mathcal{L}_S)$ concept separability in which P covers *all* positive examples in the KB, that is, inputs are learning instances $(\mathcal{K}, P, N)$ such that $N = \mathsf{ind}(\mathcal{A}) \setminus P$. We now give illustrating examples. When we claim that there is no solution, this is a direct consequence of the characterizations provided later. In the following examples and also later in this paper, we assume that $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ unless explicitly stated otherwise.

Example 1 (1) Let $(\mathcal{K}, P, N)$ be defined by $\mathcal{T} = \emptyset$,

$$\mathcal{A} = \{\mathsf{authorOf}(a_i, c_i) \mid i = 1,2,3\} \cup \{\mathsf{IJCAIpub}(c_1), \mathsf{AIJpub}(c_2), \mathsf{GraphicNovel}(c_3)\},$$

$P = \{a_1, a_2\}$, and $N = \{a_3\}$. Then $\exists \mathsf{authorOf}.(\mathsf{IJCAIpub} \sqcup \mathsf{AIJpub})$ is an $\mathcal{ALC}$ solution, but there is neither an $\mathcal{EL}$ solution nor an $\mathcal{ELI}$ solution.

(2) Let $(\mathcal{K}, P, N)$ be as in (1), but with $\mathcal{T}$ replaced by the $\mathcal{EL}$ TBox

$$\mathcal{T} = \{\exists \mathsf{authorOf}.\mathsf{IJCAIpub} \sqsubseteq \mathsf{AIResearcher}, \; \exists \mathsf{authorOf}.\mathsf{AIJpub} \sqsubseteq \mathsf{AIResearcher}\}.$$

Then $\mathsf{AIResearcher}$ is an $\mathcal{EL}$ solution.

(3) Let $(\mathcal{K}, P, N)$ be defined by $\mathcal{T} = \emptyset$,

$$\mathcal{A} = \{\mathsf{authorOf}(a, c_1), \mathsf{authorOf}(a, c_2), \mathsf{authorOf}(b, c_3)\},$$

$P = \{a\}$, and $N = \{b\}$. Due to the standard names assumption, c_1 and c_2 are distinct objects and thus $(\geqslant 2 \; \mathsf{authorOf} \; \top)$ is an $\mathcal{ALCQ}$ solution. However, there is no $\mathcal{ALCI}$ solution, even when $\mathcal{T}$ is replaced with *any* $\mathcal{ALCI}$ TBox.

(4) Let $(\mathcal{K}, P, N)$ be as in (3), but with P and N swapped. There is no $\mathcal{ALCQI}$ solution even when $\mathcal{T}$ is replaced with *any* $\mathcal{ALCQI}$ TBox. Note that $(\leqslant 1 \; \mathsf{authorOf} \; \top)$ is not a solution as the semantics does not rule out additional $\mathsf{authorOf}$ successors in concrete models.

Observe that if $\mathcal{L}_S$ is closed under conjunction, then $(\mathcal{L}_T, \mathcal{L}_S)$ concept separability can be reduced in polynomial time to the special case of $(\mathcal{L}_T, \mathcal{L}_S)$ concept separability that admits only input instances $(\mathcal{K}, P, N)$ in which N is a singleton. Indeed, if C is a solution to a learning instance $(\mathcal{K}, P, N)$, then it is a solution to $(\mathcal{K}, P, \{a\})$ for all $a \in N$. If, conversely, C_a is a solution to $(\mathcal{K}, P, \{a\})$ for all $a \in N$, then $\prod_{a \in N} C_a$ is a solution to $(\mathcal{K}, P, N)$. In what follows, we will thus mostly restrict our attention to this special case.

We study both the *combined complexity* of concept separability and definability and the *data complexity*, where only the ABox of the KB is regarded as the input to the decision problem while the TBox is fixed and thus of constant size. Note that the sizes of P and N are dominated by the size of the ABox. When not making explicit which complexity measure we speak about, we mean combined complexity.

We now give the announced characterizations of $(\mathcal{L}_S, \mathcal{L}_T)$ concept separability, starting with the case that $\mathcal{L}_S$ is an expressive DL. The characterizations use the relations $\sim_{\mathcal{L}}$ from Lemma 1, $\mathcal{L}$ being the separation language. The corresponding relation for the TBox language is not used.

Theorem 1 *Let $(\mathcal{K}, P, \{b\})$ be an $\mathcal{ALCQI}$ learning instance. For $\mathcal{L}_S \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALCQ}, \mathcal{ALCQI}\}$, the following are equivalent:*

1. *$(\mathcal{K}, P, \{b\})$ has an $\mathcal{L}_S$ solution;*
2. *there exists a model $\mathcal{I}$ of $\mathcal{K}$ of finite outdegree such that for all $a \in P$ and all models $\mathcal{J}$ of $\mathcal{K}$, $\mathcal{J}, a \not\sim_{\mathcal{L}_S} \mathcal{I}, b$.*

The proof of Theorem 1 is rather standard, based on Lemma 1 and compactness.

We now turn to the case where $\mathcal{L}_S$ is a Horn DL. The main difference to Theorem 1 is that a product construction is involved. Interestingly, our characterization is rather flexible regarding $\mathcal{L}_T$, which can even be $\mathcal{ALCQI}$. Let $\mathcal{I}_i, d_i, i \in I$, be a family of pointed interpretations. The *(direct) product* $\prod_{i \in I} \mathcal{I}_i, d$ is the pointed interpretation defined by

$$\begin{aligned} \Delta^{\prod \mathcal{I}_i} &= \{d : I \to \textstyle\bigcup_{i \in I} \Delta^{\mathcal{I}_i} \mid \text{ for } i \in I : d(i) \in \Delta^{\mathcal{I}_i}\} \\ A^{\prod \mathcal{I}_i} &= \{d \in \Delta^{\prod \mathcal{I}_i} \mid \text{for } i \in I : d(i) \in A^{\mathcal{I}_i}\}, \quad A \in \mathsf{N_C} \\ r^{\prod \mathcal{I}_i} &= \{(d, e) \mid \text{for } i \in I : (d(i), e(i)) \in r^{\mathcal{I}_i}\}, \quad r \in \mathsf{N_R} \end{aligned}$$

and $d(i) = d_i$, for all $i \in I$. We say that a set $\mathcal{M}$ of models of a KB $\mathcal{K}$ is $\mathcal{L}$ *complete* if for every $\mathcal{L}$ concept C and every $a \in \mathsf{ind}(\mathcal{K})$, $\mathcal{K} \models C(a)$ iff $a \in C^{\mathcal{I}}$ for all $\mathcal{I} \in \mathcal{M}$. If, for example, $\mathcal{K}$ is an $\mathcal{ALCQI}$ KB, then the class of all 'forest models' of $\mathcal{K}$ of finite outdegree is well-known to be $\mathcal{ALCQI}$ complete.

Theorem 2 *Let $(\mathcal{K}, P, \{b\})$ be an $\mathcal{ALCQI}$ learning instance. For $\mathcal{L}_S \in \{\mathcal{EL}, \mathcal{ELI}\}$ and $\mathcal{M}$ a set of models of $\mathcal{K}$ that is $\mathcal{L}_S$ complete, the following are equivalent:*

1. *$(\mathcal{K}, P, \{b\})$ has an $\mathcal{L}_S$ solution;*
2. *there exists a model $\mathcal{I}$ of $\mathcal{K}$ of finite outdegree such that $\prod_{a \in P}(\prod_{\mathcal{J} \in \mathcal{M}}(\mathcal{J}, a)) \not\preceq_{\mathcal{L}_S} \mathcal{I}, b$.*

The proof uses Lemma 1 and the fact that for any $\mathcal{ELI}$ concept C and family of pointed interpretations $(\mathcal{I}_i, d_i)_{i \in I}$, $d \in C^{\prod_{i \in I} \mathcal{I}_i}$ if and only if $d_i \in C^{\mathcal{I}_i}$ for all $i \in I$.

4 Characterizations for Expressive DLs

We establish characterizations of concept separability that are more refined than the initial one presented in Theorem 1. The refined characterizations establish a suprising connection between concept separability and (U)CQ-evaluation on KBs. They also provide the foundation for decision procedures for concept separability that we develop later on. We start with $\mathcal{ALCI}$ and then proceed via $\mathcal{ALCQI}$ to $\mathcal{ALC}$ and $\mathcal{ALCQ}$, for which the characterizations are slightly more technical.

A *homomorphism* h from an ABox $\mathcal{A}$ to an interpretation $\mathcal{I}$ is a mapping h from $\mathsf{ind}(\mathcal{A})$ to $\Delta^{\mathcal{I}}$ such that $A(d) \in \mathcal{A}$ implies $h(d) \in A^{\mathcal{I}}$ and $r(c, d) \in \mathcal{A}$ implies $(h(c), h(d)) \in r^{\mathcal{I}}$. We write $\mathcal{A}, a \to \mathcal{I}, b$ if there exists a homomorphism h from $\mathcal{A}$ to $\mathcal{I}$ with $h(a) = b$.

Theorem 3 *Let $(\mathcal{K}, P, \{b\})$ be an $\mathcal{ALCI}$ learning instance. Then the following conditions are equivalent:*

1. *$(\mathcal{K}, P, \{b\})$ has an $\mathcal{ALCI}$ solution;*
2. *there exists a model $\mathcal{I}$ of $\mathcal{K}$ such that for all $a \in P$, $\mathcal{A}, a \not\to \mathcal{I}, b$.*

Theorem 3 is proved by showing the equivalence of Condition 2 in Theorems 1 and 3. One first observes that it suffices to consider models $\mathcal{I}$ that contain no distinct d, e with $\mathcal{I}, d \sim_{\mathcal{ALCI}} \mathcal{I}, e$. Then the direction from Theorem 3 to Theorem 1 follows as the restriction of a bisimulation between $\mathcal{J}$ and $\mathcal{I}$ to the ABox individuals in $\mathcal{A}$ is a homomorphism to $\mathcal{I}$. For the converse direction, one carefully constructs the required model $\mathcal{J}$ of $\mathcal{K}$ from $\mathcal{I}$ using the given homomorphism.

To adapt Theorem 3 from $\mathcal{ALCI}$ to $\mathcal{ALCQI}$, we use homomorphisms h from $\mathcal{A}$ to $\mathcal{I}$ that are *locally injective* for all roles r, that is, the restriction of h to the set $\mathsf{succ}_r^{\mathcal{A}}(a) = \{b \mid r(a, b) \in \mathcal{A}\}$ is injective for all $a \in \mathsf{ind}(\mathcal{A})$. Now $\mathcal{A}, a \to^i \mathcal{I}, b$ is defined like $\mathcal{A}, a \to \mathcal{I}, b$, but based on homomorphisms that are locally injective for all roles r.

Theorem 4 *Let $(\mathcal{K}, P, \{b\})$ be an $\mathcal{ALCQI}$ learning instance. Then the following conditions are equivalent:*

1. *$(\mathcal{K}, P, \{b\})$ has an $\mathcal{ALCQI}$ solution;*
2. *there exists a model $\mathcal{I}$ of $\mathcal{K}$ such that for all $a \in P$, $\mathcal{A}, a \not\to^i \mathcal{I}, b$.*

Now for the adaptation of Theorems 3 and 4 to DLs without inverse roles. Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{ALCQI}$ KB and $a \in \mathsf{ind}(\mathcal{A})$. We use $\mathcal{A}_a^{\downarrow}$ to denote the set of assertions in $\mathcal{A}$ that use only individual names from the set $\mathsf{reach}_{\mathcal{A}}(a)$ of individual names reachable from a in $G_{\mathcal{A}}$. Now $\mathcal{A}, a \to_r \mathcal{I}, b$ means that there is a homomorphism h from $\mathcal{A}_a^{\downarrow}$ to $\mathcal{I}$ with $h(a) = b$ such that the extended KB $\mathcal{K}_{\mathcal{I},h}$, defined as

$$(\mathcal{T}, \mathcal{A} \cup \{C(c) \mid C \in \mathsf{sub}(\mathcal{K}), c \in \mathsf{reach}_{\mathcal{A}}(a), h(c) \in C^{\mathcal{I}}\}),$$

is satisfiable. We define $\mathcal{A}, a \to_r^i \mathcal{I}, b$ analogously, additionally requiring that h is locally injective for all role names r (but not necessarily for inverse roles).

Theorem 5 *Let $(\mathcal{K}, P, \{b\})$ be an $\mathcal{ALC}$ (resp. $\mathcal{ALCQ}$) learning instance. Then the following conditions are equivalent:*

1. *$(\mathcal{K}, P, \{b\})$ has an $\mathcal{ALC}$ (resp. $\mathcal{ALCQ}$) solution;*
2. *there exists a model $\mathcal{I}$ of $\mathcal{K}$ such that for all $a \in P$, $\mathcal{A}, a \not\to_r \mathcal{I}, b$ (resp. $\mathcal{A}, a \not\to_r^i \mathcal{I}, b$).*

The following example illustrates the reason for why the characterization in Theorem 5 requires $\mathcal{K}_{\mathcal{I},h}$ to be satisfiable.

Example 2 Consider $\mathcal{A} = \{r(c,a), A(a), A(b)\}$, $P = \{a\}$, $N = \{b\}$, $\mathcal{T} = \{\top \sqsubseteq \forall r.B\}$, and let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$. Then $\mathcal{K} \models B(a)$ and $\mathcal{K} \not\models B(b)$ and so B is an $\mathcal{ALC}$ solution for $(\mathcal{K}, P, N)$. Observe that $h : a \mapsto b$ is a homomorphism from $\mathcal{A}_a^{\downarrow}$ to any model $\mathcal{I}$ of $\mathcal{K}$, but for any model $\mathcal{I}$ of $\mathcal{K}$ with $b \in (\neg B)^{\mathcal{I}}$ the extended KB $\mathcal{K}_{\mathcal{I},h}$ is not satisfiable.

We observe a slightly surprising consequence of the above characterizations. Concept separability is *TBox anti-monotone* in a DL $\mathcal{L}$ if extending the TBox cannot result in a solution for a learning instance to become available, that is, for all $\mathcal{L}$ learning instances $(\mathcal{K}_1, P, N)$ and $(\mathcal{K}_2, P, N)$ with $\mathcal{T}_1 \subseteq \mathcal{T}_2$ and $\mathcal{A}_1 = \mathcal{A}_2$, $(\mathcal{K}_1, P, N)$ has an $\mathcal{L}$ solution if $(\mathcal{K}_2, P, N)$ has an $\mathcal{L}$ solution.

Theorem 6 *Concept separability is TBox anti-monotone in $\mathcal{ALCI}$ and $\mathcal{ALCQI}$, but not in $\mathcal{EL}$, $\mathcal{ELI}$, $\mathcal{ALC}$, and $\mathcal{ALCQ}$.*

Note that TBox anti-monotonicity in $\mathcal{ALCI}$ and $\mathcal{ALCQI}$ follows directly from Theorems 3 and 4. To show that concept separability is not TBox anti-monotone in $\mathcal{ALC}$ and $\mathcal{ALCQ}$ consider $(\mathcal{K}, P, N)$ from Example 2 and let $\mathcal{K}' = (\emptyset, \mathcal{A})$. Then $(\mathcal{K}', P, N)$ has no $\mathcal{ALCQ}$ solution (and thus no $\mathcal{ALC}$ solution), but $(\mathcal{K}, P, N)$ has an $\mathcal{ALC}$ solution. Example 1 (1) and (2) above shows that concept separability is not TBox anti-monotone in $\mathcal{EL}$ and $\mathcal{ELI}$.

5 Complexity: Expressive DLs

We clarify the complexity of concept separability and concept definability in expressive DLs, which turn out to be NExpTime-complete in $\mathcal{ALC}$, $\mathcal{ALCI}$, and $\mathcal{ALCQ}$, and ExpTime-complete in $\mathcal{ALCQI}$. In data complexity, they are Σ_2^p-complete in all four DLs. In both lower and upper bound proofs, we exploit a connection between separability and (appropriate versions of) UCQ evaluation on KBs that is suggested by the characterizations in the previous section.

Theorem 7 *$\mathcal{ALCI}$ concept separability and $\mathcal{ALCI}$ concept definability are* NExpTime*-complete.*

The upper bound in Theorem 7 is proved by a polynomial time reduction to the complement of rooted UCQ evaluation on $\mathcal{ALCI}$ KBs and the lower bound is proved by a polynomial time reduction from the complement of rooted CQ evaluation on $\mathcal{ALCI}$ KBs. Both problems are coNExpTime-complete [Lutz, 2008] and rather well known, so we refrain from giving definitions and only mention that a CQ is *rooted* if it is connected and has at least one answer variable, and that a UCQ is *rooted* if every CQ in it is.

For the first reduction, let $(\mathcal{K}, P, \{b\})$ be an $\mathcal{ALCI}$ learning instance with $\mathcal{K} = (\mathcal{T}, \mathcal{A})$. For every $a \in P$, let q_a be the maximal connected component $\mathcal{A}_a$ of $\mathcal{A}$ that contains a, viewed as a connected CQ with only answer variable a. Further, let q be the UCQ $\bigvee_{a \in P} q_a$. Then $(\mathcal{K}, P, \{b\})$ has a solution iff $\mathcal{K} \not\models q(b)$. In fact, $\mathcal{K} \not\models q(b)$ iff there is a model $\mathcal{I}$ of $\mathcal{K}$ such that $\mathcal{I} \not\models q_a(b)$ for all $a \in P$, that is, $\mathcal{A}_a, a \not\to \mathcal{I}, b$. This is equivalent to $\mathcal{A}, a \not\to \mathcal{I}, b$ as required by Point 2 of Theorem 3 since for all other connected components $\mathcal{A}'$ of $\mathcal{A}$, the identity is a homomorpshism to $\mathcal{I}$.

For the second reduction, let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{ALCI}$ KB, $q(x)$ a unary rooted CQ, and $a \in \mathsf{ind}(\mathcal{A})$. Let $\mathcal{A}_q$ be q viewed as an ABox where now x is an individual name, and let $\mathcal{A}'$ be the disjoint union of $\mathcal{A}$ and $\mathcal{A}_q$. Let further $\mathcal{K}' = (\mathcal{T}, \mathcal{A}')$, $P = \{x\}$, and $N = \{a\}$. Using similar arguments as above, one can show that $\mathcal{K} \not\models q(a)$ iff $(\mathcal{K}', P, N)$ has a solution. Note that this establishes hardness already for learning instances with a *single positive example* (and a single negative example). The reduction can be modified to show hardness of definability. In fact, the lower bound proof in [Lutz, 2008] applies already if we restrict ABoxes to be of the simple form $\{A_0(a_0)\}$ and can easily be modified so that the CQ $q(x)$ used is such that only the individual x can be an answer to q in $\mathcal{A}_q$ (by introducing a fresh concept name X, adding the atom $X(x)$ to q and the assertion $X(a_0)$ to the ABox). Then we can set $P = \{x\}$ as before and N is the set of all other individuals in $\mathcal{A}'$.

In the case of $\mathcal{ALCQI}$, separability closely corresponds to a version of unary rooted (U)CQ evaluation that is based on locally injective homomorphisms from the CQ to models of the KB. We introduce this problem in the appendix and prove that, remarkably, it is only ExpTime-complete. The intuitive reason is that the coNExpTime-lower bound for CQ evaluation on $\mathcal{ALCI}$ KBs requires to 'fold' the CQ in exponentially many different ways, which can only be done with locally non-injective homomorphisms. In $\mathcal{ALCQI}$, concept separability is thus no harder than standard reasoning problems such as concept satisfiability w.r.t. a TBox—these are ExpTime-complete for all expressive DLs considered in this paper [**?**].

Theorem 8 *$\mathcal{ALCQI}$ concept separability and $\mathcal{ALCQI}$ concept definability are* ExpTime*-complete.*

For the case without inverse roles, we also introduce corresponding versions of (U)CQ evaluation, based on the kinds of homomorphism used in Theorem 5. These turn out to be coNExpTime-complete. The lower bound is proved by a reduction of a suitable version of the tiling problem and crucially exploits the satisfiability condition used in the homomorphisms in Theorem 5. We work with a single positive example in the case of $\mathcal{ALC}$ while in $\mathcal{ALCQ}$ the number of positive examples is not bounded by a constant.

Theorem 9 *For $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCQ}\}$, $\mathcal{L}$ concept separability and $\mathcal{L}$ concept definability are* NExpTime*-complete.*

We close this section with clarifying data complexity.

Theorem 10 *For $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCQ}, \mathcal{ALCI}, \mathcal{ALCQI}\}$, $\mathcal{L}$ concept separability and $\mathcal{L}$ concept definability are Σ_2^p-complete in data complexity.*

The upper bounds are proved by a 'guess-coguess and check' style procedure. The lower bounds are proved by reduction of the problem to decide whether for a given undirected graph G and $k \geq 1$, there is a 2-coloring of G that does not generate a monochromatic k-clique [Rutenburg, 1986]. The proof requires two positive examples.

6 Complexity: Horn DLs

We study concept separability and concept definability in the case that the separation language is one of the Horn DLs $\mathcal{EL}$ and $\mathcal{ELI}$. As the TBox language, we consider both Horn DLs and expressive DLs. It turns out that these problems tend to be undecidable, notable exceptions being the cases that the TBox language is $\mathcal{EL}$ or $\mathcal{ELI}$ and the separation language is $\mathcal{EL}$.

We first refine the characterization from Theorem 2. For $\mathcal{L} \in \{\mathcal{EL}, \mathcal{ELI}\}$, a model $\mathcal{U}$ of a knowledge base $\mathcal{K}$ is $\mathcal{L}$ *universal* if $\mathcal{K} \models C(a)$ iff $\mathcal{U} \models C(a)$, for all $\mathcal{L}$ concepts C and $a \in \mathsf{ind}(\mathcal{A})$.

Theorem 11 *Let $(\mathcal{K}, P, \{b\})$ be an $\mathcal{ELI}$ learning instance, $\mathcal{L}_S \in \{\mathcal{EL}, \mathcal{ELI}\}$, and $\mathcal{U}$ an $\mathcal{L}_S$ universal model for $\mathcal{K}$ of finite outdegree. Then the following are equivalent:*

1. *$(\mathcal{K}, P, \{b\})$ has an $\mathcal{L}_S$ solution,*
2. *$\Pi_{a\in P}(\mathcal{U}, a) \not\preceq_{\mathcal{L}_S} \mathcal{U}, b$.*

The absence of disjunction in Horn DLs forces the product in Point 2, which captures the commonalities of the positive examples and achieves true generalization, avoiding overfitting. It is, however, also responsible for the high complexity and even undecidability of concept separability in Horn DLs. This is in fact rather surprising given that standard reasoning problems such as subsumption are only PTIME-complete in $\mathcal{EL}$ and EXPTIME-complete in $\mathcal{ELI}$.

We start with the separation language $\mathcal{EL}$, considering both $\mathcal{EL}$ and $\mathcal{ELI}$ as the TBox language.

Theorem 12 *Let $\mathcal{L}_T \in \{\mathcal{EL}, \mathcal{ELI}\}$. Then $(\mathcal{L}_T, \mathcal{EL})$ concept separability and definability are* EXPTIME*-complete, both in combined complexity and in data complexity.*

For the lower bound, we adapt a proof used in [Harel *et al.*, 2002] to prove that the simulation problem between $\mathcal{I}, d$ and $\mathcal{J}, e$ is EXPTIME-hard when $\mathcal{I}$ is a synchronized product of given interpretations and $\mathcal{J}$ is an explicitly given interpretation. Remarkably, the lower bound holds already for empty TBoxes. The number of positive examples is not bounded by a constant. For the upper bound, we build on Point 2 of Theorem 11, constructing the product of (exponential size) $\mathcal{EL}$ universal models of $\mathcal{ELI}$ knowledge bases and then deciding $\mathcal{EL}$ similarity in polynomial time. The upper bound easily extends to the case where the learned concept can use only symbols from a given signature.

