Model-Theoretic Inseparability and Modularity of Description Logic Ontologies

Boris Konev[a,*], Carsten Lutz[b], Dirk Walther[c], Frank Wolter[a]

[a] *University of Liverpool, Liverpool, UK*
[b] *Universität Bremen, Bremen, Germany*
[c] *Universidad Politécnica de Madrid, Madrid, Spain*

Abstract

The aim of this paper is to introduce and study model-theoretic notions of modularity in description logic and related reasoning problems. Our approach is based on a generalisation of logical equivalence that is called model-theoretic inseparability. Two TBoxes are inseparable w.r.t. a vocabulary Σ if they cannot be distinguished by the Σ-reducts of their models and thus can equivalently be replaced by one another in any application where only vocabulary items from Σ are relevant. We study in-depth the complexity of deciding inseparability for the description logics $\mathcal{EL}$ and $\mathcal{ALC}$ and their extensions with inverse roles. We then discuss notions of modules of a TBox based on model-theoretic inseparability and develop algorithms for extracting minimal modules from acyclic TBoxes. Finally, we provide an experimental evaluation of our module extraction algorithm based on the large-scale medical TBox Snomed ct.

1. Introduction

The main use of ontologies in computer and information science is to formalise the vocabulary of an application domain; i.e., to fix the vocabulary as a logical signature and to provide a logical theory that defines the semantics of the terms and relations in the vocabulary. The wide adoption of the

[*] Corresponding author

Email addresses: `konev@liverpool.ac.uk` (Boris Konev), `clu@informatik.uni-bremen.de` (Carsten Lutz), `dirk.walther@upm.es` (Dirk Walther), `wolter@liverpool.ac.uk` (Frank Wolter)

W3C-endorsed ontology language OWL and its profiles [1, 2], the success of logic-based reasoning support for concept classification and instance retrieval [3, 4, 5, 6], and the availability of ontology editors and management systems such as Protégé and SWOOP [7, 8] has led to the development and use of large-scale and complex ontologies that capture the vocabulary and knowledge of a wide diversity of domains. Especially in the Life Sciences and other knowledge intensive domains, many such ontologies have been created. Important examples are the national cancer institute's thesaurus and ontology (NCI), the gene ontology (GO), and Snomed ct, the Systematized Nomenclature of Medicine, Clinical Terms, which comprises about three hundred thousand vocabulary items and is used in the healthcare systems of more than twenty countries [9, 10, 11].

Engineering and maintaining professional ontologies such as the ones mentioned above is a complex and challenging task, and it has to be carried out with great care for the resulting ontology to be of high quality. Ontology design may involve a group of ontology engineers and domain experts that cooperate in order to design the ontology, update it to reflect changes/developments in the domain, and integrate it with other ontologies so as to cover larger domains. In such an environment, automated tool support for comparing, merging, updating, modularising, and re-using ontologies is of critical importance.

The aim of this paper is to propose and investigate a model-theoretic notion of inseparability between ontologies that can serve as a logical underpinning for many of these operations. From an application perspective, two ontologies are inseparable if they can be equivalently replaced by one another in any context. In logic, this can be formalised by the notion of logical equivalence, according to which two ontologies are inseparable if they have the same models. However, logical equivalence as such is clearly not sufficiently flexible to serve as a logical underpinning of modularity. For example, a module of an ontology is typically not logically equivalent to the ontology itself, and an updated ontology is typically not logically equivalent to the original ontology. By parameterising logical equivalence with a signature Σ of vocabulary items of interest, we obtain a notion of inseparability that has exactly the flexibility and properties required. We say that two ontologies are Σ-*inseparable* if the Σ-reducts of their models coincide, where the Σ-reduct of a structure is simply the restriction of that structure to the symbols in Σ. Then, a module can be defined as a subset of an ontology that is self-contained in the sense that it is Σ-inseparable from the ontology, where Σ is the set of vocabulary items that occur in the module. Similarly, an update can be categorised as not harmful for a signature Σ if the updated ontology is Σ-inseparable from the original ontology. Many other relevant notions such as controlled import of ontologies and safety of an ontology for a signature as studied in [12] can be formalised as well using Σ-inseparability, see [12, 13].

An important feature of model-theoretic inseparability is its language independence. Alternative notions of inseparability are based on deduction and use logical consequence within a given language to define inseparability of TBoxes. For example, two $\mathcal{ALC}$-TBoxes are called deductively Σ-inseparable in $\mathcal{ALC}$ iff they entail the same $\mathcal{ALC}$-concept inclusions in Σ. Distinct description logics typically define distinct versions of deductive inseparability [14] which is clearly undesirable if one moves from one language to another. In contrast, model-theoretic inseparability implies inseparability w.r.t. any standard description logic and even w.r.t. second-order logic.

The first main contribution of this paper is a systematic analysis of the complexity of deciding whether two ontologies are Σ-inseparable. We focus on ontologies that are formulated as a general or acyclic TBox in the description logic $\mathcal{EL}$ that underpins the OWL2 EL profile, the paradigmatic expressive description logic $\mathcal{ALC}$, and their extensions with inverse roles $\mathcal{ELI}$ and $\mathcal{ALCI}$. Our analysis starts with two fundamental undecidability results: Σ-inseparability is undecidable for $\mathcal{ALC}$-TBoxes even if one TBox is acyclic and the other is empty; for $\mathcal{EL}$, the undecidability result is slightly weaker, applying to the case where one TBox is empty and the other is a general TBox. It is due to such undecidability results that automated tool support for modular ontology design and maintenance is currently not based on model-theoretic inseparability, but either on deductive versions of inseparability (see, for example, [14]) or stronger inseparability notions based on locality [12]. However, the second part of our complexity analysis reveals that there are natural conditions on the TBox and/or signature that lead to a dramatic drop in complexity.

The first such condition is the restriction of the signature to a concept signature, i.e., a signature that comprises only concept names, but no role names. In this case, deciding Σ-inseparability becomes $\textsc{coNExp}^{\text{NP}}$-complete for general $\mathcal{EL}$ and $\mathcal{ALC}$-TBoxes. The proof of this result is of particular interest since it reveals a close connection between Σ-inseparability on the one hand and satisfiability in non-monotonic description logics based on circumscription on the other hand. By combining concept signatures with the additional condition that one TBox is empty, the complexity goes down to Π_2^p for $\mathcal{ALC}$ and PTime for $\mathcal{EL}$. Finally, Σ-inseparability of acyclic $\mathcal{EL}$-TBoxes from the empty TBox also turns out to be in PTime, even for signatures with role names. While these cases appear to be rather restricted at first glance, they actually play a central role in our algorithms for module checking and module extraction, discussed below. All mentioned results also hold for the extensions of $\mathcal{EL}$ and $\mathcal{ALC}$ with inverse roles. Note that, in various relevant cases, model-theoretic inseparability thus turns out to be strictly less complex than standard reasoning tasks such as subsumption, which is PSpace-complete for acyclic TBoxes formulated in $\mathcal{ALC}$ and in $\mathcal{ELI}$ [15]. It is also interesting that the difference in complexity of subsumption between $\mathcal{EL}$ and $\mathcal{ELI}$ (PTime vs. ExpTime [15]) is not reflected in the complexity of inseparability.

The PTime and Π_2^p-complexity results indicate that model-theoretic Σ-inseparability can be useful not only as a theoretical tool to define an "ideal" form of modularity, but also for practical purposes. The second aim of this paper is to apply Σ-inseparability to define several notions of a module in a TBox, and to develop algorithms for module checking and for extracting minimal modules. The former task is, given a TBox and a subset of the TBox, to decide whether the subset is a module; the latter task is, given a TBox and a signature Σ of interest, to extract an as small as possible module for the signature Σ. We give a polynomial time algorithm for module checking in acyclic $\mathcal{ELI}$-TBoxes and a Π_2^p-module checking algorithm for acyclic $\mathcal{ALCI}$-TBoxes under a natural additional syntactic condition. Note that acyclic TBoxes are used in relevant practical applications, including Snomed ct, several versions of NCI, and GO. For module extraction, we consider two approaches. First, we show that a generic module extraction algorithm can be applied to acyclic $\mathcal{ELI}$- and $\mathcal{ALCI}$-TBoxes in a black box manner by using our module module checking algorithm as an oracle. Second, we pursue a white box approach in which we directly use the module checking algorithm for acyclic $\mathcal{ELI}$-TBoxes to obtain a more direct module extraction algorithm. Finally, we introduce the module extraction software MEX that implements the white box approach and carry out a case study by extracting minimal modules from the Snomed ct ontology. The study shows that our algorithms scale effortlessly to ontologies of very large size and very often extract modules that are significantly smaller than those produced by the standard $\top\bot^*$-module extraction algorithm [16, 17] and all other existing approaches.

The paper is organised as follows. After a section introducing basic definitions and terminology we define model-theoretic inseparability in Section 3. In this section, we also introduce and investigate basic properties and applications of model-theoretic inseparability. In Sections 4 and 5, we investigate the computational complexity of deciding Σ-inseparability and then, in Sections 6 and 7 we introduce and investigate module checking and module extraction based on Σ-inseparability. A case study with experiments follows in Section 8. The paper closes with a section on related work and a discussion of future work.

This journal article is an extended version of the conference paper [18]. For the sake of readability, some proofs are deferred to an appendix.

2. Preliminaries

We introduce the syntax and semantics of the description logics considered in this paper, which are $\mathcal{EL}$, $\mathcal{ELI}$, $\mathcal{ALC}$, and $\mathcal{ALCI}$; for a more thorough introduction, the reader is referred to the DL handbook [19]. Fix two countably infinite and disjoint sets $\mathsf{N_C}$ and $\mathsf{N_R}$ whose elements are the *concept names* and the *role names*, respectively. *$\mathcal{ALCI}$-concepts* are built according to the syntax rule

$$C, D \quad ::= \quad \top \quad | \quad A \quad | \quad \neg C \quad | \quad C \sqcap D \quad | \quad \exists r.C \quad | \quad \exists r^-.C$$

where A ranges over $\mathsf{N_C}$ and r over $\mathsf{N_R}$. As usual, we use $\bot$ to abbreviate $\neg\top$, $C \sqcup D$ for $\neg(\neg C \sqcap \neg D)$, $C \rightarrow D$ for $\neg C \sqcup D$, and $\forall r.C$ for $\neg\exists r.\neg C$. A *role* is either a role name or an *inverse role*, i.e., an expression r^- with $r \in \mathsf{N_R}$. For each inverse role $s = r^-$, we set $s^- = r$. Throughout the paper, we typically use A, B to denote concept names, r, s, t to denote roles, and C, D, E to denote composite concepts.

An *$\mathcal{ALC}$-concept* is an $\mathcal{ALCI}$-concept that does not use inverse roles, i.e., the constructor $\exists r^-.C$ and the abbreviation $\forall r^-.C$ are not allowed. An *$\mathcal{ELI}$-concept* is an $\mathcal{ALCI}$-concept that does not use negation $\neg C$, and thus also disallows the abbreviations $\bot$, $C \sqcup D$, $C \rightarrow D$, and $\forall r.C$. Finally, an *$\mathcal{EL}$-concept* is an $\mathcal{ELI}$-concept that does not use inverse roles.

Concepts are used in TBoxes to describe a domain of interest. Formally, a *concept inclusion (CI)* is an expression $C \sqsubseteq D$, a *concept equality (CE)* is an expression $C \equiv D$, and a *TBox* is a finite set of CIs and CEs. When a TBox $\mathcal{T}$ uses only $\mathcal{L}$-concepts with $\mathcal{L} \in \{\mathcal{EL}, \mathcal{ELI}, \mathcal{ALC}, \mathcal{ALCI}\}$, then we call $\mathcal{T}$ an *$\mathcal{L}$-TBox*. For uniform reference, we sometimes call CIs and CEs *TBox statements*. To explicitly distinguish the TBoxes introduced here from acyclic TBoxes as introduced later on, we sometimes speak of *general TBoxes*.

We are sometimes using also first-order logic (FO) and second-order logic (SO) over the signature $\mathsf{N_C}$ of unary predicates and $\mathsf{N_R}$ of binary predicates, identifying unary predicates with concept names and binary predicates with role names. Equality is admitted. SO-formulas are thus built according to the syntax rule

$$\begin{aligned}\varphi \quad ::= \quad & x = y \mid A(x) \mid X_1(x) \mid X_2(x,y) \mid r(x,y) \mid \neg\varphi \mid \\ & \varphi \wedge \varphi \mid \exists x.\varphi \mid \exists X.\varphi\end{aligned}$$

where x, y range over first-order variables, A over $\mathsf{N_C}$, X_i over second-order variables of arity i, r over $\mathsf{N_R}$, and X over second-order variables of arity one or two. FO is the fragment of SO without second-order variables and second-order quantifiers. When speaking of an *FO-TBox* and an *SO-TBox*, we mean a finite set of FO-sentences and SO-sentences, respectively.

A *signature* Σ is a finite subset of $\mathsf{N_C} \cup \mathsf{N_R}$. In this context, we refer to both concept names and role names as *symbols*. The signature $\mathsf{sig}(C)$ of a concept C is the set of concept and role names that occur in C. If $\mathsf{sig}(C) \subseteq \Sigma$, we call C a *Σ-concept*. The same terminology is applied to TBox statements, TBoxes, and FO- and SO-formulas.

The semantics of DLs and of FO- and SO-formulas is given by *interpretations* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where the *domain* $\Delta^\mathcal{I}$ is a non-empty set and $\cdot^\mathcal{I}$ is an *interpretation function* that maps each $A \in \mathsf{N_C}$ to a subset $A^\mathcal{I}$ of $\Delta^\mathcal{I}$ and each $r \in \mathsf{N_R}$ to a binary relation $r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$. As usual, $\mathcal{I} \models \varphi$ denotes that the interpretation $\mathcal{I}$ is a *model of an* SO-sentence φ, and $\Phi \models \varphi$ denotes that the SO-sentence φ follows from the set of SO-sentences Φ. We define the semantics of DL-concepts by the standard translation $\cdot^\sharp$ into FO-formulas with one free variable shown in the upper half of Figure 1. For each concept C, and an interpretation $\mathcal{I}$, we set $C^\mathcal{I} = \{d \in \Delta^\mathcal{I} \mid \mathcal{I} \models C^\sharp[d]\}$. The semantics of TBox statements is given by a translation into FO-sentences, as shown in the lower half of Figure 1; it can be extended to TBoxes $\mathcal{T}$ in the obvious way, setting $\mathcal{T}^\sharp = \{\alpha^\sharp \mid \alpha \in \mathcal{T}\}$. We often confuse concepts with their FO-translations and do the same also for TBox statements and TBoxes, which allows us to speak of an interpretation being a *model* of a TBox $\mathcal{T}$, to write $\mathcal{T} \models C \sqsubseteq D$ when $\mathcal{T}$ is an SO-TBox and $C \sqsubseteq D$ is a CI formulated in some DL, and to write $\mathcal{T} \models \varphi$ when $\mathcal{T}$ is a DL TBox and φ an SO-sentence.

Concept C	Translation $C^\sharp$
$\top$	$x = x$
A	$A(x)$
$\neg C$	$\neg C^\sharp(x)$
$C \sqcap D$	$C^\sharp(x) \wedge D^\sharp(x)$
$\exists r.C$	$\exists y\,(r(x,y) \wedge C^\sharp(x)[y/x])$
$\exists r^-.C$	$\exists y\,(r(y,x) \wedge C^\sharp(x)[y/x])$
Inclusion α	Translation $\alpha^\sharp$
$C \sqsubseteq D$	$\forall x\,(C^\sharp(x) \rightarrow D^\sharp(x))$
$C \equiv D$	$\forall x\,(C^\sharp(x) \leftrightarrow D^\sharp(x))$

Figure 1: Standard translation $\cdot^\sharp$ of $\mathcal{ALCI}$-concept and TBox statements into FO.

The standard reasoning problem in DLs is *subsumption*: given a TBox $\mathcal{T}$ and a CI $C \sqsubseteq D$, decide whether $\mathcal{T} \models C \sqsubseteq D$. Subsumption is ExpTime-complete in $\mathcal{ELI}$, $\mathcal{ALC}$, and $\mathcal{ALCI}$ [20, 19], and in PTime in $\mathcal{EL}$ [21]. We say that TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ are *equivalent*, in symbols $\mathcal{T}_1 \equiv \mathcal{T}_2$, if they have the same models.

Acyclic TBoxes

It will often be useful to also study a weaker form of TBoxes, which we introduce next. A TBox $\mathcal{T}$ is *acyclic* if it satisfies the following conditions:

- all CEs in $\mathcal{T}$ are of the form $A \equiv C$ (*concept definitions*) and all CIs in $\mathcal{T}$ are of the form $A \sqsubseteq C$ (*primitive concept inclusions*), where A is a concept name;
- no concept name occurs more than once on the left-hand side of a statement in $\mathcal{T}$;
- $\mathcal{T}$ contains no cyclic definitions, as detailed below.

Let $\mathcal{T}$ be a TBox that contains only concept definitions and primitive concept inclusions. The relation $\prec_{\mathcal{T}} \subseteq \mathsf{N_C} \times \mathsf{sig}(\mathcal{T})$ is defined by setting $A \prec_{\mathcal{T}} X$ if there exists a TBox statement $A \bowtie C$ such that X occurs in C, where $\bowtie$ ranges over $\{\sqsubseteq, \equiv\}$. A concept name A *depends* on a symbol $X \in \mathsf{N_C} \cup \mathsf{N_R}$ if $A \prec_{\mathcal{T}}^{+} X$, where $\cdot^+$ denotes transitive closure. We use $\mathsf{depend}_{\mathcal{T}}(A)$ to denote the set of all symbols X such that A depends on X. We can now make precise what it means for $\mathcal{T}$ to *contain no cyclic definitions*: $A \notin \mathsf{depend}_{\mathcal{T}}(A)$, for all $A \in \mathsf{N_C}$.

When proving results about acyclic TBoxes $\mathcal{T}$, it is often convenient to refer to the *definitorial depth* of concept names in $\mathcal{T}$, defined as follows: $d_{\mathcal{T}}(A) = 0$ if A does not occur in any statement $A \bowtie C$ in $\mathcal{T}$; otherwise, there is a unique $A \bowtie C \in \mathcal{T}$ and $d_{\mathcal{T}}(A) = 1 + \max\{d_{\mathcal{T}}(B) \mid B$ occurs in $C\}$ with the maximum of the empty set defined as 0.

3. Model-Theoretic Inseparability

We introduce model-theoretic inseparability, explain its relevance for TBox interoperation and modularity, and establish some fundamental properties. Let $\mathcal{I}$ be an interpretation and Σ a signature. The Σ-*reduct* $\mathcal{I}|_{\Sigma}$ of $\mathcal{I}$ is the interpretation obtained from $\mathcal{I}$ by setting

- $\Delta^{\mathcal{I}|_{\Sigma}} := \Delta^{\mathcal{I}}$;
- $X^{\mathcal{I}|_{\Sigma}} := X^{\mathcal{I}}$, for all $X \in \Sigma$;
- $X^{\mathcal{I}|_{\Sigma}} := \emptyset$, for all $X \in (\mathsf{N_C} \cup \mathsf{N_R}) \setminus \Sigma$.

Two interpretations $\mathcal{I}$ and $\mathcal{J}$ *coincide on a signature* Σ if $\mathcal{I}|_{\Sigma} = \mathcal{J}|_{\Sigma}$.

Definition 1 (Σ-Inseparability)**.** Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be SO-TBoxes and Σ a signature. Then $\mathcal{T}_1$ and $\mathcal{T}_2$ are Σ-*inseparable*, in symbols $\mathcal{T}_1 \equiv_{\Sigma} \mathcal{T}_2$, if $\{\mathcal{I}|_{\Sigma} \mid \mathcal{I} \models \mathcal{T}_1\} = \{\mathcal{I}|_{\Sigma} \mid \mathcal{I} \models \mathcal{T}_2\}$.

Note that Σ-inseparability generalises logical equivalence: for every signature Σ such that $\mathsf{sig}(\mathcal{T}_1) \cup \mathsf{sig}(\mathcal{T}_2) \subseteq \Sigma$, it is easy to see that $\mathcal{T}_1 \equiv_{\Sigma} \mathcal{T}_2$ if, and only if, $\mathcal{T}_1 \equiv \mathcal{T}_2$. However, being able to choose a signature makes inseparability a more flexible and versatile tool than equivalence, as illustrated by the examples below.

Example 2. (1) Consider the following two fragments $\mathcal{T}_1$ and $\mathcal{T}_2$ of ontologies defining $\mathsf{Cystic\_fibrosis\_screening}$. $\mathcal{T}_1$ consists of the definition

$$\begin{array}{lcl} \mathsf{Cystic\_fibrosis\_screening} & \equiv & \mathsf{Screening} \sqcap \\ & & \exists \mathsf{has\_Focus}.\mathsf{Cystic\_fibrosis} \sqcap \\ & & \exists \mathsf{has\_Intent}.\mathsf{Screening\_procedure\_intent} \end{array}$$

and $\mathcal{T}_2$ consists of the inclusions

$$\begin{array}{lcl}\mathsf{Cystic\_fibrosis\_screening} & \sqsubseteq & \mathsf{Genetic\_testing} \\ \mathsf{Genetic\_testing} & \sqsubseteq & \mathsf{Molecular\_analysis} \sqcap \mathsf{Screening}\end{array}$$

Clearly, $\mathcal{T}_1 \not\equiv \mathcal{T}_2$. However, for $\Sigma = \{\mathsf{Cystic\_fibrosis\_screening}, \mathsf{Screening}\}$ we have $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$. In fact, let

$$\mathcal{T}_3 = \{\mathsf{Cystic\_fibrosis\_screening} \sqsubseteq \mathsf{Screening}\}.$$

Then one can show that $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2 \equiv_\Sigma \mathcal{T}_3$.