Surprisingly, separability and definability even become undecidable when we extend $\mathcal{EL}$ with inverse roles. This implies that no learning algorithm can be complete and terminating.

Theorem 13 *$\mathcal{ELI}$ concept separability and definability are undecidable.*

The proof is by a rather subtle reduction of the tiling problem of rectangles of unbounded size, borrowing and extending some ideas from the proof given in [Botoeva *et al.*, 2019] that the CQ entailment problem between $\mathcal{ALC}$ KBs is undecidable. It requires only two positive examples.

We remark that $\mathcal{ELI}$ concept separability is closely related to the problem of learning conjunctive queries in the context of $\mathcal{ELI}$ KBs ('query by example'). That problem was stated to be 2EXPTIME-complete in [Gutiérrez-Basulto *et al.*, 2018], but the proof of the upper bound turns out to be incorrect. In the appendix, we use a minor variation of our proof of Theorem 13 to show that the problem is actually undecidable.

Another related problem is the existence of most specific concepts (MSCs) and least common subsumers (LCSs) [Baader, 2003; Lutz *et al.*, 2010; Zarrieß and Turhan, 2013]. In fact, if $(\mathcal{K}, P, \{b\})$ is a learning instance, MSCs M_a exist of all $a \in P$, and a LCS C of the M_a exists, then $(\mathcal{K}, P, \{b\})$ has a solution if $\mathcal{K} \not\models C(b)$. We conjecture that a variation of the proof of Theorem 13 can be used to show that in $\mathcal{ELI}$, the existence of the LCS is undecidable in the presence of a TBox.

We now consider the case where the TBox is formulated in an expressive DL and the separation language is a Horn DL. For example, such a setup is natural if avoiding overfitting is a concern. Also here, separability turns out to be undecidable.

Theorem 14 *Let $\mathcal{L}_T \in \{\mathcal{ALC}, \mathcal{ALCQ}, \mathcal{ALCI}, \mathcal{ALCQI}\}$ and $\mathcal{L}_S \in \{\mathcal{EL}, \mathcal{ELI}\}$. Then $(\mathcal{L}_T, \mathcal{L}_S)$ concept separability and $(\mathcal{L}_T, \mathcal{L}_S)$ concept definability are undecidable.*

The proof is by reduction from the already mentioned CQ entailment problem between $\mathcal{ALC}$ KBs which undecidable already for (directed or undirected) tree-shaped CQs.

7 First Observations on Strong Separability

There is an alternative notion of concept learning that has been proposed in the literature [Lehmann *et al.*, 2014; Badea and Nienhuys-Cheng, 2000; Fanizzi *et al.*, 2018]. An $\mathcal{L}$ concept C is a *strong $\mathcal{L}$ solution* to a learning instance $(\mathcal{K}, P, N)$ if $\mathcal{K} \models C(a)$ for all $a \in P$ and $\mathcal{K} \models \neg C(a)$ for all $a \in N$. We then define *strong $\mathcal{L}$ concept separability* in the expected way. While we leave a thorough investigation for future work, we observe the following characterization.

Theorem 15 *For $\mathcal{L}_S \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALCQ}, \mathcal{ALCQI}\}$, an $\mathcal{ALCQI}$ learning instance $(\mathcal{K}, P, N)$ has a strong $\mathcal{L}$ solution iff for all models $\mathcal{I}$ and $\mathcal{J}$ of $\mathcal{K}$, all $a \in P$ and all $b \in N$, $\mathcal{I}, a \not\sim_{\mathcal{L}_S} \mathcal{J}, b$.*

Based on this result, one can show that for $\mathcal{ALC}$ and $\mathcal{ALCI}$ there is a tight link between strong concept separability and KB satisfiability (rather than to UCQ-evaluation on KBs) which can be used to prove the following.

Theorem 16 *For $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}\}$, strong $\mathcal{L}$ concept separability is* EXPTIME*-complete in combined complexity and* CONP*-complete in data complexity.*

Note the drop in complexity compared to the non-strong version of concept separability, from NEXPTIME to EXPTIME and from Σ_2^p to CONP. For $\mathcal{ALCQ}$ and $\mathcal{ALCQI}$, the complexity of strong $\mathcal{L}$ concept separability remains open.

8 Discussion

This paper provides characterizations and complexity results for concept separability in many important DLs. It would be interesting to further add other popular DL features such as role hierarchies, transitive roles, and nominals, and it might also be relevant to consider cases of $(\mathcal{L}_T, \mathcal{L}_S)$ concept separability in which $\mathcal{L}_T$ is less expressive than $\mathcal{L}_S$. We also plan to investigate strong separability in more detail. Finally, it would be interesting to understand in how far the characterizations given in this paper can be used or extended to analyse the version space beyond emptiness and whether they can also be used to analyse or craft refinement operators.

Acknowledgements

Lutz was supported by DFG CRC 1320 Ease.

References

[Baader, 2003] Franz Baader. Least common subsumers and most specific concepts in a description logic with existential restrictions and terminological cycles. In *Proc. of IJCAI*, pages 319–324, 2003.

[Badea and Nienhuys-Cheng, 2000] Liviu Badea and Shan-Hwei Nienhuys-Cheng. A refinement operator for description logics. In *Proc. of ILP*, pages 40–59, 2000.

[Bienvenu *et al.*, 2014] Meghyn Bienvenu, Balder ten Cate, Carsten Lutz, and Frank Wolter. Ontology-based data access: A study through disjunctive datalog, CSP, and MMSNP. *ACM Trans. Database Syst.*, 39(4):33:1–33:44, 2014.

[Botoeva *et al.*, 2019] Elena Botoeva, Roman Kontchakov, Carsten Lutz, Vladislav Ryzhikov, Frank Wolter, and Michael Zakharyaschev. Query inseparability for ALC ontologies. *Artif. Intell.*, 272:1–51, 2019.

[Bühmann *et al.*, 2016] Lorenz Bühmann, Jens Lehmann, and Patrick Westphal. DL-learner - A framework for inductive learning on the semantic web. *J. Web Sem.*, 39:15–24, 2016.

[Bühmann *et al.*, 2018] Lorenz Bühmann, Jens Lehmann, Patrick Westphal, and Simon Bin. DL-learner structured machine learning on semantic web data. In *Proc. of WWW*, pages 467–471, 2018.

[Calvanese *et al.*, 2013] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Data complexity of query answering in description logics. *Artif. Intell.*, 195:335–360, 2013.

[Chandra *et al.*, 1981] Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. Alternation. *J. ACM*, 28:114–133, 1981.

[Fanizzi *et al.*, 2018] Nicola Fanizzi, Giuseppe Rizzo, Claudia d'Amato, and Floriana Esposito. DLFoil: Class expression learning revisited. In *Proc. of EKAW*, pages 98–113, 2018.

[Goranko and Otto, 2007] Valentin Goranko and Martin Otto. Model theory of modal logic. In *Handbook of Modal Logic*, pages 249–329. Elsevier, 2007.

[Gutiérrez-Basulto *et al.*, 2018] Víctor Gutiérrez-Basulto, Jean Christoph Jung, and Leif Sabellek. Reverse engineering queries in ontology-enriched systems: The case of expressive horn description logic ontologies. In *Proc. of IJCAI-ECAI*, 2018.

[Harel *et al.*, 2002] David Harel, Orna Kupferman, and Moshe Y. Vardi. On the complexity of verifying concurrent transition systems. *Inf. Comput.*, 173(2):143–161, 2002.

[Hirsh *et al.*, 2004] Haym Hirsh, Nina Mishra, and Leonard Pitt. Version spaces and the consistency problem. *Artif. Intell.*, 156(2):115–138, 2004.

[Iannone *et al.*, 2007] Luigi Iannone, Ignazio Palmisano, and Nicola Fanizzi. An algorithm based on counterfactuals for concept learning in the semantic web. *Appl. Intell.*, 26(2):139–159, 2007.

[Kietz, 1993] Jörg-Uwe Kietz. Some lower bounds for the computational complexity of inductive logic programming. In *Proc. of ECML*, pages 115–123, 1993.

[Lehmann and Haase, 2009] Jens Lehmann and Christoph Haase. Ideal downward refinement in the $\mathcal{EL}$ description logic. In *Proc. of ILP*, pages 73–87, 2009.

[Lehmann and Hitzler, 2010] Jens Lehmann and Pascal Hitzler. Concept learning in description logics using refinement operators. *Machine Learning*, 78:203–250, 2010.

[Lehmann and Völker, 2014] Jens Lehmann and Johanna Völker. *Perspectives on Ontology Learning*, volume 18 of *Studies on the Semantic Web*. IOS Press, 2014.

[Lehmann *et al.*, 2014] Jens Lehmann, Nicola Fanizzi, Lorenz Bühmann, and Claudia d'Amato. Concept learning. In *Perspectives on Ontology Learning*, pages 71–91. AKA / IOS Press, 2014.

[Lisi, 2012] Francesca A. Lisi. A formal characterization of concept learning in description logics. In *Proc. of DL*, 2012.

[Lutz *et al.*, 2010] Carsten Lutz, Robert Piro, and Frank Wolter. Enriching $\mathcal{EL}$-concepts with greatest fixpoints. In *Proc. of ECAI*, pages 41–46, 2010.

[Lutz *et al.*, 2011] Carsten Lutz, Robert Piro, and Frank Wolter. Description logic TBoxes: Model-theoretic characterizations and rewritability. In *Proc. of IJCAI*, 2011.

[Lutz, 2008] Carsten Lutz. The complexity of conjunctive query answering in expressive description logics. In *Proc. of IJCAR*, pages 179–193, 2008.

[Pitt and Valiant, 1988] Leonard Pitt and Leslie G. Valiant. Computational limitations on learning from examples. *J. ACM*, 35(4):965–984, 1988.

[Rutenburg, 1986] Vladislav Rutenburg. Complexity of generalized graph coloring. In *Proc. of MFCS*, pages 573–581, 1986.

[Sarker *et al.*, 2017] Md. Kamruzzaman Sarker, Ning Xie, Derek Doran, Michael Raymer, and Pascal Hitzler. Explaining trained neural networks with semantic web technologies: First steps. In *Proc. of NeSy*, 2017.

[Straccia and Mucci, 2015] Umberto Straccia and Matteo Mucci. pFOIL-DL: Learning (fuzzy) $\mathcal{EL}$ concept descriptions from crisp OWL data using a probabilistic ensemble estimation. In *Proc. of SAC*, pages 345–352. ACM, 2015.

[Tran *et al.*, 2014] Thanh-Luong Tran, Quang-Thuy Ha, Thi-Lan-Giao Hoang, Linh Anh Nguyen, and Hung Son Nguyen. Bisimulation-based concept learning in description logics. *Fundam. Inform.*, 133(2-3):287–303, 2014.

[Zarrieß and Turhan, 2013] Benjamin Zarrieß and Anni-Yasmin Turhan. Most specific generalizations w.r.t. general $\mathcal{EL}$-TBoxes. In *Proc. of IJCAI*, pages 1191–1197, 2013.

A Proofs for Section 3

We prove Theorems 1 and 2. We associate every model $\mathcal{I}$ of $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ with an undirected graph $G^-_{\mathcal{I}}$ that has the set of vertices $\Delta^{\mathcal{I}}$ and an edge $\{d, e\}$ whenever $(d, e) \in r^{\mathcal{I}}$ for some role r with $\{e, d\} \not\subseteq \mathsf{ind}(\mathcal{A})$. We call $\mathcal{I}$ a *forest model* if $G^-_{\mathcal{I}}$ is a collection of trees and there are no reflexive loops and multi-edges outside of $\mathcal{A}$, that is,

1. $(d, d) \notin r^{\mathcal{I}}$ for all $d \in \Delta^{\mathcal{I}} \setminus \mathsf{ind}(\mathcal{A})$ and role names r and
2. if $(e, d) \in r^{\mathcal{I}} \cap s^{\mathcal{I}}$, r and s roles, then $\{e, d\} \subseteq \mathsf{ind}(\mathcal{A})$ or $r = s$.

The following result is well known.

Lemma 2 *Let $\mathcal{K}$ be a $\mathcal{ALCQI}$ KB and C an $\mathcal{ALCQI}$ concept. If $\mathcal{K} \not\models C(a)$, then there exists a forest model $\mathcal{I}$ of $\mathcal{K}$ of finite outdegree with $a \notin C^{\mathcal{I}}$.*

We show the following slightly more informative version of Theorem 1.

Theorem 17 *Let $\mathcal{L}_S \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALCQ}, \mathcal{ALCQI}\}$ and let $(\mathcal{K}, P, \{b\})$ be a learning instance with $\mathcal{K}$ a $\mathcal{ALCQI}$ KB. Then the following are equivalent:*

1. *$(\mathcal{K}, P, \{b\})$ has a $\mathcal{L}_S$ solution;*
2. *there exists a forest model $\mathcal{I}$ of $\mathcal{K}$ of finite outdegree such that for all $a \in P$ and all models $\mathcal{J}$ of $\mathcal{K}$, $\mathcal{J}, a \not\sim_{\mathcal{L}_S} \mathcal{I}, b$;*
3. *there exists a model $\mathcal{I}$ of $\mathcal{K}$ of finite outdegree such that for all $a \in P$ and all models $\mathcal{J}$ of $\mathcal{K}$, $\mathcal{J}, a \not\sim_{\mathcal{L}_S} \mathcal{I}, b$.*

Proof. (1) $\Rightarrow$ (2). Suppose there exists an $\mathcal{L}_S$ concept C such that $\mathcal{K} \models C(a)$ for all $a \in P$ and $\mathcal{K} \not\models C(b)$. Then there exists a model $\mathcal{I}$ of $\mathcal{K}$ such that $\mathcal{I} \not\models C(b)$. By Lemma 2, $\mathcal{I}$ can be assumed to be a forest model of finite outdegree. For all $a \in P$ and any model $\mathcal{J}$ of $\mathcal{K}$ we have $\mathcal{J} \models C(a)$. Thus, $\mathcal{J}, a \not\equiv_{\mathcal{L}_S} \mathcal{I}, b$ and, by Lemma 1, $(\mathcal{J}, a) \not\sim_{\mathcal{L}_S} (\mathcal{I}, b)$, for all $a \in P$.

(3) $\Rightarrow$ (1). For any interpretation $\mathcal{I}$ and $d \in \Delta^{\mathcal{I}}$, we set

$$\text{type}^{\mathcal{I}}_{\mathcal{L}_S}(d) = \{C \ \mathcal{L}_S \text{ concept} \mid \mathcal{I} \models C(d)\}.$$

Now assume there exists a model $\mathcal{I}$ of $\mathcal{K}$ such that for all $a \in P$ and all models $\mathcal{J}$ of $\mathcal{K}$, $\mathcal{J}, a \not\sim_{\mathcal{L}_S} \mathcal{I}, b$. We prove that there exists a finite

$$\{C_1, \dots, C_n\} \subseteq \text{type}^{\mathcal{I}}_{\mathcal{L}_S}(b)$$

such that $\mathcal{K} \models \bigvee_{1 \leq i \leq n} \neg C_i(a)$ for all $a \in P$. Then we are done as $\mathcal{K} \not\models \bigvee_{1 \leq i \leq n} \neg C_i(b)$. For a proof by contradiction, suppose this is not the case. Then, for some $a_0 \in P$, every finite subset $\{C_1, \dots, C_n\}$ of $\text{type}^{\mathcal{I}}_{\mathcal{L}_S}(b)$ satisfies $\mathcal{K} \not\models \bigvee_{1 \leq i \leq n} \neg C_i(a_0)$. Then

$$\mathcal{K} \cup \{C_1(a_0), \dots, C_n(a_0)\}$$

is satisfiable for all finite $\{C_1, \dots, C_n\} \subseteq \text{type}^{\mathcal{I}}_{\mathcal{L}_S}(b)$. By compactness,

$$\mathcal{K} \cup \{C(a_0) \mid C \in \text{type}^{\mathcal{I}}_{\mathcal{L}_S}(b)\}$$

is satisfiable. But any model $\mathcal{J}$ of $\{C(a_0) \mid C \in \text{type}^{\mathcal{I}}_{\mathcal{L}_S}(b)\}$ satisfies $\mathcal{J}, a_0 \equiv_{\mathcal{L}_S} \mathcal{I}, b$. As $\mathcal{I}$ has finite outdegree, this implies, by Lemma 1, $\mathcal{J}, a_0 \sim_{\mathcal{L}_S} \mathcal{I}, b$. We have derived a contradiction.

(2) $\Rightarrow$ (3). Immediate. ❑

For the proof of Theorem 2, first observe that for any family of pointed interpretation $\mathcal{I}_i, d_i$, $i \in I$, and $\mathcal{ELI}$ concept C the following conditions are equivalent for $d \in \Delta^{\prod_{i \in I} \mathcal{I}_i}$ with $d(i) = d_i$ for all $i \in I$:

1. $d_i \in C^{\mathcal{I}_i}$ for all $i \in I$;
2. $d \in C^{\prod_{i \in I} \mathcal{I}_i}$.

(1) $\Rightarrow$ (2). Let $\mathcal{L}_S \in \{\mathcal{EL}, \mathcal{ELI}\}$. Suppose there exists an $\mathcal{L}_S$ concept C such that $\mathcal{K} \models C(a)$ for all $a \in P$ and $\mathcal{K} \not\models C(b)$. Then there exists a model $\mathcal{I}$ of $\mathcal{K}$ such that $\mathcal{I} \not\models C(b)$. By Lemma 2, $\mathcal{I}$ can be assumed to be a model of finite outdegree. For all $a \in P$ and any model $\mathcal{J}$ of $\mathcal{K}$ we have $\mathcal{J} \models C(a)$. Denote by $\vec{a}$ the element of $\prod_{a \in P}(\prod_{\mathcal{J} \in \mathcal{M}}(\mathcal{J}, a))$ that maps every pair $(a, \mathcal{J}) \in P \times \mathcal{M}$ to a. Then

$$\prod_{a \in P}(\prod_{\mathcal{J} \in \mathcal{M}} (\mathcal{J}, a)) \models C(\vec{a}),$$

and it follows that

$$\prod_{a \in P}(\prod_{\mathcal{J} \in \mathcal{M}} (\mathcal{J}, a)) \not\leq_{\mathcal{L}_S} \mathcal{I}, b$$

and, by Lemma 1, that

$$\prod_{a \in P}(\prod_{\mathcal{J} \in \mathcal{M}} (\mathcal{J}, a)) \not\precsim_{\mathcal{L}_S} \mathcal{I}, b.$$

(2) $\Rightarrow$ (1). Assume there exists a model $\mathcal{I}$ of $\mathcal{K}$ of finite outdegree such that

$$\prod_{a \in P}(\prod_{\mathcal{J} \in \mathcal{M}} (\mathcal{J}, a)) \not\precsim_{\mathcal{L}_S} \mathcal{I}, b.$$

By Lemma 1,

$$\prod_{a \in P}(\prod_{\mathcal{J} \in \mathcal{M}} (\mathcal{J}, a)) \not\leq_{\mathcal{L}_S} \mathcal{I}, b.$$

Thus, there exists a $\mathcal{L}_S$ concept C with $\prod_{a \in P}(\prod_{\mathcal{J} \in \mathcal{M}}(\mathcal{J}, a)) \models C(\vec{a})$ and $\mathcal{I} \not\models C(b)$. Then $\mathcal{K} \not\models C(b)$ and $\mathcal{J} \models C(a)$, for all $\mathcal{J} \in \mathcal{M}$ and $a \in P$. By completeness of $\mathcal{M}$, $\mathcal{K} \models C(a)$ for all $a \in P$ and it follows that $(\mathcal{K}, P, \{b\})$ has an $\mathcal{L}_S$ solution.

B Proofs for Section 4

We first introduce notation for speaking about the relations $\sim_{\mathcal{L}}$. Let $\mathcal{I}, d$ and $\mathcal{J}, e$ be pointed interpretations.

- A relation $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$ is an *i-bisimulation* if conditions [Atom], [Forth] and [Back] from Figure 1 hold, where A ranges over all concept names and r ranges over role names and inverse roles. Then $\mathcal{I}, d$ and $\mathcal{J}, e$ are *i-bisimilar*, in symbols $\mathcal{I}, d \sim_{\mathcal{ALCI}} \mathcal{J}, e$, if there exists an i-bisimulation S between $\mathcal{I}$ and $\mathcal{J}$ containing (d, e).
- A relation $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$ is a *counting i-bisimulation* if conditions [Atom], [QForth] and [QBack] from Figure 1 hold, where A ranges over all concept names and r ranges over role names and inverse roles. Then $\mathcal{I}, d$ and $\mathcal{J}, e$ are *counting i-bisimilar*, in symbols $\mathcal{I}, d \sim_{\mathcal{ALCQI}} \mathcal{J}, e$, if there exists a counting i-bisimulation S between $\mathcal{I}$ and $\mathcal{J}$ containing (d, e).

- A relation $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{J}}$ is a *counting bisimulation* if conditions [Atom], [QForth], and [QBack] from Figure 1 hold, where A ranges over all concept names and r ranges over *role names*. Then $\mathcal{I}, d$ and $\mathcal{J}, e$ are *counting bisimilar*, in symbols $\mathcal{I}, d \sim_{\mathcal{ALCQ}} \mathcal{J}, e$, if there exists a counting bisimulation S between $\mathcal{I}$ and $\mathcal{J}$ containing (d, e).

Rather than proving the refined characterizations directly, we first introduce new model-theoretic relations between pointed ABoxes $\mathcal{A}, a$ and pointed interpretations $\mathcal{I}, b$ that bridge the gap between the various types of bisimulations characterizing the existence of $\mathcal{L}_S$ solutions and the existence of (various types of) homomorphisms between pointed ABoxes and interpretations used in the refined characterizations. In what follows we often identify an ABox $\mathcal{A}$ with the interpretation $\mathcal{I}_{\mathcal{A}}$ with domain $\mathsf{ind}(\mathcal{A})$ and

- $d \in A^{\mathcal{I}_{\mathcal{A}}}$ if $A(d) \in \mathcal{A}$, for all $A \in \mathsf{N_C}$, and
- $(d, d') \in r^{\mathcal{I}_{\mathcal{A}}}$ if $r(d, d') \in \mathcal{A}$, for all $r \in \mathsf{N_R}$.

Let $\mathcal{A}$ be an ABox and $\mathcal{I}$ an interpretation. For any DL $\mathcal{L}$ and relation $S \subseteq \mathsf{ind}(\mathcal{A}) \times \Delta^{\mathcal{I}}$ we consider the following condition:

($\sim_{\mathcal{L}}$) if $(d, d_1), (d, d_2) \in S$, then $d_1 \sim_{\mathcal{L}} d_2$.

Then S is called

(1) an $\mathcal{ALCI}$ *simulation* if the conditions [AtomR] and [Forth] hold for all concept names A and roles r and the condition ($\sim_{\mathcal{ALCI}}$) holds. We write $\mathcal{A}, a \preceq_{\mathcal{ALCI}} \mathcal{I}, b$ if there exists an $\mathcal{ALCI}$ simulation S between $\mathcal{A}$ and $\mathcal{I}$ with $(a, b) \in S$;

(2) an $\mathcal{ALCQI}$ *simulation* if the conditions [AtomR] and [QForth] hold for all concept names A and all roles r and the condition ($\sim_{\mathcal{ALCQI}}$) holds. We write $\mathcal{A}, a \preceq_{\mathcal{ALCQI}} \mathcal{I}, b$ if there exists an $\mathcal{ALCQI}$ simulation S between $\mathcal{A}$ and $\mathcal{I}$ with $(a, b) \in S$;

(3) if $S \subseteq \mathsf{reach}_{\mathcal{A}}(a) \times \Delta^{\mathcal{I}}$, then S is called an $\mathcal{ALC}$ *simulation* if the following conditions hold: (a) if $\mathcal{A}_a^{\downarrow} \neq \emptyset$, then the conditions [AtomR] and [Forth] hold for all concept names A and all role names r and the condition ($\sim_{\mathcal{ALC}}$) holds. (b) The extended KB $\mathcal{K}_{\mathcal{I},S} = (\mathcal{T}, \mathcal{A}_{\mathcal{I},S})$ with $\mathcal{A}_{\mathcal{I},S}$ defined as the union of $\mathcal{A}$ and

$$\{C(c) \mid C \in \mathsf{sub}(\mathcal{K}), c \in \mathsf{reach}_{\mathcal{A}}(a), (c, c') \in S, c' \in C^{\mathcal{I}}\}$$

is satisfiable. We write $\mathcal{A}, a \preceq_{\mathcal{ALC}} \mathcal{I}, b$ if there exists an $\mathcal{ALC}$ simulation S between $\mathcal{A}$ and $\mathcal{I}$ with $(a, b) \in S$;

(4) if $S \subseteq \mathsf{reach}_{\mathcal{A}}(a) \times \Delta^{\mathcal{I}}$, then S is called an $\mathcal{ALCQ}$ *simulation* if the following conditions hold: (a) if $\mathcal{A}_a^{\downarrow} \neq \emptyset$, then the conditions [AtomR] and [QForth] hold for all concept names A and all role names r and the condition ($\sim_{\mathcal{ALCQ}}$) holds. (b) The extended KB $\mathcal{K}_{\mathcal{I},S} = (\mathcal{T}, \mathcal{A}_{\mathcal{I},S})$ with $\mathcal{A}_{\mathcal{I},S}$ defined as the union of $\mathcal{A}$ and

$$\{C(c) \mid C \in \mathsf{sub}(\mathcal{K}), c \in \mathsf{reach}_{\mathcal{A}}(a), (c, c') \in S, c' \in C^{\mathcal{I}}\}$$

is satisfiable. We write $\mathcal{A}, a \preceq_{\mathcal{ALCQ}} \mathcal{I}, b$ if there exists an $\mathcal{ALCQ}$ simulation S between $\mathcal{A}$ and $\mathcal{I}$ with $(a, b) \in S$.

We are in the position now to formulate the bridge characterizations.

Lemma 3 *Let $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCQ}, \mathcal{ALCI}, \mathcal{ALCQI}\}$. Let $(\mathcal{K}, P, \{b\})$ be a $\mathcal{L}$ learning instance. Then the following conditions are equivalent for any model $\mathcal{I}$ of $\mathcal{K}$ of finite outdegree:*

1. *there exists $a \in P$ and a model $\mathcal{J}$ of $\mathcal{K}$ such that $\mathcal{J}, a \sim_{\mathcal{L}} \mathcal{I}, b$;*
2. *there exists $a \in P$ such that $\mathcal{A}, a \preceq_{\mathcal{L}} \mathcal{I}, b$.*

Proof. We first consider $\mathcal{ALCI}$.

(1.) $\Rightarrow$ (2.) Observe that the restriction $S|_{\mathsf{ind}(\mathcal{A})}$ of any i-bisimulation between models $\mathcal{J}$ and $\mathcal{I}$ of $\mathcal{K}$ with $(a, b) \in S$ defines an $\mathcal{ALCI}$ simulation between $\mathcal{A}$ and $\mathcal{I}$ containing (a, b).

(2.) $\Rightarrow$ (1.) Assume that $\mathcal{A}, a \preceq_{\mathcal{ALCI}} \mathcal{I}, b$ for some $a \in P$. Let $S \subseteq \mathsf{ind}(\mathcal{A}) \times \Delta^{\mathcal{I}}$ be an $\mathcal{ALCI}$ simulation between $\mathcal{A}$ and $\mathcal{I}$ and assume $(a, b) \in S$. Take for every c in the maximally connected component $\mathcal{A}_a$ of a in $\mathcal{A}$ an element $c' \in \Delta^{\mathcal{I}}$ such that $(c, c') \in S$. Let $\mathcal{I}_c$ be an isomorphic copy of $\mathcal{I}$ with c' replaced by c. Let $S_{c,c'}$ be the i-bisimulation between $\mathcal{I}_c$ and $\mathcal{I}$ with $(d, e) \in S_{c,c'}$ iff d is a copy of e. In particular $(c, c') \in S$. Also, for any $c'' \neq c'$ with $(c, c'') \in S$ let $S_{c,c''}$ be an i-bisimulation between $\mathcal{I}_c$ and $\mathcal{I}$ with $(c, c'') \in S_{c,c''}$ witnessing Condition ($\sim_{\mathcal{ALCI}}$). For $\mathcal{A}' := \mathcal{A} \setminus \mathcal{A}_a$, let $\mathcal{J}'$ be any model of $\mathcal{T}$ and $\mathcal{A}'$ (for example, an isomorphic copy of $\mathcal{I}$). Now let $\mathcal{J}$ be defined by hooking to $\mathcal{A}_a$ the interpretations $\mathcal{I}_c$, $c \in \mathcal{A}_a$, at c and adding a copy of $\mathcal{J}'$. In detail, we set

$$A^{\mathcal{J}} = \bigcup_{c \in \mathsf{ind}(\mathcal{A}_a)} A^{\mathcal{I}_c} \cup A^{\mathcal{J}'} \cup \{d \mid A(d) \in \mathcal{A}_a\}$$

for all $A \in \mathsf{N_C}$, and

$$r^{\mathcal{J}} = \bigcup_{c \in \mathsf{ind}(\mathcal{A}_a)} r^{\mathcal{I}_c} \cup r^{\mathcal{J}'} \cup \{(d, d') \mid r(d, d') \in \mathcal{A}_a\}$$

for all $r \in \mathsf{N_R}$. Using the conditions [AtomR] and [Forth] for S, it is not difficult to show that the maximal i-bisimulation S' between $\mathcal{J}$ and $\mathcal{I}$ contains

$$S \cup \bigcup_{(c,c') \in S} S_{c,c'}$$

Moreover, as $\mathcal{J}'$ is a model of $\mathcal{A}'$ and $\mathcal{T}$, the domain of S' contains $\Delta^{\mathcal{J}} \setminus \Delta^{\mathcal{J}'}$, and $\mathcal{I}$ is a model of $\mathcal{K}$, it follows that $\mathcal{J}$ is a model of $\mathcal{K}$.

We now consider $\mathcal{ALCQI}$. The *i-unfolding* $\mathcal{J}$ of an interpretation $\mathcal{I}$ at node $d \in \Delta^{\mathcal{I}}$ is defined as follows. A *path* ρ in $\mathcal{I}$ starting at d is any sequence $d_0 r_0 d_1 \cdots r_{n-1} d_n$ such that $d_0 = d$, r_i is a role for all $0 \leq i < n$, and $(d_i, d_{i+1}) \in r_i^{\mathcal{I}}$, for all $0 \leq i < n$. We set $\mathsf{tail}(\rho) = d_n$. Then the domain $\Delta^{\mathcal{J}}$ of $\mathcal{J}$ is the set of all path $\rho = d_0 r_0 d_1 \cdots r_{n-1} d_n$ starting at d such that $r_i \neq r_{i+1}^-$ for $0 \leq i < n$, and

$$A^{\mathcal{J}} = \{\rho \in \Delta^{\mathcal{J}} \mid \mathsf{tail}(\rho) \in A^{\mathcal{I}}\}$$

and

$$\begin{aligned} r^{\mathcal{J}} \quad = \quad & \{(\rho, \rho r d) \mid \rho r d \in \Delta^{\mathcal{I}}\} \cup \\ & \{(\rho r^- d, \rho) \mid \rho r^- d \in \Delta^{\mathcal{I}}\} \end{aligned}$$

It is folklore and easy to prove that $\mathcal{I}, d \sim_{\mathcal{ALCQI}} \mathcal{J}, d$. We return to the proof of the equivalence.

(1.) $\Rightarrow$ (2.) Observe that the restriction $S|_{\mathsf{ind}(\mathcal{A})}$ of any counting i-bisimulation between models $\mathcal{J}$ and $\mathcal{I}$ of $\mathcal{K}$ with

$(a, b) \in S$ defines an $\mathcal{ALCQI}$ simulation between $\mathcal{A}$ and $\mathcal{I}$ containing (a, b).

(2.) $\Rightarrow$ (1.) Assume that $\mathcal{A}, a \preceq_{\mathcal{ALCQI}} \mathcal{I}, b$ for some $a \in P$. Let $S \subseteq \mathsf{ind}(\mathcal{A}) \times \Delta^{\mathcal{I}}$ be an $\mathcal{ALCQI}$ simulation between $\mathcal{A}$ and $\mathcal{I}$ with $(a, b) \in S$. We construct a model $\mathcal{J}$ of $\mathcal{K}$ such that $\mathcal{J}, a \sim_{\mathcal{ALCQI}} \mathcal{I}, b$.

Take for every c in the maximally connected component $\mathcal{A}_a$ of a in $\mathcal{A}$ an element $c' \in \Delta^{\mathcal{I}}$ such that $(c, d) \in S$. Let $c_1, \ldots, c_n$ be the set r-successors of c in $\mathcal{A}$ for some role r. Let $d_1, \ldots, d_m$ be the r-successors of d in $\mathcal{I}$. There exists a subset D of $\{d_1, \ldots, d_m\}$ such that S contains a bijection f between $\{c_1, \ldots, c_n\}$ and D. Assume without loss of generality that $f = \{(c_1, d_1), \ldots, (c_n, d_n)\}$. Now take for $n < i \leq m$ the i-unfolding $\mathcal{I}^*_{d_i}$ of $\mathcal{I}$ at d_i. Remove from each $\mathcal{I}^*_{d_i}$ the r^--successor $d_i r^- d$ together with the subtree below $d_i r^- d$ and denote the resulting interpretation by $\mathcal{I}'_{d_i}$. Now expand $\mathcal{A}_a$ by adding c to $A^{\mathcal{J}}$ for all concept names A with $d \in A^{\mathcal{I}}$ and connecting for every $n < i \leq m$ a fresh copy of $\mathcal{I}'_{d_i}$ to $\mathcal{A}_a$ by adding (c, d_i) to $r^{\mathcal{J}}$. This is done for all $c \in \mathsf{ind}(\mathcal{A}_a)$. It is not difficult to show that the maximal counting i-bisimulation S' between the resulting interpretation $\mathcal{J}_0$ and $\mathcal{I}$ contains (a, b). For $\mathcal{A}' := \mathcal{A} \setminus \mathcal{A}_a$, let $\mathcal{J}'$ be any model of $\mathcal{T}$ and $\mathcal{A}'$ and let $\mathcal{J}$ be defined by taking the disjoint union of $\mathcal{J}_0$ and $\mathcal{J}'$. Then $\mathcal{J}$ is as required.

The now consider $\mathcal{ALC}$.

(1.) $\Rightarrow$ (2.) Consider the restriction $S|_{\mathsf{ind}(\mathcal{A}^{\downarrow}_a)}$ of any bisimulation S between models $\mathcal{J}$ and $\mathcal{I}$ of $\mathcal{K}$ with $(a, b) \in S$ to $\mathsf{ind}(\mathcal{A}^{\downarrow}_a)$. Then S satisfies condition (a). Moreover, the extended KB $\mathcal{K}_{\mathcal{I},S} = (\mathcal{T}, \mathcal{A}_{\mathcal{I},S})$ from condition (b) is satisfiable as it is satisfied by $\mathcal{J}$.

(2.) $\Rightarrow$ (1.) A straightforward adaptation of the proof for $\mathcal{ALCI}$.

The proof for $\mathcal{ALCQ}$ is similar to the proof for $\mathcal{ALCQI}$ and omitted. ❑

Lemma 4 *Let $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALCQ}, \mathcal{ALCQI}\}$ and let $(\mathcal{K}, P, \{b\})$ be a $\mathcal{L}$ learning instance. Then the following conditions are equivalent:*

1. *there exists a model $\mathcal{I}$ of $\mathcal{K}$ such that for all $a \in P$, $\mathcal{A}, a \not\preceq_{\mathcal{L}} \mathcal{I}, b$;*
2. *there exists a model $\mathcal{I}$ of $\mathcal{K}$ such that for all $a \in P$, $\mathcal{A}, a \not\to_{\mathcal{L}} \mathcal{I}, b$,*

where $\to_{\mathcal{ALCI}} := \to$, $\to_{\mathcal{ALCQI}} := \to^i$, $\to_{\mathcal{ALC}} := \to^r$, and $\to_{\mathcal{ALCQ}} := \to^i_r$. Moreover, one can choose in Conditions (1) and (2) models $\mathcal{I}$ that coincide regarding their domain and the interpretation of symbols used in $\mathcal{K}$.

Proof. We give the proof for $\mathcal{ALCI}$. (1.) $\Rightarrow$ (2.) is trivial. (2.) $\Rightarrow$ (1.). Assume $\mathcal{I}$ is a model of $\mathcal{K}$ such that for all $a \in P$, $\mathcal{A}, a \not\to \mathcal{I}, b$. Define $\mathcal{I}'$ by extending $\mathcal{I}$ by taking for every $d \in \Delta^{\mathcal{I}}$ a fresh concept name C_d and setting $C_d^{\mathcal{I}'} = \{d\}$. $\mathcal{I}'$ is as required for (1.) since any two i-bisimilar nodes in $\mathcal{I}'$ are identical. ❑

We are now in the position to prove Theorems 3, 4, and 5. As the proofs are similar, we only give the proof of Theorem 3.

Proof of Theorem 3. Assume first that $(\mathcal{K}, P, \{b\})$ has an $\mathcal{ALCI}$ solution. By Theorem 17, there exists a model $\mathcal{I}$ of $\mathcal{K}$ of finite outdegree such that for all $a \in P$ and all models $\mathcal{J}$ of $\mathcal{K}$, $\mathcal{J}, a \not\sim_{\mathcal{L}_S} \mathcal{I}, b$. By Lemma 3, $\mathcal{A}, a \not\preceq_{\mathcal{ALCI}} \mathcal{I}, b$ for all $a \in P$. By Lemma 4, there exists a model $\mathcal{I}'$ of $\mathcal{K}$ such that $\mathcal{A}, a \not\to \mathcal{I}', b$ for all $a \in P$, as required. Conversely, assume there exists a model $\mathcal{I}$ of $\mathcal{K}$ such that $\mathcal{A}, a \not\to \mathcal{I}, b$ for all $a \in P$. Clearly we may assume that $\mathcal{I}$ has finite outdegree. By Lemma 4, there exists a model $\mathcal{I}'$ of $\mathcal{K}$ of finite outdegree such that $\mathcal{A}, a \not\preceq_{\mathcal{ALCI}} \mathcal{I}', b$ for all $a \in P$. By Lemma 3, for all model $\mathcal{J}$ of $\mathcal{K}$ and all $a \in P$, $\mathcal{J}, a \not\sim_{\mathcal{ALCI}} \mathcal{I}', b$. By Theorem 17, $(\mathcal{K}, P, \{b\})$ has an $\mathcal{ALCI}$ solution.

C Proofs for Section 5

We start with introducing the syntax and semantics of conjunctive queries and unions of conjunctive queries, as well as the associated evaluation problems. A *conjunctive query (CQ)* is of the form $q = \exists \mathbf{y}\, \varphi(\mathbf{x}, \mathbf{y})$, where $\mathbf{x}$ and $\mathbf{y}$ are tuples of variables and $\varphi(\mathbf{x}, \mathbf{y})$ is a conjunction of *atoms* of the form $A(x)$ or $r(x, y)$ with A a concept name, r a role name, and $x, y \in \mathbf{x} \cup \mathbf{y}$. We call $\mathbf{x}$ the *answer variables* of q and $\mathbf{y}$ the *quantified variables*, writing $q(\mathbf{x})$ to emphasize that the answer variables of q are $\mathbf{x}$. The CQ q gives rise to an ABox $\mathcal{A}_q$ simply by viewing variables as individuals names and atoms as assertions. This also associates q with the directed graph $G_{\mathcal{A}_q}$ and thus we can use standard terminology from graph theory also for CQs. A CQ is *rooted* if it is connected and has at least one answer variable.

A *homomorphism* from q to an interpretation $\mathcal{I}$ is a function $h : \mathbf{x} \cup \mathbf{y} \to \Delta^{\mathcal{I}}$ such that $h(x) \in A^{\mathcal{I}}$ for every atom $A(x)$ of $q(\mathbf{x})$ and $(h(x), h(y)) \in r^{\mathcal{I}}$ for every atom $r(x, y)$ of $q(\mathbf{x})$. We write $\mathcal{I} \models q(\mathbf{a})$ and call $\mathbf{a}$ an *answer to* $q(\mathbf{x})$ *on* $\mathcal{I}$ if there is a homomorphism from $q(\mathbf{x})$ to $\mathcal{I}$ with $h(\mathbf{x}) = \mathbf{a}$.

A *union of conjunctive queries (UCQ)* is a disjunction of one or more CQs that all have the same answer variables. The *arity* of a (U)CQ is the number of answer variables in it and a CQ is *unary* if it is of arity 1. A UCQ is *rooted* if every CQ in it is. For a UCQ q and an interpretation $\mathcal{I}$, we write $\mathcal{I} \models q(\mathbf{a})$ if there is a CQ p in q with $\mathcal{I} \models p(\mathbf{a})$. For a KB $\mathcal{K}$, we write $\mathcal{K} \models q(\mathbf{a})$ if $\mathcal{I} \models q(\mathbf{a})$ for every model $\mathcal{I}$ of $\mathcal{K}$.

Let $\mathcal{L}$ be a DL. With *rooted (U)CQ evaluation on $\mathcal{L}$ KBs*, we mean the problem to decide, given an $\mathcal{L}$ KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, a rooted (U)CQ q, and a candidate answer $\mathbf{a} \subseteq \mathsf{ind}(\mathcal{A})$, whether $\mathcal{K} \models q(\mathbf{a})$. Theorem 3 and the remark made directly after it closely relates $\mathcal{ALCI}$ concept separability to rooted (U)CQ evaluation on $\mathcal{ALCI}$ KBs. This is crucially exploited in the proof of Theorem 7 given in the main part of the paper. In what follows we provide proofs for the other theorems in Section 5. Our strategy will be to identify a version of (U)CQ evaluation that corresponds to the concept separability problem at hand, determine its complexity, and finally provide mutual reductions as in the proof of Theorem 7.

We start with Theorem 8, concerned with $\mathcal{ALCQI}$. $\mathcal{ALCQI}$ concept separability is related to rooted (U)CQ evaluation based on locally injective homomorphisms, introduced next. Let q be a UCQ. For an interpretation $\mathcal{I}$, we write $\mathcal{I} \models^i q(\mathbf{a})$ if there is a CQ p in q and a locally injective homomorphism from $p(\mathbf{x})$ to $\mathcal{I}$ with $h(\mathbf{x}) = \mathbf{a}$. For a KB $\mathcal{K}$, we write $\mathcal{K} \models^i q(\mathbf{a})$ if $\mathcal{I} \models^i q(\mathbf{a})$ for all models $\mathcal{I}$ of $\mathcal{K}$. Let $\mathcal{L}$ be a DL. *Injective rooted (U)CQ evaluation on $\mathcal{L}$ KBs* means to decide, given an $\mathcal{L}$ KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, a rooted (U)CQ $q(\mathbf{x})$, and a candidate answer $\mathbf{a} \subseteq \mathsf{ind}(\mathcal{A})$, whether $\mathcal{K} \models^i q(\mathbf{a})$.

We need some preliminaries. A concept C is *satisfiable* w.r.t. a TBox $\mathcal{T}$ if $\mathcal{T}$ has a model with $C^\mathcal{I} \neq \emptyset$. We use $\mathsf{sub}(\mathcal{T})$ to denote the set of concepts that occur in $\mathcal{T}$, closed under subconcepts. Let $\mathcal{K}$ be an $\mathcal{ALCQI}$ KB. We associate each model $\mathcal{I}$ of $\mathcal{K}$ an undirected graph $G^-_\mathcal{I}$ that has the set of vertices $\Delta^\mathcal{I}$ and an edge $\{d, e\}$ whenever $(d, e) \in r^\mathcal{I}$ for some role r with $\{e, d\} \not\subseteq \mathsf{ind}(\mathcal{A})$. We call $\mathcal{I}$ a *forest model* if $G^-_\mathcal{I}$ is a collection of trees and there are no reflexive loops and multi-edges outside of $\mathcal{A}$, that is,

1. $(d, d) \notin r^\mathcal{I}$ for all $d \in \Delta^\mathcal{I} \setminus \mathsf{ind}(\mathcal{A})$ and role names r and
2. if $(e, d) \in r^\mathcal{I} \cap s^\mathcal{I}$, r and s roles, then $\{e, d\} \subseteq \mathsf{ind}(\mathcal{A})$ or $r = s$.

The *outdegree* of forest model $\mathcal{I}$ is the outdegree of $G^-_\mathcal{I}$.

The next theorem determines the complexity of injective unary rooted CQ evaluation. Once it is established, we can prove Theorem 8 in exactly the same way as Theorem 7, based on Theorems 4 and 8. Details are omitted.

Theorem 18 *Injective unary rooted CQ evaluation on $\mathcal{ALCQI}$ KBs is* ExpTime*-complete. The same is true for UCQ evaluation.*

Proof. The lower bound is easy by reduction from unsatisfiability in $\mathcal{ALC}$, which is ExpTime-complete [**?**]. Indeed, an $\mathcal{ALC}$ concept C is unsatisfiable w.r.t. an $\mathcal{ALC}$ TBox $\mathcal{T}$ iff

$$(\mathcal{T} \cup \{\neg C \sqsubseteq A\}, \{B(a)\}) \models^i A(a).$$

where A and B are fresh concept names.

For the upper bound, let an $\mathcal{ALCQI}$ KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, a unary rooted UCQ $q_0(x_0)$, and a candidate answer $a_0 \in \mathsf{ind}(\mathcal{A})$ be given.

We use Γ to denote the set of pairs $(r(x_1, x_2), q_2(x_2))$ such that there exists a CQ q' in q_0 with $r(x_1, x_2)$ an atom in q' and q_2 a set of atoms in q' with the following properties: the set $q' \setminus \{r(x_1, x_2)\}$ consists of two (potentially empty) disconnected components q_1 and q_2 of atoms where

- x_i does not occur in q_{3-i} for $i \in \{1, 2\}$,
- $x_0 \neq x_2$ and x_0 does not occur in q_2, and
- $q_{r,q_2}(x_1) = r(x_1, x_2) \cup q_2$ is a tree-shaped unary CQ with root x_1.

Clearly, the number of pairs in Γ is polynomial in the size of q_0. Define the following sets of unary tree-shaped CQs:

$$\begin{aligned}\Gamma_0 &= \{q_2(x_2) \mid (r(x_1, x_2), q_2) \in \Gamma\} \\ \Gamma_1 &= \{q_{r,q_2}(x_1) \mid (r(x_1, x_2), q_2) \in \Gamma\}\end{aligned}$$

Let $\mathsf{con}(\mathcal{T})$ be the closure under single negation of the set of subconcepts of concepts in $\mathcal{T}$. A *type* is a maximal set of concepts $t \subseteq \mathsf{con}(\mathcal{T})$ such that $\bigsqcap t$ is satisfiable w.r.t. $\mathcal{T}$. An *extended type* t additionally contains for every $q(x) \in \Gamma_0$ either the expression $\mathsf{isat}(q(x))$ or $\neg\mathsf{isat}(q(x))$.