(2) Assume that a TBox $\mathcal{T}'$ is a *definitorial extension* of a TBox $\mathcal{T}$; i.e., $\mathcal{T}'$ is obtained from $\mathcal{T}$ by adding new concept definitions $A \equiv C$ so that A does neither occur in $\mathcal{T}$ nor on the right-hand side of any of the new definitions. For example, suppose that $\mathcal{T}_1$ from above has been extended with

$$\begin{array}{lcl}\mathsf{Tuberculosis\_screening} & \equiv & \mathsf{Bacterial\_disease\_screening} \sqcap \\ & & \exists \mathsf{has\_Focus}.\mathsf{Tuberculosis} \sqcap \\ & & \exists \mathsf{has\_Intent}.\mathsf{Screening\_procedure\_intent},\end{array}$$

where $\mathsf{Tuberculosis\_screening}$ is a new concept name. Intuitively, $\mathcal{T}'$ does not interfere with the meaning of symbols in $\mathcal{T}$, but only introduces new names for complex concepts. Indeed, one can show that $\mathcal{T} \equiv_{\mathsf{sig}(\mathcal{T})} \mathcal{T}'$ whenever $\mathcal{T}'$ is a definitorial extension of $\mathcal{T}$.

One can show that $\equiv_\Sigma$ is an equivalence relation on the set of SO-TBoxes. Note that $\Sigma \subseteq \Sigma'$ implies $\equiv_\Sigma \supseteq \equiv_{\Sigma'}$, i.e., shrinking the signature cannot result in additional TBoxes to become separable. In what follows, we call this property the *monotonicity property* of inseparability.

In the same way that logical equivalence can be decomposed into two implications, inseparability can be decomposed into two Σ-*entailments* [13], defined next. We say that a TBox $\mathcal{T}_1$ Σ-*entails* a TBox $\mathcal{T}_2$, in symbols $\mathcal{T}_1 \models_\Sigma \mathcal{T}_2$, if $\{\mathcal{I}|_\Sigma \mid \mathcal{I} \models \mathcal{T}_1\} \subseteq \{\mathcal{I}|_\Sigma \mid \mathcal{I} \models \mathcal{T}_2\}$. Although we are mainly concerned with Σ-inseparability, it will sometimes be convenient to also use Σ-entailment. The following observation shows that, as far as computational complexity is concerned, there is no difference between the two notions.

Lemma 3. *Σ-entailment and Σ-inseparability are mutually reducible in polynomial time.*

Proof. Obviously, $\mathcal{T}_1$ and $\mathcal{T}_2$ are Σ-inseparable if $\mathcal{T}_1$ Σ-entails $\mathcal{T}_2$ and vice versa.[1] Now assume that we want to decide $\mathcal{T}_1 \models_\Sigma \mathcal{T}_2$. By replacing every non-Σ-symbol X shared by $\mathcal{T}_1$ and $\mathcal{T}_2$ with a fresh symbol X_1 in $\mathcal{T}_1$ and a distinct fresh symbol X_2 in $\mathcal{T}_2$, we can achieve that $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_1) \cap \mathsf{sig}(\mathcal{T}_2)$ without changing the original (non-)Σ-entailment of $\mathcal{T}_2$ by $\mathcal{T}_1$. We then have $\mathcal{T}_1 \models_\Sigma \mathcal{T}_2$ iff $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_1 \cup \mathcal{T}_2$. □

The further discussion of inseparability is guided by three use cases.

Equivalent Replacement. In applications, it can be necessary or beneficial to replace an existing TBox $\mathcal{T}_1$ with a new TBox $\mathcal{T}_2$, for instance with the aim to speed up reasoning when $\mathcal{T}_2$ is significantly smaller than $\mathcal{T}_1$ or when a reference TBox $\mathcal{T}_1$ is updated to a new version $\mathcal{T}_2$. In such a replacement, one would like to ensure some form of equivalence of $\mathcal{T}_1$ and $\mathcal{T}_2$ to guarantee that correctness is not compromised. Σ-inseparability provides a very strong form of equivalence as $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$ guarantees that $\mathcal{T}_1$ can be replaced with $\mathcal{T}_2$ in any application that refers only to symbols from Σ. For example, the main purpose of many bio-medical ontologies is to produce a systematic *concept hierarchy*, the partial order on concept names that is induced by subsumption. When we are only interested in computing the Σ-part of this hierarchy and we know that the TBox $\mathcal{T}_1$ is Σ-inseparable from $\mathcal{T}_2$, then we can classify $\mathcal{T}_2$ instead of $\mathcal{T}_1$ since $\mathcal{T}_1 \models A \sqsubseteq B$ iff $\mathcal{T}_2 \models A \sqsubseteq B$ for all $A, B \in \mathsf{N_C} \cap \Sigma$. Several authors have used this observation to optimise practical reasoning, see for example [22, 23, 24]. However, $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$ does not only mean that the same subsumptions between concept *names* from Σ are entailed: the same is true for composite Σ-concepts formulated in *any* DL. This is a consequence of the following useful characterisation of Σ-inseparability in terms of logical consequence, taken from [13].

Theorem 4. *Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be SO-TBoxes and Σ a signature. The following conditions are equivalent:*

1. *$\mathcal{T}_1$ and $\mathcal{T}_2$ are Σ-inseparable;*
2. *$\mathcal{T}_1 \models \varphi$ iff $\mathcal{T}_2 \models \varphi$ for every a SO-sentence φ with $\mathsf{sig}(\varphi) \subseteq \Sigma$.*

[1] Note that this is a Turing reduction. All complexity classes considered in this paper are closed under Turing reductions.

Proof. (1) ⇒ (2). Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be Σ-inseparable and φ an SO-sentence such that $\mathsf{sig}(\varphi) \subseteq \Sigma$. Then $\mathcal{T}_1 \not\models \varphi$ iff there is a model $\mathcal{I}$ of $\mathcal{T}_1$ with $\mathcal{I} \not\models \varphi$ iff there is a model $\mathcal{J}$ of $\mathcal{T}_2$ with $\mathcal{J} \not\models \varphi$ (use the definition of Σ-inseparability and choose $\mathcal{I}$ (respectively $\mathcal{J}$) such that $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$) iff $\mathcal{T}_2 \not\models \varphi$.

(2) ⇒ (1). Assume (2) holds and let $\mathcal{I}$ be a model of $\mathcal{T}_1$ (for models $\mathcal{J}$ of $\mathcal{T}_2$ we can proceed in the same way). We have to show that there exists a model $\mathcal{J}$ of $\mathcal{T}_2$ such that $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$. To this end, it is sufficient to show that $\mathcal{I} \models \exists S_1' \cdots \exists S_n' \ (\bigwedge_{\alpha \in \mathcal{T}_2} \alpha')$, where $\{S_1, \dots, S_n\} = \mathsf{sig}(\mathcal{T}_2) \setminus \Sigma$, S_i' is a second-variable of the same arity as S_i $(1 \leq i \leq n)$ and α' is the result of replacing in α each S_i with S_i'. We clearly have $\mathcal{T}_2 \models \exists S_1' \cdots \exists S_n' \ (\bigwedge_{\alpha \in \mathcal{T}_2} \alpha')$ and thus (2) yields $\mathcal{T}_1 \models \exists S_1' \cdots \exists S_n' \ (\bigwedge_{\alpha \in \mathcal{T}_2} \alpha')$. Since $\mathcal{I}$ is a model of $\mathcal{T}_1$, we are done. □

As another application in which equivalent replacement is useful, consider *ontology-based data access (OBDA)* where queries are posed against instance data that is stored in an ABox while a TBox is used to enrich the data and to obtain more complete answers [25, 26]. Typical query languages include conjunctive queries, positive existential queries, datalog, and regular path queries. When using a reference ontology such as SNOMED CT in an OBDA application, it is often the case that only a fragment Σ of the overall signature is used in the data and in the query—for instance only symbols that describe patients and findings, but no symbols that concern medical legislation, drug compounds, and so on. As reference ontologies tend to be updated regularly, suppose that the currently used TBox $\mathcal{T}_1$ has to be replaced with a revised version $\mathcal{T}_2$ in which the Σ-part was not changed, formalised by $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$. Then Theorem 4 guarantees that, for any Σ-instance data and for any Σ-query formulated in one of the query languages mentioned above, query answers coincide relative to $\mathcal{T}_1$ and $\mathcal{T}_2$. More details and formal definitions can be found in [13].

Controlled Merging and Import. During ontology development and when customising ontologies for applications, it may be necessary to merge different ontologies into a single one. Then, a major challenge is to control the consequences of the merged ontology and to prevent the merged ontologies from interacting in undesired ways. For example, if an application needs ontologies for two different domains that are formalised by TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ with signatures Σ_1 and Σ_2, then it is natural to take the union $\mathcal{T}_1 \cup \mathcal{T}_2$. To formalise that any interaction between $\mathcal{T}_1$ and $\mathcal{T}_2$ is avoided, one can demand that $\mathcal{T}_1 \cup \mathcal{T}_2$ is Σ_i-inseparable from $\mathcal{T}_i$ for $i = 1, 2$. Interestingly, this can be ensured by demanding that $\mathcal{T}_1$ and $\mathcal{T}_2$ are $\Sigma_1 \cap \Sigma_2$-inseparable. This property of an inseparability relation is called *robustness under joins* [13, 14].

As another example, consider the import of a TBox $\mathcal{T}_{im}$ into a TBox $\mathcal{T}$ during the development of $\mathcal{T}$, with the aim of reusing the existing formalisation for a particular subdomain instead of modelling it from scratch—an operation that is used frequently in the development of thematically broad reference ontologies such as SNOMED CT. As in the previous example, the resulting operation is taking the union $\mathcal{T} \cup \mathcal{T}_{im}$. The difference between merging and import is in the desired interactions: while $\mathcal{T}$ is not supposed to interfere with the meaning of the symbols from $\mathcal{T}_{im}$, $\mathcal{T}$ would usually define new concepts based on the symbols in $\mathcal{T}_{im}$ (cf. Point 2 of Example 2) and thus it *is* expected that importing $\mathcal{T}_{im}$ has an impact on the symbols in $\mathcal{T}$. Using inseparability, we can formalise the resulting requirement as $\mathcal{T} \cup \mathcal{T}_{im} \equiv_\Sigma \mathcal{T}_{im}$, for $\Sigma = \mathsf{sig}(\mathcal{T}_{im})$.

Finally, we consider the case where a TBox $\mathcal{T}_1$ is replaced with a TBox $\mathcal{T}_2$ as in *equivalent replacement* above, but where this happens in the context of a TBox $\mathcal{T}$ into which $\mathcal{T}_1$ was imported. The central observation is that $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$ (for an appropriate Σ) yields a notion of equivalence strong enough to also prevent undesired interactions in the presence of the importing TBox $\mathcal{T}$, provided that $\mathcal{T}$ uses no symbols from $\mathcal{T}_1$ and $\mathcal{T}_2$ except those in Σ. Formally, this can be captured by the following property of *robustness under replacement*, taken from [13]. We state it as a theorem since robustness under replacements will be used in several proofs later on.

Theorem 5 (Robustness under Replacement). *Let $\mathcal{T}$, $\mathcal{T}_1$, and $\mathcal{T}_2$ be SO-TBoxes and Σ a signature. If $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$ and $\mathsf{sig}(\mathcal{T}) \cap \mathsf{sig}(\mathcal{T}_1 \cup \mathcal{T}_2) \subseteq \Sigma$, then $\mathcal{T}_1 \cup \mathcal{T} \equiv_\Sigma \mathcal{T}_2 \cup \mathcal{T}$.*

Proof. Assume that $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$ and $\mathsf{sig}(\mathcal{T}) \cap \mathsf{sig}(\mathcal{T}_1 \cup \mathcal{T}_2) \subseteq \Sigma$. To show $\mathcal{T}_1 \cup \mathcal{T} \equiv_\Sigma \mathcal{T}_2 \cup \mathcal{T}$, let $\mathcal{I}$ be a model of $\mathcal{T}_1 \cup \mathcal{T}$. We have to show that there is a model $\mathcal{J}$ of $\mathcal{T}_2 \cup \mathcal{T}$ such that $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$. By $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$, there exists a model $\mathcal{J}$ of $\mathcal{T}_2$ such that $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$. We may assume w.l.o.g. that, additionally, $\mathcal{J}$ coincides with $\mathcal{I}$ on all symbols that are not in $\mathsf{sig}(\mathcal{T}_2)$. Since $\mathsf{sig}(\mathcal{T}) \cap \mathsf{sig}(\mathcal{T}_2) \subseteq \Sigma$, $\mathcal{J}$ coincides with $\mathcal{I}$ even on $\mathsf{sig}(\mathcal{T})$. It follows that $\mathcal{J}$ is also a model of $\mathcal{T}$, as required. □

Safe Import / Signature Interfaces. In the ontology import scenario discussed above, it is possible that the imported ontology $\mathcal{T}_{im}$ is revised frequently. In this case, one would like to design the importing TBox $\mathcal{T}$ such that *any* TBox $\mathcal{T}_{im}$ can be imported into $\mathcal{T}$ without undesired interaction, as long as the signature of $\mathcal{T}_{im}$ is not changed. Intuitively, $\mathcal{T}$ provides a safe interface for importing ontologies that only share symbols from some fixed signature Σ with $\mathcal{T}$. This idea has led to the definition of safety for a signature in [12]: an SO-TBox $\mathcal{T}$ is *safe for a signature* Σ if $\mathcal{T} \cup \mathcal{T}_{im} \equiv_{\mathsf{sig}(\mathcal{T}_{im})} \mathcal{T}_{im}$ holds for all SO-TBoxes $\mathcal{T}_{im}$ with $\mathsf{sig}(\mathcal{T}) \cap \mathsf{sig}(\mathcal{T}_{im}) \subseteq \Sigma$. Using robustness under replacement, safety can be formulated as inseparability from the empty TBox, eliminating the quantification over TBoxes used in the original definition.

Theorem 6. *An SO-TBox $\mathcal{T}$ is safe for a signature Σ iff $\mathcal{T} \equiv_\Sigma \emptyset$.*

Proof. Assume first that $\mathcal{T} \not\equiv_\Sigma \emptyset$. Then $\mathcal{T} \cup \mathcal{T}_{im} \not\equiv_{\mathsf{sig}(\mathcal{T}_{im})} \mathcal{T}_{im}$ for the trivial TBox $\mathcal{T}_{im} = \{A \sqsubseteq A \mid A \in \Sigma \cap \mathsf{N_C}\} \cup \{\exists r.\top \sqsubseteq \top \mid r \in \Sigma \cap \mathsf{N_R}\}$ with $\mathsf{sig}(\mathcal{T}_{im}) = \Sigma$. Hence $\mathcal{T}$ is not safe for Σ. Now assume $\mathcal{T} \equiv_\Sigma \emptyset$ and let $\mathcal{T}_{im}$ be a finite set of SO-sentences such that $\mathsf{sig}(\mathcal{T}) \cap \mathsf{sig}(\mathcal{T}_{im}) \subseteq \Sigma$. Then $\mathcal{T} \cup \mathcal{T}_{im} \equiv_{\mathsf{sig}(\mathcal{T}_{im})} \mathcal{T}_{im}$ follows from robustness under replacement. □

Theorem 6 paves the way towards deciding whether a TBox $\mathcal{T}$ is safe for a signature Σ, by checking Σ-inseparability from the empty TBox. We will thus study the latter problem as an important special case of Σ-inseparability, with deciding safety and Theorem 6 as our main motivation.

Example 7. To illustrate the notion of safety, we consider a scenario inspired by [12]. An ontology engineer wants to build an ontology on scientific projects and intends to import the concept names in the signature $\Sigma = \{\mathsf{Nasopharyngitis}, \mathsf{Cystic\_Fibrosis}\}$ from an ontology on medical terms. Statements that could be part of the project ontology are

$$\begin{array}{rcl}
\mathsf{Nasopharyngitis\_Project} & \equiv & \mathsf{Project} \sqcap \exists \mathsf{has\_Focus}.\mathsf{Nasopharyngitis} \\
\mathsf{Cystic\_Fibrosis\_Project} & \equiv & \mathsf{Project} \sqcap \exists \mathsf{has\_Focus}.\mathsf{Cystic\_Fibrosis} \\
\mathsf{Project} & \sqsubseteq & \exists \mathsf{has\_Focus}.\mathsf{Disease} \\
\mathsf{Disease\_of\_HighPriority} & \sqsubseteq & \mathsf{Disease} \sqcap \exists \mathsf{has\_Focus}^-.\mathsf{Project} \\
\mathsf{Cystic\_Fibrosis} & \sqsubseteq & \mathsf{Disease\_of\_HighPriority} \\
\mathsf{Nasopharyngitis} & \sqsubseteq & \mathsf{Disease}
\end{array}$$

This TBox if safe for Σ. To see this, let $\mathcal{I}$ be any interpretation. Define $\mathcal{J}$ by interpreting $\mathsf{Nasopharyngitis}$ and $\mathsf{Cystic\_Fibrosis}$ as in $\mathcal{I}$, selecting some $d \in \Delta^{\mathcal{I}}$, and setting

$$\begin{array}{rcl}
\mathsf{Disease\_of\_HighPriority}^{\mathcal{J}} & = & \{d\} \cup \mathsf{Cystic\_Fibrosis}^{\mathcal{I}} \cup \mathsf{Nasopharyngitis}^{\mathcal{I}} \\
\mathsf{Disease}^{\mathcal{J}} & = & \{d\} \cup \mathsf{Cystic\_Fibrosis}^{\mathcal{I}} \cup \mathsf{Nasopharyngitis}^{\mathcal{I}} \\
\mathsf{Project}^{\mathcal{I}} & = & \{d\} \\
\mathsf{has\_focus}^{\mathcal{I}} & = & \{d\} \times \mathsf{Disease}^{\mathcal{I}}
\end{array}$$

and finally interpreting $\mathsf{Genetic\_Disorder\_Project}$ and $\mathsf{Cystic\_Fibrosis\_Project}$ according to their definitions:

$$\begin{array}{rcl}
\mathsf{Nasopharyngitis\_Project}^{\mathcal{J}} & = & (\mathsf{Project} \sqcap \exists \mathsf{has\_Focus}.\mathsf{Nasopharyngitis})^{\mathcal{J}} \\
\mathsf{Cystic\_Fibrosis\_Project}^{\mathcal{J}} & = & (\mathsf{Project} \sqcap \exists \mathsf{has\_Focus}.\mathsf{Cystic\_Fibrosis})^{\mathcal{J}}
\end{array}$$

Then $\mathcal{J}$ is a model of $\mathcal{T}$ and $\mathcal{I}|_{\Sigma} = \mathcal{J}|_{\Sigma}$, as required.

We consider two natural extensions of $\mathcal{T}$ that are not safe for Σ. First, after adding the concept inclusion

$$\mathsf{Nasopharyngitis} \sqsubseteq \neg \mathsf{Disease\_of\_HighPriority}$$

to $\mathcal{T}$ the resulting TBox $\mathcal{T}'$ implies the disjointness of the imported concepts $\mathsf{Cystic\_Fibrosis}$ and $\mathsf{Nasopharyngitis}$ and is, therefore, not safe for Σ. More formally, there does not exist any model $\mathcal{J}$ of $\mathcal{T}'$ that coincides with the interpretation $\mathcal{I}$ defined by $\Delta^{\mathcal{I}} = \{d\}$ and $\mathsf{Cystic\_Fibrosis}^{\mathcal{I}} = \mathsf{Nasopharyngitis}^{\mathcal{I}} = \{d\}$ on Σ.

Second, by adding the disjointness condition

$$\mathsf{Nasopharyngitis\_Project} \sqcap \mathsf{Cystic\_Fibrosis\_Project} \sqsubseteq \bot$$

to $\mathcal{T}$ one obtains a TBox $\mathcal{T}''$ which is not safe for Σ. This can be seen by showing that there does not exist a model of $\mathcal{T}''$ that coincides with the interpretation $\mathcal{I}$ with $\Delta^{\mathcal{I}} = \{d\}$ and $\mathsf{Nasopharyngitis}^{\mathcal{I}} = \mathsf{Cystic\_Fibrosis}^{\mathcal{I}} = \{d\}$ on Σ.

4. Deciding Inseparability in $\mathcal{ALC}$ and $\mathcal{ALCI}$

We study inseparability as a decision problem for $\mathcal{ALC}$ and $\mathcal{ALCI}$: given TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ and a signature Σ, decide whether $\mathcal{T}_1 \equiv_{\Sigma} \mathcal{T}_2$. Since this

	acyclic $\mathcal{EL}/\mathcal{ELI}$	general $\mathcal{EL}/\mathcal{ELI}$	acyclic and general $\mathcal{ALC}/\mathcal{ALCI}$
$\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$	?	undecidable (Thm. 15)	undecidable (Thm. 8)
$\mathcal{T} \equiv_\Sigma \emptyset$	PTime (Thm. 25)	undecidable (Thm. 15)	undecidable (Thm. 8)
$\mathcal{T}_1 \equiv_{\Sigma \cap \mathsf{N_C}} \mathcal{T}_2$	?	coNExp$^{\text{NP}}$-com. (Thm. 16)	coNExp$^{\text{NP}}$-com. (Thms. 9 and 10)
$\mathcal{T} \equiv_{\Sigma \cap \mathsf{N_C}} \emptyset$	PTime (Thm. 25/16)	PTime (Thm. 16)	Π_2^p-complete (Thm. 14)

Table 1: Results of Section 4

general version of the problem turns out to be undecidable even for acyclic $\mathcal{ALC}$-TBoxes, we also consider three more restricted versions of the original problem: (i) The case where Σ contains only concept names, but no role names. We argue in Section 4.2 that this case captures several relevant applications. We exhibit an interesting connection to DLs with circumscription [27, 28] and use it to show coNExp$^{\text{NP}}$-completeness for the inseparability of acyclic and general TBoxes formulated in $\mathcal{ALC}$ and $\mathcal{ALCI}$.[2] (ii) The case where $\mathcal{T}_2 = \emptyset$, which corresponds to deciding *safety for a signature* as discussed in the previous section, and which is still undecidable for acyclic $\mathcal{ALC}$-TBoxes. We also consider (iii) the combination of the cases (i) and (ii), which turns out to be Π_2^p-complete for acyclic and general TBoxes formulated in $\mathcal{ALC}$ and $\mathcal{ALCI}$. The results obtained in this and the subsequent section are summarised in Table 1 where $\equiv_{\Sigma \cap \mathsf{N_C}}$ means that Σ contains only concept names.