Let $\mathcal{I}$ be a forest model of $\mathcal{K}$ and $d \notin \mathsf{ind}(\mathcal{A})$ an r-successor of some $a \in \mathsf{ind}(\mathcal{A})$. We say that $\mathsf{isat}(q(x))$ *is satisfied in d in $\mathcal{I}$* if $\mathcal{I} \models^i q(d)$ and this is witnessed by a locally injective homomorphism h into $\Delta^\mathcal{I} \setminus \mathsf{ind}(\mathcal{A})$. Observe that then h is locally injective if, and only if, it is injective. An extended type t *is satisfied in $\mathcal{I}$ in d* if its concepts are satisfied in d and $\mathsf{isat}(q(x))$ is satisfied in d iff $\mathsf{isat}(q(x)) \in t$, for all $q(x) \in \Gamma_0$. An *extended type assignment for $\mathcal{A}$* is a function μ that assigns to every $a \in \mathsf{ind}(\mathcal{A})$ a type $\mu(a)$ and to every triple (a, r, t) with $a \in \mathsf{ind}(\mathcal{A})$, r a role from $\mathcal{K}$, and t an extended type, a natural number $\mu(a, r, t)$. An extended type assignment is *small* if for every a and role r we have $\sum_t \mu(a, r, t) \leq |\mathcal{T}|$. An extended type assignment for $\mathcal{A}$ is *realized by a forest model $\mathcal{I}$* if for every $a \in \mathsf{ind}(\mathcal{A})$:

1. a satisfies $\mu(a)$ in $\mathcal{I}$;
2. the number of r-successors of a in $\mathcal{I}$ outside $\mathsf{ind}(\mathcal{A})$ satisfying an extended type t is $\mu(a, r, t)$.

An extended type assignment μ is *$\mathcal{K}$-realizable* iff there exists a forest model $\mathcal{I}$ of $\mathcal{K}$ that realizes it.

A *forest decomposition of a CQ q* in q_0 is a partition

$$\hat{q} \cup q_1(x_1) \cup \cdots \cup q_n(x_n)$$

of (the set of atoms in) q such that $q_i(x_i) \in \Gamma_0$. We assume that the variables $x_1, \ldots, x_n$ all occur in $\hat{q}$, which can be achieved by adding 'dummy atoms' of the form $\top(x_i)$. It can be verified that $\hat{q}$ and q_i share only the variable x_i, that x_0 occurs in $\hat{q}$, and that if x_0 occurs in q_i, then $x_i = x_0$. A forest decomposition of the UCQ q_0 is any forest decomposition of any of its CQs.

Given a forest decompostion $\hat{q} \cup q_1 \cup \cdots \cup q_n$ and x_i in $\hat{q}$ and a role r we obtain the tree-shaped CQ $q_r(x_i)$ as the conjunction of all queries of the form $r(x_i, y) \cup q(y)$ in $\{q_1, \ldots, q_n\}$. For an extended type assignment μ we write $\mathcal{K} \models^i_\mu q_r(a)$ if $\mathcal{I} \models^i q_r(a)$ for some model (equivalently, all models) $\mathcal{I}$ of $\mathcal{K}$ realizing μ and this is witnessed by an injective homomorphism from the variables of $q_r(x_i)$ into the interpretation induced by the subtree generated by a in $\mathcal{I}$ (so a is the only individual in $\mathsf{ind}(\mathcal{A})$ in the range of h).

We say that an extended type assignment μ *avoids* q_0 if for every forest decomposition

$$\hat{q} \cup q_1(x_1) \cup \cdots \cup q_n(x_n)$$

there is no locally injective homomorphism $h : \hat{q} \rightarrow \mathcal{A} \cup \{A(a) \mid A \in \mu(a)\}$ such that $h(x_0) = a_0$ and $\mathcal{K} \models^i_\mu q_r(h(x_i))$ for all roles r and for $1 \leq i \leq n$.

Claim. $\mathcal{K} \not\models^i q_0(a_0)$ iff there is a small $\mathcal{K}$ realizable extended type assignment that avoids q_0.

We only sketch the proof of the claim. In the 'if' direction, we are given a $\mathcal{K}$ realizable small extended type assignment μ that avoids q_0. Take any forest model $\mathcal{I}$ of $\mathcal{K}$ that realizes μ. Then $\mathcal{I} \not\models^i q_0(a_0)$. In the 'only if' direction, we can read from any forest model $\mathcal{I}$ of $\mathcal{K}$ such that $\mathcal{I} \not\models^i q_0(a_0)$ first such a model of small outdegree and then a small extended type assignment that avoids q_0 by simply using as $\mu(a)$ the type realized at $\mathcal{I}$ at a and as $\mu(a, r, t)$ the number of r-successors outside $\mathcal{A}$ that satisfy t, for every $a \in \mathsf{ind}(\mathcal{A})$.

It now remains to observe that the number of small extended types assigments for $\mathcal{K}$ is at most exponential in the size of $\mathcal{K}$, that it can be checked in exponential time in the size of $\mathcal{K}$ whether a small extended type assignment is $\mathcal{K}$ realizable, and that is can be checked in exponential time whether a small $\mathcal{K}$ realizable extended type assignment avoids q_0.

❏

To treat $\mathcal{ALC}$ concept separability, we introduce a special form of (U)CQ evaluation based on the kind of homomorphism also used in Theorem 5. Let q be a unary UCQ. For an interpretation $\mathcal{I}$, we write $\mathcal{I} \models^r q(a)$ if there is a CQ $p(x)$ in q and a homomorphism from $p_x^\downarrow$ to $\mathcal{I}$ with $h(x) = a$, with $p_x^\downarrow$ the restriction of p to the variables (directedly) reachable from x, such that the extended ABox

$$\mathcal{A}_p \cup \{C(y) \mid C \in \mathsf{sub}(\mathcal{K}), y \in \mathsf{var}(q), h(y) \in C^\mathcal{I}\}$$

is satisfiable w.r.t. $\mathcal{T}$. For a KB $\mathcal{K}$, we write $\mathcal{K} \models^r q(\mathbf{a})$ if $\mathcal{I} \models^r q(\mathbf{a})$ for all models $\mathcal{I}$ of $\mathcal{K}$. Let $\mathcal{L}$ be a DL. *Reachable unary rooted (U)CQ evaluation on $\mathcal{L}$ KBs* means to decide, given an $\mathcal{L}$ KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, a unary rooted (U)CQ $q(x)$, and a candidate answer $\mathbf{a} \subseteq \mathsf{ind}(\mathcal{A})$, whether $\mathcal{K} \models^r q(\mathbf{a})$.

Theorem 19 *Reachable unary rooted CQ evaluation on $\mathcal{ALC}$ KBs is* CONEXPTIME*-complete. The same is true for UCQ evaluation.*

Proof. We start with the lower bound, which is by reduction of a NEXPTIME-complete tiling problem. Since the reduction is rather similar to the one in [Lutz, 2008], we describe it only on a high level of abstraction.

An *(exponential torus) tiling problem* P is a triple (T, H, V), where $T = \{0, \dots, k\}$ is a finite set of *tile types* and $H, V \subseteq T \times T$ represent the *horizontal and vertical matching conditions*. An *initial condition* for P takes the form $c = (c_0, \dots, c_{n-1}) \in T^n$. A mapping $\tau : \{0, \dots, 2^n - 1\} \times \{0, \dots, 2^n - 1\} \to T$ is a *solution* for P given c if for all $x, y < 2^n$, the following holds (where $\oplus_i$ denotes addition modulo i):

- if $\tau(x, y) = t_1$ and $\tau(x \oplus_{2^n} 1, y) = t_2$, then $(t_1, t_2) \in H$
- if $\tau(x, y) = t_1$ and $\tau(x, y \oplus_{2^n} 1) = t_2$, then $(t_1, t_2) \in V$
- $\tau(i, 0) = c_i$ for all $i < n$.

There exists a tiling problem P such that, given an initial condition c, it is NEXPTIME-complete to decide whether there exists a solution for P given c. We fix such a $P = (T, H, V)$.

Let c be an initial condition for P. It is straightforward to construct an $\mathcal{ALC}$ TBox $\mathcal{T}$ that achieves the following, see [Lutz, 2008] for a similar but more complicated construction:[1]

- Below each instance of a distinguished concept name A_0, there is a binary tree of depth $2n$ whose edges are represented by a role name r.
- The leaves of the tree correspond to the positions in the $2^n \times 2^n$-torus and the position of each leaf is represented in binary using the concept names $X_1, \dots, X_n$ for the x-coordinate and $Y_1, \dots, Y_n$ for the y-coordinate.
- Each 'leaf' has three additional successors, all attached via the role name r, marked with the concept names H (for 'here'), U (for 'up'), and R (for 'right').
- The three successors are also associated with torus positions represented via $X_1, \dots, X_n$ and $Y_1, \dots, Y_n$. If the position of the leaf that the successors are attached to is (i, j), then the H-successor also has position (i, h), the U-successor has position $(i, j \oplus_{2^n} 1)$ and the R-successor has position $(i \oplus_{2^n}, j)$.
- At each of the successor nodes, disjunction is used to 'guess' a tile from T, where tile m is represented by concept name T_m. The guess is compatible with the initial condition c given and are 'locally compatible' with the matching conditions, that is, if tiles i, j, ℓ are assigned to the H-, U-, and R-successor of the same tree leaf, then $(i, j) \in V$ and $(i, \ell) \in H$.

Now define the CQ q to consist of the following atoms, where x_0 is the only answer variable:

$$\begin{array}{l} r(x_0, x_1), \dots, r(x_{2n}, x_{2n+1}), \\ r(x_0, x_1'), r(x_1', x_2'), \dots, r(x_{2n}, x_{2n+1}) \\ s_1(y, x_{2n+1}), s_2(y, x_{2n+1}'), B_0(y) \end{array}$$

and further extend $\mathcal{T}$ with the following, for $1 \leq i \leq n$:

$$\begin{array}{l} B_0 \sqsubseteq (\forall s_1.X_i \sqcap \forall s_2.X_i) \sqcup (\forall s_1.\neg X_i \sqcap \forall s_2.\neg X_i) \\ B_0 \sqsubseteq (\forall s_1.Yi \sqcap \forall s_2.Y_i) \sqcup (\forall s_1.\neg Y_i \sqcap \forall s_2.\neg Y_i) \\ B_0 \sqsubseteq \bigsqcup_{i,j \in T, i \neq j} (\forall s_1.T_i \sqcap \forall s_2.T_j) \end{array}$$

The construction of $\mathcal{T}$ and q can be implemented in polynomial time.

Claim. $(\mathcal{T}, \{A_0(a)\}) \models q(a)$ iff P has no solution given c.

For the 'if' direction, assume that P has no solution given c. Let $\mathcal{I}$ be a model of $\mathcal{T}$ with $a \in A_0^\mathcal{I}$. Then $\mathcal{T}$ generates a tree below a as described above. Since P has no solution given c, this tree must contain a tiling defect, that is, there must be two elements d_1, d_2 reachable from a along an r-path of length $2n + 1$ such that d_1, d_2

1. are associated with the same position, that is, $d_1 \in X_i^\mathcal{I}$ iff $d_2 \in X_i^\mathcal{I}$ for $1 \leq i \leq n$ and $d_1 \in Y_i^\mathcal{I}$ iff $d_2 \in Y_i^\mathcal{I}$ for $1 \leq i \leq n$; and
2. are tiled differently, that is, $d_1 \in T_i^\mathcal{I}$ and $d_2 \in T_j^\mathcal{I}, i \neq j$.

Note that $q_{x_0}^\downarrow$ is the restriction of q to all variables except y. Clearly, there is a homomorphism h from $q_{x_0}^\downarrow$ to $\mathcal{I}$ with $h(x_{2n+1}) = d_1$ and $h(x_{2n+1}') = d_2$. It can be verified that, because d_1, d_2 satisfy Conditions 1 and 2 above, h satisfies the required satisfiability condition.

For the (contrapositive of the) 'only if' direction, assume that P has a solution given c. We can then find a model $\mathcal{I}$ of $\mathcal{T}$ with $a \in A_0^\mathcal{I}$ such that the tree enforced by $\mathcal{T}$ below a represents that solution. In particular, there is no tiling defect. Consequently, all homomorphisms from $q_{x_0}^\downarrow$ to $\mathcal{I}$ violate the satisfiability condition.

We now give a sketch of the upper bound. The following is well known for the standard semantics of (U)CQ evaluation, see for example [Lutz, 2008]. The proof is by a straightforward unraveling construction which applies without changes also to the special semantics that we are interested in here.

[1]The construction is more complicated because it must use a symmetric role for the edges of the tree, which is not the case here.

Claim. Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{ALC}$ KB, q a unary UCQ, and $a \in \mathsf{ind}(\mathcal{A})$. Then $\mathcal{K} \models^r q(a_0)$ iff for every forest model $\mathcal{I}$ of $\mathcal{K}$ of outdegree at most $|\mathcal{T}|$, $\mathcal{I} \models^r q(a)$.

A NEXPTIME algorithm for the complement of reachable unary rooted UCQ evaluation on $\mathcal{ALC}$ KBs is now as follows. Let $\mathcal{K}$, q, and a be given as an input. We guess an initial piece of a forest model $\mathcal{I}$ of $\mathcal{K}$ of outdegree at most $|\mathcal{T}|$ that is of depth at most $|q_0|$, that is, the maximum length of a (simple) path in $G^-_{\mathcal{I}}$ is $|q_0|$. Note that the number of elements in such an initial piece is single exponential, more precisely bounded by $|\mathcal{A}| + |\mathcal{A}| \cdot |\mathcal{T}|^{|q_0|}$. Along with $\mathcal{I}$, we guess an adornment $\mu : \Delta^{\mathcal{I}} \rightarrow 2^{\mathsf{sub}(\mathcal{T})}$ that specifies which subconcepts of $\mathcal{T}$ are satisfied at wich element in $\mathcal{I}$. It is required that for all $d \in \Delta^{\mathcal{I}}$, $\bigsqcap \mu(d)$ is consistent w.r.t. $\mathcal{T}$, which can be checked in EXPTIME. The adornment must also be compatible with $\mathcal{I}$, that is

- $d \in A^{\mathcal{I}}$ iff $A \in \mu(d)$ for all concept names A;
- if $(d, e) \in r^{\mathcal{I}}$, $C \in \mu(e)$ and $\exists r.C \in \mathsf{sub}(\mathcal{T})$, then $\exists r.C \in \mu(d)$.

The adornment serves the purpose to ensure that the guessed initial piece of $\mathcal{I}$ can be extended to a full forest model of $\mathcal{K}$. Since $q(x)$ is rooted, however, only the guessed initial piece can be in the range of a homomorphism h from $q^{\downarrow}_x$ to $\mathcal{I}$ that maps x to an ABox individual. The number of homomorphisms from $q^{\downarrow}_x$ to $\mathcal{I}$ is single exponential, more precisely bounded by $|\Delta^{\mathcal{I}}|^{|q|}$, and thus we can iterate through all candidates. For each candidate that turns out to be a homomorphism, we additionally verify the required satisfiability condition in EXPTIME. We accept if every homomorphism violates the satisfiability condition and reject otherwise. ❑

Theorem 9 *For $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCQ}\}$, $\mathcal{L}$ concept separability and $\mathcal{L}$ concept definability are* NEXPTIME*-complete.*

Proof. For $\mathcal{ALC}$, we can prove this in exactly the same way as Theorem 7, based on Theorems 5 and 19. In particular, we can argue as in the proof of Theorem 7 that the lower bound also applies to definability.

For $\mathcal{ALCQ}$, we need a slight variation of Theorem 19. We only sketch the required modifications. By Theorem 5, we need a version of reachable unary rooted (U)CQ evaluation in which the homomorphisms are locally injective. Note that, in contrast to the $\mathcal{ALCQI}$ case, local injectivity here only concerns role names, but not inverse roles. This has no impact on the upper bound part of the proof of Theorem 19, which goes through exactly as before.

For the lower bound, we need a minor modification because the homomorphisms used in the correctness proof of the reduction for the tiling problem are not injective. The problem is that the CQ q branches directly at the 'root' while the corresponding 'real branching' in the tree might occur on a deeper lever. Our solution is to replace q with a UCQ $q_0 \vee \cdots \vee q_{2n-1}$ that contains one CQ for each possible level on which the branching occurs. In detail, we define q_i to be

$$r(x_0, x_1), \ldots, r(x_{2n}, x_{2n+1}),$$
$$r(x_i, x'_{i+1}), r(x'_{i+1}, x'_{i+2}), \ldots, r(x_{2n}, x_{2n+1})$$
$$s_1(y, x_{2n+1}), s_2(y, x'_{2n+1}), B_0(y)$$

Note that we get the lower bound only for UCQ evaluation rather than for CQ evaluation, but this is also sufficient for obtaining Theorem 9. In fact, in the reduction from CQ evaluation to separability in the proof of Theorem 7, we can include in $\mathcal{A}'$ an ABox $\mathcal{A}_p$ for every CQ p in the input UCQ q and then choose P to contain all invididuals that correspond to the answer variable in some $\mathcal{A}_p$. ❑

Theorem 10 *For $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCQ}, \mathcal{ALCI}, \mathcal{ALCQI}\}$, $\mathcal{L}$ concept separability and $\mathcal{L}$ concept definability are Σ^p_2-complete in data complexity.*

Proof. As in the case of combined complexity, these results can be established via corresponding versions of (rooted unary) UCQ evaluation, but now assuming that the TBox is fixed and thus of constant size while both the ABox and the query remain an input. All four upper bounds are obtained in a uniform way, and so are all four lower bounds.

For the upper bounds, we establish Σ^p_2 upper bounds for all relevant versions of UCQ evaluation using the same algorithm as in the NEXPTIME upper bound established in the proof of Theorem 9. In particular, we also start with guessing an initial piece $\mathcal{I}$ of a model of $\mathcal{K}$ which is now of polynomial size since $|\mathcal{T}|$ is a constant. For verifying that the adornment μ is as desired, we need to know whether concepts of the form $\bigsqcap \mu(d)$ are satisfiable w.r.t. $\mathcal{T}$. This, however, is now trivial since there are only constantly many concepts of this form. We then coguess the homomorphisms h from $q^{\downarrow}_x$ to $\mathcal{I}$ rather than enumerating all candidates. If $\mathcal{L} \in \{\mathcal{ALCI}, \mathcal{ALCQI}\}$, then this already yields the Σ^p_2 upper bound. For $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCQ}\}$, checking that the coguessed homomorphisms h are as desired requires checking that certain ABox of polynomial size are inconsistent w.r.t. $\mathcal{T}$. This can be done by coguessing an adornment of $\mathcal{A}$ with a set of types plus a set of types that are realized in a potential model outside the ABox and then verifying that the guessed configuration does actually not give rise to a model. We need to coguess only polynomially since $|\mathcal{T}|$ is constant. This gives the Σ^p_2 upper bound.

Now for the lower bounds. We prove these directly for UCQ evaluation rather than for the complement, thus aiming at Π^p_2-hardness. The proof is by reduction from the following Π^p_2-complete problem [Rutenburg, 1986]. Given an undirected graph G and a $k \geq 1$, decide whether all 2-colorings of (the vertices of) G contain a monochromatic k-clique. Let $\mathcal{T} = \{C \sqsubseteq C_1 \sqcup C_2\}$. Given $G = (V, E)$ and k, we construct an ABox $\mathcal{A}$ and UCQ q as follows:

$$\mathcal{A} = \{r(a_u, a_v), r(a_v, a_u) \mid \{u, v\} \in E\} \cup \{C(a_v), r(a_0, a_v), r(a_v, a_0) \mid v \in V\}$$

$$q(x) = \bigwedge_{1 \leq i,j \leq k} (C_1(y_i) \wedge r(x, y_i) \wedge r(y_i, x) \wedge r(y_i, y_j)) \vee \bigwedge_{1 \leq i,j \leq k} (C_2(y_i) \wedge r(x, y_i) \wedge r(y_i, x) \wedge r(y_i, y_j))$$

where in both CQs in q, all variables except x are existentially quantified. It is now easy to verify the following.

Claim. $(\mathcal{T}, \mathcal{A}) \models q(a_0)$ iff all 2-colorings of G contain a monochromatic k-clique. ❑

D Proofs for Section 6

We recall the standard $\mathcal{ELI}$ universal model, and additionally introduce an $\mathcal{EL}$ universal model for $\mathcal{ELI}$ knowledge bases. Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and $\text{sub}(\mathcal{T})$ be the set of all sub-concepts occurring in $\mathcal{T}$. A *type for* $\mathcal{T}$ is a subset $t \subseteq \text{sub}(\mathcal{T})$ closed under consequences from $\mathcal{T}$, that is, whenever $\mathcal{T} \models \bigsqcap t \sqsubseteq D$ for some $D \in \text{sub}(\mathcal{T})$, then $D \in t$. When $a \in \mathsf{ind}(\mathcal{A})$, t, t' are types for $\mathcal{T}$, and r is a role, we write

- $a \leadsto_r^{\mathcal{T},\mathcal{A}} t$ if $\mathcal{T}, \mathcal{A} \models \exists r. \bigsqcap t(a)$ and t is maximal with this condition, and
- $t \leadsto_r^{\mathcal{T}} t'$ if $\mathcal{T} \models \bigsqcap t \sqsubseteq \exists r. \bigsqcap t'$ and t' is maximal with this condition.

A *path for* $\mathcal{K}$ is a finite sequence $\pi = ar_0t_1 \cdots t_{n-1}r_{n-1}t_n$, $n \geq 0$, with $a \in \mathsf{ind}(\mathcal{A})$, $r_0, \ldots, r_{n-1}$ roles, and $t_1, \ldots, t_n$ types for $\mathcal{T}$ such that

(i) $a \leadsto_{r_0}^{\mathcal{T},\mathcal{A}} t_1$ and (ii) $t_i \leadsto_{r_i}^{\mathcal{T}} t_{i+1}$ for every $1 \leq i < n$.

We use $\mathsf{tail}(\pi)$ to denote the last element of a path π. Let Paths be the set of all paths for $\mathcal{K}$, and define an interpretation $\mathcal{U}_\mathcal{K}$ as follows:

$$\begin{aligned}
\Delta^{\mathcal{U}_\mathcal{K}} &= \mathsf{Paths} \\
A^{\mathcal{U}_\mathcal{K}} &= \{a \in \mathsf{ind}(\mathcal{A}) \mid \mathcal{T}, \mathcal{A} \models A(a)\} \cup \\
&\quad\ \{\pi \in \mathsf{Paths} \setminus \mathsf{ind}(\mathcal{A}) \mid A \in \mathsf{tail}(\pi)\} \\
r^{\mathcal{U}_\mathcal{K}} &= \{(a,b) \in \mathsf{ind}(\mathcal{A})^2 \mid r(a,b) \in \mathcal{A}\} \cup \\
&\quad\ \{(\pi, \pi r t) \mid \pi r t \in \mathsf{Paths}\} \cup \\
&\quad\ \{(\pi r^- t, \pi) \mid \pi r^- t \in \mathsf{Paths}\}
\end{aligned}$$

Moreover, define an interpretation $\mathcal{G}_\mathcal{K}$ by taking:

$$\begin{aligned}
\Delta^{\mathcal{G}_\mathcal{K}} &= \mathsf{ind}(\mathcal{A}) \cup \{t, t' \mid t \text{ is a type for } \mathcal{T}\} \\
A^{\mathcal{G}_\mathcal{K}} &= \{a \in \mathsf{ind}(\mathcal{A}) \mid \mathcal{T}, \mathcal{A} \models A(a)\} \cup \{t_a \mid A \in t\} \\
r^{\mathcal{G}_\mathcal{K}} &= \{(a,b) \in \mathsf{ind}(\mathcal{A})^2 \mid r(a,b) \in \mathcal{A}\} \cup \\
&\quad\ \{(a,t) \mid a \leadsto_r^{\mathcal{T},\mathcal{A}} t\} \cup \{(t', a) \mid a \leadsto_{r^-}^{\mathcal{T},\mathcal{A}} t\} \cup \\
&\quad\ \{(t_1, t_2), (t_1', t_2') \mid t_1 \leadsto_r^{\mathcal{T}} t_2\} \cup \\
&\quad\ \{(t_2', t_1'), (t_2', t_1) \mid t_1 \leadsto_{r^-}^{\mathcal{T}} t_2\}
\end{aligned}$$

We have the following properties of $\mathcal{U}_\mathcal{K}$ and $\mathcal{G}_\mathcal{K}$.