We briefly comment on the two open problems in Table 1: nothing is known about the complexity of deciding $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$ if $\mathcal{T}_1$ and $\mathcal{T}_2$ are acyclic $\mathcal{EL}$ or $\mathcal{ELI}$-TBoxes. For all we know, the problem could be undecidable (as in the case of general $\mathcal{EL}$-TBoxes) or in PTime (which holds if one TBox is empty). If Σ is a concept signature, then a coNExp$^{\text{NP}}$ upper bound follows from the corresponding upper bounds for $\mathcal{ALCI}$-TBoxes but no non-trivial lower bound is known.

[2] Interestingly, the import-by-query approach presented in [29, 30] shows a similar drop in complexity when signatures consisting of concept names only are considered.

4.1. The General Case

We start with the fundamental observation that inseparability is undecidable even when $\mathcal{T}_1$ is an acyclic $\mathcal{ALC}$-TBox and $\mathcal{T}_2$ is empty.

Theorem 8. *Given an acyclic $\mathcal{ALC}$-TBox $\mathcal{T}$ and a signature Σ, it is undecidable whether $\mathcal{T} \equiv_\Sigma \emptyset$.*

Proof. The proof is by reduction of the *universal consistency problem* in bimodal logic [31]. The DL version of this problem can be stated as follows. Fix two role names r_1 and r_2. A *frame* is a structure $\mathcal{F} = (\Delta^\mathcal{F}, r_1^\mathcal{F}, r_2^\mathcal{F})$ with $\Delta^\mathcal{F}$ a non-empty domain and $r_i^\mathcal{F} \subseteq \Delta^\mathcal{F} \times \Delta^\mathcal{F}$ for $i \in \{1, 2\}$. An interpretation $\mathcal{I}$ is *based on* $\mathcal{F}$ if $\Delta^\mathcal{I} = \Delta^\mathcal{F}$ and $r_i^\mathcal{I} = r_i^\mathcal{F}$ for $i \in \{1, 2\}$. In other words, $\mathcal{I}$ is based on $\mathcal{F}$ iff $\mathcal{I}|_\Sigma = \mathcal{F}$ for $\Sigma = \{r_1, r_2\}$. We say that an $\mathcal{ALC}$-concept C is *valid* on $\mathcal{F}$ and write $\mathcal{F} \models C$ if $C^\mathcal{I} = \Delta^\mathcal{I}$ for every interpretation $\mathcal{I}$ based on $\mathcal{F}$. Then C is *universally consistent* if there is a frame $\mathcal{F}$ with $\mathcal{F} \models C$. The universal consistency problem is to decide, given an $\mathcal{ALC}$-concept C with $\mathsf{sig}(C) \subseteq \mathsf{N_C} \cup \{r_1, r_2\}$, whether C is universally consistent. By a result of S. Thomason, this problem is undecidable; see Theorem 1(b) in [31].

For the reduction, let C be an $\mathcal{ALC}$-concept with $\mathsf{sig}(C) \subseteq \mathsf{N_C} \cup \{r_1, r_2\}$. Fix an $A \in \mathsf{N_C}$ that does not occur in C and an $u \in \mathsf{N_R}$ distinct from r_1, r_2.

Claim. C is universally consistent iff $\mathcal{T} \not\equiv_\Sigma \emptyset$, where $\mathcal{T} = \{A \sqsubseteq \exists u.\neg C\}$ and $\Sigma = \{A, r_1, r_2\}$.

Assume first that C is universally consistent. Then there is a frame $\mathcal{F}$ with $\mathcal{F} \models C$. To show that $\mathcal{T} \not\equiv_\Sigma \emptyset$, we have to find an interpretation $\mathcal{J}$ such that for all interpretations $\mathcal{I}$ with $\mathcal{I} \equiv_\Sigma \mathcal{J}$, we have $\mathcal{I} \not\models \mathcal{T}$. Choose as $\mathcal{J}$ any interpretation based on $\mathcal{F}$ such that $A^\mathcal{J} \neq \emptyset$ and let $\mathcal{I}$ be such that $\mathcal{I} \equiv_\Sigma \mathcal{J}$. Then $\mathcal{I}$ is based on $\mathcal{F}$ and thus $C^\mathcal{I} = \Delta^\mathcal{I}$ and $(\exists u.\neg C)^\mathcal{I} = \emptyset$. We also have $A^\mathcal{I} \neq \emptyset$, and consequently $\mathcal{I} \not\models \mathcal{T}$ as required.

Conversely, suppose that $\mathcal{T} \not\equiv_\Sigma \emptyset$. Then there is an interpretation $\mathcal{J}$ such that $\mathcal{I} \not\models \mathcal{T}$ for all $\mathcal{I}$ with $\mathcal{I} \equiv_\Sigma \mathcal{J}$. Let $\mathcal{F}$ be the frame on which $\mathcal{J}$ is based. To prove that C is universally consistent, we show that $\mathcal{F} \models C$. Assume to the contrary of what is to be shown that there is an interpretation $\mathcal{I}'$ based on $\mathcal{F}$ with $C^{\mathcal{I}'} \neq \Delta^\mathcal{F}$. Let $\mathcal{I}$ be identical to $\mathcal{I}'$ except that $A^\mathcal{I} = A^\mathcal{J}$ and $u^\mathcal{I} = \Delta^\mathcal{F} \times \Delta^\mathcal{F}$. Obviously, $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$. Moreover, $\mathcal{I}$ is a model of $\mathcal{T}$: it suffices to note that we have $(\neg C)^\mathcal{I} = (\neg C)^{\mathcal{I}'} \neq \emptyset$ and thus $(\exists u.\neg C)^\mathcal{I} = \Delta^\mathcal{F}$. This is a contradiction to the choice of $\mathcal{I}$. □

4.2. Concept Signatures

We now consider the case where the signature Σ contains only concept names, but no role names. From now on, we will call such a signature Σ a *concept signature*. This case is of interest for two reasons. First, it lays the technical foundations for the results about ontology modules obtained in Sections 6 and 7. And second, it covers relevant special cases of ontology merging and import. For ontology merging recall that, by robustness under joins, $\mathcal{T}_1 \cup \mathcal{T}_2$ is $\mathsf{sig}(\mathcal{T}_i)$-inseparable from $\mathcal{T}_i$ for $i = 1, 2$ if, and only if, $\mathcal{T}_1$ and $\mathcal{T}_2$ are $\mathsf{sig}(\mathcal{T}_1) \cap \mathsf{sig}(\mathcal{T}_2)$-inseparable. If $\mathcal{T}_1$ and $\mathcal{T}_2$ cover different domains, it might well be that they do not share any role names. In this case, the signature $\mathsf{sig}(\mathcal{T}_1) \cap \mathsf{sig}(\mathcal{T}_2)$ relevant for testing inseparability is a concept signature. For safe ontology import, the focus of description logics on modeling concepts rather than roles suggest that importing only concept names is a relevant special case. By Theorem 6, checking safety can in this case be realized by deciding $\mathcal{T} \equiv_\Sigma \emptyset$ with Σ a concept signature.[3]

In the following, we show that deciding Σ-inseparability w.r.t. concept signatures is $\text{coNExp}^{\text{NP}}$-complete for both $\mathcal{ALC}$ and $\mathcal{ALCI}$. Our main technical tool is a close correspondence between Σ-inseparability w.r.t. concept signatures and satisfiability of a certain kind of TBoxes with circumscription [28], which is interesting in its own right. We start with introducing TBoxes with circumscription.

A *circumscribed TBox*[4] $\mathsf{Circ}_{M,F}(\mathcal{T})$ consists of a (general) TBox $\mathcal{T}$ and disjoint finite sets of concept names M and F, where M identifies those concept names that are circumscribed (i.e., whose extension is minimised) and F identifies those concept names whose extension must remain fixed during circumscription. The extension of all other symbols, including those of all role names, can be varied freely during circumscription. To formally define the semantics of circumscribed TBoxes, we define a preference relation $<_{M,F}$ on interpretations by setting $\mathcal{I} <_{M,F} \mathcal{I}'$ if $\Delta^\mathcal{I} = \Delta^{\mathcal{I}'}$, $A^\mathcal{I} = A^{\mathcal{I}'}$ for all $A \in F$, $A^\mathcal{I} \subseteq A^{\mathcal{I}'}$ for all $A \in M$, and there exists an $A \in M$ such that $A^\mathcal{I} \subsetneq A^{\mathcal{I}'}$. An interpretation $\mathcal{I}$ is a *model* of $\mathsf{Circ}_{M,F}(\mathcal{T})$ if $\mathcal{I}$ is a model of $\mathcal{T}$ and there is no model $\mathcal{I}'$ of $\mathcal{T}$ with $\mathcal{I}' <_{M,F} \mathcal{I}$. A concept C is *satisfiable*

[3] Note that, by definition of safety, a *concrete* import TBox $\mathcal{T}_{im}$ should not share any non-Σ-symbols with the importing TBox $\mathcal{T}$, including role names. In practical cases this can be attainable by renaming roles in $\mathcal{T}_{im}$ [29].

[4] Called *simple concept circumscribed TBox* in [28].

w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$ if there is a model $\mathcal{I}$ of $\mathsf{Circ}_{M,F}(\mathcal{T})$ with $C^{\mathcal{I}} \neq \emptyset$. It is shown in [28] that satisfiability of concepts w.r.t. circumscribed TBoxes formulated in $\mathcal{ALC}$ or $\mathcal{ALCI}$ is $\textsc{NExp}^{\text{NP}}$-complete.

We establish an upper bound for the inseparability of general $\mathcal{ALCI}$-TBoxes w.r.t. concept signatures in Theorem 9 and a corresponding lower bound for acyclic $\mathcal{ALC}$-TBoxes in Theorem 10. Together, the proofs of these theorems establish the announced connection between inseparability and circumscription.

Theorem 9. *Given $\mathcal{ALCI}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ and a concept signature Σ, it is in* $\textsc{coNExp}^{\text{NP}}$ *to decide whether $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$.*

Proof. By Lemma 3, it suffices to give a polynomial reduction of Σ-entailment for $\mathcal{ALCI}$-TBoxes to concept unsatisfiability w.r.t. circumscribed $\mathcal{ALCI}$-TBoxes. Assume that we want to decide whether $\mathcal{T}_1 \models_\Sigma \mathcal{T}_2$ with $\Sigma \subseteq \mathsf{N_C}$. We may assume that $\mathcal{T}_2 = \{C \equiv \top\}$, for some $\mathcal{ALCI}$-concept C, and that $\mathsf{sig}(\mathcal{T}_1) \cap \mathsf{sig}(\mathcal{T}_2) \subseteq \Sigma$ (see proof of Lemma 3). Set $M = \{A\}$ with A a fresh concept name, $F = \Sigma$, and $\mathcal{T} = \mathcal{T}_1 \cup \{A \equiv \neg C\}$.

Claim. $\mathcal{T}_1 \models_\Sigma \mathcal{T}_2$ iff A is unsatisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$.

Proof of Claim. Assume that A is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$ and let $\mathcal{I}$ be a model of $\mathsf{Circ}_{M,F}(\mathcal{T})$ with $A^{\mathcal{I}} \neq \emptyset$. Then $\mathcal{I}$ is a model of $\mathcal{T}_1$. To show that $\mathcal{T}_1 \not\models_\Sigma \mathcal{T}_2$, it is enough to show that there is no model $\mathcal{J}$ of $\mathcal{T}_2$ with $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. Assume to the contrary that there is such a $\mathcal{J}$. Since $\mathsf{sig}(\mathcal{T}_1) \cap \mathsf{sig}(\mathcal{T}_2) \subseteq \Sigma$ and $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$, we can take the $\Sigma \cup \mathsf{sig}(\mathcal{T}_1)$-part of $\mathcal{I}$ and the $\Sigma \cup \mathsf{sig}(\mathcal{T}_2)$-part of $\mathcal{J}$ to build a new interpretation $\mathcal{J}'$ that is a model of both $\mathcal{T}_1$ and $\mathcal{T}_2$. Since A does not occur in Σ, $\mathcal{T}_1$, and $\mathcal{T}_2$, we can further assume $A^{\mathcal{J}'} = \emptyset$. Since $\mathcal{J}'$ is a model of $\mathcal{T}_2$, we have $C^{\mathcal{J}'} = \Delta^{\mathcal{J}'}$, thus $\mathcal{J}'$ is not only a model of $\mathcal{T}_1$, but also of $\mathcal{T}$. It can be checked that $\mathcal{J}' <_{M,F} \mathcal{I}$, in contradiction to $\mathcal{I}$ being a model of $\mathsf{Circ}_{M,F}(\mathcal{T})$.

Conversely, assume that $\mathcal{T}_1$ does not Σ-entail $\mathcal{T}_2$. Let $\widehat{\mathcal{I}}$ be a model of $\mathcal{T}_1$ such that there is no model $\mathcal{J}$ of $\mathcal{T}_2$ with $\mathcal{J}|_\Sigma = \widehat{\mathcal{I}}|_\Sigma$. Using filtration, we show in Appendix A that $\widehat{\mathcal{I}}$ can be assumed to be finite. Consider the class of models

$$\mathfrak{J} := \{\mathcal{I} \mid \mathcal{I} \models \mathcal{T}_1,\ \Delta^{\mathcal{I}} = \Delta^{\widehat{\mathcal{I}}},\ \text{and } \widehat{\mathcal{I}}|_\Sigma = \mathcal{I}|_\Sigma\}.$$

Since A does not occur in Σ, $\mathcal{T}_1$, and $\mathcal{T}_2$, we may assume that $A^{\mathcal{I}} = \neg C^{\mathcal{I}}$ for all $\mathcal{I} \in \mathfrak{J}$. Thus, all interpretations in $\mathfrak{J}$ are models of $\mathcal{T}$. Moreover, no interpretation $\mathcal{I} \in \mathfrak{I}$ is a model of $\mathcal{T}_2$, thus $C \neq \Delta^{\mathcal{I}}$ which implies $A^{\mathcal{I}} \neq \emptyset$. Since $\widehat{\mathcal{I}}$ is finite, we find an $\mathcal{I} \in \mathfrak{I}$ such that $A^{\mathcal{I}}$ is minimal, i.e., there is no $\mathcal{I}' \in \mathfrak{I}$ with $A^{\mathcal{I}'} \subsetneq A^{\mathcal{I}}$. It can be verified that there is no model $\mathcal{I}''$ of $\mathcal{T}$ with $\mathcal{I}' <_{M,F} \mathcal{I}$ (any such model would have to be in $\mathfrak{I}$ and contradict the choice of $\mathcal{I}$). Hence, $\mathcal{I}$ witnesses that A is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$. □

We now establish a matching lower bound. It applies to acyclic $\mathcal{ALC}$-TBoxes, and even to the case where the *union* of the input TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ is acyclic. Note that this need not be the case even if $\mathcal{T}_1$ and $\mathcal{T}_2$ are both acyclic by themselves. Acyclicity of $\mathcal{T}_1 \cup \mathcal{T}_2$ will play a role in the proof of Theorem 33 below.

Theorem 10. *Given acyclic $\mathcal{ALC}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ and a concept signature Σ, it is* co$\textsc{NExp}^{\text{NP}}$*-hard to decide whether $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$. This is even true when $\mathcal{T}_1 \cup \mathcal{T}_2$ is acyclic.*

Proof. We only treat general $\mathcal{ALC}$-TBoxes here. The proof for $\mathcal{T}_1 \cup \mathcal{T}_2$ acyclic is significantly more technical and deferred to the appendix.

By Lemma 3, it is sufficient to give a polynomial reduction of concept satisfiability w.r.t. circumscribed $\mathcal{ALC}$-TBoxes to the complement of Σ-entailment. Assume that we want to decide whether an $\mathcal{ALC}$-concept C is satisfiable w.r.t. a circumscribed TBox $\mathsf{Circ}_{M,F}(\mathcal{T})$, where $\mathcal{T}$ is also formulated in $\mathcal{ALC}$. Let

$$\mathcal{T}_1 = \mathcal{T} \cup \{\top \sqsubseteq \exists \mathsf{aux}.C\},$$

with aux a fresh role name. Note that C is satisfiable w.r.t. the non-circumscribed TBox $\mathcal{T}$ iff $\mathcal{T}_1$ has a model. We shall introduce a second TBox $\mathcal{T}_2$ and inseparability-signature Σ to simulate circumscription, i.e., we want to achieve that $\mathcal{T}_1 \not\models_\Sigma \mathcal{T}_2$ iff $\mathcal{T}_1$ is satisfiable in a model that is minimal w.r.t. $<_{M,F}$. Set $\Sigma = M \cup F$. To construct $\mathcal{T}_2$, introduce for each symbol $X \in \mathsf{sig}(\mathcal{T}) \cup M \cup F$ a primed copy X' of X and denote by $\mathcal{T}'$ the TBox resulting from $\mathcal{T}$ when every occurrence of X is replaced with X'. Take a fresh role name aux' and define $\mathcal{T}_2$ by taking the union of

- $\mathcal{T} \cup \mathcal{T}'$
- $A' \sqsubseteq A$, for every $A \in M$;
- $A' \equiv A$, for every $A \in F$;

- $\top \sqsubseteq \exists \mathsf{aux}'. \bigsqcup_{A \in M}(A \sqcap \neg A')$.

Claim. C is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$ iff $\mathcal{T}_1$ does not Σ-entail $\mathcal{T}_2$.

Proof of Claim. Assume that C is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$ and let $\mathcal{I}$ be a model of $\mathsf{Circ}_{M,F}(\mathcal{T})$ with $C^{\mathcal{I}} \neq \emptyset$. By setting $\mathsf{aux}^{\mathcal{I}} = \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, we may assume that $\mathcal{I}$ is a model of $\mathcal{T}_1$. We prove that there is no model $\mathcal{J}$ of $\mathcal{T}_2$ with $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. Assume to the contrary that there is such a $\mathcal{J}$. Let $\mathcal{J}'$ be obtained from $\mathcal{J}$ by setting $X^{\mathcal{J}'} = (X')^{\mathcal{J}}$, for all $X \in \mathsf{sig}(\mathcal{T}) \cup M \cup F$. We show that $\mathcal{J}' <_{M,F} \mathcal{I}$:
(i) For $A \in M$, we have $A^{\mathcal{J}'} = (A')^{\mathcal{J}} \subseteq A^{\mathcal{J}} = A^{\mathcal{I}}$, with the inclusion due to $A' \sqsubseteq A \in \mathcal{T}_2$ and the last equation due to $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$ and $A \in \Sigma$.
(ii) For $A \in F$, we have $A^{\mathcal{J}'} = (A')^{\mathcal{J}} = A^{\mathcal{J}} = A^{\mathcal{I}}$.
(iii) There is an $A \in M$ with $A^{\mathcal{J}'} \subsetneq A^{\mathcal{I}}$ since $\top \sqsubseteq \exists \mathsf{aux}'. \bigsqcup_{A \in M}(A \sqcap \neg A') \in \mathcal{T}_2$.
In summary, $\mathcal{J}' <_{M,F} \mathcal{I}$. Moreover, $\mathcal{J}'$ is a model of $\mathcal{T}$ since $\mathcal{J}$ is a model of $\mathcal{T}'$. We have thus obtained a contradiction to $\mathcal{I}$ being a model of $\mathsf{Circ}_{M,F}(\mathcal{T})$.

Conversely, assume that $\mathcal{T}_1 \not\models_\Sigma \mathcal{T}_2$. Take a model $\mathcal{I}$ of $\mathcal{T}_1$ such that there is no model $\mathcal{J}$ of $\mathcal{T}_2$ with $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. Then $C^{\mathcal{I}} \neq \emptyset$. To show that C is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$, it thus suffices to show that there is no model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{J} <_{M,F} \mathcal{I}$. Assume to the contrary that there is such a $\mathcal{J}$. Consider the interpretation $\mathcal{J}'$ defined by setting $X^{\mathcal{J}'} = X^{\mathcal{I}}$ and $(X')^{\mathcal{J}'} = X^{\mathcal{J}}$, for every $X \in \mathsf{sig}(\mathcal{T}) \cup M \cup F$. It can be checked that $\mathcal{J}'$ is a model of $\mathcal{T}_2$ such that $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$, in contradiction to the non-existence of such a model. □

4.3. Concept Signatures and the Empty TBox

We now consider deciding whether $\mathcal{T} \equiv_\Sigma \emptyset$ for a concept signature Σ and an $\mathcal{ALC}$- or $\mathcal{ALCI}$-TBox $\mathcal{T}$. As mentioned before, this problem corresponds to deciding safety of $\mathcal{T}$ w.r.t. Σ. It turns out that $\mathcal{T} \equiv_\Sigma \emptyset$ is equivalent to the property that, for every interpretation $\mathcal{I}$ *of cardinality one*, there is a model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. This suggests a Π_2^p-upper bound for deciding $\mathcal{T} \equiv_\Sigma \emptyset$, which turns out to be tight. Both the characterisation of inseparability in terms of interpretations of cardinality one and the Π_2^p-completeness carry over to FO-TBoxes whose classes of models are closed under disjoint unions.

We start with introducing some preliminaries. A *one-point interpretation* is an interpretation $\mathcal{I}$ with $|\Delta^{\mathcal{I}}| = 1$. A model of $\mathcal{T}$ is a *one-point-model* if it is a one-point interpretation. Let $\mathcal{I}_i, i \in I$ be a family of interpretations. The *disjoint union* $\mathcal{I} = \biguplus_{i \in I} \mathcal{I}_i$ of $\mathcal{I}_i, i \in I$, is defined as

- $\Delta^{\mathcal{I}} = \bigcup_{i \in I} \{i\} \times \Delta^{\mathcal{I}_i}$;
- $A^{\mathcal{I}} = \bigcup_{i \in I} \{i\} \times A^{\mathcal{I}_i}$, for all $A \in \mathsf{N_C}$;
- $r^{\mathcal{I}} = \bigcup_{i \in I} \{((i,x),(i,y)) \mid (x,y) \in r^{\mathcal{I}_i}\}$, for all $r \in \mathsf{N_R}$.

A TBox $\mathcal{T}$ is *preserved under disjoint unions* if the disjoint union of any family of models of $\mathcal{T}$ is again a model of $\mathcal{T}$. A description logic $\mathcal{L}$ is *preserved under disjoint unions* if all $\mathcal{L}$-TBoxes are preserved under disjoint unions. The following is well known (see for example [32]) and even applies to much more expressive DLs such as $\mathcal{SHIQ}$.

Lemma 11. *$\mathcal{ALCI}$ is preserved under disjoint unions.*

The following lemma states the announced characterisation of inseparability from the empty TBox w.r.t. concept signatures in terms of one-point interpretations.

Lemma 12 (One-Point Criterion)**.** *Let $\mathcal{T}$ be an FO-TBox preserved under disjoint unions and Σ a concept signature. Then $\mathcal{T} \equiv_\Sigma \emptyset$ iff for every one-point interpretation $\mathcal{I}$ there exists a model $\mathcal{J}$ of $\mathcal{T}$ such that $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$.*

Proof. The implication from left to right is trivial. For the converse direction, assume that for every one-point interpretation $\mathcal{I}$ there exists a model $\mathcal{J}$ of $\mathcal{T}$ such that $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. We have to show that, then, the same holds for *any* interpretation $\mathcal{I}$. This can be done as follows. For every $d \in \Delta^{\mathcal{I}}$, let $\mathcal{I}_d$ be the restriction of $\mathcal{I}$ to the singleton domain $\{d\}$: $\Delta^{\mathcal{I}_d} = \{d\}$, $A^{\mathcal{I}_d} = \{d \mid d \in A^{\mathcal{I}}\}$ for all $A \in \mathsf{N_C}$ and $r^{\mathcal{I}_d} = \{(d,d) \mid (d,d) \in r^{\mathcal{I}}\}$ for all $r \in \mathsf{N_R}$. For every $d \in \Delta^{\mathcal{I}}$, by assumption there is a model $\mathcal{J}_d$ of $\mathcal{T}$ with $\mathcal{J}_d|_\Sigma = \mathcal{I}_d|_\Sigma$. Let $\mathcal{J} = \biguplus_{d \in \Delta^{\mathcal{I}}} \mathcal{J}_d$. By preservation under disjoint unions $\mathcal{J}$ is a model of $\mathcal{T}$. We have $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$ because Σ consists of concept names only. □

Note that Lemma 12 fails if Σ contains a role name or we want to check inseparability from unrestricted TBoxes rather than from the empty one. The former is illustrated by the following example.

Example 13. *Consider the $\mathcal{EL}$-TBox $\mathcal{T} = \{\exists r.A \sqsubseteq A\}$ and $\Sigma = \{A, r\}$. For every one-point interpretation $\mathcal{I}$ we have $\mathcal{I} \models \mathcal{T}$. However, for the interpretation $\mathcal{J}$ with $\Delta^{\mathcal{J}} = \{d_1, d_2\}$, $r^{\mathcal{J}} = \{(d_1, d_2)\}$ and $A^{\mathcal{J}} = \{d_2\}$, we have $\mathcal{J} \not\models \mathcal{T}$. Since $\mathcal{T}$ does not contain any non-Σ symbols, there is no $\mathcal{J}'$ such that $\mathcal{J}|_\Sigma = \mathcal{J}'|_\Sigma$ and $\mathcal{J}' \models \mathcal{T}$.*

Lemma 12 suggests a straightforward alternating procedure for deciding $\mathcal{T} \equiv_\Sigma \emptyset$ that yields a Π_2^p-upper bound; note that when $\mathcal{I}$ is a singleton and $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$, then $\mathcal{J}$ must also be a singleton, so all quantification in Lemma 12 is over objects of polynomial size. A matching lower bound can be established by reduction of $\forall\exists$-QBF, i.e., the validity of QBF formulas of the form $\forall\vec{q}\exists\vec{p}\varphi$, which is well-known to be Π_2^p-complete. In the proof of the following theorem, we actually show the upper bound by reduction to $\forall\exists$-QBF instead of by an alternating procedure. This paves the way to using highly optimised QBF solvers [33] for deciding $\mathcal{T} \equiv_\Sigma \emptyset$.

Theorem 14. *Given an $\mathcal{ALCI}$-TBox $\mathcal{T}$ and a concept signature Σ, it is in Π_2^p to decide $\mathcal{T} \equiv_\Sigma \emptyset$. The same problem is Π_2^p-hard for acyclic $\mathcal{ALC}$-TBoxes.*

Proof. We start with the upper bound. Assume we want to decide $\mathcal{T} \equiv_\Sigma \emptyset$ with $\mathcal{T}$ an $\mathcal{ALCI}$-TBox and Σ a concept signature. Take a propositional variables p_A for each concept name $A \in \Sigma$ and a (distinct) propositional variable q_X for each symbol $X \in \mathsf{sig}(\mathcal{T}) \setminus \Sigma$. Translate concepts D in the signature $\mathsf{sig}(\mathcal{T})$ into propositional formulas $D^\dagger$ by setting

$$\begin{array}{rcll} A^\dagger & = & p_A & \text{for all } A \in \Sigma \cap \mathsf{N_C} \\ A^\dagger & = & q_A & \text{for all } A \in (\mathsf{sig}(\mathcal{T}) \setminus \Sigma) \cap \mathsf{N_C} \\ (D_1 \sqcap D_2)^\dagger & = & D_1^\dagger \wedge D_2^\dagger & \\ (\neg D)^\dagger & = & \neg D^\dagger & \\ (\exists r.D)^\dagger & = & q_r \wedge D^\dagger & \text{for all } r \in \mathsf{sig}(\mathcal{T}) \cap \mathsf{N_R} \\ (\exists r^-.D)^\dagger & = & q_r \wedge D^\dagger & \text{for all } r \in \mathsf{sig}(\mathcal{T}) \cap \mathsf{N_R} \end{array}$$

Note that if $\mathcal{I}$ is a one-point interpretation with $\Delta^\mathcal{I} = \{d\}$ and v is a propositional truth assignment such that

- $d \in A^\mathcal{I}$ iff $v(p_A) = 1$, for all $A \in \mathsf{sig}(\mathcal{T}) \cap \Sigma \cap \mathsf{N_C}$,
- $d \in A^\mathcal{I}$ iff $v(q_A) = 1$, for all $A \in (\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_C}) \setminus \Sigma$, and
- $(d, d) \in r^\mathcal{I}$ iff $v(q_r) = 1$, for all $r \in (\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_C})$,

then $d \in D^\mathcal{I}$ iff $v(D^\dagger) = 1$ holds for all $\mathcal{ALCI}$-concepts D over $\mathsf{sig}(\mathcal{T})$. Now let

$$\mathcal{T}^\dagger = \bigwedge_{C \sqsubseteq D \in \mathcal{T}} C^\dagger \to D^\dagger \wedge \bigwedge_{C \equiv D \in \mathcal{T}} C^\dagger \leftrightarrow D^\dagger$$

and let $\vec{p}$ denote the sequence of variables p_A, $A \in \Sigma$, and $\vec{q}$ denote the sequence of variables q_X, $X \in \mathsf{sig}(\mathcal{T}) \setminus \Sigma$. Then it can be verified that the QBF $\varphi_{\mathcal{T}} := \forall\vec{p}\exists\vec{q}\mathcal{T}^\dagger$ is valid iff for every one-point interpretation $\mathcal{I}$, there is a model $\mathcal{J}$ of $\mathcal{T}$ such that $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$, as required.

For the lower bound, we first reduce $\forall\exists$-QBF to the following problem:

(‡) Given a finite set M of propositional formulas, is $\chi_P = \bigwedge_{\varphi\in P}\varphi \wedge \bigwedge_{\varphi\in M\setminus P}\neg\varphi$ satisfiable for all $P \subseteq M$?

For the reduction, let $\psi = \forall p_1 \cdots \forall p_n \exists q_1 \cdots \exists q_m \varphi$ be a $\forall\exists$-QBF. Then ψ is valid iff (‡) holds for $M = \{p_1, \ldots, p_n, \varphi \wedge p\}$ with p a fresh propositional variable. To prove this assume that ψ is not valid and let $v(p_i) \in \{0, 1\}$ for $1 \leq i \leq n$ be a truth assignment such that there does not exist any truth assignment extending v and satisfying φ. Then χ_P is not satisfiable for $P = \{p_i \mid v(p_i) = 1\} \cup \{p \wedge \varphi\}$. Conversely, assume that χ_P is not satisfiable for some $P \subseteq M$. Then P contains $\varphi \wedge p$. Let $v(p_i) = 1$ iff $p_i \in P$, for $1 \leq i \leq n$. Then there does not exist any extension of v satisfying φ and so ψ is not valid.

Now we reduce (‡) to $\mathcal{T} \equiv_\Sigma \emptyset$. Assume $M = \{\varphi_1, \ldots, \varphi_n\}$ is given. We can regard each φ_i as an $\mathcal{ALC}$-concept C_{φ_i} by replacing each propositional variable p with a concept name A_p. Let $A_1, \ldots, A_n$ be fresh concept names and set $\Sigma = \{A_1, \ldots, A_n\}$ and $\mathcal{T} = \{A_i \equiv C_{\varphi_i} \mid 1 \leq i \leq n\}$. Then, $\chi_P = \bigwedge_{\varphi_i\in P}\varphi_i \wedge \bigwedge_{\varphi_i\in M\setminus P}\neg\varphi_i$ is satisfiable for every $P \subseteq M$ iff $\mathcal{T} \equiv_\Sigma \emptyset$. To prove this, assume that χ_P is not satisfiable for some $P \subseteq M$. Let $\mathcal{I}$ be the interpretation with domain $\Delta^\mathcal{I} = \{d\}$ and $d \in A_i^\mathcal{I}$ iff $\varphi_i \in P$. Then there does not exist any model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$. Conversely, assume that $\mathcal{I}$ is a model of $\mathcal{T}$ such that there does not exist any model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$. We may assume that $\mathcal{I}$ is a one-point interpretation with domain $\Delta^\mathcal{I} = \{d\}$. But then χ_P is not satisfiable for $P = \{\varphi_i \mid d \in A_i^\mathcal{I}\}$. □

Note that the reductions in the proof of Theorem 14 are quite robust under modifications to the description logic used. For example, the Π_2^p-lower bound already holds for acyclic $\mathcal{ALC}$-TBoxes without role names and the Π_2^p-upper bound still holds for very expressive DLs such as $\mathcal{SHIQ}$ and even for FO-TBoxes preserved under disjoint unions.

5. Inseparability in $\mathcal{EL}$ and $\mathcal{ELI}$

We switch from the description logics $\mathcal{ALC}$ and $\mathcal{ALCI}$ to the less expressive variants $\mathcal{EL}$ and $\mathcal{ELI}$ that lack negation, disjunction, and universal quantification. Our first results are that inseparability of $\mathcal{EL}$-TBoxes from the empty TBoxes w.r.t. unrestricted signatures is still undecidable and inseparability of $\mathcal{EL}$- and $\mathcal{ELI}$-TBoxes w.r.t. concept signatures is still $\text{coNExp}^{\text{NP}}$-complete, showing that in these cases the reduction of expressive power does not seem to pay off in terms of reduced complexity. One difference, however, is that in the $\mathcal{EL}$-case, our lower bounds only apply to general TBoxes while they work even for acyclic TBoxes in the $\mathcal{ALC}$-case. We then show that there *is* a complexity reduction for inseparability from the empty TBox when only concept signatures or only acyclic TBoxes are allowed. In the first case, the complexity drops from Π_2^p to PTime and in the second case from undecidable to PTime. Both PTime upper bounds apply to $\mathcal{ELI}$-TBoxes and thus, in all analysed cases, the complexity coincides for $\mathcal{EL}$ and $\mathcal{ELI}$. This is remarkable since the standard reasoning problem of concept subsumption w.r.t. TBoxes is in PTime for $\mathcal{EL}$, while it is ExpTime-complete for general $\mathcal{ELI}$-TBoxes and PSpace-complete for acyclic $\mathcal{ELI}$-TBoxes [15].

5.1. The General Case and Concept Signatures

We start with the announced undecidability result. Note that, in contrast to the case of $\mathcal{ALC}$, the undecidability result for $\mathcal{EL}$ applies only to general TBoxes, but not to acyclic ones. In fact, we leave it open whether inseparability of acyclic $\mathcal{EL}$-TBoxes is decidable.

Theorem 15. *Given an $\mathcal{EL}$-TBox $\mathcal{T}$ and signature Σ, it is undecidable whether $\mathcal{T} \equiv_\Sigma \emptyset$.*

The proof of Theorem 15 is by reduction of the word problem for semigroups and somewhat technical. Details are given in the appendix.

We now consider inseparability w.r.t. concept signatures and show that there is no gain in complexity when replacing $\mathcal{ALC}/\mathcal{ALCI}$ with $\mathcal{EL}/\mathcal{ELI}$ as far as general TBoxes are concerned. Again, the case of acyclic TBoxes is left open.

Theorem 16. *Given $\mathcal{EL}$- or $\mathcal{ELI}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ and concept signatures Σ, it is* $\text{coNExp}^{\text{NP}}$*-complete to decide $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$.*

The upper bound in Theorem 16 is immediate from Theorem 9. The lower bound is proved by a reduction of (a slight variation of) inseparability of $\mathcal{ALC}$-TBoxes w.r.t. concept signatures. The technical details of this reduction are given in the appendix.

5.2. Concept Signatures and the Empty TBox

By Theorem 15, inseparability of $\mathcal{EL}$-TBoxes is still undecidable when one of the TBoxes must be empty. Like in $\mathcal{ALC}/\mathcal{ALCI}$, the situation improves when we allow only concept signatures. In contrast to $\mathcal{ALC}/\mathcal{ALCI}$, we can also attain decidability by replacing general TBoxes with acyclic ones. In both cases, we even obtain PTime complexity. The next theorem analyses the case of concept signatures.

Theorem 17. *Given an $\mathcal{ELI}$-TBox $\mathcal{T}$ and a concept signature Σ, it is in PTime to decide $\mathcal{T} \equiv_\Sigma \emptyset$.*

Proof. Let $\mathcal{T}$ be an $\mathcal{ELI}$-TBox and Σ a concept signature. Consider the $\forall\exists$-QBF $\varphi_\mathcal{T}$ constructed in the proof of Theorem 14. As proved there, we have $\mathcal{T} \equiv_\Sigma \emptyset$ iff $\varphi_\mathcal{T} = \forall\vec{p}\exists\vec{q}\mathcal{T}^\dagger$ is valid. It is easy to check that, when $\mathcal{T}$ is an $\mathcal{ELI}$-TBox, then $\mathcal{T}^\dagger$ is a conjunction of propositional *Horn formulas*, i.e., formulas of the form $w_1 \wedge \cdots \wedge w_k \rightarrow v_1 \wedge \cdots \wedge v_l$ where the w_i and v_i are propositional variables. It remains to recall that the validity of quantified Boolean Horn formulas can be decided in PTime [34].

□

5.3. Acyclic TBoxes and the Empty TBox

We now explore the second way of overcoming undecidability of inseparability of $\mathcal{EL}$- and $\mathcal{ELI}$-TBoxes from the empty TBox: restrict the input to acyclic TBoxes. Note that, in this case, we do *not* limit ourselves to concept signatures. As in the previous section, we obtain a PTime upper bound, though with a different and more subtle approach. The approach is based on a characterisation of $\mathcal{T} \equiv_\Sigma \emptyset$ in terms of certain syntactic and semantic dependencies that will also play a central role when we deal with modules and module extraction in Sections 6 and 7.

We start with introducing syntactic dependencies. The following example shows two cases of how an acyclic $\mathcal{EL}$-TBox can fail to be Σ-inseparable from the empty TBox. These two cases will then give rise to two types of syntactic dependencies.

Example 18. (a) Let $\mathcal{T} = \{A \sqsubseteq \exists r.B\}$ and $\Sigma = \{A, B\}$. Then $\mathcal{T} \not\equiv_\Sigma \emptyset$: for the interpretation $\mathcal{I}$ with $\Delta^\mathcal{I} = \{d\}$, $A^\mathcal{I} = \{d\}$, and $B^\mathcal{I} = \emptyset$, there is no model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$.

(b) Let $\mathcal{T} = \{A_1 \sqsubseteq \exists r.B_1, A_2 \sqsubseteq \exists r.B_2, A \equiv B_1 \sqcap B_2\}$ and $\Sigma = \{A_1, A_2, A\}$. Then $\mathcal{T} \not\equiv_\Sigma \emptyset$: for the interpretation $\mathcal{I}$ with $\Delta^\mathcal{I} = \{d\}$, $A_1^\mathcal{I} = A_2^\mathcal{I} = \{d\}$, and $A^\mathcal{I} = \emptyset$, there is no model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$.

Intuitively, in part (a) of Example 18, the reason for separability from the empty TBox is that one Σ-symbol occurs in the definition of another one; in part (b), this is not the case, but there are interacting definitions of different Σ-concept names. To generalise these examples, we introduce some notation. For an acyclic TBox $\mathcal{T}$,

- $\mathsf{Lhs}(\mathcal{T})$ denotes the set of concept names A such that there is a statement $A \bowtie C \in \mathcal{T}$;
- $\mathsf{Def}(\mathcal{T})$ denotes the set of concept names such that there is a definition $A \equiv C \in \mathcal{T}$;
- $\mathsf{depend}^\equiv_\mathcal{T}(A)$ is defined exactly as $\mathsf{depend}_\mathcal{T}(A)$ (introduced in Section 2), except that only concept definitions $A \equiv C$ are considered while concept inclusions $A \sqsubseteq C$ are disregarded.

Definition 19. Let $\mathcal{T}$ be an acyclic $\mathcal{ELI}$-TBox, Σ a signature, and $A \in \Sigma$. We say that

- *A has a direct Σ-dependency in $\mathcal{T}$* if $\mathsf{depend}_\mathcal{T}(A) \cap \Sigma \neq \emptyset$.
- *A has an indirect Σ-dependency in $\mathcal{T}$* if $A \in \mathsf{Def}(\mathcal{T}) \cap \Sigma$ and there are $A_1, \ldots, A_n \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma$ such that $A \notin \{A_1, \ldots, A_n\}$ and

$$\mathsf{depend}^\equiv_\mathcal{T}(A) \setminus \mathsf{Def}(\mathcal{T}) \subseteq \bigcup_{1 \leq i \leq n} \mathsf{depend}_\mathcal{T}(A_i).$$

We say that $\mathcal{T}$ *contains an (in)direct Σ-dependency* when there is an $A \in \Sigma$ that has an (in)direct Σ-dependency in $\mathcal{T}$.

To understand the "$\setminus\mathsf{Def}(\mathcal{T})$"-part in indirect Σ-dependencies, reconsider part (b) of Example 18, replace $A \equiv B_1 \sqcap B_2$ with $A \equiv B_1 \sqcap B_1' \sqcap B_2$ and $B_1' \equiv B_1$, and observe that $\mathcal{T} \not\equiv_\Sigma \emptyset$ still holds. Note that many A that have a direct Σ-dependency also have an indirect Σ-dependency. The following lemma can be proved by a straightforward generalisation of the arguments in Example 18 and shows that, as expected, syntactic dependencies result in separability from the empty TBox.

Lemma 20. *Let $\mathcal{T}$ be an acyclic $\mathcal{ELI}$-TBox and Σ a signature. If $\mathcal{T}$ contains a direct or indirect syntactic Σ-dependency, then $\mathcal{T} \not\equiv_\Sigma \emptyset$.*

In the remainder of this section, we show that the converse of Lemma 20 is also true and then use this observation to devise the announced PTime decision procedure.

To prove the converse of Lemma 20 (and also to prepare for the use of Σ-dependencies in the context of acyclic $\mathcal{ALCI}$-TBoxes in Section 6), it is convenient to replace indirect Σ-dependencies with the following semantic condition.

Definition 21. Let $\mathcal{T}$ be an acyclic $\mathcal{ELI}$-TBox, Σ a signature. We say that $\mathcal{T}$ contains a *Boolean Σ-constraint* if there is a $P \subseteq \mathsf{Lhs}(\mathcal{T}) \cap \Sigma$ such that the concept

$$C_P = \bigsqcap_{A \in P} A \sqcap \bigsqcap_{A \in (\mathsf{Lhs}(\mathcal{T}) \cap \Sigma) \setminus P} \neg A$$

is not satisfiable in a one-point model of $\mathcal{T}$.

It can be seen that Boolean Σ-constraints are actually not an equivalent replacement of indirect Σ-dependencies. There are direct Σ-dependencies that give rise to Boolean Σ-constraints but do not give rise to indirect Σ-dependencies.

Example 22. *Let $\mathcal{T} = \{A \sqsubseteq \exists r.B, B \sqsubseteq B'\}$ and $\Sigma = \{A, B\}$. Then $\mathcal{T}$ contains a direct Σ-dependency since $B \in \mathsf{depend}_\mathcal{T}(A)$ and $\mathcal{T}$ contains a Boolean Σ-constraint since $A \sqcap \neg B$ is not satisfiable in any one-point model of $\mathcal{T}$. However, $\mathcal{T}$ contains no indirect Σ-dependency since $\mathcal{T}$ contains no concept equations.*

However, we have the following (joint) equivalence.

Lemma 23. *Let $\mathcal{T}$ be an acyclic $\mathcal{ELI}$-TBox and Σ a signature. Then the following are equivalent:*

(a) $\mathcal{T}$ contains neither a direct nor an indirect Σ-dependency;
(b) $\mathcal{T}$ contains neither a direct Σ-dependency nor a Boolean Σ-constraint.