Lemma 5 *For every $\mathcal{ELI}$ knowledge base $\mathcal{K}$, we have:*

1. *$\mathcal{U}_\mathcal{K}$ is an $\mathcal{ELI}$ universal model for $\mathcal{K}$ and has finite outdegree.*
2. *$\mathcal{G}_\mathcal{K}$ is an $\mathcal{EL}$ universal model for $\mathcal{K}$ and has finite outdegree.*

Proof. Point 1 is well-known, so we concentrate on Point 2. We show that $\mathcal{G}_\mathcal{K}$ is an $\mathcal{EL}$ universal model by constructing an $\mathcal{EL}$ simulation from $\mathcal{G}_\mathcal{K}$ to $\mathcal{U}_\mathcal{K}$ that contains all (a, a) for all $a \in \mathsf{ind}(\mathcal{A})$. $\mathcal{U}_\mathcal{K}$ is known to be $\mathcal{ELI}$ universal and therefore $\mathcal{EL}$ universal. Let

$$S = \{(\mathsf{tail}(\pi), \pi) \mid \pi \in \Delta^{\mathcal{U}_\mathcal{K}}\}.$$

We will show that S is an $\mathcal{EL}$ simulation from $\mathcal{G}_\mathcal{K}$ to $\mathcal{U}_\mathcal{K}$. Note that for any path π we have that $\mathsf{tail}(\pi) \in \Delta^{\mathcal{G}_\mathcal{K}}$, however the $t' \in \Delta^{\mathcal{G}_\mathcal{K}}$ are not contained in S. For the condition [AtomR] of simulations assume that $t \in A^{\mathcal{G}_\mathcal{K}}$ and $(t, \pi) \in S$. It follows that $t = \mathsf{tail}(\pi)$. Consider the first case of $t \in \mathsf{ind}(\mathcal{A})$, then we have that $\mathcal{T}, \mathcal{A} \models A(t)$ and thus $\pi \in A^{\mathcal{U}_\mathcal{K}}$. Consider the second case that t is a type for $\mathcal{T}$, then we have that $A \in t$ and thus $\pi \in A^{\mathcal{U}_\mathcal{K}}$.

For the [Forth] condition on simulations, assume that $(t, u) \in r^{\mathcal{G}_\mathcal{K}}$ and $(t, \pi) \in S$. We need to show that there is a π' such that $(\pi, \pi') \in r^{\mathcal{U}_\mathcal{K}}$ and $(u, \pi') \in S$. It follows from construction of S that $t = \mathsf{tail}(\pi)$. We distinguish cases on the shape of t and u. First, assume that $(t, u) \in \mathsf{ind}(\mathcal{A})^2$, then we have that $r(t, u) \in \mathcal{A}$ and thus we can let $\pi' = u$ and have $(\pi, \pi') \in r^{\mathcal{U}_\mathcal{K}}$ as well as $(u, \pi') \in S$. For the second case, assume that $t \in \mathsf{ind}(\mathcal{A})$ and u is a type for $\mathcal{T}$. It follows that $t \leadsto_r^{\mathcal{T},\mathcal{A}} u$ and thus there is a path $\pi' = \pi r u$ in $\Delta^{\mathcal{U}_\mathcal{K}}$. By construction we have that $(u, \pi') \in S$ and $(\pi, \pi') \in r^{\mathcal{U}_\mathcal{K}}$ by construction of $\mathcal{U}_\mathcal{K}$. For the final case we assume that both t and u are types for $\mathcal{T}$. It follows that $t \leadsto_r^{\mathcal{T}} u$ and thus there is a path $\pi' = \pi r u$ in $\Delta^{\mathcal{U}_\mathcal{K}}$. By construction, we have that $(u, \pi') \in S$ and $(\pi, \pi') \in r^{\mathcal{U}_\mathcal{K}}$ by construction of $\mathcal{U}_\mathcal{K}$.

Thus S is a simulation from $\mathcal{G}_\mathcal{K}$ to $\mathcal{U}_\mathcal{K}$ that by construction contains all (a, a) for all $a \in \mathsf{ind}(\mathcal{A})$ and therefore $\mathcal{G}_\mathcal{K}$ is $\mathcal{EL}$ universal. ❑

Theorem 12 *Let $\mathcal{L}_T \in \{\mathcal{EL}, \mathcal{ELI}\}$. Then $(\mathcal{L}_T, \mathcal{EL})$ concept separability and definability are* ExpTime*-complete, both in combined complexity and in data complexity.*

We start with the upper bound, that is, we provide an ExpTime-algorithm for deciding whether $(\mathcal{K}, P, \{b\})$ has an $\mathcal{EL}$-solution, based on Theorem 11. It consists of the following steps:

1. Construct the exponentially sized $\mathcal{EL}$-universal model $\mathcal{G}_\mathcal{K}$.
2. Construct the exponentially sized product $\prod_{a \in P}(\mathcal{G}_\mathcal{K}, a)$.
3. Check if there exists a simulation from $\prod_{a \in P}(\mathcal{G}_\mathcal{K}, a)$ to $(\mathcal{G}_\mathcal{K}, b)$ with a PTime-algorithm for finding simulations in finite structures.

Correctness is a direct consequence of the characterization provided in Theorem 11.

For showing ExpTime-hardness of $(\mathcal{L}_T, \mathcal{EL})$ concept separability we adapt a proof of ExpTime-hardness of the simulation problem for concurrent transition systems [Harel *et al.*, 2002]. More precisely, we reduce the word problem for alternating, linear space bounded Turing machines (TMs), that is, given such a TM M with linear space bound $s(n)$, we construct an ABox $\mathcal{A}$ and sets of positive and negative examples P and N, such that $((\emptyset, \mathcal{A}), P, N)$ is $\mathcal{EL}$ concept separable iff M does not accept w. It is well-known that there is a fixed alternating TM whose word problem is ExpTime-complete [Chandra *et al.*, 1981].

For our purposes, an alternating Turing machine $M = (\Gamma, Q_\forall, Q_\exists, \mapsto, q_0, F_{acc}, F_{rej})$ consists of a finite set of tape symbols Γ, a set of universal states $Q_\forall$, a set of existential states $Q_\exists$, a set of accepting states F_{acc}, a set of rejecting states F_{rej} (these sets of states are disjoint, their union is the set of all states Q), an initial state q_0 and a transition relation

$\mapsto \subseteq Q \times \Gamma \times Q \times \Gamma \times \{L, R, H\}$. L, R and H correspond to the head moving to the left, to the right and staying at the same cell, respectively. We call the accepting and rejecting states $F_{acc} \cup F_{rej}$ final states. In our model of alternation, $\mapsto$ has a branching degree of 2. It is in an existential state in even-numbered steps and in a universal state in odd-numbered steps. We use $(q, a) \mapsto ((q_l, b_l, \Delta_l), (q_r, b_r, \Delta_r))$ when M is in state $q \in Q_\forall \cup Q_\exists$ reading symbol a to indicate that it branches to the left with (q_l, b_l, Δ_l) and to the right with (q_r, b_r, Δ_r). These directions are not related to the movement of the head which is determined by Δ_l or Δ_r. We call q_l the $\swarrow$*-child* of q and q_r the $\searrow$*-child* of q. Additionally, we assume that once M reaches a final state, it loops there forever and that it always reaches a final state.

The computation of M on an input word w can be represented as a graph, whose nodes are configurations of M. With each node in the graph we associate an acceptance value of 1 or 0 as follows. Configurations that are in an accepting state have the acceptance value 1, configurations that are in a rejecting state have acceptance value 0. The acceptance value of a configuration in a universal state is the minimum value of its two children and the value of an configuration in an existential state is the maximum value of its children. A TM accepts its input if the initial configuration has an acceptance value of 1 and rejects its input if the initial configuration has an acceptance value of 0.

Given a particular TM M and a word w, we construct an ABox $\mathcal{A}$ consisting of two parts $\mathcal{A}_{M,w}$ and $\mathcal{B}_M$. Let $M = (\Gamma, Q_\forall, Q_\exists, \mapsto, q_0, F_{acc}, F_{rej})$ be the given TM, and let $s(n)$ be the space bound of M on the word w as input. We use concept names Reject and Accept and role names $r_{q,a,d,i}$, for all $q \in Q$, $a \in \Gamma$, $d \in \{\swarrow, \searrow\}$ and $1 \leq i \leq s(n)$ in both parts of the ABox.

We start with $\mathcal{B}_M$. The individuals in $\mathcal{B}_M$ that correspond to universal and existential states are $(\forall, 0, 0, 0)$, $(\forall, 0, 1, 0)$, $(\forall, 1, 0, 0)$, $(\forall, 1, 1, 1)$. $(\exists, 0, 0, 0)$, $(\exists, 0, 1, 1)$, $(\exists, 1, 0, 1)$, $(\exists, 1, 1, 1)$. The intuition is that an internal element $(*, l, r, v)$ corresponds to a configuration of a TM with the following properties: its $\swarrow$-child has acceptance value l, its $\searrow$-child has acceptance value r and its own acceptance value therefore is v. Furthermore, there are two individuals that represent an acceptance or a rejection state of the TM, they are called 1 and 0, which corresponds to their acceptance value. The only concept assertions in $\mathcal{B}_M$ are $\mathsf{Reject}(0)$ and $\mathsf{Accept}(1)$.

We add the following role assertions to $\mathcal{B}_M$ in order to represent the transitions between configurations. For left branches, we have $r_{q,a,\swarrow,i}(e, e') \in \mathcal{B}_M$ for all $q \in Q, a \in \Gamma, 1 \leq i \leq n$ and for $e = (*, l, r, v)$ if either $e' = (*', l', r', v')$, $*$ is the opposite type of state as $*'$, and $l = v'$ or alternatively $e' = l$. For right branches, we have $r_{q,a,\searrow,i}(e, e') \in \mathcal{B}_M$ for all $q \in Q, a \in \Gamma, 1 \leq i \leq n$, and for $e = (*, l, r, v)$ if either $e' = (*', l', r', v')$, $*$ is the opposite type of state as $*'$ and $r = v'$ or alternatively $e' = r$. We additionally, have $r(0, 0), r(1, 1) \in \mathcal{B}_M$ for all role names r.

The second part $\mathcal{A}_{M,w}$ has a number of concept and role assertions for each of the $s(n)$ tape cells. For every cell i we have individuals of the form (q, a, i) and (a, i), for all $q \in Q, a \in \Gamma$. An individual (a, i) represents that the content of cell i is a and the head of the TM is not on cell i. An individual (q, a, i) represents that the content of cell i is a, that the head of the TM is on cell i and that the TM is in state q. In the following description the cases $i = 1$ and $i = s(n)$ are not treated in a special way, since we can assume that that M does not move its head beyond cell 1 or $s(n)$.

Informally, a role assertion $r_{q,a,d,i}(e, e')$ is included in $\mathcal{A}_{M,w}$ if in state q with the head at cell i and reading tape symbol a, M can change the tape cell represented by e to e' by taking a d-branch. Note that e and e' may be identical, meaning that the TM transition does not affect the tape cell.

More formally, each transition $(q, a) \mapsto ((q_l, b_l, \Delta_l), (q_r, b_r, \Delta_r))$ of M results in the following role assertions for each tape cell i in $\mathcal{A}_{M,w}$:

1. Role assertions that correspond to the head moving from cell i to cell $i-1$ or $i+1$. For each individual $(q, a, i) \in \mathsf{ind}(\mathcal{A}_{M,w})$, we include:
$$r_{q,a,\swarrow,i}((q, a, i), (b_l, i)), \quad r_{q,a,\searrow,i}((q, a, i), (b_r, i))$$
2. Role assertions that correspond to the head moving from cell $i-1$ or $i+1$ to cell i. For each individual $(b, i) \in \mathsf{ind}(\mathcal{A}_{M,w})$, we include:
$$\begin{aligned} &r_{q,a,\swarrow,i-1}((b, i), (q_l, b, i)), \text{ if } \Delta_l = R \\ &r_{q,a,\searrow,i-1}((b, i), (q_r, b, i)), \text{ if } \Delta_r = R \\ &r_{q,a,\swarrow,i+1}((b, i), (q_l, b, i)), \text{ if } \Delta_l = L \\ &r_{q,a,\searrow,i+1}((b, i), (q_r, b, i)), \text{ if } \Delta_r = L \end{aligned}$$
3. Role assertions that correspond to the transition not modifying the cell. For each $(b, i) \in \mathsf{ind}(\mathcal{A}_{M,w})$, we include:
$$\begin{aligned} &r_{q,a,\swarrow,j}((b, i), (b, i)), \text{ for all } j \notin \{i-1, i\} \text{ if } \Delta_l = R \\ &r_{q,a,\searrow,j}((b, i), (b, i)), \text{ for all } j \notin \{i-1, i\} \text{ if } \Delta_r = R \\ &r_{q,a,\swarrow,j}((b, i), (b, i)), \text{ for all } j \notin \{i, i+1\} \text{ if } \Delta_l = L \\ &r_{q,a,\searrow,j}((b, i), (b, i)), \text{ for all } j \notin \{i, i+1\} \text{ if } \Delta_r = L \\ &r_{q,a,\swarrow,j}((b, i), (b, i)), \text{ for all } j \neq i \text{ if } \Delta_l = H \\ &r_{q,a,\searrow,j}((b, i), (b, i)), \text{ for all } j \neq i \text{ if } \Delta_r = H \end{aligned}$$
4. Role assertions that modify the current cell without moving the head. For each $(q, a, i) \in \mathsf{ind}(\mathcal{A}_{M,w})$, we include:
$$\begin{aligned} &r_{q,a,\swarrow,i}((q, a, i), (q_l, b_l, i)), \text{ if } \Delta_l = H \\ &r_{q,a,\searrow,i}((q, a, i), (q_r, b_r, i)), \text{ if } \Delta_r = H \end{aligned}$$

Additionally, we need a number of role assertions for the final transitions $(q, a) \mapsto (q, a, H)$, $q \in F_{acc} \cup F_{rej}$ of M. For each such transition and each possible cell i, we include the assertions
$$r_{q,a,\swarrow,i}((q, a, i), (q, a, i)), \quad r_{q,a,\searrow,i}((q, a, i), (q, a, i)).$$

It remains to add concept assertions that mark accepting and rejecting states. We include, for all $a \in \Gamma, 1 \leq i \leq n$:
$$\begin{aligned} &\mathsf{Reject}((q, a, i)) \in \mathcal{A}_M, \text{ for all } q \in F_{rej}, \\ &\mathsf{Reject}((a, i)) \in \mathcal{A}_M, \\ &\mathsf{Accept}((q, a, i)) \in \mathcal{A}_M, \text{ for all } q \in F_{acc}, \\ &\mathsf{Accept}((a, i)) \in \mathcal{A}_M. \end{aligned}$$

This finishes the construction of $\mathcal{A}_{M,w}$. It remains to give P and N. Intuitively, P is a set of individuals representing the initial configuration of M on input w. Moreover, N consists of a single individual b. Formally, let w_k denote the k-th symbol of w and β be the symbol for an empty tape cell. We then define:

$$P = \{(q_0, w_1, 1)\} \cup \{(w_k, k) \mid 2 \leq k \leq n\} \cup \{(\beta, k) \mid n+1 \leq k \leq s(n)\}$$
$$b = (\forall, 1, 1, 1).$$

Lemma 6 *$(\mathcal{K}, P, \{b\})$ is $\mathcal{EL}$ concept separable for $\mathcal{K} = (\emptyset, \mathcal{A}_M \cup \mathcal{B}_M)$ iff M does not accept w.*

Proof. By Theorem 11 and the construction of $\mathcal{A}_{M,w}$ and $\mathcal{B}_M$, it suffices to show that

$$M \text{ accepts } w \quad \text{iff} \quad \prod_{a \in P} (\mathcal{I}_{\mathcal{A}_{M,w}}, a) \preceq_{\mathcal{EL}} \mathcal{I}_{\mathcal{B}_M}, b, \qquad (*)$$

where, for any ABox $\mathcal{A}$, $\mathcal{I}_\mathcal{A}$ is $\mathcal{A}$ viewed as interpretation. Before we give the formal proof, we provide some insight in the construction of $\mathcal{A}_{M,w}$. For this purpose, let us denote with $\mathcal{I}$ the product $\prod_{a \in P} \mathcal{I}_{\mathcal{A}_{M,w}}$, where we assume a fixed order on the elements in P. Moreover, for a configuration α of M, let x_α denote the element of $\Delta^\mathcal{I}$ corresponding to this configuration (in the natural way). We claim that elements of $\mathcal{I}$ that are reachable from P (read as an element of $\Delta^\mathcal{I}$ based on the order) correspond precisely to the configurations in the computation of M on input w. Indeed, $P = x_{\alpha_0}$ for the initial configuration α_0 of M on input w. Moreover, we observe that, for any (non-final) configuration α of M, x_α has precisely two successors $x_{\alpha_l}, x_{\alpha_r}$ in $\mathcal{I}$ where α_l, α_r are the successor configurations of α according to M's transition relation Δ. To see this, let q be the state, i be the head position and a be the current tape symbol in α. By construction of $\mathcal{A}_{M,w}$, the element $(q, a, i) \in x_\alpha$ has precisely two successors, one for $r_{q,a,\swarrow,i}$ and one for $r_{q,a,\searrow,i}$, and we have $(x_\alpha, x') \in r^\mathcal{I}_{q,a,\swarrow,i}$ if and only if $x' = x_{\alpha_l}$ is the $\swarrow$-child of x_α and $(x_\alpha, x') \in r^\mathcal{I}_{q,a,\searrow,i}$ if and only if $x' = x_{\alpha_r}$ is the $\searrow$-child of x_α.

We proceed with the proof of $(*)$.

($\Leftarrow$) Suppose S is a simulation between $\prod_{a \in P}(\mathcal{I}_{\mathcal{A}_{M,w}}, a)$ and $\mathcal{I}_{\mathcal{B}_M}, b$. The goal is to show that the initial configuration α_0 of M on input w is accepting. For proving this, we associate with every element $d \in \mathsf{ind}(\mathcal{B}_M)$ a value v_d by taking $v_d = v$ if d is of the shape $(*, l, r, v)$ and $v_d = d$ if $d \in \{0, 1\}$. Now, the desired statement is a consequence of the following claim.

Claim. For every configuration α reachable from α_0, we have that, if $(x_\alpha, d) \in S$, then v_d is the acceptance value of α.

Proof of the Claim. We prove this by induction on the length of the longest path from α to a final configuration. First, consider the case when α is a final configuration. By construction of $\mathcal{A}_{M,w}$, we have $x_\alpha \in \mathsf{Accept}^\mathcal{I}$ if α is accepting and $x_\alpha \in \mathsf{Reject}^\mathcal{I}$ if α is rejecting. By definition of $\mathcal{B}_M$ and $(x_\alpha, d) \in S$, we know $d = 1$ or $d = 0$, respectively. Consider now the case that α is not a final configuration and $(x_\alpha, d) \in S$. By definition of the role assertions in $\mathcal{B}_M$, we have that d is a universal (resp., existential) element $(\forall, *, *, *)$ (resp., $(\exists, *, *, *)$) if α is a universal (resp., existential) configuration. By what was said above, x_α has precisely two successors x_{α_l} and x_{α_r} in $\mathcal{I}$. By the simulation condition [Forth], we know that $(x_{\alpha_l}, d_l) \in S$ and $(x_{\alpha_r}, d_r) \in S$ for some elements d_l, d_r. By induction, we know that v_{d_l} and v_{d_r} are the acceptance values of α_l and α_r. By definition of $\mathcal{B}_M$, the acceptance value of α is v_d. This finishes the proof of the Claim.

($\Rightarrow$) Suppose M accepts the word w. Define a relation S from elements x_α of $\mathcal{I}$ to individuals of $\mathcal{B}_M$. Let $(x_\alpha, v) \in S$ for configurations α that are final with acceptance value v and $(x_\alpha, (*, l, r, v)) \in S$ for configurations α if and only if $*$ is the type of α, l is the acceptance value of the $\swarrow$-child of α, r is the acceptance value of the $\searrow$-child of α, v is the acceptance value of α. We prove that S is a simulation between $\mathcal{I}, P$ and $\mathcal{I}_{\mathcal{B}_M}, b$. For condition [AtomR], we only have to consider the final elements that are in the extensions of Accept or Reject since there are no other concept names. S fulfills [AtomR] since it relates final elements to final elements in $\mathcal{B}_M$. For [Forth], consider an element x_α with $(x_\alpha, d) \in S$ for some non-final configuration α; the argument for final configurations is similar. By the remark above, we have to show that $(x_{\alpha_l}, d_l) \in S$ and $(x_{\alpha_r}, d_r) \in S$ for two elements d_l, d_r and the possible successor configurations α_l and α_r of α. But this is clear from the definition of S. ❑

ExpTime-completeness of $(\mathcal{L}_T, \mathcal{EL})$ concept separability follows from the given ExpTime-algorithm and the proof of ExpTime-hardness, $\mathcal{L}_T \in \{\mathcal{EL}, \mathcal{ELI}\}$. For concept definability, note first that the above reduction remains valid if we add the assertion $X(a)$ for every $a \in P \cup \{b\}$ to the ABox, for some fresh concept name X. Let $\mathcal{K}'$ be the updated knowledge base. It can be verified that P is definable in $\mathcal{K}'$ by an $\mathcal{EL}$ concept iff P and $\{b\}$ can be separated by an $\mathcal{EL}$ concept in $\mathcal{K}$. Thus we get the same hardness result for concept definability.

Theorem 13 *$\mathcal{ELI}$ concept separability and definability are undecidable.*

The proof is by reduction of the rectangle tiling problem. An instance of the *rectangle tiling problem* is a tuple (T, H, V, t_I, t_F) where T is a finite set of *tile types*, $H, V \subseteq T \times T$ are the *horizontal and vertical compatibility relations*, and $t_I, t_F \in T$ are the *initial and final tile*. A *solution* consists of a tiling τ of some $n \times m$-grid, $n, m \geq 1$, that is, a function $\tau : \{1, \ldots, n\} \times \{1, \ldots, m\} \to T$ such that the following conditions are satisfied:

1. $\tau(1, 1) = t_I$ and $\tau(n, m) = t_F$;
2. $(\tau(i, j), \tau(i+1, j)) \in H$ for $1 \leq i < n$ and $1 \leq j \leq m$;
3. $(\tau(i, j), \tau(i, j+1)) \in V$ for $1 \leq i \leq n$ and $1 \leq j < m$.

We assume that T is partitioned into $T_0 \uplus T_1 \uplus T_2$ and that the following conditions are satisfied:

C1 if $(t, t') \in H$ and $t \in T_i$, $i \in \{0, 1, 2\}$, then $t' \in T_{i+1 \bmod 3}$;

C2 if $(t, t') \in V$ and $t \in T_i$, $i \in \{0, 1, 2\}$, then $t' \in T_i$;

C3 $t_I \in T_0$ and $t_F \in T_2$.

C4 t_F can only be used in the upper right corner, that is neither H nor V contains a pair of the form (t_F, t);

C5 there is a unique tile t'_F that must be placed to the left of t_F and cannot be used anywhere else, that is, $(t'_F, t_F) \in H$, $(t'_F, t) \in H$ implies $t = t_F$, and $(t, t_F) \in H$ implies $t = t'_F$.

It is easy to show that this version of the tiling problem is undecidable by reduction of the halting problem for Turing machines. To avoid dealing with special cases, we also assume that if there is a tiling of some $n \times m$-grid, then there is a tiling of an $n \times m$-grid with $m > 2$. Note that, due to the assumed conditions, all tiles on the left-most column must be from T_0 and all tiles on the right-most column must be from T_2. Moreover, n must be divisible by 3.