Proof. "(a) ⇒ (b)". Assume that $\mathcal{T}$ contains neither a direct nor an indirect Σ-dependency. We have to show that $\mathcal{T}$ does not contain a Boolean Σ-constraint, i.e., that for every $P \subseteq \mathsf{Lhs}(\mathcal{T}) \cap \Sigma$ the concept C_P from Definition 21 is satisfiable in a one-point model of $\mathcal{T}$. Assume $P \subseteq \mathsf{Lhs}(\mathcal{T}) \cap \Sigma$ is given. We construct a one-point model $\mathcal{I}$ of $\mathcal{T}$ satisfying C_P as follows. Set $\Delta^{\mathcal{I}} = \{d\}$ and for all $r \in \mathsf{N_R}$,

$$r^{\mathcal{I}} = \begin{cases} \{(d,d)\} & : \quad r \in \bigcup_{B \in P} \mathsf{depend}_{\mathcal{T}}(B) \\ \emptyset & : \quad \text{otherwise} \end{cases}$$

The extension $A^{\mathcal{I}}$ of concept names A is defined by induction on the definitorial depth $d_{\mathcal{T}}(A)$ of A in $\mathcal{T}$: if $A \in \mathsf{N_C}$ with $d_{\mathcal{T}}(A) = 0$, then set

$$A^{\mathcal{I}} = \begin{cases} \{d\} & : \quad A \in \bigcup_{B \in P} \mathsf{depend}_{\mathcal{T}}(B) \\ \emptyset & : \quad \text{otherwise} \end{cases}$$

Now let $A \in \mathsf{N_C}$ such that $d_{\mathcal{T}}(A) > 0$ and assume that $B^{\mathcal{I}}$ has already been defined for all concept names B with $d_{\mathcal{T}}(B) < d_{\mathcal{T}}(A)$. Then define $A^{\mathcal{I}}$ as follows:

(1) if $A \in P \cup \bigcup_{B \in P} \mathsf{depend}_{\mathcal{T}}(B)$, then set $A^{\mathcal{I}} = \{d\}$;
(2) otherwise,
  (2.1) if $A \notin \mathsf{Def}(\mathcal{T})$, then set $A^{\mathcal{I}} = \emptyset$;
  (2.2) if $A \in \mathsf{Def}(\mathcal{T})$, then set $A^{\mathcal{I}} = C^{\mathcal{I}}$ for the unique C such that $A \equiv C \in \mathcal{T}$.

Note that $C^{\mathcal{I}}$ is defined in (2.2) since $d_{\mathcal{T}}(B) < d_{\mathcal{T}}(A)$ for all concept names B in C. This finishes the definition of $\mathcal{I}$. We show that $\mathcal{I}$ is a model of $\mathcal{T}$ and satisfies C_P.

Claim 1. $\mathcal{I}$ satisfies $\mathcal{T}$.

To prove Claim 1, let $A \bowtie C \in \mathcal{T}$. We distinguish the following cases:

- $A \in P \cup \bigcup_{B \in P} \mathsf{depend}_{\mathcal{T}}(B)$.

  By definition of $A^{\mathcal{I}}$ and $r^{\mathcal{I}}$, we have $A^{\mathcal{I}} = \{d\}$, $r^{\mathcal{I}} = \{(d,d)\}$ for all $r \in \mathsf{depend}_{\mathcal{T}}(A)$, and $B^{\mathcal{I}} = \{d\}$ for all $B \in \mathsf{depend}_{\mathcal{T}}(A)$. Thus $C^{\mathcal{I}} = \{d\}$ and $A \bowtie C$ is satisfied in $\mathcal{I}$.

- $A \notin P \cup \bigcup_{B \in P} \mathsf{depend}_{\mathcal{T}}(B)$.

  If $\bowtie = \sqsubseteq$, then $A \notin \mathsf{Def}(\mathcal{T})$. By (2.1), $A^{\mathcal{I}} = \emptyset$ and so $\mathcal{I}$ satisfies $A \sqsubseteq C$. If $\bowtie = \equiv$, then $A^{\mathcal{I}} = C^{\mathcal{I}}$ by (2.2), and so $\mathcal{I}$ satisfies $A \equiv C$.

Claim 2. $d \in C_P^{\mathcal{I}}$.

Recall that $C_P = \bigsqcap_{A \in P} A \sqcap \bigsqcap_{A \in (\mathsf{Lhs}(\mathcal{T}) \cap \Sigma) \setminus P} \neg A$. Since $P \subseteq \mathsf{Lhs}(\mathcal{T})$ we have $d_{\mathcal{T}}(A) > 0$ for all $A \in P$. Thus $A^{\mathcal{I}}$ is defined in (1) for all $A \in P$. It follows that $A^{\mathcal{I}} = \{d\}$ for all $A \in P$. Thus, $d \in (\bigsqcap_{A \in P} A)^{\mathcal{I}}$. To show that $d \in (\bigsqcap_{A \in (\mathsf{Lhs}(\mathcal{T}) \cap \Sigma) \setminus P} \neg A)^{\mathcal{I}}$, let $A \in (\mathsf{Lhs}(\mathcal{T}) \cap \Sigma) \setminus P$. We have to show that $A^{\mathcal{I}} = \emptyset$. Since $\mathcal{T}$ does not contain direct Σ-dependencies, $A \notin P \cup \bigcup_{B \in P} \mathsf{depend}_{\mathcal{T}}(B)$. Hence Case (2) of the inductive definition of $A^{\mathcal{I}}$ applies. If Case (2.1) applies then $A^{\mathcal{I}} = \emptyset$, as required. If Case (2.2) applies then let $A \equiv C \in \mathcal{T}$ and $P = \{A_1, \ldots, A_n\}$. Since $\mathcal{T}$ does not contain indirect Σ-dependencies, there is a $B \in \mathsf{depend}_{\mathcal{T}}^{\equiv}(A) \setminus \mathsf{Def}(\mathcal{T})$ such that $B \notin \bigcup_{1 \leq i \leq n} \mathsf{depend}_{\mathcal{T}}(A_i)$. Since $\mathcal{T}$ does not contain direct Σ-dependencies and $A \in (\mathsf{Lhs}(\mathcal{T}) \cap \Sigma)$, we have $B \notin \Sigma$. Thus $B \notin P$ since $P \subseteq \Sigma$. We thus obtain that $B \notin P \cup \bigcup_{B' \in P} \mathsf{depend}_{\mathcal{T}}(B')$ and so $B^{\mathcal{I}} = \emptyset$ follows from the definition of $\mathcal{I}$. Since $\mathcal{I}$ is a model of $\mathcal{T}$, this yields $A^{\mathcal{I}} = \emptyset$, as required.

"(b) $\Rightarrow$ (a)". Assume $\mathcal{T}$ contains neither a direct Σ-dependency nor a Boolean Σ-constraint. We have to show that $\mathcal{T}$ does not contain an indirect Σ-dependency. Assume to the contrary that there are $A_1, \ldots, A_n \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma$ and $A \in \mathsf{Def}(\mathcal{T}) \cap \Sigma$ such that $A \notin \{A_1, \ldots, A_n\}$ and

$$\mathsf{depend}_{\mathcal{T}}^{\equiv}(A) \setminus \mathsf{Def}(\mathcal{T}) \subseteq \bigcup_{1 \leq i \leq n} \mathsf{depend}_{\mathcal{T}}(A_i). \qquad (*)$$

We show that the concept $C_P = A_1 \sqcap \cdots \sqcap A_n \sqcap \neg A$ is not satisfiable in a one-point model $\mathcal{I}$ of $\mathcal{T}$, in contradiction to the fact that $\mathcal{T}$ does not contain a Boolean Σ-constraint. We first note the following observation, which is easy to prove by induction on the definitorial depth of A in $\mathcal{T}$.

Claim 3. For an acyclic $\mathcal{ELI}$-TBox $\mathcal{T}$, one-point model $\mathcal{I}$ of $\mathcal{T}$ and $A \in$

$\mathsf{Def}(\mathcal{T})$ with $A^{\mathcal{I}} = \emptyset$, we have $B^{\mathcal{I}} = \emptyset$ for some $B \in \mathsf{depend}^{\equiv}_{\mathcal{T}}(A) \setminus \mathsf{Def}(\mathcal{T})$.

Now assume that there is a one-point model $\mathcal{I}$ of $\mathcal{T}$ with $\Delta^{\mathcal{I}} = \{d\}$ and $d \in C_P^{\mathcal{I}}$. We show that this leads to a contradiction. From $d \in C_P^{\mathcal{I}}$ we obtain $A^{\mathcal{I}} = \emptyset$. By Claim 3 there exists $B \in \mathsf{depend}^{\equiv}_{\mathcal{T}}(A) \setminus \mathsf{Def}(\mathcal{T})$ such that $B^{\mathcal{I}} = \emptyset$. By $(*)$, we have $B \in \bigcup_{1 \leq i \leq n} \mathsf{depend}_{\mathcal{T}}(A_i)$. On the other hand, from the condition that $\mathcal{I}$ is a model of $\mathcal{T}$ with $d \in A_i^{\mathcal{I}}$ for $1 \leq i \leq n$ we obtain $d \in E^{\mathcal{I}}$ for all $E \in \bigcup_{1 \leq i \leq n} \mathsf{depend}_{\mathcal{T}}(A_i)$. We have derived a contradiction. □

We are now ready to establish the converse of Lemma 20. Since we will use direct Σ-dependencies and Boolean Σ-constraints also in the context of acyclic $\mathcal{ALCI}$-TBoxes in Section 6, we formulate the result for this more general type of TBox.

Lemma 24. *Let $\mathcal{T}$ be an acyclic $\mathcal{ALCI}$-TBox and Σ a signature. If $\mathcal{T}$ contains no direct Σ-dependencies and no Boolean Σ-constraints, then $\mathcal{T} \equiv_\Sigma \emptyset$.*

Proof. Let $\mathcal{T}$ be an acyclic $\mathcal{ALCI}$-TBox containing no direct Σ-dependencies and no Boolean Σ-constraints. We show that $\mathcal{T} \equiv_\Sigma \emptyset$. Let $\mathcal{I}$ be an interpretation. We have to show that there is a model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. For each $d \in \Delta^{\mathcal{I}}$, let

$$P_d = \{A \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma \mid d \in A^{\mathcal{I}}\}$$

and let C_{P_d} be the concept defined in Definition 21. Since $\mathcal{T}$ does not contain any Boolean Σ-constraints, for each $d \in \Delta^{\mathcal{I}}$ there is a one-point model $\mathcal{I}_d$ of $\mathcal{T}$ satisfying C_{P_d}. We may assume that $\Delta^{\mathcal{I}_d} = \{d\}$ and so $d \in C_{P_d}^{\mathcal{I}_d}$, for all $d \in \Delta^{\mathcal{I}}$. The construction of $\mathcal{J}$ uses both the interpretation $\mathcal{I}$ and the one-point interpretations $\mathcal{I}_d$ for $d \in \Delta^{\mathcal{I}}$. In detail, $\mathcal{J}$ is defined as follows. Set $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}}$ and define for each symbol $X \in (\mathsf{N_C} \setminus \mathsf{Lhs}(\mathcal{T})) \cup \mathsf{N_R}$,

$$X^{\mathcal{J}} = \begin{cases} X^{\mathcal{I}} & \text{if } X \in \Sigma \\ \bigcup_{d \in \Delta^{\mathcal{I}}} X^{\mathcal{I}_d} & \text{otherwise} \end{cases}$$

It remains to define the extension $A^{\mathcal{J}}$ for all concept names $A \in \mathsf{Lhs}(\mathcal{T})$. The definition is by induction on the definitorial depth $d_{\mathcal{T}}(A)$ of A in $\mathcal{T}$. Note that $A^{\mathcal{J}}$ has been defined already for all A with $d_{\mathcal{T}}(A) = 0$ (since then $A \notin \mathsf{Lhs}(\mathcal{T})$). Now let $A \in \mathsf{Lhs}(\mathcal{T})$ and assume that $B^{\mathcal{J}}$ has already been defined for all concept names B with $d_{\mathcal{T}}(B) < d_{\mathcal{T}}(A)$. Let $A \bowtie C$ be the definition of A in $\mathcal{T}$ and set

$$A^{\mathcal{J}} = \begin{cases} \bigcup_{d \in \Delta^{\mathcal{I}}} A^{\mathcal{I}_d} & \text{if } A \in \Sigma \cup \bigcup_{B \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma} \mathsf{depend}_{\mathcal{T}}(B) \\ C^{\mathcal{J}} & \text{otherwise} \end{cases}$$

Note that $A^{\mathcal{J}}$ is well-defined since if $A \bowtie C \in \mathcal{T}$ then $C^{\mathcal{J}}$ is defined before $A^{\mathcal{J}}$. We first show that $\mathcal{J}|_{\Sigma} = \mathcal{I}|_{\Sigma}$. By definition, $X^{\mathcal{I}} = X^{\mathcal{J}}$ for all symbols $X \in \Sigma$ with $X \in (\mathsf{N_C} \setminus \mathsf{Lhs}(\mathcal{T})) \cup \mathsf{N_R}$. It thus remains to consider the concept names $A \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma$. By definition of the interpretations $\mathcal{I}_d$, we have $A^{\mathcal{I}} = \bigcup_{d \in \Delta^{\mathcal{I}}} A^{\mathcal{I}_d}$. By definition of $\mathcal{J}$, we have $A^{\mathcal{J}} = \bigcup_{d \in \Delta^{\mathcal{I}}} A^{\mathcal{I}_d}$. Thus $A^{\mathcal{I}} = A^{\mathcal{J}}$, as required.

We now show that $\mathcal{J}$ is a model of $\mathcal{T}$. Let $A \bowtie C \in \mathcal{T}$. We distinguish two cases:

- $A \in \Sigma \cup \bigcup_{B \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma} \mathsf{depend}_{\mathcal{T}}(B)$.

  By definition of $\mathcal{J}$, we have $A^{\mathcal{J}} = \bigcup_{d \in \Delta^{\mathcal{I}}} A^{\mathcal{I}_d}$. Since $\mathcal{T}$ does not contain a direct Σ-dependency, $\mathsf{depend}_{\mathcal{T}}(A)$ does not contain any Σ-symbols. Observe that $X^{\mathcal{J}} = \bigcup_{d \in \Delta^{\mathcal{I}}} X^{\mathcal{I}_d}$ for all symbols $X \in \mathsf{depend}_{\mathcal{T}}(A)$: if $X \in (\mathsf{N_C} \setminus \mathsf{Lhs}(\mathcal{T})) \cup \mathsf{N_R}$, then this follows from $X \notin \Sigma$ and the definition of $\mathcal{J}$. If $X \in \mathsf{Lhs}(\mathcal{T})$, then $X \in \mathsf{depend}_{\mathcal{T}}(A)$ yields $X \in \bigcup_{B \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma} \mathsf{depend}_{\mathcal{T}}(B)$ and so $X^{\mathcal{J}} = \bigcup_{d \in \Delta^{\mathcal{I}}} X^{\mathcal{I}_d}$ by definition of $\mathcal{J}$. By the semantics of $\mathcal{ALCI}$-concepts, it follows that $C^{\mathcal{J}} = \bigcup_{d \in \Delta^{\mathcal{I}}} C^{\mathcal{I}_d}$. Since every $\mathcal{I}_d$ is a model of $\mathcal{T}$, $\mathcal{J}$ satisfies $A \bowtie C$.

- $A \notin \Sigma \cup \bigcup_{B \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma} \mathsf{depend}_{\mathcal{T}}(B)$.

  Then $A^{\mathcal{J}} = C^{\mathcal{J}}$ by definition of $\mathcal{J}$. Thus $\mathcal{J}$ satisfies $A \bowtie C$.

This finishes the proof since we have shown that for every interpretation $\mathcal{I}$ there exists a model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{J}|_{\Sigma} = \mathcal{I}|_{\Sigma}$. □

We now prove the main result of this section.

Theorem 25. *(1) For every acyclic $\mathcal{ELI}$-TBox $\mathcal{T}$ and signature Σ, $\mathcal{T} \equiv_{\Sigma} \emptyset$ iff $\mathcal{T}$ has no direct nor indirect Σ-dependencies.*

(2) For acyclic $\mathcal{ELI}$-TBoxes $\mathcal{T}$ and signatures Σ, it is in PTime *to decide whether $\mathcal{T} \equiv_{\Sigma} \emptyset$.*

Proof. (1) follows from Lemmas 20, 23, and 24. To show (2), it is sufficient to prove that the presence of direct and indirect Σ-dependencies in an acyclic $\mathcal{ELI}$-TBox can be decided in PTime. First note that for each concept name A, the set $\mathsf{depend}_{\mathcal{T}}(A)$ and $\mathsf{depend}^{\equiv}_{\mathcal{T}}(A)$ can be computed by a straightforward reachability analysis. Thus, the existence of direct Σ-dependencies can be checked in polynomial time. For indirect Σ-dependencies, the same is true because if a concept name $A \in \Sigma$ has an indirect Σ-dependency in $\mathcal{T}$ induced by concept names $A_1, \dots, A_n \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma$, then A has an indirect Σ-dependency in $\mathcal{T}$ induced by the set of concept names $(\mathsf{Lhs}(\mathcal{T}) \cap \Sigma) \setminus \{A\}$. Thus, to check weather $\mathcal{T}$ contains an indirect Σ-dependency it is sufficient to check whether there is $A \in \mathsf{Def}(\mathcal{T}) \cap \Sigma$ such that

$$\mathsf{depend}^{\equiv}_{\mathcal{T}}(A) \setminus \mathsf{Def}(\mathcal{T}) \subseteq \bigcup_{(\mathsf{Lhs}(\mathcal{T}) \cap \Sigma) \setminus \{A\}} \mathsf{depend}_{\mathcal{T}}(A_i),$$

which is in polynomial time. □

6. Ontology Modules

We employ Σ-inseparability to define different notions of a module in a TBox, analyse their relationship and properties, and determine the complexity of checking whether a given subset $\mathcal{T}' \subseteq \mathcal{T}$ is a module (from now on called *module checking*). Each notion of a module is parameterised with a signature Σ of interest. This is motivated by the main applications of modules such as replacing the original TBox in an application that uses only a certain subsignature Σ or summarising what a given TBox says about a given signature Σ. The signature Σ is also important for module extraction studied in Section 7.

Arguably, the most direct way to define a Σ-module $\mathcal{M}$ in a TBox $\mathcal{T}$ is thus to demand that $\mathcal{M} \subseteq \mathcal{T}$ and $\mathcal{M} \equiv_\Sigma \mathcal{T}$. By Theorem 4, we then have that $\mathcal{M}$ and $\mathcal{T}$ entail the same SO-sentences over Σ, and by Theorem 5, this even holds in the context of other ontologies used together with $\mathcal{M}$ and $\mathcal{T}$. From now on, we call this kind of module a *plain* Σ-module. Note that it might clearly happen that a plain Σ-module $\mathcal{M}$ must contain also non-Σ-symbols, simply because they occur together with Σ-symbols in the same statement in $\mathcal{T}$. However, according to its definition, $\mathcal{M}$ does not need to be inseparable from $\mathcal{T}$ regarding these additional symbols, which might be regarded as $\mathcal{M}$ being incomplete. Moreover, a plain Σ-module $\mathcal{M}$ does not need to contain *all* non-tautological statements from $\mathcal{T}$ that concern Σ because those statements might be implied by other statements that *are* contained in $\mathcal{M}$. Again, this might be viewed as a kind of incompleteness of $\mathcal{M}$.

To address these issues, we define two additional kinds of modules: *self-contained* Σ-modules $\mathcal{M}$ that must be inseparable from $\mathcal{T}$ not only regarding Σ, but regarding $\Sigma \cup \mathsf{sig}(\mathcal{M})$; and *depleting* Σ-modules $\mathcal{M}$ for which we require that $\mathcal{T} \setminus \mathcal{M}$ is inseparable from the empty TBox, again regarding $\Sigma \cup \mathsf{sig}(\mathcal{M})$.

Definition 26. Let $\mathcal{M} \subseteq \mathcal{T}$ be TBoxes and Σ a signature. Then $\mathcal{M}$ is a

- *plain* Σ*-module* of $\mathcal{T}$ if $\mathcal{M} \equiv_\Sigma \mathcal{T}$;
- *self-contained* Σ*-module* of $\mathcal{T}$ if $\mathcal{M} \equiv_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \mathcal{T}$;
- *depleting* Σ*-module* of $\mathcal{T}$ if $\mathcal{T} \setminus \mathcal{M} \equiv_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \emptyset$.

A (plain, self-contained, depleting) Σ-module of $\mathcal{T}$ is *minimal* if no $\mathcal{N} \subsetneq \mathcal{M}$ is such a Σ-module of $\mathcal{T}$.

Plain and depleting Σ-modules were introduced in [18], where they are called weak and strong Σ-modules. Self-contained and depleting Σ-modules are implicit in [16] and were first explicitly studied in [35, 14]. Note that if $\mathcal{M}$ is a depleting Σ-module, then $\mathcal{T} \setminus \mathcal{M}$ is safe for $\mathsf{sig}(\mathcal{M})$ and so the module $\mathcal{M}$ can be maintained separately outside of $\mathcal{T}$ without the risk of unintended interactions with the rest of $\mathcal{T}$. Also note that checking depleting Σ-modules is exactly the same problem as deciding Σ-inseparability from the empty TBox. The relationship between the different kinds of modules is as follows, as first observed in [35].

Proposition 27.

1. *If $\mathcal{M}$ is a self-contained Σ-module of $\mathcal{T}$, then it is a plain Σ-module.*
2. *If $\mathcal{M}$ is a depleting Σ-module of $\mathcal{T}$, then it is a self-contained Σ-module.*

Proof. Point 1 follows from the monotonicity of inseparability, and Point 2 follows from robustness under replacement: assume $\mathcal{M} \subseteq \mathcal{T}$ and let $\Sigma' = \Sigma \cup \mathsf{sig}(\mathcal{M})$. If $\mathcal{T} \setminus \mathcal{M} \equiv_{\Sigma'} \emptyset$, then $(\mathcal{T} \setminus \mathcal{M}) \cup \mathcal{M} \equiv_{\Sigma'} \mathcal{M}$ and so $\mathcal{T} \equiv_{\Sigma'} \mathcal{M}$, as required. □

In this paper, we concentrate on self-contained and depleting modules, and do not further consider plain modules. One reason is that we do not have much positive to say about the latter. For example, the basic task of module checking is equivalent to deciding Σ-inseparability in the case of plain modules. As can be seen in Table 1, this problem is undecidable in all cases that we were able to solve, the only open case being acyclic $\mathcal{EL}$- and $\mathcal{ELI}$-TBoxes. Another reason for preferring self-contained and depleting modules is that, as first observed in [16], these types of modules are very useful for module extraction, as studied in Section 7. The following examples show that depleting modules and self-contained modules are not the same notion.

Example 28. (1) Let $\mathcal{T} = \{A \sqsubseteq B \sqcap A_1, A \sqsubseteq B \sqcap A_2\}$, $\mathcal{M} = \{A \sqsubseteq B \sqcap A_1\}$, and $\Sigma = \{A, B\}$. $\mathcal{M}$ is a self-contained Σ-module of $\mathcal{T}$, but not a depleting Σ-module of $\mathcal{T}$.

(2) The difference between depleting and self-contained modules is 'felt' even by acyclic $\mathcal{EL}$-TBoxes. Take $\mathcal{T} = \{B \sqsubseteq A, A \equiv \top\}$, $\mathcal{M} = \{A \equiv \top\}$, and $\Sigma = \{A, B\}$. Then $\mathcal{M}$ is a self-contained Σ-module of $\mathcal{T}$, but not a depleting one. We will see below that no such example exists if concept definitions the form $A \equiv \top \sqcap \cdots \sqcap \top$ are disallowed.

6.1. Modules in acyclic $\mathcal{EL}/\mathcal{ELI}$

We consider module checking in acyclic $\mathcal{EL}$- and $\mathcal{ELI}$- TBoxes. Note that, by Theorem 25, depleting modules can be checked in PTime. Our main result is that, for acyclic $\mathcal{ELI}$-TBoxes that do not contain *trivial* concept definitions of the form $A \equiv \top \sqcap \cdots \sqcap \top$, self-contained Σ-modules coincide with depleting Σ-modules. Note that, in applications, such concept definitions are rather exotic and it should always be possible to remove them by replacing A with $\top$ throughout $\mathcal{T}$. In summary, depleting and (equivalently) self-contained modules can be checked in PTime for acyclic $\mathcal{EL}$- and $\mathcal{ELI}$-TBoxes. As we mentioned before, it is interesting to contrast this with the PSpace-completeness of subsumption in acyclic $\mathcal{ELI}$-TBoxes [15].

Theorem 29. *Let $\mathcal{T}$ be an acyclic $\mathcal{ELI}$-TBox, $\mathcal{M} \subseteq \mathcal{T}$, and Σ a signature. If $\mathcal{T}$ does not contain trivial concept definitions, then the following are equivalent:*

1. *$\mathcal{M}$ is a depleting Σ-module of $\mathcal{T}$;*
2. *$\mathcal{M}$ is a self-contained Σ-module of $\mathcal{T}$.*

Proof. The implication from Point 1 to Point 2 follows from Theorem 27. Consider the implication from Point 2 to Point 1. Assume that $\mathcal{M}$ is a self-contained Σ-module of $\mathcal{T}$. Let $\Sigma' = \Sigma \cup \mathsf{sig}(\mathcal{M})$. We have to show that $\mathcal{T} \setminus \mathcal{M} \equiv_{\Sigma'} \emptyset$. By Lemma 24, it is sufficient to show

(a) $\mathcal{T} \setminus \mathcal{M}$ does not contain a Boolean Σ'-constraint;

(b) $\mathcal{T} \setminus \mathcal{M}$ does not contain a direct Σ'-dependency.

We start with Point (a). For a proof by contradiction, assume that (a) does not hold. Then there is a set $P \subseteq \mathsf{Lhs}(\mathcal{T} \setminus \mathcal{M}) \cap \Sigma'$ such that

$$C_P = \bigsqcap_{A \in P} A \sqcap \bigsqcap_{A \in (\mathsf{Lhs}(\mathcal{T} \setminus \mathcal{M}) \cap \Sigma') \setminus P} \neg A$$

is not satisfiable in a one-point model of $\mathcal{T} \setminus \mathcal{M}$. We construct a one-point interpretation $\mathcal{I}$ with $\Delta^{\mathcal{I}} = \{d\}$ satisfying $\mathcal{M}$ and C_P. Set $r^{\mathcal{I}} = \emptyset$, for all $r \in \mathsf{N_R}$. The interpretation of concept names A is defined by induction on the definitorial depth $d_{\mathcal{M}}(A)$ of A in $\mathcal{M}$. For every $A \in \mathsf{N_C}$ with $d_{\mathcal{M}}(A) = 0$, let

$$A^{\mathcal{I}} = \begin{cases} \{d\} & \text{if } A \in P \\ \emptyset & \text{otherwise} \end{cases}$$

Note that $d_{\mathcal{M}}(A) = 0$ for all $A \in \mathsf{Lhs}(\mathcal{T} \setminus \mathcal{M})$ and so we have already defined the interpretation of all concept names A in C_P in such a way that $\mathcal{I}$ satisfies C_P. It remains to complete the definition of $\mathcal{I}$ for concept names of positive definitorial depth in $\mathcal{M}$ and ensure that $\mathcal{M}$ is satisfied. This is straightforward, however: assume that $A \bowtie C \in \mathcal{M}$ and that $B^{\mathcal{I}}$ has been defined for all B in C. Then set $A^{\mathcal{I}} = C^{\mathcal{I}}$. The resulting interpretation $\mathcal{I}$ is a model of $\mathcal{M}$ with $d \in C_P^{\mathcal{I}}$.

As $\mathsf{sig}(C_P) \subseteq \Sigma'$, we have that C_P is satisfied in any interpretation $\mathcal{J}$ with $\mathcal{I}|_{\Sigma'} = \mathcal{J}|_{\Sigma'}$. Thus, since we assume that C_P is not satisfiable in any one-point model of $\mathcal{T} \setminus \mathcal{M}$ and $\mathsf{sig}(\mathcal{M}) \subseteq \Sigma'$, there is no model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{I}|_{\Sigma'} = \mathcal{J}|_{\Sigma'}$. This contradicts the assumption that $\mathcal{M}$ is a self-contained Σ-module of $\mathcal{T}$.

We prove Point (b). For a proof by contradiction, assume that Point (b) does not hold. There is an $A \in \mathsf{Lhs}(\mathcal{T} \setminus \mathcal{M}) \cap \Sigma'$ such that $X \in \mathsf{depend}_{\mathcal{T} \setminus \mathcal{M}}(A) \cap \Sigma'$, for some symbol $X \in \Sigma'$. We construct a one-point model $\mathcal{I}$ of $\mathcal{M}$ with $X^{\mathcal{I}} = \emptyset$ and $A^{\mathcal{I}} \neq \emptyset$. Let $\Delta^{\mathcal{I}} = \{d\}$ and set $r^{\mathcal{I}} = \emptyset$, for all $r \in \mathsf{N_R}$. The interpretation of concept names is defined by induction over the definitorial depth in $\mathcal{M}$. Observe that $d_{\mathcal{M}}(A) = 0$ and set $A^{\mathcal{I}} = \{d\}$ and $B^{\mathcal{I}} = \emptyset$ for all $B \in \mathsf{N_C} \setminus \{A\}$ with $d_{\mathcal{M}}(B) = 0$. We complete this interpretation in the obvious way: assume that $B \bowtie C \in \mathcal{M}$ and that $E^{\mathcal{I}}$ has been defined for all E in C. Then set $B^{\mathcal{I}} = C^{\mathcal{I}}$.

By construction, $\mathcal{I}$ is a model of $\mathcal{M}$. Since $\mathcal{M}$ does not contain trivial concept definitions, $B^{\mathcal{I}} = \emptyset$ for all concept names B distinct from A with

$A \notin \mathsf{depend}_{\mathcal{M}}(B)$. It follows that $X^{\mathcal{I}} = \emptyset$: if X is a role name, then this follows from the definition of $\mathcal{I}$. If X is a concept name, then note that $A \notin \mathsf{depend}_{\mathcal{M}}(X)$ since $\mathcal{T}$ is acyclic and so $X^{\mathcal{I}} = \emptyset$, as required.

Since $A^{\mathcal{I}} = \{d\}$, $X^{\mathcal{I}} = \emptyset$ and $X \in \mathsf{depend}_{\mathcal{T}\setminus\mathcal{M}}(A) \cap \Sigma'$ there is no model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{I}|_{\Sigma'} = \mathcal{J}|_{\Sigma'}$. This contradicts the assumption that $\mathcal{M}$ is a self-contained Σ-module of $\mathcal{T}$. □

The following result now follows with Theorem 25.

Theorem 30. *Checking depleting and (equivalently) self-contained Σ-modules in acyclic $\mathcal{ELI}$-TBoxes is in* PTime.

6.2. Modules in acyclic $\mathcal{ALC}/\mathcal{ALCI}$

For module checking in acyclic $\mathcal{ALC}$- and $\mathcal{ALCI}$-TBoxes, we cannot hope to obtain general positive results that parallel those obtained for $\mathcal{EL}$ and $\mathcal{ELI}$ in the previous section: by Theorem 8, checking depleting modules in acyclic $\mathcal{ALC}$-TBoxes is undecidable, and the same is true for checking self-contained modules since inseparability from the empty TBox can be reduced to checking self-contained modules by observing that a TBox $\mathcal{T}$ is Σ-inseparable from the empty TBox iff the empty TBox is a self-contained Σ-module of $\mathcal{T}$. Interestingly, though, the direct Σ-dependencies used for $\mathcal{EL}$ and $\mathcal{ELI}$ in Sections 5.3 and 6.1 still turn out to be useful also in the context of $\mathcal{ALC}$ and $\mathcal{ALCI}$ (while indirect ones appear to be less relevant). In particular, we identify two syntactic conditions that both result in checking depleting and self-contained modules in $\mathcal{ALC}$- and $\mathcal{ALCI}$-TBoxes to become decidable: one requires that the TBox $\mathcal{T} \setminus \mathcal{M}$ is free of direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependencies and the other is a relaxation of this condition. We actually start with the latter condition and show that checking self-contained modules is coNExp$^{\mathrm{NP}}$-complete while checking depleting modules is Π_2^p-complete. Under the first mentioned condition, checking depleting and self-contained modules is both Π_2^p-complete, and the two kinds of module coincide.

We believe that the syntactic restrictions considered in this section are very natural from an application perspective, i.e., the question whether $\mathcal{M} \subseteq \mathcal{T}$ is a Σ-module of $\mathcal{T}$ should arise only if $\mathcal{T} \setminus \mathcal{M}$ contains no direct Σ-dependencies. This is illustrated by the following example.

Example 31. Let $\mathcal{T} = \{A \sqsubseteq B \sqcup \neg B\} \cup \mathcal{M}$ with $\mathcal{M} = \{E \sqsubseteq A \sqcap B\}$, and let $\Sigma = \{A, B, E\} = \mathsf{sig}(\mathcal{M})$. Note that $\mathcal{T} \setminus \mathcal{M}$ contains a direct Σ-dependency since $B \in \mathsf{depend}_{\mathcal{T}\setminus\mathcal{M}}(A)$. If $\mathcal{T}$ was an acyclic $\mathcal{EL}$-TBox, this would immediately imply that $\mathcal{M}$ is not a depleting Σ-module of $\mathcal{T}$. However, for the acyclic $\mathcal{ALC}$-TBox $\mathcal{T}$ given above, we clearly have $\mathcal{T} \setminus \mathcal{M} \equiv_\Sigma \emptyset$ (even $\mathcal{T} \setminus \mathcal{M} \equiv \emptyset$), so $\mathcal{M}$ is a depleting Σ-module of $\mathcal{T}$. The example can clearly be generalised by replacing $B \sqcup \neg B$ with any concept C that is valid (i.e., $C^\mathcal{I} = \Delta^\mathcal{I}$ in every model $\mathcal{I}$) and contains a Σ-symbol. However, the example relies on the fact that the TBox statements in $\mathcal{T} \setminus \mathcal{M}$ define the meaning of one Σ-symbol by employing another Σ-symbol without asserting any non-trivial semantic relationship between the two Σ-symbols. This seems to be rather untypical for real-world ontologies and most likely indicates a modelling error.

As announced, we start with a syntactic condition that is weaker than $\mathcal{T} \setminus \mathcal{M}$ not containing a direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency, but turns out to be sufficient to regain decidability of module checking.

Definition 32. Let $\mathcal{T}$ be an acyclic $\mathcal{ALCI}$-TBox and Σ a signature. We say that $\mathcal{T}$ *contains a direct* Σ*-dependency between concept and role names* if there exists $A \in \Sigma \cap \mathsf{N_C}$ and $r \in \Sigma \cap \mathsf{N_R}$ such that $r \in \mathsf{depend}_\mathcal{T}(A)$.

Now, the weaker condition is that $\mathcal{T} \setminus \mathcal{M}$ does not contain a direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency between concept and role names. Interestingly, self-contained and depleting modules turn out to not coincide under the weaker condition, while they do under the stronger one. In fact, for the weaker condition, checking self-contained and depleting modules is of rather different complexity. It is also interesting that, as shown by the proof of the following theorem, having no direct Σ-dependencies between concept and role names is very closely related to dealing with a concept signature Σ.

Theorem 33. *Given an acyclic* $\mathcal{ALC}$ *or* $\mathcal{ALCI}$*-TBox* $\mathcal{T}$*, a signature* Σ*, and a subset* $\mathcal{M} \subseteq \mathcal{T}$ *such that* $\mathcal{T} \setminus \mathcal{M}$ *contains no direct* $\Sigma \cup \mathsf{sig}(\mathcal{M})$*-dependency between concept and role names, it is*

1. Π^p_2*-complete to decide whether* $\mathcal{M}$ *is a depleting* Σ*-module of* $\mathcal{T}$*;*
2. coNExp$^{\text{NP}}$*-complete to decide whether* $\mathcal{M}$ *is a self-contained* Σ*-module of* $\mathcal{T}$*.*

Proof. The lower bound for Point 1 is a consequence of Theorem 14, which states Π^p_2-hardness for Σ-inseparability of acyclic $\mathcal{ALC}$-TBoxes from the empty TBox. It suffices to observe that (i) for all TBoxes $\mathcal{T}$ and signatures Σ, we have $\mathcal{T} \equiv_\Sigma \emptyset$ iff $\mathcal{T} \setminus \mathcal{M} \equiv_\Sigma \emptyset$ with $\mathcal{M} := \emptyset$ and (ii) $\mathcal{T} \setminus \mathcal{M} = \mathcal{T}$ does not contain direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependencies between concept and role names since Σ is a concept signature.

Now for the lower bound for Point 2. By Theorem 10 and Lemma 3, Σ-entailment between acyclic $\mathcal{ALC}$-TBoxes is $\textsc{coNExp}^{\text{NP}}$-hard when Σ is a concept signature. It thus remains to note that we can use a small variation of the reduction from Σ-entailment to Σ-inseparability given in the proof of Lemma 3, replacing Σ-inseparability with checking self-contained modules. In detail, let $\mathcal{T}_1, \mathcal{T}_2$ be acyclic TBoxes and Σ a concept signature. By our strengthened formulation of Theorem 10 (see comment before that theorem), we can assume that $\mathcal{T}_1 \cup \mathcal{T}_2$ is acyclic. We show that we can also assume that $\Sigma = \mathsf{sig}(\mathcal{T}_1) \cap \mathsf{sig}(\mathcal{T}_2)$. First, as in the proof of Lemma 3, we can assume that $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_1) \cap \mathsf{sig}(\mathcal{T}_2)$. Now, since we can also add fresh inclusions of the form $A_0 \sqsubseteq A$ for $A \in \Sigma \cap \mathsf{N_C}$ (and with A_0 fresh) to $\mathcal{T}_1$ and $\mathcal{T}_2$ we can assume that $\Sigma = \mathsf{sig}(\mathcal{T}_1) \cap \mathsf{sig}(\mathcal{T}_2)$. Set $\mathcal{M} = \mathcal{T}_1$, $\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2$, and $\Sigma' = \mathsf{sig}(\mathcal{T}_1)$. Then $\mathcal{T} \setminus \mathcal{M}$ contains no direct $\Sigma' \cup \mathsf{sig}(\mathcal{M})$-dependency between concept and role names since Σ is a concept signature. Now it can be verified that $\mathcal{T}_1 \models_\Sigma \mathcal{T}_2$ iff $\mathcal{T}_1 \equiv_{\mathsf{sig}(\mathcal{T}_1)} \mathcal{T}_1 \cup \mathcal{T}_2$ iff $\mathcal{M} \equiv_{\Sigma' \cup \mathsf{sig}(\mathcal{M})} \mathcal{T}$ iff $\mathcal{M}$ is a self-contained Σ'-module. In fact, the only interesting equivalence is the first one, where we exploit that $\Sigma = \mathsf{sig}(\mathcal{T}_1) \cap \mathsf{sig}(\mathcal{T}_2)$.

The upper bound proofs are by reduction to Σ-inseparability with Σ a concept signature, for which corresponding upper bounds have been established in Theorems 14 and 9. We start with identifying the part of an acyclic TBox $\mathcal{T}$ that is 'relevant' for deciding Σ-inseparability from $\mathcal{T}$. For an acyclic TBox $\mathcal{T}$ and signature Σ, let

$$\mathsf{Lhs}_\Sigma(\mathcal{T}) = \{A \bowtie C \in \mathcal{T} \mid A \in \Sigma \text{ or } \exists X \in \Sigma \ (A \in \mathsf{depend}_\mathcal{T}(X))\}.$$

Intuitively, $\mathsf{Lhs}_\Sigma(\mathcal{T})$ is the set of all inclusions in $\mathcal{T}$ that are influenced by symbols from Σ. The following claim makes this observation precise. It shows that, as far as Σ-inseparability is concerned, $\mathcal{T}$ can be equivalently replaced with $\mathsf{Lhs}_\Sigma(\mathcal{T})$.

Claim. For every interpretation $\mathcal{I}$, the following conditions are equivalent:

(a) there is a model $\mathcal{J}$ of $\mathcal{T}$ with $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$;

(b) there is a model $\mathcal{J}$ of $\mathsf{Lhs}_\Sigma(\mathcal{T})$ with $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$.

The implication "(a) ⇒ (b)" is immediate, and "(b) ⇒ (a)" is shown in the appendix. The main reason for replacing certain TBoxes $\mathcal{T}$ with $\mathsf{Lhs}_\Sigma(\mathcal{T})$ in the remaining proof is the following property:

($*$) if a TBox $\mathcal{T}$ contains no direct Σ-dependency between concept and role names, then $\mathsf{Lhs}_\Sigma(\mathcal{T})$ does not contain a role name from Σ.

Clearly, this is not true for $\mathcal{T}$ itself. We now present the announced reductions.

To prove the upper bound for Point 1, we have to show that it is in Π_2^p to decide whether $\mathcal{T} \setminus \mathcal{M} \equiv_{\Sigma'} \emptyset$, with $\Sigma' = \Sigma \cup \mathsf{sig}(\mathcal{M})$. By the claim, $\mathcal{T} \setminus \mathcal{M} \equiv_{\Sigma'} \emptyset$ iff $\mathsf{Lhs}_{\Sigma'}(\mathcal{T} \setminus \mathcal{M}) \equiv_{\Sigma'} \emptyset$. By ($*$), $\mathsf{Lhs}_{\Sigma'}(\mathcal{T} \setminus \mathcal{M})$ does not contain a role name from Σ'. Thus, we have $\mathsf{Lhs}_{\Sigma'}(\mathcal{T} \setminus \mathcal{M}) \equiv_{\Sigma'} \emptyset$ iff $\mathsf{Lhs}_{\Sigma'}(\mathcal{T} \setminus \mathcal{M}) \equiv_{\Sigma' \cap \mathsf{N_C}} \emptyset$. By Theorem 14, the latter can be decided in Π_2^p.

To prove the upper bound for Point 2, we have to show that it is in $\textsc{coNExp}^{\text{NP}}$ to decide whether $\mathcal{M} \equiv_{\Sigma'} \mathcal{T}$, with $\Sigma' = \Sigma \cup \mathsf{sig}(\mathcal{M})$. Clearly, $\mathcal{M} \equiv_{\Sigma'} \mathcal{T}$ iff $\mathcal{M} \models_{\Sigma'} \mathcal{T} \setminus \mathcal{M}$ iff $\mathcal{M} \models_{\Sigma'} \mathsf{Lhs}_{\Sigma'}(\mathcal{T} \setminus \mathcal{M})$. By ($*$), $\mathsf{Lhs}_{\Sigma'}(\mathcal{T} \setminus \mathcal{M})$ does not contain a role name from Σ'. Thus, $\mathcal{M} \models_{\Sigma'} \mathsf{Lhs}_{\Sigma'}(\mathcal{T} \setminus \mathcal{M})$ iff $\mathcal{M} \models_{\Sigma' \cap \mathsf{N_C}} \mathsf{Lhs}_{\Sigma'}(\mathcal{T} \setminus \mathcal{M})$. By Theorem 9, the latter can be decided in $\textsc{coNExp}^{\text{NP}}$. □

We now consider the stronger syntactic condition that $\mathcal{T} \setminus \mathcal{M}$ does not contain a direct Σ-dependency in the original sense of Definition 19. We prove that, in this case, there is no difference between depleting and self-contained modules, and that module checking is Π_2^p-complete.

Theorem 34. *For all acyclic $\mathcal{ALCI}$-TBox $\mathcal{T}$, signatures Σ, and subsets $\mathcal{M} \subseteq \mathcal{T}$ such that $\mathcal{T} \setminus \mathcal{M}$ contains no direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependencies, the following are equivalent:*

1. *$\mathcal{M}$ is a depleting Σ-module of $\mathcal{T}$.*
2. *$\mathcal{M}$ is a self-contained Σ-module of $\mathcal{T}$;*

Given $\mathcal{T}$, Σ, and $\mathcal{M}$ as above, it is Π_2^p-complete to decide whether $\mathcal{M}$ is a self-contained/depleting Σ-module of $\mathcal{T}$.

Proof. The proof of the equivalence of Points 1 and 2 is similar to the proof of Theorem 29. The implication "Point 1 ⇒ Point 2" again follows from Theorem 27. The proof of "Point 2 ⇒ Point 1" is identical to the proof of Theorem 29 with the only exception that the absence of direct Σ-dependencies (Point (b) in the proof of Theorem 29) is stated explicitly in the conditions of Theorem 34 rather than deduced from the assumption that $\mathcal{M}$ is a self-contained Σ-module of $\mathcal{T}$. Note that here we need that Lemma 24 applies not only to $\mathcal{ELI}$-TBoxes, but also to $\mathcal{ALCI}$-TBoxes (see comment before that lemma).