Let $\mathcal{P} = (T, H, V, t_I, t_F)$ be an instance of the rectangle tiling problem. We aim to construct an $\mathcal{ELI}$ learning instance $(\mathcal{K}, P, N)$, $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, such that $\mathcal{P}$ has a solution if and only $\Pi_{a \in P}(\mathcal{U}_\mathcal{K}, a) \not\preceq_{\mathcal{ELI}} \mathcal{U}_\mathcal{K}, b$. By Theorem 11, the latter is the case if and only if $(\mathcal{K}, P, N)$ has a solution.

Set $\mathcal{A} = \{P_1(a_1), P_2(a_2), N_1(b)\}$, $P = \{a_1, a_2\}$, and $N = \{b\}$. The concept names P_1, P_2, N_1 trigger the construction of different trees below a_1, a_2, b in $\mathcal{U}_\mathcal{K}$, via the TBox $\mathcal{T}$ that is at the heart of the construction. We define it next, but first informally explain the symbols used in $\mathcal{T}$:

- a single reflexive and symmetric role name S, represented via the role composition $r^-; r$; that is, we use $\exists S.C$ as an abbreviation for $\exists r^-.\exists r.C$;
- for each tile type $t \in T$, three concept names B^0_t, B^1_t, B^2_t; additionally, concept names B^0_d, B^1_d, B^2_d where $d \notin T$ is a dummy tile;
- a concept name E that marks the last node of the first row in a row by row traversal of the grid, from bottom to top;
- a concept name N that marks intermediate nodes inserted between any two rows in the traversal;
- auxiliary concept names I to mark the bottom-most row F to mark the top-most row, X to mark all intermediate rows, and a further auxiliary concept name G.

We start with the tree rooted at a_1, writing $i \oplus k$ as an abbreviation for $i + k$ modulo 3:

1. $P_1 \sqsubseteq \bigsqcap_{(t_I,t) \in V} \exists S.(N \sqcap \exists S.(B^0_{t_I} \sqcap B^1_t \sqcap I))$
2. for all $t_1, t_2 \in T_j \setminus \{t_F\}$, $j \in \{0, 2\}$:
$$B^0_{t_1} \sqcap B^1_{t_2} \sqcap I \sqsubseteq \bigsqcap_{(t_1,t_3) \in H, (t_3,t_4) \in V} \exists S.(B^0_{t_3} \sqcap B^1_{t_4} \sqcap I)$$
3. for all $t_1, t_2 \in T_1$:
$$B^0_{t_1} \sqcap B^1_{t_2} \sqcap I \sqsubseteq \bigsqcap_{(t_1,t_3) \in H, (t_3,t_4) \in V} \exists S.(B^0_{t_3} \sqcap B^1_{t_4} \sqcap I) \sqcap \bigsqcap_{(t_1,t_3) \in H, (t_3,t_4) \in V} \exists S.(B^0_{t_3} \sqcap B^1_{t_4} \sqcap E)$$
4. for all $t_1, t_2 \in T_2 \setminus \{t_F\}$:
$$B^0_{t_1} \sqcap B^1_{t_2} \sqcap E \sqsubseteq \bigsqcap_{t_3 \in T_0, (t_3,t_4) \in V} \exists S.(N \sqcap \exists S.(B^1_{t_3} \sqcap B^2_{t_4} \sqcap X))$$
5. for all $i \in \{0, 1, 2\}$ and $t_1, t_2 \in T_j$, $j \in \{0, 1\}$:
$$B^i_{t_1} \sqcap B^{i\oplus 1}_{t_2} \sqcap X \sqsubseteq \bigsqcap_{(t_1,t_3) \in H, (t_3,t_4) \in V} \exists S.(B^i_{t_3} \sqcap B^{i\oplus 1}_{t_4} \sqcap X)$$
6. for all $i \in \{0, 1, 2\}$ and $t_1, t_2 \in T_2 \setminus \{t_F\}$:
$$B^i_{t_1} \sqcap B^{i\oplus 1}_{t_2} \sqcap X \sqsubseteq \bigsqcap_{(t_1,t_3) \in H, (t_3,t_4) \in V} \exists S.(B^i_{t_3} \sqcap B^{i\oplus 1}_{t_4} \sqcap X) \sqcap \bigsqcap_{t_3 \in T_0, (t_3,t_4) \in V} \exists S.(N \sqcap \exists S.(B^{i\oplus 1}_{t_3} \sqcap B^{i\oplus 2}_{t_4} \sqcap X))$$
7. for all $i \in \{0, 1, 2\}$ and $t \in T_2$:
$$B^i_t \sqcap B^{i\oplus 1}_{t_F} \sqcap X \sqsubseteq \bigsqcap_{t' \in T_0} \exists S.(N \sqcap \exists S.(B^{i\oplus 1}_{t'} \sqcap F))$$
8. for all $i \in \{0, 1, 2\}$ and $t \in T \setminus \{t'_F, t_F\}$:
$$B^i_t \sqcap F \sqsubseteq \bigsqcap_{(t,t') \in H} \exists S.(B^i_{t'} \sqcap F)$$
9. for all $i \in \{0, 1, 2\}$:
$$B^i_{t'_F} \sqcap F \sqsubseteq B^i_d \sqcap \exists S.B^i_{t_F}$$

A *tiling word* is a word over the alphabet $T \cup \{N\}$. Let τ be the tiling of some $n \times m$-grid. The *row by row unfolding of* τ is the tiling word

$$\tau(1, 1) \cdots \tau(n, 1)N \cdots N\tau(1, m) \cdots \tau(n, m).$$

Note that we use the symbol N to separate the rows. The concept inclusions above generate a tree in which for every tiling τ of some $n \times m$-grid, we we find a path p that describes the row by row unfolding of τ. Here and in what follows, a *path* is a sequence of domain elements $p = d_0 \cdots d_n$ such that

$$(d_i, d_{i+1}) \in S^{\mathcal{U}_\mathcal{K}} := (r^-)^{\mathcal{U}_\mathcal{K}} \circ r^{\mathcal{U}_\mathcal{K}}$$

for all $i < n$. We now describe the path p in more detail, see also the left-hand side of Figure 2. Each element d on p is labeled with N or with two concept names B^i_t and $B^{i\oplus 1}_{t'}$ with $(t, t') \in V$ to indicate that the grid position represented by d carries tile t' and that the grid position directly below the position represented by d carries tile t. The first part of p (between the first two occurrences of N) uses concept names B^0_t and $B^1_{t'}$ and gives the tiling of rows 0 and 1. It's last element satisfies the concept name E. The next part of p uses concept names B^1_t and $B^2_{t'}$ and once again gives the tiling of row 1 and the tiling of row 2. The next part uses B^2_t and $B^0_{t'}$ to describe rows 2 and 3, and so on. The horizontal tiling condition is satisfied on the entire path. After the last part of the path, which gives the tiling of the topmost row and repeats the tiling of the row below it, there is another segment that is labeled with F and in which each node is labeled only with a single concept name B^i_t, repeating the labeling of the topmost row. A notable difference is that the position before the last one is not only labeled with a concept name $B^i_{t'_F}$ (c.f. condition C5 above), but also with B^i_d. The last element on that segment is a leaf, that is, it has no $\mathcal{S}$-successors. Note that while the vertical matching condition is satisfied locally, at this point we have no guarantee that the repeated row tilings are actually identical or that all rows have the same length.

We next define the tree rooted at a_2:

10. $P_2 \sqsubseteq E \sqcap \bigsqcap_{t \in T_0} \exists S.(N \sqcap \exists S.(B^1_t \sqcap G))$

11. for all $i \in \{0,1,2\}$ and $t \in T_j \setminus \{t'_F\}$, $j \in \{0,1\}$:

$$B^i_t \sqcap G \sqsubseteq \bigsqcap_{t' \in T_{j+1}} \exists S.(B^i_{t'} \sqcap G)$$

12. for all $i \in \{0,1,2\}$ and $t \in T_2 \setminus \{t_F\}$:

$$\begin{aligned} B^i_t \sqcap G \quad \sqsubseteq \quad & \bigsqcap_{t' \in T_0} \exists S.(B^i_{t'} \sqcap G) \sqcap \\ & \bigsqcap_{t' \in T_0} \exists S.(N \sqcap \exists S.(B^{i\oplus 1}_{t'} \sqcap G)) \end{aligned}$$

13. for all $i \in \{0,1,2\}$:

$$B^i_{t'_F} \sqcap G \sqsubseteq \exists S.(B^i_{t_F} \sqcap \exists S.(N \sqcap \bigsqcap_{t \in (T \cup \{d\}) \setminus \{t_F, t'_F\}} B^i_t))$$

The generated tree contains every path p on which every element is labeled with N or with a single concept names B^i_t, subject to the following conditions. The path starts with an N. The part between the first two occurrences with N is labeled with concept names B^1_t, the part between the second two occurrences with concept names B^2_t, and so on. Moreover, Condition C1 must be respected. Nodes labeled with a concept $B^i_{t'_F}$ are special. They have a successor d that we call a *pre-cycle node* and that satisfies $B^i_{t_F}$ (and no other successor). The pre-cycle node d has as its (only) successor a leaf node d' that we call a *cycle node* and that satisfies B^i_t for all $t \in T$ except t_F and t'_F, and also N and B^i_d. See the middle part of Figure 2. The name 'cycle node' refers to the fact that, as explained in more detail later, it is crucial that (like any other node) this node can reach itself via S. This part of the TBox also labels a_2 with E. Informally, the E-labeling in the tree below a_2 is offset by -1 row compared to the E-labeling in the tree below a_1, and this plays a central role in the reduction.

For the tree rooted at b, define a set $\mathcal{C}$ of concept names as

$$\mathcal{C} = \{N\} \cup \{B^j_t \mid j \in \{0,1,2\} \text{ and } t \in T \cup \{d\}\}.$$

and include the following concept inclusion in $\mathcal{T}$:

14.

$$N_1 \quad \sqsubseteq \quad \exists S.(\bigsqcap_{A \in \mathcal{C}} A \sqcap \exists S.\exists S.(E \sqcap \bigsqcap_{A \in \mathcal{C}} A))$$

The tree generated below b is actually a path of length three, see the right-hand side of Figure 2. The first node b_1 on the path makes true all relevant concept names except E, the second one b_2 is a 'hole', meaning that it makes no concept names true, and the leaf b_3 makes true all relevant concept names including E.

We now describe the main idea behind the construction. Recall that we aim to show that $\mathcal{P}$ has a solution iff $\Pi_{i\in\{1,2\}}(\mathcal{U}_\mathcal{K}, a_i) \not\preceq_{\mathcal{ELI}} \mathcal{U}_\mathcal{K}, b$ where $\mathcal{U}_\mathcal{K}$ is the universal model of $\mathcal{K}$. Since we deal with simulations, we can as well replace $\Pi_{i\in\{1,2\}}(\mathcal{U}_\mathcal{K}, a_i)$ with its unraveling into a tree. In what follows, we will this generally assume $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$ to be tree-shaped. A crucial observation is that the subtree in $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$ rooted at (a_1, a_2) admits a simulation to the tree below b in $\mathcal{U}_\mathcal{K}$ if and only if the former does not contain an $\mathcal{S}$-path in which every node satisfies at least one concept name from $\mathcal{C}$ (we say that the path has no 'holes') and whose last node satisfies E. It thus suffices to argue that $\mathcal{P}$ has a solution if and only if there is such a path in $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$ that starts at an S-successor of (a_1, a_2).

If $\mathcal{P}$ has a solution, then this solution gives rise to a path in $\mathcal{U}_\mathcal{K}$ that starts at a successor of a_1, as described above, and proceeds all the way to the second node on the path labeled with some concept name of the form $B^i_{t_F}$, $i \in \{0,1,2\}$. We find a corresponding path in $\mathcal{U}_\mathcal{K}$ that starts at a successor of a_2 and carries B^i_t labels that are identical to the 'larger superscript' labeling of the a_1-path (see Figure 2). It only follows that path up to the first node labeled with a concept name of the form $B^i_{t_F}$ and then enters a cycle node. These two synchronous paths give rise to an initial piece of a path p in $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$ that starts at (a_1, a_2). We call this the 'downwards part' of p. We can extend the initial piece by following the a_1-path until the second node labeled with some $B^i_{t_F}$ while 'cycling' reflexively at the cycle node of the second path. Afterwards, we synchronously follow both paths upwards, giving rise to an 'upwards part' of p . In this way, p is extended to a hole free path whose end node satisfies E. Consequently, $\Pi_{i\in\{1,2\}}(\mathcal{U}_\mathcal{K}, a_i) \not\preceq_{\mathcal{ELI}} \mathcal{U}_\mathcal{K}, b$.

The other direction is more laborious. We carry out a careful case analysis to show that any hole free path in $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$ that starts at a successor of (a_1, a_2) and ends in an E-node must follow the pattern described above. To avoid holes in the upwards part of p, the labeling of the a_2-path must match up with the 'smaller superscript' labeling of the a_1-path. This ensures that the repeated labelings of each row in the a_1-path (c.f. the description of the tree in $\mathcal{U}_\mathcal{K}$ below a_1) actually coincide. Since the vertical matching condition is satisfied 'locally' on the a_1-path, it is thus also satisfied 'globally' in the tiling described by that path. Moreover, when an N-node d of the a_2-path was paired with some N-node of the a_1-path in the downwards part of p, then d is now paired with the subsequent N-node on the a_1-path. This ensures that all rows have the same length. All this guarantees that we can read off a solution to $\mathcal{P}$ from the a_1-path.

In order to establish another undecidability result later on, we show a slightly stronger statement than the actual correctness; for this, we need to introduce homomorphisms. As usual, a *homomorphism* from $\mathcal{I}$ to $\mathcal{J}$ is a map $h : \Delta^\mathcal{I} \to \Delta^\mathcal{J}$ such that $d \in A^\mathcal{I}$ implies $h(d) \in A^\mathcal{J}$, and $(d, e) \in r^\mathcal{I}$ im-

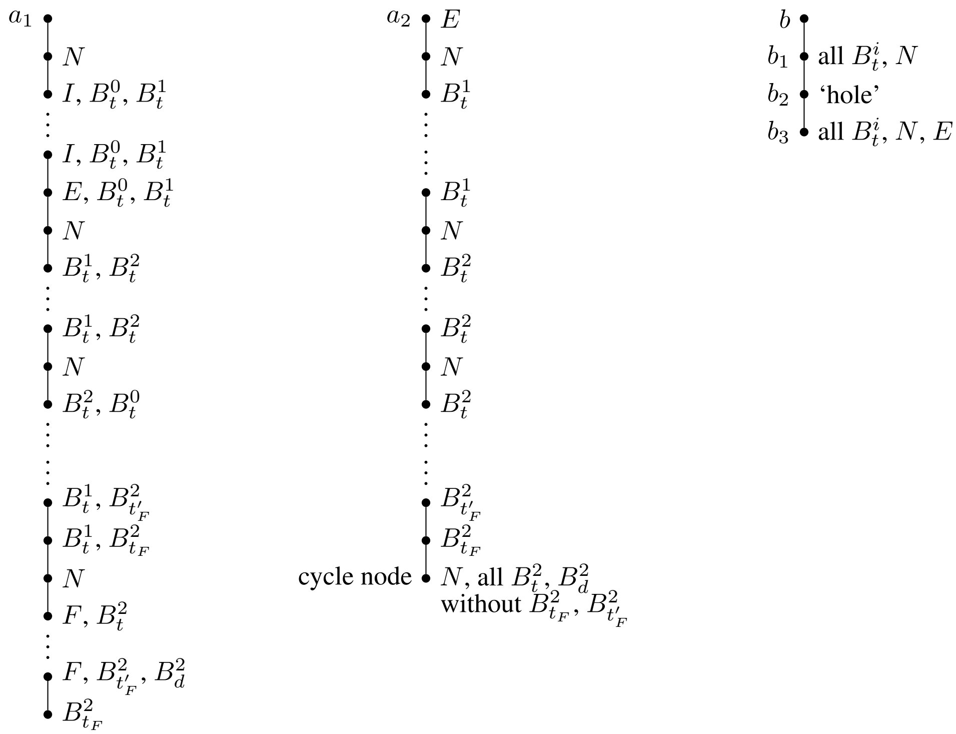

Figure 2: The image shows two paths in $\mathcal{U}_\mathcal{K}$. Every drawn edge is an S-edge.

plies $(h(d), h(e)) \in r^\mathcal{J}$, for all $d, e \in \Delta^\mathcal{I}$ and all concept names A and role names r. For pointed interpretations $\mathcal{I}, d$ and $\mathcal{J}, e$, we write $\mathcal{I}, d \to \mathcal{J}, e$ if there is a homomorphism h from $\mathcal{I}$ to $\mathcal{J}$ with $h(d) = e$.

Lemma 7 *The following are equivalent:*

(1) $\mathcal{P}$ has a solution;

(2) $\Pi_{i\in\{1,2\}}(\mathcal{U}_\mathcal{K}, a_i) \not\preceq_{\mathcal{ELI}} \mathcal{U}_\mathcal{K}, b$;

(3) $\Pi_{i\in\{1,2\}}(\mathcal{U}_\mathcal{K}, a_i) \not\to \mathcal{U}_\mathcal{K}, b$.

Proof. (2) $\Rightarrow$ (3) is trivial, so we prove only (3) $\Rightarrow$ (1) and (1) $\Rightarrow$ (2). Consider two elements $d, d' \in \Delta^{\mathcal{U}_\mathcal{K}}$ that are both part of the subtree in $\mathcal{U}_\mathcal{K}$ rooted at a_i, $i \in \{1, 2\}$, and such that $(d, d') \in S^{\mathcal{U}_\mathcal{K}}$. We call d' a *successor* of d if d' is further away from a_i than d, and the *predecessor* of d if d is further away from a_i than d'.

(3) $\Rightarrow$ (1). Assume that $\Pi_{i\in\{1,2\}}(\mathcal{U}_\mathcal{K}, a_i) \not\to \mathcal{U}_\mathcal{K}, b$. We claim that $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$ must contain a path $p = (d_0, e_0)\cdots(d_n, e_n)$ that starts at an S-successor of (a_1, a_2) and such that (d_n, e_n) satisfies E in $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$ and p has no holes, that is, for all $i < n$, (d_i, e_i) satisfies in $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$ at least one concept name from $\mathcal{C}$.

In order to prove the existence of such a path, let B be the set of all nodes different from (a_1, a_2) that can be reached by an S-path starting at (a_1, a_2). We define subsets B_1, B_2, B_3 of B as follows:

- B_1 contains all direct S-neighbors of (a_1, a_2) different from (a_1, a_2), and the set of all non-holes $(d, e) \neq (a_1, a_2)$ for which there is a S-path from (a_1, a_2) to (d, e) that does not have a hole;
- B_2 is the set of all holes that are S-neighbors of elements in B_1, but not in B_1;
- B_3 is $B \setminus (B_1 \cup B_2)$.

Note that B_1, B_2, B_3 is a partition of B. Moreover, B_2 separates B_1 and B_3 in the sense that every S-path from an element in B_1 to an element in B_3 must pass an element of B_2. Similarly, B_1 separates (a_1, a_2) and B_2.

Claim. There is an S-path in $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$ from (a_1, a_2) to an element satisfying E that visits only elements in $B \setminus B_2$.

Proof of Claim. Suppose this is not the case, that is, every S-path from (a_1, a_2) to an element satisfying E visits some element in B_2. We construct a homomorphism h showing $\Pi_{i\in\{1,2\}}(\mathcal{U}_\mathcal{K}, a_i) \to \mathcal{U}_\mathcal{K}, b$. It suffices to define h on elements in $B \cup \{(a_1, a_2)\}$; for unreachable elements, we can just take the projection to the first component (the second would also work). We start to define h by setting:

- $h(a_1, a_2) = b$;
- for all $i \in \{1, 2, 3\}$ and all $(d, e) \in B_i$, $h(d, e) = b_i$,

where b_1, b_2, b_3 are the elements from the path in $\mathcal{U}_\mathcal{K}$ below b, as shown in Figure 2.

Recall that the nodes in B can only make true concept names from $\mathcal{C}$, that b_1 makes true all these concept names except E, and that b_3 makes true all concept names from $\mathcal{C}$. When $(d, e) \in B_3$, it is thus clear that $(d, e) \in A^{\mathcal{U}_\mathcal{K}\times\mathcal{U}_\mathcal{K}}$ for some concept name A implies $h(d, e) \in A^{\mathcal{U}_\mathcal{K}}$. When $(d, e) \in B_2$, then it is a hole and thus this condition is vacuously true. When $(d, e) \in B_1$, then (d, e) does not satisfy E due the definition of B_1 and our assumption that every path to an element satisfying E visits some element in B_2. Moreover, we have that

$(*)$ $((d,e),(d',e')) \in S^{\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}}$ implies $(h(d,e),h(d',e')) \in S^{\mathcal{U}_\mathcal{K}}$.

It remains to define h on the intermediate elements of S-paths, that is, r^--successors of elements reachable via an S-path. Let (d,e) be such an element. Since (d,e) is an intermediate element, it has at precisely four r-neighbors $(d_1,e_1),\ldots,(d_4,e_4)$. Note that h is already defined for all of them, and we have $((d_i,e_i),(d_j,e_j)) \in S^{\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}}$, for all $i,j \in \{1,\ldots,4\}$. By $(*)$, we have $(h(d_i,e_i),h(d_j,e_j)) \in S^{\mathcal{U}_\mathcal{K}}$, for all $i,j \in \{1,\ldots,4\}$. Thus, one of the following is the case:

- $\{h(d_1,e_1),\ldots,h(d_4,e_4)\} \subseteq \{b,b_1\}$;
- $\{h(d_1,e_1),\ldots,h(d_4,e_4)\} \subseteq \{b_1,b_2\}$;
- $\{h(d_1,e_1),\ldots,h(d_4,e_4)\} \subseteq \{b_2,b_3\}$.

Depending on which case applies, we complete the definition of h by setting $h(d,e)$ to the intermediate node between b and b_1, b_1 and b_2, and b_2 and b_3, respectively. This finishes the proof of the Claim.

Let $(d_0,e_0),\ldots,(d_n,e_n)$ be some S-path from (a_1,a_2) to an element satisfying E which does not visit any element from B_2. Moreover, let $i \in \{1,\ldots,n\}$ be the largest index such that (d_i,e_i) is an S-neighbor of (a_1,a_2). We claim that (d_i,e_i) is an $\uparrow\uparrow$-successor of (a_1,a_2). Indeed, if this is not the case, then $d_i = a_1$ or $e_i = a_2$. In both cases we derive that (d_{i+1},e_{i+1}) is not an S-neighbor of (a_1,a_2) (due to the choice of i), and thus a hole (due to the construction of $\mathcal{K}$), a contradiction. Since the path does not visit any element from B_2 and B_2 separates B_1 from B_3, it does also not visit any element from B_3. By definition of B_1 and choice of i, there are no holes on the path $p = (d_i,e_i),\ldots,(d_n,e_n)$. Hence, this path satisfies all the required conditions. We might clearly assume that p is a simple path, that is, $i \neq j$ implies $(d_i,e_i) \neq (d_j,e_j)$.

By construction of the trees below a_1 and a_2, every node (d_i,e_i) on p satisfies a unique concept name from $\mathcal{C}$. We will later show how to read off from this unique labeling of p a tiling word that is a row by row unfoldung of a tiling of some finite grid.

It is important to carefully analyse how p lies within $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$. We say that (d_{i+1},e_{i+1}) is a *$\downarrow\downarrow$-successor* of (d_i,e_i) if d_{i+1} is a successor of d_i and e_{i+1} is a successor of e_i, and likewise for *$\downarrow\uparrow$-successors*, *$\downarrow\circlearrowleft$-successors*, and so on, with $\uparrow$ indicating the transition to a predecessor in the respective component and $\circlearrowleft$ indicating identity of the component (recall that S is reflexive). We observe that several kinds of successors cannot occur on p:

(i) $\uparrow\downarrow$-successors (d_{i+1},e_{i+1}) with e_{i+1} not a cycle node.