Now for the complexity. The Π^p_2-upper bound is inherited from Theorem 33. The lower bound can be established by an analysis of the proof of Theorem 14, where we show that Σ-inseparability of acyclic $\mathcal{ALCI}$-TBoxes from the empty TBox is Π^p_2-hard. We have already noted that this problem is equivalent to checking depleting modules, and the TBoxes used in the proof do obviously not contain direct Σ-dependencies. □

7. Module Extraction

In this section we consider the problem to compute, given a TBox $\mathcal{T}$ and a signature Σ, a subset $\mathcal{M}$ of $\mathcal{T}$ that is a minimal self-contained or depleting Σ-module of $\mathcal{T}$. We call this problem the module extraction problem. We present algorithms for module extraction from acyclic $\mathcal{ELI}$-TBoxes and from acyclic $\mathcal{ALCI}$-TBoxes, building on the results for module checking obtained in the previous section (Theorems 30 and 34). For acyclic $\mathcal{ELI}$-TBoxes, we extract the *unique minimal* depleting Σ-module and for acyclic $\mathcal{ALCI}$-TBoxes we extract the *unique minimal* depleting Σ-module whose complement contains no direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency. Note that self-contained Σ-modules and depleting Σ-modules coincide in these cases (if there are no trivial concept definitions). For brevity, we will from now on only speak about depleting Σ-modules. All presented algorithms run in polynomial time when module checking is available as an oracle. For acyclic $\mathcal{ELI}$-TBoxes, by Theorem 30 we thus obtain overall polynomial runtime.

We first consider the *black box approach* from [14], where module checking is used as an oracle in a generic module extraction algorithm. Together with Theorems 30 and 34, this approach yields polynomial algorithms for module extraction in both of the $\mathcal{ELI}$- and the $\mathcal{ALCI}$-case mentioned above, modulo the oracle. We complement this approach by presenting a *white box approach* for module extraction from acyclic $\mathcal{ELI}$-TBoxes which also runs in polynomial time, but more tightly integrates the module checking into the module extraction and can be expected to be more efficient in practice.

7.1. Black Box Approach

The black box approach to module extraction described in [14] is based on the algorithm shown in Figure 2. Notice that $\mathcal{W}$ is used to find a TBox statement $\alpha \in \mathcal{T} \setminus \mathcal{M}$ such that α is contained in some minimal subset $\mathcal{S}$

```
Input: TBox $\mathcal{T}$ and signature $\Sigma$.
$\mathcal{M} := \emptyset$;
$\mathcal{W} := \emptyset$;
while $(\mathcal{T} \setminus \mathcal{M}) \neq \mathcal{W}$ do
    choose $\alpha \in (\mathcal{T} \setminus \mathcal{M}) \setminus \mathcal{W}$
    $\mathcal{W} := \mathcal{W} \cup \{\alpha\}$;
    if $\mathcal{W} \not\equiv_{(\Sigma \cup \mathsf{sig}(\mathcal{M}))} \emptyset$ then
        $\mathcal{M} := \mathcal{M} \cup \{\alpha\}$;
        $\mathcal{W} := \emptyset$
    endif
end while
output $\mathcal{M}$
```

Figure 2: Extracting depleting Σ-modules

of $\mathcal{T} \setminus \mathcal{M}$ with $\mathcal{S} \not\equiv_\Sigma \emptyset$. This subset $\mathcal{S}$ is contained in $\mathcal{W}$, but need not be identical to it. It is proved in [14] that for any TBox $\mathcal{T}$ formulated in first-order logic and signature Σ, there is a *unique* minimal depleting Σ-module of $\mathcal{T}$, and that this module is the output of the algorithm when started on $\mathcal{T}$ and Σ.[5] The runtime of the algorithm is $O((|\mathcal{T}| + |\Sigma|)^2 \times T_c(\mathcal{T}, \Sigma))$, where $T_c(\mathcal{T}, \Sigma)$ is the time needed to check for a TBox $\mathcal{T}$ and signature Σ whether $\mathcal{T} \equiv_\Sigma \emptyset$ (which is equivalent to checking depleting Σ-modules). As announced, by Theorem 30 we thus obtain an algorithm that runs in overall polynomial time. It also follows that minimal depleting Σ-modules of $\mathcal{ELI}$-TBoxes and of $\mathcal{ALCI}$-TBoxes are unique. We summarise this in the following theorem.

Theorem 35. *For any $\mathcal{ALCI}$-TBox and signature Σ, there is a unique minimal depleting Σ-module of $\mathcal{T}$. For $\mathcal{ELI}$-TBoxes, these unique minimal modules can be extracted in polynomial time using the algorithm in Figure 2.*

Since checking depleting Σ-modules is undecidable for $\mathcal{ALCI}$, the generic algorithm in Figure 2 cannot be applied directly to acyclic $\mathcal{ALCI}$-TBoxes. However, as observed in Theorem 34, decidability is regained when the remaining TBox $\mathcal{T} \setminus \mathcal{M}$ (respectively the set $\mathcal{W}$ in the algorithm) does not

[5] These results are actually shown for a whole class of inseparability relations, of which "$\equiv_\Sigma$" can easily be verified to be a member. In particular, we have already seen that "$\equiv_\Sigma$" satisfies robustness under replacements).

```
Input: acyclic $\mathcal{ELI}$-TBox $\mathcal{T}$ and signature $\Sigma$.
Initialise: $\mathcal{M} = \emptyset$.
Apply Rules 1 and 2 exhaustively, preferring Rule 1.
Output: $\mathcal{M}$.

(R1) if $A \in \Sigma \cup \mathsf{sig}(\mathcal{M})$ has a direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency in $\mathcal{T} \setminus \mathcal{M}$
         then set $\mathcal{M} := \mathcal{M} \cup \{A \bowtie C\}$ for $A \bowtie C \in \mathcal{T} \setminus \mathcal{M}$.

(R2) if $A \in \Sigma \cup \mathsf{sig}(\mathcal{M})$ has an indirect $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency in $\mathcal{T} \setminus \mathcal{M}$
         then set $\mathcal{M} := \mathcal{M} \cup \{A \equiv C\}$ for $A \equiv C \in \mathcal{T} \setminus \mathcal{M}$.
```

Figure 3: Module extraction in $\mathcal{ELI}$

contain a direct Σ-dependency. This suggests a variation of the algorithm in Figure 2 that is capable of extracting, from an acyclic $\mathcal{ALCI}$-TBox $\mathcal{T}$, a depleting Σ-module $\mathcal{M} \subseteq \mathcal{T}$ that is not minimal in general, but minimal with the property that $\mathcal{T} \setminus \mathcal{M}$ does not contain a direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency. The following is proved in the appendix.

Theorem 36. *Let $\mathcal{T}$ be an acyclic $\mathcal{ALCI}$-TBox and Σ a signature. Modify the algorithm in Figure 2 by replacing the condition "$\mathcal{W} \not\equiv_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \emptyset$" with*

$$\text{"}\mathcal{W} \textit{ contains a direct } \Sigma \cup \mathsf{sig}(\mathcal{M})\textit{-dependency or } \mathcal{W} \not\equiv_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \emptyset\text{".}$$

The resulting algorithm computes the unique minimal depleting Σ-module $\mathcal{M}$ of $\mathcal{T}$ such that $\mathcal{T} \setminus \mathcal{M}$ contains no direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency.

A central step in the proof of Theorem 36 is to show that if $\mathcal{W} \subseteq \mathcal{T}$ is minimal such that $\mathcal{W}$ contains a direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency, then *all* of $\mathcal{W}$ must be contained in $\mathcal{M}$. Note that this justifies including α in $\mathcal{M}$ in the modified algorithm.

7.2. White Box Approach

The algorithm used for module extraction in the previous section uses the Σ-inseparability check in a black box manner. In the case of acyclic $\mathcal{ELI}$-TBoxes, it is possible to integrate the Σ-inseparability check described in the proof of Theorem 25 more tightly into the module extraction algorithm, which results in an algorithm that is more transparent and should be expected to be more efficient in practical cases. This algorithm is given in Figure 3.

Theorem 37. *Let $\mathcal{T}$ be an acyclic $\mathcal{ELI}$-TBox and Σ a signature. The output $\mathcal{M}$ of the algorithm in Figure 3 is the unique minimal depleting Σ-module of $\mathcal{T}$.*

Proof. Let $\mathcal{M}$ be the output of the algorithm in Figure 3. Then Rules (R1) and (R2) are not applicable and so $\mathcal{T} \setminus \mathcal{M}$ does not contain any direct or indirect $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency. By Lemmas 20, 23, and 24, it follows that $\mathcal{T} \setminus \mathcal{M} \equiv_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \emptyset$, as required.

It remains to prove that $\mathcal{M}$ is a minimal depleting Σ-module of $\mathcal{T}$, and that it is unique with this property.[6] To this end, let $\mathcal{M}_0$ be a depleting Σ-module of $\mathcal{T}$. We prove by induction on the number of rule applications that, at every iteration of the algorithm, we have $\mathcal{M} \subseteq \mathcal{M}_0$. It follows that the output $\mathcal{M}$ of the algorithm is contained in every depleting Σ-module of $\mathcal{T}$ and, therefore, $\mathcal{M}$ is the unique minimal depleting Σ-module of $\mathcal{T}$.

The induction start is trivial since $\mathcal{M} = \emptyset \subseteq \mathcal{M}_0$. For the induction step, assume that, by induction hypothesis, $\mathcal{M} \subseteq \mathcal{M}_0$. We make a case distinction according to which rule is applied.

Rule (R1) is applied. Then there is an $A \in \Sigma \cup \mathsf{sig}(\mathcal{M})$ that has a direct $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency in $\mathcal{T} \setminus \mathcal{M}$ and $A \bowtie C \in \mathcal{T} \setminus \mathcal{M}$ is added to $\mathcal{M}$. Assume for a proof by contradiction that $A \bowtie C \notin \mathcal{M}_0$. Let $X \in \mathsf{depend}_{\mathcal{T} \setminus \mathcal{M}}(A) \cap (\Sigma \cup \mathsf{sig}(\mathcal{M}))$ (here and in what follows, $\mathcal{M}$ refers to the state before the addition of $A \bowtie C$). We make a case distinction as follows.

Case 1. $X \in \mathsf{depend}_{\mathcal{T} \setminus \mathcal{M}_0}(A)$. We have $A \bowtie C \in \mathcal{T} \setminus \mathcal{M}_0$. As $\mathcal{M} \subseteq \mathcal{M}_0$ and so $\{A, X\} \subseteq \Sigma \cup \mathsf{sig}(\mathcal{M}_0)$, we conclude that $\mathcal{T} \setminus \mathcal{M}_0$ has a direct $\Sigma \cup \mathsf{sig}(\mathcal{M}_0)$-dependency. By Lemma 20, this contradicts the assumption that $\mathcal{M}_0$ is a depleting Σ-module of $\mathcal{T}$.

Case 2. $X \notin \mathsf{depend}_{\mathcal{T} \setminus \mathcal{M}_0}(A)$. Let $A = Y_1, \ldots, Y_n = B$ be such that $Y_i \prec_{\mathcal{T} \setminus \mathcal{M}} Y_{i+1}$, that is, $Y_i \bowtie C_i \in (\mathcal{T} \setminus \mathcal{M})$, for some C_i, and Y_{i+1} occurs in C_i. We have $Y_1 = A$ and $C_1 = C$, thus $Y_1 \bowtie C_1 \in \mathcal{T} \setminus \mathcal{M}_0$. Let i be the smallest index such that $Y_i \bowtie C_i \notin \mathcal{T} \setminus \mathcal{M}_0$. Then $Y_i \bowtie C_i \in \mathcal{M}_0$ and thus $Y_i \in \mathsf{sig}(\mathcal{M}_0)$. Then we have $A \prec^+_{\mathcal{T} \setminus \mathcal{M}_0} Y_i$, that is, $\mathcal{T} \setminus \mathcal{M}_0$ has a direct $\Sigma \cup \mathsf{sig}(\mathcal{M}_0)$-dependency, which again yields a contradiction.

[6] We already know the latter from Theorem 35. However, the natural proof of minimality of $\mathcal{M}$ given here yields uniqueness of minimal depleting Σ-modules as a by-product.

Rule (R2) is applied. Then there is an $A \in \Sigma \cup \mathsf{sig}(\mathcal{M})$ that has an indirect $\Sigma \cup \mathsf{sig}(\mathcal{M})$-dependency in $\mathcal{T} \setminus \mathcal{M}$, and $A \equiv C \in \mathcal{T} \setminus \mathcal{M}$ is added to $\mathcal{M}$. We have

$$\mathsf{depend}^{\equiv}_{\mathcal{T}\setminus\mathcal{M}}(A) \setminus \mathsf{Def}(\mathcal{T} \setminus \mathcal{M}) \subseteq \bigcup_{1 \leq i \leq n} \mathsf{depend}_{\mathcal{T}\setminus\mathcal{M}}(A_i), \tag{1}$$

for some set $\{A_1, \ldots, A_n\} \subseteq \mathsf{Lhs}(\mathcal{T} \setminus \mathcal{M}) \cap (\Sigma \cup \mathsf{sig}(\mathcal{M}))$ not containing A (where, again, $\mathcal{M}$ generally refers to the state before the addition). Assume for a proof by contradiction that $A \equiv C \notin \mathcal{M}_0$. We show that $\mathcal{M}_0$ contains an indirect $\Sigma \cup \mathsf{Sig}(\mathcal{M}_0)$-dependency, i.e.,

$$\mathsf{depend}^{\equiv}_{\mathcal{T}\setminus\mathcal{M}_0}(A) \setminus \mathsf{Def}(\mathcal{T} \setminus \mathcal{M}_0) \subseteq \bigcup_{B \in (\Sigma \cup \mathsf{sig}(\mathcal{M}_0)) \setminus \{A\}} \mathsf{depend}_{\mathcal{T}\setminus\mathcal{M}_0}(B). \tag{2}$$

This establishes a contradiction to Lemma 20 and the assumption that it is a depleting $\Sigma \cup \mathsf{sig}(\mathcal{M}_0)$-module of $\mathcal{T}$. We first show that the left-hand side of Inclusion (1) is contained in that of Inclusion (2), i.e.,

$$\mathsf{depend}^{\equiv}_{\mathcal{T}\setminus\mathcal{M}_0}(A) \setminus \mathsf{Def}(\mathcal{T} \setminus \mathcal{M}_0) \subseteq \mathsf{depend}^{\equiv}_{\mathcal{T}\setminus\mathcal{M}}(A) \setminus \mathsf{Def}(\mathcal{T} \setminus \mathcal{M}) \tag{3}$$

As $\mathcal{M} \subseteq \mathcal{M}_0$, we have $(\mathcal{T} \setminus \mathcal{M}_0) \subseteq (\mathcal{T} \setminus \mathcal{M})$ and so $\mathsf{depend}^{\equiv}_{\mathcal{T}\setminus\mathcal{M}_0}(A) \subseteq \mathsf{depend}^{\equiv}_{\mathcal{T}\setminus\mathcal{M}}(A)$. Suppose there exists $X \in \mathsf{depend}^{\equiv}_{\mathcal{T}\setminus\mathcal{M}_0}(A) \setminus \mathsf{Def}(\mathcal{T} \setminus \mathcal{M}_0)$ such that $X \in \mathsf{Def}(\mathcal{T} \setminus \mathcal{M})$. Then $\mathcal{M}_0 \setminus \mathcal{M}$ contains a definition $X \equiv C'$, for some C'. Thus $X \in \mathsf{sig}(\mathcal{M}_0)$ and so $\mathcal{T} \setminus \mathcal{M}_0$ has a direct $\Sigma \cup \mathsf{sig}(\mathcal{M}_0)$-dependency in contradiction with $\mathcal{M}_0$ being a depleting Σ-module. This finishes the proof of Inclusion (3).

To prove Inclusion (2), let $X \in \mathsf{depend}^{\equiv}_{\mathcal{T}\setminus\mathcal{M}_0}(A) \setminus \mathsf{Def}(\mathcal{T} \setminus \mathcal{M}_0)$. We have to show the following.

Claim 1. $X \in \mathsf{depend}_{\mathcal{T}\setminus\mathcal{M}_0}(B)$ for some $B \in (\Sigma \cup \mathsf{sig}(\mathcal{M}_0)) \setminus \{A\}$.

It follows from Inclusion (3) that $X \in \mathsf{depend}^{\equiv}_{\mathcal{T}\setminus\mathcal{M}}(A) \setminus \mathsf{Def}(\mathcal{T} \setminus \mathcal{M})$. We have $X \notin \Sigma \cup \mathsf{sig}(\mathcal{M}_0)$ since $\mathcal{M}_0$ does not contain a direct $\Sigma \cup \mathsf{sig}(\mathcal{M}_0)$-dependency.

By Inclusion (1), there is an $A_i \in (\Sigma \cup \mathsf{sig}(\mathcal{M})) \setminus \{A\}$ such that $X \in \mathsf{depend}_{\mathcal{T}\setminus\mathcal{M}}(A_i)$. Let $Z_1, \ldots, Z_n$ be such that $A_i = Z_1 \prec_{\mathcal{T}\setminus\mathcal{M}} \cdots \prec_{\mathcal{T}\setminus\mathcal{M}} Z_n = X$. If $X \in \mathsf{depend}_{\mathcal{T}\setminus\mathcal{M}_0}(A_i)$, then Claim 1 is established by choosing $B = A_i$. Thus assume $X \notin \mathsf{depend}_{\mathcal{T}\setminus\mathcal{M}_0}(A_i)$. Since $X \notin (\Sigma \cup \mathsf{sig}(\mathcal{M}_0))$, we have $Z_{n-1} \prec_{\mathcal{T}\setminus\mathcal{M}_0} Z_n = X$. Let j be the largest index such that for all $j < i < n$, we have $Z_i \prec_{\mathcal{T}\setminus\mathcal{M}_0} Z_{i+1}$ and $Z_j \not\prec_{\mathcal{T}\setminus\mathcal{M}_0} Z_{j+1}$. Then $Z_j \bowtie C'' \in \mathcal{M}_0$ for some

C'' and $Z_{j+1} \in \mathsf{sig}(C'') \subseteq \mathsf{sig}(\mathcal{M}_0)$. Thus Claim 1 is established by choosing $B = Z_{j+1}$ since $X \in \mathsf{depend}_{\mathcal{T}\setminus\mathcal{M}_0}(Z_{j+1})$ and $Z_{j+1} \in \mathsf{sig}(\mathcal{M}_0)$. Note that we require $Z_{j+1} \neq A$ which is the case since otherwise $A \in \mathsf{depend}_{\mathcal{T}\setminus\mathcal{M}}(A_i)$ and so Rule (R1) would have been applied to A_i first. □

Note that preference of (R1) over (R2) is necessary to ensure correctness of the module extraction algorithm. To see this, consider $\mathcal{T} = \{A \sqsubseteq B, B \equiv E\}$ and $\Sigma = \{A, B\}$. We can apply Rule (R2) to $B \equiv E$ since $B \in \mathsf{Def}(\mathcal{T})$ and $\mathsf{depend}(B) \subseteq \mathsf{depend}(A)$. Afterwards, we can apply Rule (R1) to A and obtain $\mathcal{M} = \mathcal{T}$ as the extracted module. This module is not minimal, however. By applying Rule (R1) first to $A \sqsubseteq B$, one obtains the module $\{A \sqsubseteq B\}$ which is a minimal depleting Σ-module of $\mathcal{T}$.

8. Case Study

We perform a case study by extracting minimal depleting (equivalently: self-contained) Σ-modules from the medical ontology Snomed ct [11], using (a variation of) the algorithm in Figure 3. The purpose of the case study is twofold: first, we want to find out whether the extraction algorithm scales to very large real-world TBoxes with several hundred thousand concepts, and second we are interested in the typical size of the extracted modules. In particular, we compare the size of modules extracted by our approach with the size of modules extracted using the locality-based approach from [16, 17], concentrating on the $\top\bot^*$-version of those modules which from now on we call *STAR-modules* for easier pronunciation. We use the implementation of the STAR-module extraction algorithm available as part of OWL API [36] Release 3.2.4. Locality-based module extraction computes an *approximation* of the minimal depleting Σ-module: the STAR-module for Σ is always a depleting Σ-module and, therefore, contains the minimal one extracted by our approach. In general, this inclusion is proper and so the STAR-module for Σ can be properly larger than the minimal depleting Σ-module. The following result determines an important class of acyclic $\mathcal{ELI}$-TBoxes in which they coincide.[7]

[7] We thank C. Del Vescovo, P. Klinov, B. Parsia, U. Sattler, T. Schneider, and D. Tsarkov for helpful discussions about the relationship between STAR-modules and minimal depleting Σ-modules. Additional experiments comparing the two module extraction approaches based on ontologies different from Snomed ct can be found in [37].

Proposition 38. *Let $\mathcal{T}$ be an acyclic $\mathcal{ELI}$-TBox containing no concept definitions. Then the STAR-module of $\mathcal{T}$ w.r.t. Σ coincides with the minimal depleting Σ-module of $\mathcal{T}$, for every signature Σ.*

Note that, though not always minimal, the STAR-modules have the advantage of being efficiently computable also for TBoxes formulated in very expressive description logics. We do not include any other module extraction approach in the comparison as all currently available algorithms that are applicable to Snomed ct extract modules which are at least as large as STAR-modules; this applies, for example, to the algorithm from [38].

Snomed ct is an acyclic $\mathcal{EL}$-TBox, extended with *role inclusion statements* of the form $r \sqsubseteq s$ and one *right-identity statement* of the form $r \circ s \sqsubseteq r$. To take into account the role inclusions, we extend the module extraction algorithm from Figure 3 to acyclic $\mathcal{ELI}$-TBoxes with role inclusions, which we treat using a $\bot$-local approximation (see [16, 17]). The extracted modules are depleting and self-contained Σ-modules but are, because of the approximation approach to role inclusions, not necessarily minimal. Details of the modified algorithm and a correctness proof can be found in in the appendix. The algorithm was implemented in the MEX system, which is written in OCaml. All experiments were carried out on a PC with Intel Core2 CPU running at 2.13 Ghz and with 3GB of RAM.