Towards a proof by contradiction, assume that (d_{i+1},e_{i+1}) is an $\uparrow\downarrow$-successor with e_{i+1} not a cycle node. We know that (d_i,e_i) satisfies a concept name from $\mathcal{C}$. Since e_{i+1} is not a cycle node, this concept name is not of the form B^j_d. First assume that it has the form B^j_t, $t \in T$. Let $t \in T_w$. Both d_i and e_i also satisfy B^j_t. By construction of the trees below a_1 and a_2 and since (d_{i+1},e_{i+1}) is an $\uparrow\downarrow$-successor of (d_i,e_i),

- d_{i+1} satisfies N or some $B^\ell_{t'}$ with $t' \in T_{w\ominus 1}$, but no other tile from $\mathcal{C}$ except possibly B^ℓ_d;
- e_{i+1} satisfies N or some $B^{\ell'}_{t'}$ with $t' \in T_{w\oplus 1}$ and no other concept name from $\mathcal{C}$;
- d_{i+1} and e_{i+1} do not both satisfy N (as there are at least three non-N-nodes between any two consecutive N-nodes in $\mathcal{U}_\mathcal{K}$).

As a consequence, (d_{i+1},e_{i+1}) is a hole. Contradiction.

Now assume that (d_i,e_i) satisfies N. Then so do d_i and e_i. By construction of the trees below a_1 and a_2 and since e_{i+1} is not a cycle node, d_{i+1} satisfies some $B^\ell_{t'}$ with $t' \in T_2$ and no other concept name from $\mathcal{C}$, and e_{i+1} satisfies some $B^{\ell'}_{t'}$ with $t' \in T_0$ and no other concept name from $\mathcal{C}$. As a consequence, (d_{i+1},e_{i+1}) is a hole. Contradiction.

(ii) $\circlearrowleft *$-successors.

First for the $\circlearrowleft\downarrow$ case. Towards a proof by contradiction, assume that (d_{i+1},e_{i+1}) is a $\circlearrowleft\downarrow$-successor of (d_i,e_i). First assume that e_{i+1} is not a cycle node. Then e_i and e_{i+1} satisfy unique but different concept names from $\mathcal{C}$ that are not of the form B^ℓ_d. The first such concept name is also satisfied by (d_i,e_i), thus by d_i, and the second concept name is also satisfied by (d_{i+1},e_{i+1}), thus by $d_{i+1} = d_i$. But by construction of the tree below a_1, d_i does not satisfy two such different concept names. Contradiction.

Now assume that e_{i+1} is a cycle node. By construction of the subtree below a_2, e_i satisfies a concept name of the form $B^j_{t_F}$, and thus so does d_i. Moreover, (d_{i+1},e_{i+1}) satisfies a concept name from $\mathcal{C}$ also satisfied by e_{i+1} and since e_{i+1} is a cycle node, this concept name cannot be of the form $B^j_{t_F}$ or $B^j_{t'_F}$. Thus $d_{i+1} = d_i$ satisfies both concept names, which is not possible by the construction of the tree below a_1.

The case $\circlearrowleft\uparrow$ is similar. Furthermore, there are no $\circlearrowleft\circlearrowleft$-successors since p is simple.

(iii) $\uparrow\circlearrowleft$-successors (d_{i+1},e_{i+1}) with (d_i,e_i) not an $\uparrow\downarrow$-successor.

Towards a proof by contradiction, assume that (d_{i+1},e_{i+1}) is a $\uparrow\circlearrowleft$-successor of (d_i,e_i) with (d_i,e_i) not an $\uparrow\downarrow$-successor, and that it is the first such node on p. If e_i is not a cycle node, then we can argue as in (ii) above. Thus assume that e_i is a cycle node. Consider the successor type of (d_i,e_i). Since e_i is a cycle node, it is a leaf in $\mathcal{U}_\mathcal{K}$. Together with Point (ii) above and since (d_i,e_i) is the first $\uparrow\circlearrowleft$-successor, (d_i,e_i) can thus only be a $\downarrow\circlearrowleft$-successor or a $\downarrow\downarrow$-successor. The former implies $(d_{i-1},e_{i-1}) = (d_{i+1},e_{i+1})$ in contradiction to p being simple. In the latter case, $d_{i-1} = d_{i+1}$ and e_{i-1} is the predecessor of e_i in $\mathcal{U}_\mathcal{K}$. By construction of the tree below a_1, the latter implies that e_{i-1} is labeled with some concept name $B^j_{t_F}$. Since (d_{i-1},e_{i-1}) is not a hole in p, d_{i-1} is also labeled with $B^j_{t_F}$ and thus so is

$d_{i+1} = d_{i-1}$. However, by construction of the subtree below a_2, the cycle node e_{i+1} is not labeled with $B^j_{t_F}$ and thus (d_{i+1}, e_{i+1}) is a hole in p. Contradiction.

(iv) $\downarrow\uparrow$-successors (d_{i+1}, e_{i+1}) with e_i not a cycle node. Similar to the the proof of (i).

(v) $\downarrow\circlearrowleft$-successors (d_{i+1}, e_{i+1}) with e_i not a cycle node. Similar to the proof of (ii).

The remaining kinds of successors are $\downarrow\downarrow$, $\uparrow\uparrow$, $\uparrow\downarrow$, $\uparrow\circlearrowleft$, $\downarrow\uparrow$, $\downarrow\circlearrowleft$, and Points (i) to (v) impose strong restrictions on the latter four types of successors. We aim to show that p must follow the pattern

$$\downarrow\downarrow^+\downarrow\circlearrowleft^+\downarrow\uparrow\ \uparrow\uparrow^+$$

with the $\downarrow\circlearrowleft^+$-subpath having a cycle node in the second component and the $\downarrow\uparrow\ \uparrow\uparrow$-subpath escaping from that cycle node and its pre-cycle node back to a regular node.[2]

We first note that p must start with a $\downarrow\downarrow^+$-prefix. In fact, (d_1, e_1) cannot be a $\downarrow\circlearrowleft$- or $\downarrow\uparrow$-successor of (d_0, e_0) by Points (iv) and (v) and since (d_0, e_0) cannot be a cycle node. It cannot be an $\uparrow\downarrow$-successor by Point (i) and since we cannot reach a cycle node from e_0 in one step. It cannot be an $\uparrow\circlearrowleft$-successor by Point (iii). And it cannot be an $\uparrow\uparrow$-successor as then p would contain the hole (a_1, a_2). By construction of the trees below a_1 and a_2, the $\downarrow\downarrow^+$-prefix of p cannot contain an element that satisfies E. Thus the $\downarrow\downarrow^+$-prefix must be followed by some other kind of successor (d_{i_0+1}, e_{i_0+1}). This cannot be an $\uparrow\uparrow$-successor since p is simple and not an $\uparrow\circlearrowleft$-successor by Point (iii). The remaining candidates are $\uparrow\downarrow$, $\downarrow\circlearrowleft$, and $\downarrow\uparrow$. We show that the first and last option are impossible.

Assume towards a proof by contradiction that the $\downarrow\downarrow^+$-prefix is followed by an $\uparrow\downarrow$-successor (d_{i_0+1}, e_{i_0+1}). By Point (i), e_{i_0+1} is a cycle node. By construction of the tree below a_2, e_{i_0} must be labeled with a concept name $B^j_{t_F}$. Thus, d_{i_0} is also labeled with $B^j_{t_F}$ and by construction of the subtree below a_1, $d_{i_0+1} = d_{i_0-1}$ is labeled with $B^j_{t'_F}$. In fact, d_{i_0-1} is the first element on the path $d_0, \dots, d_{i_0-1}$ in $\mathcal{U}_\mathcal{K}$ that is labeled with a concept name of the form $B^\ell_{t'_F}$: if some d_s with $s < i_0 - 1$ was the first element on the path $d_0, \dots, d_{i_0-1}$ labeled with $B^\ell_{t'_F}$, then e_{s+2} is a cycle node due to the construction of the tree below a_2 and since we travel $\downarrow\downarrow^2$ from (d_s, e_s) to (d_{s+2}, e_{s+2}); this contradicts the fact that e_{i_0} is reachable from e_s by traveling downwards. As d_{i_0-1} is the first element of its kind, it is not labeled with B^ℓ_d and in fact $B^\ell_{t'_F}$ is the only concept name from $\mathcal{C}$ satisfied by $d_{i_0-1} = d_{i_0+1}$. However, the cycle node e_{i_0+1} is not labeled with $B^j_{t'_F}$, thus (d_{i_0+1}, e_{i_0+1}) is a hole in p. Contradiction.

Now assume that the $\downarrow\downarrow^+$-prefix is followed by a $\downarrow\uparrow$-successor (d_{i_0+1}, e_{i_0+1}). Then, e_{i_0} is a cycle node by Point (iv) and thus e_{i_0+1} is a pre-cycle node and actually $e_{i_0+1} = e_{i_0-1}$. Thus (d_{i_0-1}, e_{i_0-1}) and (d_{i_0+1}, e_{i_0+1}) both satisfy a concept name $B^j_{t_F}$ and also d_{i_0-1} and d_{i_0+1} both satisfy $B^j_{t_F}$. This, however, is impossible by construction of the subtree below a_1 and in particular due to the partitioning of T into $T_0 \uplus T_1 \uplus T_2$.

We have thus shown that p starts with a $\downarrow\downarrow^+\downarrow\circlearrowleft^+$-prefix. By Point (v), the first $\downarrow\circlearrowleft$-successor (d_{i_0+1}, e_{i_0+1}) is such that e_{i_0} is a cycle node. By construction of the subtree below a_2, (d_{i_0-1}, e_{i_0-1}) satisfies a concept name $B^j_{t_F}$ and (d_{i_0-2}, e_{i_0-2}) satisfies $B^j_{t'_F}$. This will be used later.

Since e_{i_0} is a leaf in $\mathcal{U}_\mathcal{K}$, this prefix can only be followed by a successor of type $\downarrow\uparrow$, $\uparrow\uparrow$, and $\uparrow\circlearrowleft$, say (d_{i_1+1}, e_{i_1+1}). However, $\uparrow\circlearrowleft$ is impossible by Point (iii). Moreover, $\uparrow\uparrow$ is impossible too. Assume to the contrary that (d_{i_1+1}, e_{i_1+1}) is an $\uparrow\uparrow$-successor. We know that (d_{i_1-1}, e_{i_1-1}) satisfies a unique concept name from $\mathcal{C}$. Since $e_{i_1-1} = e_{i_1}$ is a leaf node, this concept name is not of the form $B^j_{t_F}$, and it is also satisfied by d_{i_1-1}. But e_{i_1+1} is a pre-cycle node and thus the only concept name it satisfies is of the form $B^j_{t_F}$; the same concept name must be satisfied by $d_{i_1+1} = d_{i_1-1}$. But no element in the subtree below a_1 satisfies two such concept names.

It follows that (d_{i_1+1}, e_{i_1+1}) is a $\downarrow\uparrow$-successor. It thus satisfies a concept name $B^j_{t_F}$. In fact, we have only moved downwards in the first component so far, and thus d_{i_1+1} is the second node on a path in $\mathcal{U}_\mathcal{K}$ that satisfies a concept name $B^j_{t_F}$, the first one being d_{i_0-1}. As a consequence and by construction of the tree below a_1, d_{i_1+1} is a leaf in $\mathcal{U}_\mathcal{K}$ and d_{i_1} satisfies both $B^j_{t'_F}$ and B^j_d. This will be used later.

We next analyze the type of successor that (d_{i_1+2}, e_{i_1+2}) is. Since d_{i_1+1} is a leaf and by Point (iii), the only options are $\uparrow\uparrow$ and $\uparrow\downarrow$. The latter, however is impossible since p is simple. We have shown that p starts with a $\downarrow\downarrow^+\downarrow\circlearrowleft^+\downarrow\uparrow\ \uparrow\uparrow$-prefix.

We can proceed to travel $\uparrow\uparrow$. We argue that we can never switch to any other kind of successor again. $\uparrow\downarrow$, $\downarrow\uparrow$, and $\downarrow\circlearrowleft$ are ruled out by Points (i), (iv), and (v) and since we can never reach a cycle node in the second component while traveling upwards. $\uparrow\circlearrowleft$ is ruled out by Point (iii). The only remaining candidate is $\downarrow\downarrow$. But we can never switch to $\downarrow\downarrow$ before reaching a_1 in the first component or a_2 in the second component because p is simple. The former cannot happen since a_1 satisfies neither E nor any concept name from $\mathcal{C}$ and p has no holes. The latter can (and in fact does) only happen at the final element of p since a_2 satisfies (E but) no concept name from $\mathcal{C}$ and thus seeing a_2 before the end means that p has a hole. We have thus shown that p indeed follows the pattern

$$\downarrow\downarrow^+\downarrow\circlearrowleft^+\downarrow\uparrow\ \uparrow\uparrow^+ .$$

Moreover the last element (d_n, e_n) of p must be such that $e_n = a_2$ because by construction of the tree below a_2 and what we have said about the structure of p, this is the only way for (d_n, e_n) to satisfy E.

To proceed, consider the prefix

$$p' = (d_0, e_0), \dots, (d_{i_0-1}, e_{i_0-1})$$

of p. As already pointed out, each node on p is associated with a unique concept name from $\mathcal{C}$. For the nodes on p', this concept name cannot be of the form B^j_d (recall that d is the dummy tile) since $d_0, \dots, d_{i_0-1}$ constitutes a path in $\mathcal{U}_\mathcal{K}$ that

[2] This also implies that neither $\uparrow\downarrow$- nor $\uparrow\circlearrowleft$-successors occur at all, but we are not yet in a position to show this directly.

travels purely downwards and sees only one concept name of the form $B^j_{t_F}$ at the very end. In fact, we have already argued that d_{i_0-1} satisfies a concept name $B^j_{t_F}$. No earlier node does so since by construction of the tree below a_2 we would otherwise have reached a cycle node in the second component earlier than at e_{i_0}.

We can thus read off from p' a unique tiling word $t_0 \cdots t_{i_0-1}$. By construction of the trees below a_1 and a_2, $t_0 = N$. Let $t_1 \cdots t_{n_1}$ be the longest prefix of $t_1 \cdots t_{i_0-1}$ that does not contain N. Since (d_1, e_1) is a $\downarrow\downarrow$-successor and again by construction of the trees below a_1 and a_2, this prefix is not empty. Moreover, each node d_i with $1 \leq i \leq n_1$ satisfies a concept name of the form B^0_t and a concept name of the form B^1_t. It is the latter concept that is also satisfied by (d_i, e_i) and thus defines the tiles $t_1 \cdots t_{n_1}$. We obtain another sequence of tiles $t^{(0)}_1 \cdots t^{(0)}_{n_1}$ from the B^0_t labeling. We aim to show that

$$t^{(0)}_1 \cdots t^{(0)}_{n_1} t_0 \cdots t_{i_0-1}$$

is a row by row unfolding of a tiling of some $n_1 \times m$-grid. By construction of the tree below a_1, the following is not hard to verify:

1. $t^{(0)}_1 = t_I$;
2. the horizontal matching condition is satisfied; more formally, whenever tt' is a subword of $t^{(0)}_0 \cdots t^{(0)}_{n_1} t_0 \cdots t_{i_0-1}$ and none of t and t' is N, then $(t, t') \in H$;
3. the first two rows in $t^{(0)}_0 \cdots t^{(0)}_{n_1} N t_0 \cdots t_{n-1}$ are of the same length n_1 and all vertically neighboring tiles on these two rows satisfy V (because the double labeling with B^0_t and $B^1_{t'}$ in the tree below a_1 respects V).

It remains to show that the vertical matching condition is satisfied beyond the first two rows and that all rows rather than only the first two have the intended length n_1.

We associate an *offset* with each (d_i, e_i) on p, defined as the difference $D_2 - D_1$ where D_1 is the distance of d_i from a_1 in $\mathcal{U}_\mathcal{K}$ and D_2 the distance of e_i from a_2 in $\mathcal{U}_\mathcal{K}$. Clearly, the offset of (d_0, e_0) is 0. By construction of the tree below a_1 and choice of n_1, d_{n_1} satisfies E and no other element among $d_0, \ldots, d_{i_0-1}$ does. Moreover, since we first travel only downwards and then only upwards in the first component and d_n satisfies E, we must have $d_n = d_{n_1}$. By construction of the tree below a_2, the only element among $e_0, \ldots, e_n$ satisfying E is $e_n = a_2$. Consequently, the offset of (d_n, e_n) is $n_1 + 1$.

Since the offset of (d_0, e_0) is 0 and until (d_{i_0}, e_{i_0}) we have only seen $\downarrow\downarrow$-successors, the offset of (d_{i_0}, e_{i_0}) must also be 0. Likewise, the offset of (d_n, e_n) being n_1+1 and the fact that from (d_{i_1+1}, e_{i_1+1}) on we have only seen $\uparrow\uparrow$-successors implies that the offset of (d_{i_1+1}, e_{i_1+1}) must also be $n_1 + 1$. Consequently and since the single $\downarrow\uparrow$-step adds an offset of 2, the $\downarrow\circlearrowleft^+$-subpath of p has length $n_1 - 1$. Clearly, the distance of d_{i_0-1} from a_1 is i_0. To reach (d_{i_1+1}, e_{i_1+1}) from (d_{i_0-1}, e_{i_0-1}), we make one $\downarrow\downarrow$-step, $n_1 - 1$ $\downarrow\circlearrowleft$-steps, and one $\downarrow\uparrow$-step. As a consequence, the distance of d_{i_1+1} from a_1 in $\mathcal{U}_\mathcal{K}$ is $i_0 + n_1 + 1$. Since from (d_{i_1+1}, e_{i_1+1}) we make only $\uparrow\uparrow$-steps and $e_{i_1+1} = e_{i_0-1}$, this implies the following crucial conditions:

(a) if (d_i, e_i) is a node in p with $i < i_0$, then (d_{i+n_1+1}, e_i) is also a node in p;

(b) if (d_i, e_i) is a node in p with $n_1 < i < i_0$, then $(d_{i-(n_1+1)}, e_i)$ is also a node in p.

This, in turn, implies that all rows are of the same length and that the vertical matching condition is satisfied, as follows.

We start with row length. Consider the tiling word $t^{(0)}_1 \cdots t^{(0)}_{n_1} t_0 \cdots t_{i_0-1}$. We already know by choice of n_1 that $t_0 = t_{n_1+1} = N$. We have to show that

1. $t_{\ell \cdot (n_1+1)} = N$ for $1 < \ell < \frac{i_0-1}{n_1+1}$ and
2. for no other t_i, $t_i = N$.

For Point 1, we concentrate on $t_{2(n_1+1)}$, the same argument can be applied inductively for $\ell > 2$. The argument is in fact easy based on (a). We know that $t_{n_1+1} = N$, thus the unique concept name from $\mathcal{C}$ satisfied by (d_{n_1+1}, e_{n_1+1}) is N. It follows from (a) that the unique concept name from $\mathcal{C}$ satisfied by $(d_{2(n_1+1)}, e_{n_1+1})$ is also N, and thus the same is true for $(d_{2(n_1+1)}, e_{2(n_1+1)})$. Consequently, $t_{2(n_1+1)} = N$. The proof of Point 2 is similar, using (b) instead of (a) and showing that if $t_i = N$ for some t_i not covered by Point 1, then $t_i = N$ for some $i \in \{1, \ldots, n_1\}$ which we know is not the case.

Now for the vertical matching condition. Take any $t_i \neq N$ from $t^{(0)}_1 \cdots t^{(0)}_{n_1} t_0 \cdots t_{i_0-1}$ that is neither on the bottommost nor on the topmost row. We know that (d_i, e_i) satisfies a unique concept name $B^j_{t_i}$ from $\mathcal{C}$. By (a), (d_{i+n_1+1}, e_i) is a node on p. It must clearly also satisfy $B^j_{t_i}$ and no other concept name from $\mathcal{C}$, and the same is true for d_{i+n_1+1}. By construction of the tree below a_1, d_{i+n_1+1} satisfies, apart from $B^j_{t_i}$, also a concept name $B^{j\oplus 1}_{t'}$ with $(t_i, t') \in V$. Using the construction of the subtree below a_1 and a_2 and the fact that the prefix p' of p has only $\downarrow\downarrow$-successors, it can be seen that $t_{i+n_1+1} = t'$, which is exactly what we had to show.

$(1) \Rightarrow (2)$. Assume that $\mathcal{P}$ has a solution, that is, there is a tiling τ of some $n \times m$-grid, $n, m \geq 1$. By our assumption on $\mathcal{P}$, we may assume that $m \geq 2$. Let w be a tiling word that is a row by row unfolding of τ, with an additional leading N symbol (that is, every row in w is prefixed by N). The length of w is $(n+1) \cdot m$. We start with showing that $\mathcal{U}_\mathcal{K}$ contains a path p_1 that starts at an S-successor of a_1 and whose labeling with the concept names from $\mathcal{C}$ gives rise to w.

The length of p_1 will be $k := (n+1) \cdot (m-1)$; we shall explain later why p_1 is short of one row. We number the columns of the grid from 0 to $n-1$ and the rows of the grid from 0 to $m-1$. For all positions $i \leq k$ on p_1, let

- $\mathsf{row}(i) = (i \,\mathsf{div}(n+1)) + 1$ and
- $\mathsf{col}(i) = i - 1 \,\mathsf{mod}(n+1)$ if $i \,\mathsf{mod}(n+1) > 0$ while $\mathsf{col}(i)$ is undefined otherwise.

The '+1' in the first item ensures that the first elements of p_1 corresponds to row 1 rather than to the bottommost row 0. The extra condition in the second items avoids assigning a column to positions in w that carry the symbol N.

By construction of the tree in $\mathcal{U}_\mathcal{K}$ below a_1, we can find a path $p_1 = d_0 \cdots d_k$ that satisfies the following conditions for all $i \leq k$:

1. $(a_1, d_0) \in S^{\mathcal{U}_\mathcal{K}}$;
2. $d_n \in E^{\mathcal{U}_\mathcal{K}}$ (this corresponds to the last position of the first row represented by p_1);
3. if $\tau(\mathsf{col}(i), \mathsf{row}(i)) = t$, then $d_i \in (B_t^{\mathsf{row}(i) \bmod 3})^{\mathcal{U}_\mathcal{K}}$;
4. if $\tau(\mathsf{col}(i), \mathsf{row}(i) - 1) = t$, then $d_i \in (B_t^{\mathsf{row}(i)-1 \bmod 3})^{\mathcal{U}_\mathcal{K}}$;
5. if $\mathsf{col}(i)$ is undefined, then $d_i \in N^{\mathcal{U}_\mathcal{K}}$.

Note that the first $n+1$ elements on p_1 represent the tiling of row 0 via concept names B_t^0 and the tiling of row 1 via concept names B_t^1. The next $n+1$ elements represent row 1 via concept names B_t^1 and the tiling of row 2 via concept names B_t^2, and so on.

Again by construction of the tree below a_1, we can extend p_1 into a path $p_1^+ = d_0 \cdots d_{k+(n+1)}$ that repeats the topmost row $m-1$ in the sense that the following are satisfied for $k < i \leq k + (n+1)$:

1. $(d_k, d_{k+1}) \in S^{\mathcal{U}_\mathcal{K}}$;
2. if $\tau(\mathsf{col}(i), m-1) = t$, then $d_i \in (B_t^{m-1 \bmod 3})^{\mathcal{U}_\mathcal{K}}$;
3. $d_{k+n} \in (B_d^{m-1 \bmod 3})^{\mathcal{U}_\mathcal{K}}$ (this corresponds to the second last position of the repeated row $m-1$).

So p_1^+ simply repeats the representation of the topmost row from the end of p_1, using the same concept name. The only difference is the labeling with the dummy tile described in Point 3, which is not present in p_1.

By construction of the tree below a_2, $\mathcal{U}_\mathcal{K}$ contains a path $p_2 = e_0 \cdots e_{k+1}$ that starts at an S-successor of $a_2 \in E^{\mathcal{U}_\mathcal{K}}$ and satisfies the following conditions for all $i < k$:

1. $(a_2, e_0) \in S^{\mathcal{U}_\mathcal{K}}$;
2. if $\tau(\mathsf{col}(i), \mathsf{row}(i) + 1) = t$, then $e_i \in (B_t^{\mathsf{row}(i)+1 \bmod 3})^{\mathcal{U}_\mathcal{K}}$;
3. if $\mathsf{col}(i)$ is undefined, then $d_i \in N^{\mathcal{U}_\mathcal{K}}$;
4. e_{k+1} is a cycle node.

Note that the labeling in Point 2 of p_2 is exactly the same as the labeling in Point 3 of p_1.

Now consider the following path in $\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}$ that starts at the S-successor (d_0, e_0) of (a_1, a_2):

- first follow p_1^+ and p_2 synchronously:
$$(d_0, e_0) \cdots (d_{k+1}, e_{k+1})$$
- then proceed to follow p_1^+ while remaining stationary in the cycle node at the end of p_2:
$$(d_{k+2}, e_{k+2}) \cdots (d_k - 1, e_{k+1})$$
- then make a single step downwards in p_+^1, reaching the end of this path, while making a single step upwards in p_2:
$$(d_k, e_k)$$
- then synchronously follow both paths backwards, even stepping up to a_2:
$$(d_{k-1}, e_{k-1}) \cdots (d_n, a_2).$$

By what was said above, it can be verified that (i) the end of this path (d_n, a_2) is in $E^{\mathcal{U}_\mathcal{K} \times \mathcal{U}_\mathcal{K}}$ and (ii) every element of the path satisfies a concept name from $\mathcal{C}$. The only slightly subtle point for the latter is the element (d_{k-1}, e_{k+1}), which is the predecessor of (d_k, e_k) on the constructed path. It satisfies a concept name of the form B_d^j and in fact achieving this is the reason for introducing the dummy tile (as no other concept name from $\mathcal{C}$ is satisfied by (d_k, e_k)).

However, there is no path in $\mathcal{U}_\mathcal{K}$ that starts at an S-successor of b and satisfies Properties (i) and (ii); in fact, every path that starts at an S-successor of b and whose end is in $E^{\mathcal{U}_\mathcal{K}}$ must pass an element that does not satisfy any concept name in $\mathcal{C}$. Consequently, $\Pi_{i \in \{1,2\}}(\mathcal{U}_\mathcal{K}, a_i) \not\preceq_{\mathcal{ELI}} \mathcal{U}_\mathcal{K}, b$. ❏

We show now that also *query by example (QBE)* over $\mathcal{ELI}$ knowledge bases is undecidable, as claimed in the main text and in contrast to what is claimed in [Gutiérrez-Basulto *et al.*, 2018]. Formally, QBE is the following problem:

- **Input**: $\mathcal{ELI}$ knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, sets of positive and negative examples $P, N \subseteq \mathsf{ind}(\mathcal{A})^n$
- **Question:** Is there a conjunctive query q with n free variables that *separates* P and N over $\mathcal{K}$, that is:
  - $\mathcal{K} \models q(\mathbf{a})$, for all $\mathbf{a} \in P$, and
  - $\mathcal{K} \not\models q(\mathbf{a})$, for all $\mathbf{a} \in N$?

It has been shown in [Gutiérrez-Basulto *et al.*, 2018] that there is a separating conjunctive query for an instance $(\mathcal{K}, P, N)$ with $P, N \subseteq \mathsf{ind}(\mathcal{A})$ iff $\prod_{a \in P}(\mathcal{U}_\mathcal{K}, a) \not\to \mathcal{U}_\mathcal{K}, b$, for every $b \in N$. Thus, Lemma 7 establishes that the reduction to concept separability is also a reduction to query-by-example. Hence, we obtain:

Theorem 20 *Query-by-example over $\mathcal{ELI}$ knowledge bases is undecidable.*

Theorem 14 *Let $\mathcal{L}_T \in \{\mathcal{ALC}, \mathcal{ALCQ}, \mathcal{ALCI}, \mathcal{ALCQI}\}$ and $\mathcal{L}_S \in \{\mathcal{EL}, \mathcal{ELI}\}$. Then $(\mathcal{L}_T, \mathcal{L}_S)$ concept separability and $(\mathcal{L}_T, \mathcal{L}_S)$ concept definability are undecidable.*

Proof. As indicated in the main paper, the proof is by reduction from the CQ entailment problem between $\mathcal{ALC}$ KBs, proved undecidable already for (directed or undirected) tree-shaped CQs in [Botoeva *et al.*, 2019]. For our purposes, the undecidable problem can be stated as follows, for $\mathcal{L} \in \{\mathcal{EL}, \mathcal{ELI}\}$: given an ABox $\mathcal{A}$ using a single individual a and $\mathcal{ALC}$ KBs $\mathcal{K}_1 = (\mathcal{T}_1, \mathcal{A})$ and $\mathcal{K}_2 = (\mathcal{T}_2, \mathcal{A})$, does $\mathcal{K}_1 \models C(a)$ imply $\mathcal{K}_2 \models C(a)$, for all $\mathcal{L}$ concepts C?

For the reduction, we define the *relativisation* C^A of an $\mathcal{ALCQI}$ concept C to a concept name A by induction as follows:

$$\begin{array}{rcl} \top^A &=& A \\ C^A &=& C \sqcap A,\ C \text{ a concept name} \\ (\neg C)^A &=& A \sqcap \neg C^A \\ (C \sqcap D)^A &=& C^A \sqcap D^A \\ (\geqslant n\ r\ C)^A &=& A \sqcap (\geqslant n\ r\ C^A) \end{array}$$

Define the relativization $\mathcal{T}^A$ of a TBox $\mathcal{T}$ to A as

$$\mathcal{T}^A := \{C^A \sqsubseteq D^A \mid C \sqsubseteq D \in \mathcal{T}\}.$$

We use the following property of relativizations: for any interpretation $\mathcal{I}$ and $d \in \Delta^{\mathcal{I}}$, $d \in (C^A)^{\mathcal{I}}$ if, and only if, $d \in C^{\mathcal{I}_{|A}}$, where $\mathcal{I}_{|A}$ denotes the restriction of $\mathcal{I}$ to $A^{\mathcal{I}}$.

Now assume that $\mathcal{L} \in \{\mathcal{EL}, \mathcal{ELI}\}$ and assume that $\mathcal{ALC}$ KBs $\mathcal{K}_1 = (\mathcal{T}_1, \mathcal{A})$ and $\mathcal{K}_2 = (\mathcal{T}_2, \mathcal{A})$ using a single individual a are given. Let A_1, A_2, and B be fresh concept names and set

$$\mathcal{T} = \mathcal{T}_1^{A_1} \cup \mathcal{T}_1^{A_2} \cup \mathcal{T}_2^{B}$$

Define $\mathcal{A}'$ to be the union of copies $\mathcal{A}^c$ of $\mathcal{A}$ in which a is replaced with $c \in \{a_1, a_2, b\}$, extended with the assertions $A_1(a_1), A_2(a_2), B(b)$. Let $\mathcal{K} = (\mathcal{T}, \mathcal{A}')$. We show that the following two conditions are equivalent:

- $(\mathcal{K}, P, N)$ has an $\mathcal{L}$ solution for $P = \{a_1, a_2\}$ and $N = \{b\}$;
- there exists an $\mathcal{L}$ concept C such that $\mathcal{K}_1 \models C(a)$ and $\mathcal{K}_2 \not\models C(a)$.

To prove the equivalence we make three observations.

Claim 1. The following hold for any $\mathcal{ELI}$ concept C:

1. if $\mathcal{K} \models C(a_1)$, then C does not contain A_2 or B;
2. if $\mathcal{K} \models C(a_2)$, then C does not contain A_1 or B;
3. if $\mathcal{K} \models C(b)$, then C does not contain A_1 or A_2.

We only show Item (1). Let $\mathcal{I}_1$ be a model of $(\mathcal{T}_1^{A_1}, \mathcal{A}_{a_1})$, $\mathcal{I}_1'$ a model of $(\mathcal{T}_1^{A_2}, \mathcal{A}_{a_2})$, and $\mathcal{I}_2$ a model of $(\mathcal{T}_2^{B}, \mathcal{A}_b)$. Assume without loss of generality that

- $a_2, b \notin \Delta^{\mathcal{I}_1}$, $a_1, b \notin \Delta^{\mathcal{I}_1'}$, and $a_1, a_2 \notin \Delta^{\mathcal{I}_2}$;
- $A_2^{\mathcal{I}_1} = B^{\mathcal{I}_1} = \emptyset$, $A_1^{\mathcal{I}_1'} = B^{\mathcal{I}_1'} = \emptyset$, and $A_1^{\mathcal{I}_2} = A_2^{\mathcal{I}_2} = \emptyset$.

Now define $\mathcal{I}$ as the disjoint union of $\mathcal{I}_1$, $\mathcal{I}_1'$, and $\mathcal{I}_2$. Then $\mathcal{I}$ is a model of $\mathcal{K}$ and $\mathcal{I} \not\models C(a_1)$ if A_2 or B occur in C because no node in $A_2^{\mathcal{I}}$ or $B^{\mathcal{I}}$ is reachable from a_1 in $\mathcal{I}$ and C is an $\mathcal{ELI}$ concept. This finishes the proof of Claim 1.

The following claim can also be proved using the same reachability argument for $\mathcal{ELI}$ concepts.

Claim 2. The following hold for any $\mathcal{ELI}$ concept C:

1. $\mathcal{K} \models C(a_1)$ if and only if $(\mathcal{T}_1^{A_1}, \mathcal{A}_{a_1}) \models C(a_1)$;
2. $\mathcal{K} \models C(a_2)$ if and only if $(\mathcal{T}_1^{A_2}, \mathcal{A}_{a_2}) \models C(a_2)$;
3. $\mathcal{K} \models C(b)$ if and only if $(\mathcal{T}_2^{B}, \mathcal{A}_b) \models C(b)$.

Finally, the following follows from the properties of relativizations.

Claim 3. The following hold for any $\mathcal{ELI}$ concept C:

1. if C does not contain A_1, then $\mathcal{K}_1 \models C(a)$ iff $(\mathcal{T}_1^{A_1}, \mathcal{A}_{a_1}) \models C(a_1)$;
2. if C does not contain A_2, then $\mathcal{K}_1 \models C(a)$ iff $(\mathcal{T}_1^{A_2}, \mathcal{A}_{a_2}) \models C(a_2)$;
3. if C does not contain B, then $\mathcal{K}_2 \models C(a)$ iff $(\mathcal{T}_2^{B}, \mathcal{A}_b) \models C(b)$.

We now prove the equivalence. First suppose there is an $\mathcal{L}$ concept C such that $\mathcal{K} \models C(a_1)$ and $\mathcal{K} \models C(a_2)$ but $\mathcal{K} \not\models C(b)$. By Claim 1, C does not contain A_1, A_2 or B. By Claims 2 and 3, $\mathcal{K}_1 \models C(a)$ and $\mathcal{K}_2 \not\models C(a)$, as required. Conversely, suppose $\mathcal{K}_1 \models C(a)$ and $\mathcal{K}_2 \not\models C(a)$. Since A_1, A_2, B do not occur in $\mathcal{K}_1$, they do not occur in C either. Then Claims 2 and 3 yield $\mathcal{K} \models C(a_1)$ and $\mathcal{K} \models C(a_2)$ as well as $\mathcal{K} \not\models C(b)$ from $\mathcal{K}_2 \not\models C(a)$. ❑

E Proofs for Section 7

We first give detailed definitions of strong solutions to learning instances and the strong separability problem.

Definition 2 *Let $\mathcal{L}_T, \mathcal{L}_S$ be DLs and $(\mathcal{K}, P, N)$ a learning instance with $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ an $\mathcal{L}_T$ KB. A* strong $\mathcal{L}_S$ solution *to $(\mathcal{K}, P, N)$ is an $\mathcal{L}_S$ concept C such that*

1. *$\mathcal{K} \models C(a)$ for all $a \in P$ and*
2. *$\mathcal{K} \models \neg C(a)$ for all $a \in N$.*

Any TBox language $\mathcal{L}_T$ and separation language $\mathcal{L}_S$ give rise to an associated *strong concept separability* problem.

PROBLEM :	strong $(\mathcal{L}_T, \mathcal{L}_S)$ concept separability
INPUT :	$\mathcal{L}_T$ learning instance $(\mathcal{K}, P, N)$
QUESTION :	Does $(\mathcal{K}, P, N)$ have a strong $\mathcal{L}_S$ solution?

We simply speak of *strong $\mathcal{L}$ concept separability* when $\mathcal{L}_T = \mathcal{L}_S = \mathcal{L}$.

Theorem 15 *For $\mathcal{L}_S \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALCQ}, \mathcal{ALCQI}\}$, an $\mathcal{ALCQI}$ learning instance $(\mathcal{K}, P, N)$ has a strong $\mathcal{L}$ solution iff for all models $\mathcal{I}$ and $\mathcal{J}$ of $\mathcal{K}$, all $a \in P$ and all $b \in N$, $\mathcal{I}, a \not\sim_{\mathcal{L}_S} \mathcal{J}, b$.*

Proof. The implication $(\Rightarrow)$ follows directly from Lemma 1. For the converse direction, assume $(\mathcal{K}, P, N)$ has no strong $\mathcal{L}_S$ solution. Let

$$\begin{aligned}\Gamma_P &= \{C \in \mathcal{L}_S \mid \forall a \in P : \mathcal{K} \models C(a)\}\\ \Gamma_N &= \{C \in \mathcal{L}_S \mid \forall a \in N : \mathcal{K} \models C(a)\}\end{aligned}$$

In what follows we use the fact that Γ_P and Γ_N are closed under conjunction. We say that a set Γ of concepts is satisfiable in $a \in \mathsf{ind}(\mathcal{A})$ w.r.t. a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ if the extended (possibly infinite) KB

$$\mathcal{K}' = (\mathcal{T}, \mathcal{A} \cup \{C(a) \mid C \in \Gamma\})$$

is satisfiable.

Claim 1. (1) There exists $a \in P$ such that $\Gamma_P \cup \Gamma_N$ is satisfiable in a w.r.t. $\mathcal{K}$. (2) There exists $a \in N$ such that $\Gamma_P \cup \Gamma_N$ is satisfiable in a w.r.t. $\mathcal{K}$.

We prove Condition (1). The proof of Condition (2) is dual. Assume $\Gamma_P \cup \Gamma_N$ is not satisfiable in any $a \in P$ w.r.t. $\mathcal{K}$. Then Γ_N is not satisfiable in any $a \in P$ w.r.t. $\mathcal{K}$. By compactness, there exist $D_a \in \Gamma_N$ such that $\mathcal{K} \models \neg D_a(a)$, for all $a \in N$. Thus, $\mathcal{K} \models \neg(\bigsqcap_{b \in P} D_b)(a)$ for all $a \in P$ and $\mathcal{K} \models (\bigsqcap_{b \in P} D_b)(a)$ for all $a \in N$. We have derived a contradiction. to the assumption that $(\mathcal{K}, P, N)$ has no strong $\mathcal{L}_S$ solution.

Now let $\Gamma_0 = \Gamma_P \cup \Gamma_N$ and consider an enumeration $C_1, C_2, \ldots$ of the remaining concepts in $\mathcal{L}_S$. Then we set inductively, $\Gamma_{i+1} = \Gamma_i \cup \{C_{i+1}\}$ if there exist $a \in P$ and $b \in N$ such that $\Gamma_i \cup \{C_{i+1}\}$ is satisfiable in both a and b w.r.t. $\mathcal{K}$. Set $\Gamma_{i+1} = \Gamma_i \cup \{\neg C_{i+1}\}$, otherwise.

Claim 2. For all $i > 0$: there are $a \in P$ and $b \in N$ such that $\Gamma_i \cup \{C_{i+1}\}$ is satisfiable in both a and b w.r.t. $\mathcal{K}$ or there are $a \in P$ and $b \in N$ such that $\Gamma_i \cup \{\neg C_{i+1}\}$ is satisfiable in both a and b w.r.t. $\mathcal{K}$.

Assume Claim 2 has been proved for $i - 1$. Let w.l.o.g., $\Gamma_i = \Gamma_P \cup \Gamma_n \cup \{C_1, \ldots, C_i\}$. Assume Claim 2 does not hold for i. Then, again w.l.o.g., there is no $a \in P$ such that $\Gamma_i \cup \{C_{i+1}\}$ is satisfiable in a w.r.t. $\mathcal{K}$ and there is no $b \in N$ such that $\Gamma_i \cup \{\neg C_{i+1}\}$ is satisfiable in b w.r.t. $\mathcal{K}$. By compactness, there exists $D \in \Gamma_N$ such that $\mathcal{K} \models D'(a)$ for all $a \in P$ and

$$D' = ((D \sqcap C_1 \sqcap \cdots \sqcap C_i) \rightarrow \neg C_{i+1}).$$

Then, by definition, we have $D' \in \Gamma_P$. Then $D' \in \Gamma_i$ and so there is no $b \in N$ such that Γ_i is satisfiable in b w.r.t. $\mathcal{K}$. We have derived a contradiction.

Let $\Gamma = \bigcup_{i>0} \Gamma_i$. Then there exist models $\mathcal{I}$ and $\mathcal{J}$ of $\mathcal{K}$ and $a \in P$ and $b \in P$ such that $\mathcal{I} \models C(a)$ for all $C \in \Gamma$ and $\mathcal{J} \models C(b)$ for all $C \in \Gamma$. Thus, $\mathcal{I}, a \equiv_{\mathcal{L}_S} \mathcal{J}, b$. We may assume that $\mathcal{I}$ and $\mathcal{J}$ are ω-saturated in the sense of classical model theory. Then $\mathcal{I}, a \sim_{\mathcal{L}_S} \mathcal{J}, b$, as required. For ω-saturated interpretations and the implication $\mathcal{I}, a \equiv_{\mathcal{L}_S} \mathcal{J}, b \Rightarrow \mathcal{I}, a \sim_{\mathcal{L}_S} \mathcal{J}, b$ for ω-saturated interpretations, see [Lutz *et al.*, 2011; Goranko and Otto, 2007] and references therein. ❑

Theorem 16 *For $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}\}$, strong $\mathcal{L}$ concept separability is* ExpTime*-complete in combined complexity and* coNP*-complete in data complexity.*

Proof. We give the proof for $\mathcal{ALCI}$. The proof for $\mathcal{ALC}$ is the same and omitted. By Theorem 15, we have to decide whether there exist pointed models $\mathcal{I}, a$ and $\mathcal{J}, b$ of $\mathcal{K}$ with $a \in P$ and $b \in N$ such that $\mathcal{I}, a \sim_{\mathcal{ALCI}} \mathcal{J}, b$. A *$\mathcal{K}$-type* t is a maximal subset of the closure of $\mathsf{sub}(\mathcal{K})$ under single negation that is satisfiable. We prove that the following are equivalent for any $a, b \in \mathsf{ind}(\mathcal{A})$:

1. there exist $\mathcal{I}, a$ and $\mathcal{J}, b$ with $\mathcal{I}, a \sim_{\mathcal{ALCI}} \mathcal{J}, b$;
2. there exists a $\mathcal{K}$-type t such that both $\mathcal{K} \cup \{C(a) \mid C \in t\}$ and $\mathcal{K} \cup \{C(b) \mid C \in t\}$ are satisfiable.

The direction (1) $\Rightarrow$ (2) is trivial. Conversely, take a $\mathcal{K}$-type t such that for some $a \in P$ and $b \in N$ the extended KBs $\mathcal{K} \cup \{C(a) \mid C \in t\}$ and $\mathcal{K} \cup \{C(b) \mid C \in t\}$ are satisfiable. Take models $\mathcal{I}_a$ and $\mathcal{J}_b$ of $\mathcal{K} \cup \{C(a) \mid C \in t\}$ and $\mathcal{K} \cup \{C(b) \mid C \in t\}$, respectively. Define new models $\mathcal{I}$ and $\mathcal{J}$ as follows: obtain $\mathcal{I}$ by hooking $\mathcal{J}_b$ to $\mathcal{I}_a$ by identifying a and b (and replacing all individual names of $\Delta^{\mathcal{J}_b} \setminus \{b\}$ by fresh non individuals names) and obtain $\mathcal{J}$ by hooking $\mathcal{I}_a$ to $\mathcal{J}_b$ by identifying b and a (and replacing all individual names of $\Delta^{\mathcal{I}_a} \setminus \{a\}$ by fresh non individuals names). Then both $\mathcal{I}$ and $\mathcal{J}$ are models of $\mathcal{K}$ and, modulo renaming of individuals, $\mathcal{I}, a$ and $\mathcal{J}, b$ are isomorphic, and so $\mathcal{I}, a \sim_{\mathcal{ALCI}} \mathcal{J}, b$. We give the explicit construction of $\mathcal{I}$. Take for any $c \in \Delta^{\mathcal{J}_b} \setminus \{b\}$ a fresh individual c'. Then set

$$\Delta^{\mathcal{I}} = \Delta^{\mathcal{I}_a} \cup \{c' \mid c \in \Delta^{\mathcal{J}_b} \setminus \{b\}\}$$

and

$$A^{\mathcal{I}} = A^{\mathcal{I}_a} \cup \{c' \mid c \in A^{\mathcal{J}_b} \setminus \{b\}\}$$

and

$$\begin{aligned} r^{\mathcal{I}} \;=\; & r^{\mathcal{I}_a} \cup \\ & \{(c_1', c_2') \mid (c_1, c_2) \in r^{\mathcal{J}_b}, b \notin \{c_1, c_2\}\} \cup \\ & \{(a, c') \mid (b, c) \in r^{\mathcal{J}_b}, c \neq b\} \cup \\ & \{(c', a) \mid (c, b) \in r^{\mathcal{J}_b}, c \neq b\} \cup \\ & \{(a, a) \mid (b, b) \in r^{\mathcal{J}_b}\} \end{aligned}$$

The ExpTime upper bound now follows from the fact that satisfiability of $\mathcal{ALCI}$ KBs can be decided in ExpTime and the coNP-upper bound is immediate.

For the lower bounds, recall that the satisfiability of $\mathcal{ALCI}$ KBs is ExpTime-complete in combined complexity and NP-complete in data complexity. Let for any $\mathcal{ALCI}$ KB $\mathcal{K}$, $\mathcal{K}' = \mathcal{K} \cup \{A(a), A(b)\}$ with a, b fresh individual names and A a concept name. Then $\mathcal{K}$ is satisfiable iff $(\mathcal{K}', P, N)$ has no strong $\mathcal{ALCI}$ solution, where $P = \{a\}$ and $N = \{b\}$. The ExpTime-lower bound in combined complexity and coNP-lower bound in data complexity now follow by polynomial reduction of the $\mathcal{ALCI}$ KB unsatisfiability problem.

The lower bound proof above is slightly unsatisfactory as it is based on having learning instances with unsatisfiable KBs. Here is an argument that uses satisfiable KBs only. The problem whether $\mathcal{T} \models A \sqsubseteq \neg B$ for $\mathcal{ALCI}$ TBoxes $\mathcal{T}$ and concept names A, B is well known to be ExpTime-complete. Now let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, $\mathcal{A} = \{A(a), B(b)\}$, $P = \{a\}$, and $N = \{b\}$. Then $(\mathcal{K}, P, N)$ has a strong $\mathcal{ALCI}$ solution iff $\mathcal{T} \models A \sqsubseteq \neg B$.

For the coNP-lower bound note that it is well known that it is coNP-hard in data complexity to decide $\mathcal{K} \models \neg B(a)$ for a $\mathcal{ALCI}$ KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ and concept name B. Now let $\mathcal{K}' = (\mathcal{T}, \mathcal{A} \cup \{B(b)\})$ for a fresh individual names b and set $P = \{a\}$ and $N = \{b\}$. Then $(\mathcal{K}', P, N)$ has a strong $\mathcal{ALCI}$ solution iff $\mathcal{K} \models \neg B(a)$. ❑