We start with comparing MEX-modules with STAR-modules for signatures Σ that are Snomed ct *subsets.* Such subsets are sets of concept names that are appropriate to deployment or use of Snomed ct for a particular language, dialect, country, speciality, organisation, user or context. They usually represent groups of concepts or other content which share specific characteristics.[8] For the experiments, we extracted modules from version July 2009 of Snomed ct, which has 245 476 concept inclusions, 62 217 concept equations, 12 role inclusions, 307 694 concept names, and 62 role names. To facilitate the comparison, we added all role names from Snomed ct to the subsets. For 83 out of 159 cases, the STAR-module coincides with the MEX-module. Table 2 presents the sizes of STAR and MEX-modules for the remaining 76 signatures, where the size is measured as the number of concept definitions and inclusions in the module. For all shown signatures, the MEX-modules are smaller than the STAR-modules, with the column labelled

[8] For this and more information about Snomed ct subsets see `http://www.connectingforhealth.nhs.uk/systemsandservices/data/uktc/snomed/subsets`.

Signature	STAR	MEX	Diff.
Admin	6682	6073	609
Adminmed	189	144	45
Adminpr.	3158	2774	384
Advreap.	19	17	2
Allereve	34	22	12
Allerpr.	38	27	11
Cdaenct.	326	292	34
Childbe.	487	374	113
Childrf.	108	97	11
Crcaren.	6769	3662	3107
Crcarer.	5402	2808	2594
Devicet.	3635	3605	30
Diagimg	11288	4135	7153
Diagnos.	96593	87564	9029
Encodis.	711	693	18
Enctyp	31	11	20
Endosfi.	1897	807	1090
Endospr.	1145	403	742
Epcona.4	231	217	14
Epcre.6a	1302	1192	110
Epcre.6b	52	48	4
Epen.11b	40	24	16
Epene.7a	76	58	18
Epene.7b	26	16	10
Epeyee.4	733	687	46
Epfoa.28	46	33	13
Epgra.9a	52	28	24
Epgra.9b	156	98	58
Epin.22a	2417	2410	7
Epin.22b	255	215	40
Epinha.3	449	384	65
Epinj.20	2802	2765	37
Epin.29a	46	36	10
Epins.21	63	57	6
Epmult.1	1883	1745	138
Epne.13a	126	100	26
Epneve.3	93	86	7
Epnot.5a	99	74	25
Epnot.5b	120	85	35
Epnovt.2	124	123	1
Epor.2a1	1489	1403	86
Epora.2b	84	52	32
Eporal.1	4903	4865	38
Epsche.6	359	323	36
Epsup.10	390	335	55
Eptop.5a	366	355	11
Eptop.5b	55	53	2
Famhist	3299	1547	1752
Finding	185409	180089	5320
Foodadr.	2712	2706	6
Gf	43	18	25
Invest	35098	21604	13494
Labcoll.	591	274	317
Labinve.	6835	6093	742
Labinvm.	8628	6437	2191
Labmorph	4568	4546	22
Labspec	5191	2998	2193
Lifesty.	20320	16017	4303
Manumat	9786	9573	213
Nonhuman	11813	5124	6689
Nrel	29	26	3
Pbcl	6869	5686	1183
Pbhllng	3896	1639	2257
Perpref	89	86	3
Pf	324	322	2
Provadv	2647	1944	703
Rapa	85	82	3
Sf	1552	1524	28
Socperc.	6568	6488	80
Specmat.	14070	8668	5402
Srvcarp.	89	78	11
Supply	26	25	1
Supreq	17	14	3
Treatme.	70980	54975	16005
Vmp	10076	9871	205
Vtm	6400	5859	541

Table 2: Comparison of module sizes computed by STAR and MEX.

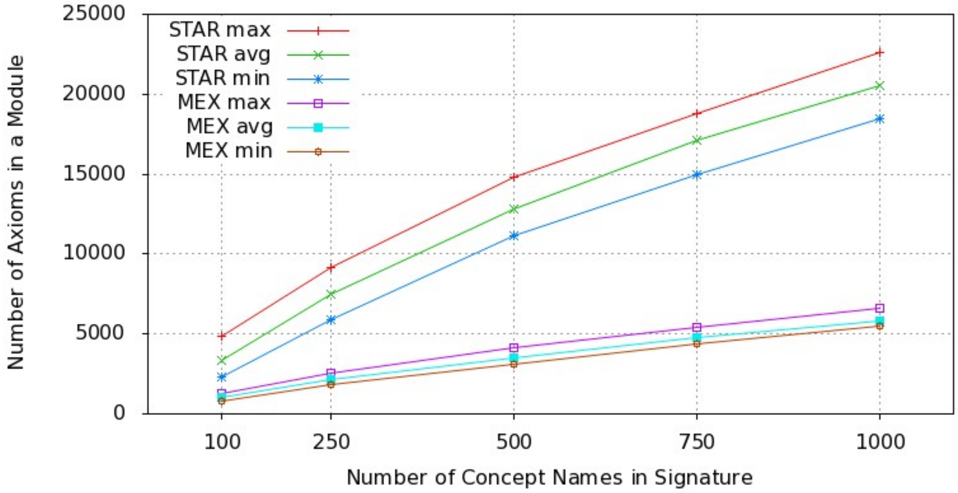


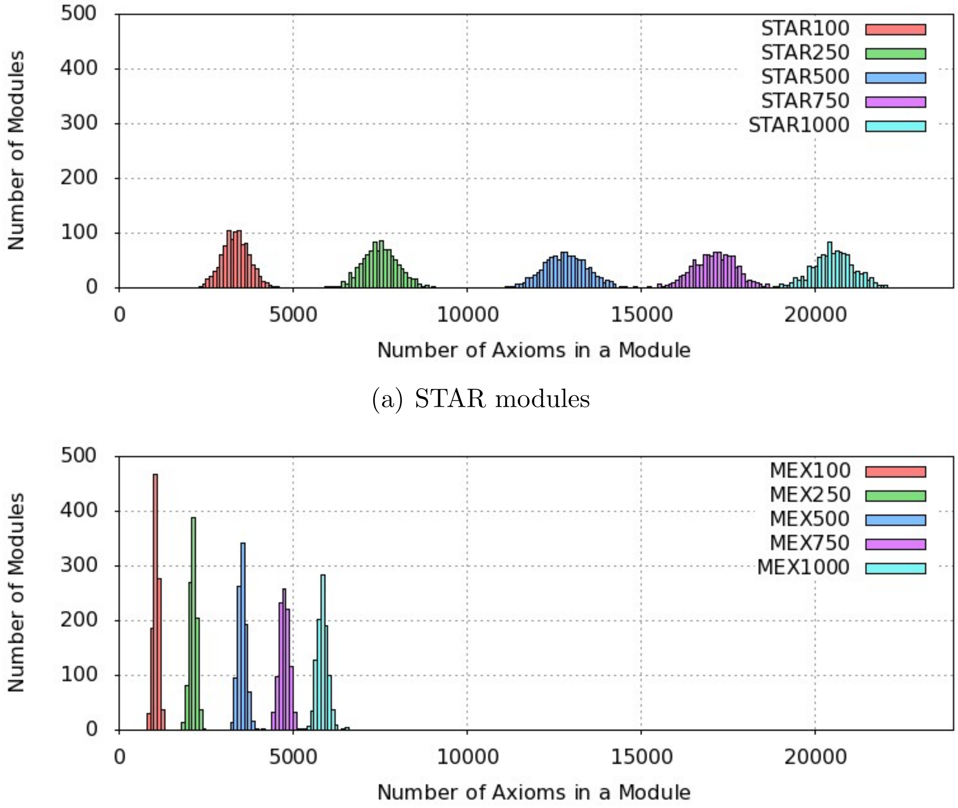
Figure 4: Sizes of STAR modules and MEX modules

'Diff' showing how much smaller.

Most of the cases in which there is no difference between the STAR-module and the MEX-module can be explained by Proposition 38. In the experiments above, only 89 out of 159 STAR-modules contain concept definitions. In all remaining cases the STAR-module coincides with the MEX-module simply because the extraction algorithm processes no concept definitions. Thus, there are only 13 cases in which the STAR-module coincides with the MEX-module and in which the module contains concept definitions.

For further experiments based on randomly generated signatures, we have used a more recent version of Snomed ct from 2012, which has 227 962 concept inclusions, 66 507 concept equations, 12 role inclusions, 294 470 concept names, and 62 role names. The experiments are based on randomly selected signatures of size between 100 and 1 000 concept names and 20 role names and they were carried out for 1 000 different signatures of each size. The special role name RoleGroup, which is used for an encoding trick, was included in every signature.

Figure 4 shows the maximal, minimal, and average module sizes depending on the size of the signature. Figure 5 shows the frequency distribution of the MEX-modules and STAR-modules. In each chart, there are five different histograms, one for each of the signature sizes ranging over 100, 250, 500, 750, and 1 000. Each of these histograms displays the distribution of the module



(a) STAR modules

(b) MEX modules

Figure 5: Size distribution of modules extracted from Snomed ct 2012

sizes of 1 000 extracted Snomed ct modules for randomly selected signatures of a certain size. For instance, the histogram labelled with STAR100 in Figure 5 shows the distribution of the size of 1 000 STAR modules for the signature size 100 extracted from Snomed ct.

In the random experiments, for every signature size the largest MEX-module is smaller than the smallest STAR-module and the average size of MEX-modules is up to 3.5 times smaller than the average size of STAR-modules. These figures do not match the outcome of the experiments with non-random subsets of signatures of Snomed ct and demand an explanation. By Proposition 38, both extraction methods will show little difference in the output when the TBox contains very few concept definitions. As mentioned above already, only 89 out of 159 STAR-modules for the non-random subset signatures contain concept definitions. This suggests that the subset signatures are highly non-random and can ‘hit’ the part of Snomed ct where terms are not fully defined. On the other hand, since roughly one fifth of Snomed ct statements are concept definitions, it is very likely that sufficiently large random signatures ’hit’ enough concept definitions to cause significant differences in the module sizes.

Finally, we report on the resource consumption by both methods. The maximum time and space consumed by MEX when extracting a module for 10 000 symbols is 1.35 seconds and 105.3 MByte. For 100 000 symbols (which is rather unrealistic), it is 4.48 seconds and 122.6 MByte. For STAR-modules the maximum time and space consumed when extracting a module for 10 000 symbols is 5.2 seconds and 552.9 MByte. When extracting a module for 100 000 symbols is 8.64 seconds and 556.4 MByte. (In both cases, we do not include the time needed to read the ontology.) The significantly higher memory requirements of the STAR-module computation is due to the fact that the OWL API is capable of handling TBoxes in expressive description logics, which requires some overhead in the data structures compared with MEX.

In summary, MEX thus scales extremely well to large TBoxes and rather often extracts significantly smaller modules than the best locality-based approach available.

9. Related Work

Modularity is currently a very active research topic in description logics and in ontologies in general, and a large number of articles have been published on the subject. For this reason, it is not possible to provide a fully comprehensive literature survey in this section. Instead, we focus on work that is closely related to the material presented in this paper and refer the reader to [39] for a recent collection of articles on modular ontologies, and to the proceedings of the workshop on modular ontologies (WoMO) which is held annually since 2006. We start by discussing related work on inseparability and conservative extensions and then proceed to modules, module extraction, and ontology decomposition.

A distinguishing feature of the notion of inseparability used in this paper is its model-theoretic nature. An important alternative is to define a deductive version of inseparability based on logical consequence within a languages. For example, one might call two $\mathcal{ALC}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ deductively

Σ-inseparable if for all Σ-concept inclusions $C \sqsubseteq D$ formulated in $\mathcal{ALC}$, we have $\mathcal{T}_1 \models C \sqsubseteq D$ iff $\mathcal{T}_2 \models C \sqsubseteq D$. Note that deductive inseparability is language dependent in the sense that the description logic language in which the logical consequences $C \sqsubseteq D$ are formulated has to be fixed and can be varied. For instance, two $\mathcal{ALC}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ may be Σ-inseparable w.r.t. $\mathcal{EL}$-consequences, but not w.r.t. $\mathcal{ALC}$-consequences. In contrast, there is no such language dependency in the notion of model-theoretic inseparability. In the following, we discuss both model-theoretic and deductive inseparability, as well as the closely related notion of (Σ-)conservative extension, i.e., Σ-inseparability of two TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ with $\mathcal{T}_1 \subseteq \mathcal{T}_2$. Just like inseparability, conservative extensions can be defined in a model-theoretic way and in a deductive way.

To the best of our knowledge, a model-theoretic notion of inseparability was first considered in a description logic context in [40], where it is shown that deciding model-theoretic conservative extensions is undecidable for general $\mathcal{ALC}$-TBoxes. Model-theoretic inseparability was then studied in more depth in [18], of which this paper is an extended version. Moreover, model-theoretic inseparability was considered in [14] in the context of the DL-Lite family of description logics, where it is shown that for TBoxes formulated in DL-Lite$^{\mathcal{N}}_{bool}$ and DL-Lite$^{\mathcal{N}}_{horn}$, model-theoretic inseparability is decidable and coNExp-hard. The precise complexity remains open.

In the context of ontologies formulated in first-order logic, model-theoretic notions of conservative extension have been proposed in [41] and used to modularise consistency proofs for ontologies in [22]. This line of research in modularity is partly based on ideas from modular software specification and verification, where conservative extensions and inseparability have been used since more than twenty years. For more details, we refer the reader to [42, 43, 44, 45, 46] where, in particular, the distinction between model-theoretic and deductive versions of conservative extensions and inseparability is introduced and discussed.

In description logic research, deductive notions of conservative extension and inseparability are investigated in [47, 40, 12, 14, 48, 49]. In contrast to model-theoretic inseparability, deductive inseparability is typically decidable; for example, it is ExpTime-complete for general $\mathcal{EL}$-TBoxes and 2ExpTime-complete for general $\mathcal{ALC}$ and $\mathcal{ALCI}$-TBoxes. In contrast, deductive conservative extensions turn out to be undecidable for the extension of $\mathcal{ALC}$ with nominals [50, 40].

We now discuss related work on modules and module extraction. As already mentioned in Section 8, the locality-based modules of [50] can be viewed as an approximation of model-theoretic modules and have the advantage that they can be efficiently computed also for TBoxes formulated in expressive description logics. Deductive inseparability-based module extraction algorithms are presented and discussed in [17, 35, 14, 49]. This work is based on the algorithm presented in Figure 2 and uses algorithms deciding deductive inseparability as an oracle. To the best of our knowledge the module extraction algorithms presented in this paper are the first work so far based on model-theoretic inseparability. A significant amount of work on modules and module extraction for ontologies is based on the syntactic structure of ontologies rather than on their interpretation as logical theories. We mention [51, 52, 53, 54] and refer also to the references therein.

Other related lines of research are the computation of uniform interpolants for ontologies and the automatic decomposition of ontologies into a set of modules. When computing uniform interpolants, one is interested in producing, for a given TBox $\mathcal{T}$ and signature Σ, a new TBox $\mathcal{T}'$ that uses only symbols in Σ and is Σ-inseparable from $\mathcal{T}$. Note that, in contrast to module extraction, (i) the TBox $\mathcal{T}'$ need not be a subset of $\mathcal{T}$, and (ii) the resulting TBox $\mathcal{T}'$ is not allowed to include any non-Σ-symbols at all. Unlike a module, the uniform interpolant $\mathcal{T}'$ asked for need in general not exist. Since uniform interpolants based on model-theoretic Σ-inseparability are typically not computable, research has mostly concentrated on deductive conservative extensions, see [55, 56] for the case of $\mathcal{ALC}$-TBoxes, [57, 58, 59] for the case of $\mathcal{EL}$-TBoxes, and [60, 14] for the case of DL-Lite TBoxes. A combination of uniform interpolation and module extraction has been studied in [61].

In ontology decomposition, the goal is to provide a partitioning of the ontology into independent modules, or of the ontology signature into independent subsignatures. Note that, in contrast to the kind of module extraction studied in this paper, the signatures of the desired modules are not given by the user; to the contrary, it is a main purpose of automatic ontology decomposition to reveal the symbols in the ontologies signature that 'belong together' and those that do not. Relevant work in this area can, for example, be found in [62, 63, 64, 65, 66].

10. Conclusion and Future Work

We have introduced a model-theoretic notion of inseparability for description logic TBoxes, analysed the complexity of deciding inseparability in various cases, and explored several forms of modules based on inseparability. This has led to module extraction algorithms for acyclic $\mathcal{ALCI}$-TBoxes and acyclic $\mathcal{ELI}$-TBoxes. Experiments for the $\mathcal{ELI}$ case show that our algorithm often extracts modules of significantly smaller size than other state-of-the-art techniques, and that our algorithms scale very well to large TBoxes such as SNOMED CT.

There are two main conclusions one can draw from this investigation: firstly, deciding model-theoretic inseparability for DLs that contain $\mathcal{ALC}$ is undecidable or of very high computational complexity, even for acyclic TBoxes and even if one TBox is empty. For such expressive DLs, practical tools for checking inseparability and extracting modules should thus be based on approximations of model-theoretic inseparability. One important known methodology for approximations is based on locality. The results in this paper suggest to consider, in addition, approximations based on syntactic dependencies.

Secondly, for the description logics $\mathcal{EL}$ and its extension $\mathcal{ELI}$, model-theoretic inseparability becomes tractable in important cases and can be deployed in practical applications without resorting to approximations. In addition, the tractability results for checking inseparability of $\mathcal{ELI}$-TBoxes from the empty TBox show that the complexity of standard reasoning (subsumption) can be significantly higher than the complexity of model-theoretic inseparability.

As future research, it would be interesting to fill the gaps in Table 1 by determining the decidability/complexity of inseparability of acyclic $\mathcal{EL}$ and $\mathcal{ELI}$-TBoxes in the general case. From an application perspective, it would also be relevant to study inseparability of acyclic TBoxes formulated in mild extensions of $\mathcal{EL}$ and $\mathcal{ELI}$-TBoxes, such as with role inclusions, transitive roles, range restrictions, and nominals. In addition, it is of interest to investigate model-theoretic inseparability for DL-Lite dialects introduced in [67] and for which inseparability has not been investigated in [14]. Examples include DL-Lite$_{core}$ and DL-Lite$_{core}$ with role inclusions.

From a practical viewpoint, it would be interesting to implement the module extraction algorithm for acyclic $\mathcal{ALCI}$-TBoxes. We are optimistic that the straightforward implementation using off-the-shelf QBF solvers can be applied to large scale acyclic non-$\mathcal{ELI}$-TBoxes such as the NCI thesaurus. A further extension to practical minimal module extraction with mild syntactic restrictions seems to be in reach even for acyclic $\mathcal{SHIQ}$-TBoxes.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments and suggestions for improvement of the submitted version. Boris Konev and Frank Wolter were supported by EPSRC grant EP/E065279/1. Carsten Lutz was supported by the DFG-funded SFB/TR8 "Spatial Cognition". Dirk Walther was supported by the MICINN project TIN2009-14562-CO5.

References

[1] G. Antoniou, F. van Harmelen, Web ontology language: OWL, in: Handbook on Ontologies, International Handbooks on Information Systems, Springer, 2004, pp. 67–92.

[2] The WWW Consortium, OWL 2 web ontology language: Profiles, Tech. rep., available from `http://www.w3.org/TR/owl2-profiles/`.

[3] D. Tsarkov, I. Horrocks, FaCT++ description logic reasoner: System description, in: Proceedings of the 3rd International Joint Conference on Automated Reasoning, IJCAR 2006, Vol. 4130 of Lecture Notes in Artificial Intelligence, Springer, Berlin, Heidelberg, 2006, pp. 292–297.

[4] V. Haarslev, K. Hidde, R. Möller, M. Wessel, The RacerPro knowledge representation and reasoning system, Semantic Web 3 (3) (2012) 267–277.

[5] E. Sirin, B. Parsia, B. Cuenca Grau, A. Kalyanpur, Y. Katz, Pellet: A practical OWL-DL reasoner, Journal of Web Semantics 5 (2) (2007) 51–53.

[6] R. Shearer, B. Motik, I. Horrocks, HermiT: A highly-efficient OWL reasoner, in: Proceedings of the 5th Workshop on OWL: Experiences and Directions, OWLED 2008, Vol. 432 of CEUR Workshop Proceedings, CEUR-WS.org, 2008.

[7] H. Knublauch, M. Musen, A. Rector, Editing description logic ontologies with the Protégé OWL plugin., in: Proceedings of the 17th international workshop on description logic, DL 2004, Vol. 104 of CEUR Workshop Proceedings, CEUR-WS.org, 2004.

[8] A. Kalyanpur, B. Parsia, E. Sirin, B. Cuenca Grau, J. Hendler, SWOOP: a web ontology editing browser, Journal of Web Semantics 4 (2) (2006) 144–153.

[9] J. Golbeck, G. Fragoso, F. Hartel, J. Hendler, J. Oberthaler, B. Parsia, The national cancer institute's thesaurus and ontology, Web Semantics: Science, Services and Agents on the World Wide Web 1 (1) (2011) 75–80.

[10] The Gene Ontology Consortium, The gene ontology: enhancements for 2011, Nucleic Acids Research 40 (D1) (2012) D559–D564.

[11] R. Cornet, N. de Keizer, Forty years of SNOMED: a literature review, BMC Medical Informatics and Decision Making 8 (Suppl 1) (2008) 1–6.

[12] B. Cuenca Grau, I. Horrocks, Y. Kazakov, U. Sattler, Modular reuse of ontologies: theory and practice, Journal of Artificial Intelligence Research (JAIR) 31 (2008) 273–318.

[13] B. Konev, C. Lutz, D. Walther, F. Wolter, Formal properties of modularisation, in: Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization, Vol. 5445 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2009, pp. 25–66.

[14] R. Kontchakov, F. Wolter, M. Zakharyaschev, Logic-based ontology comparison and module extraction, with an application to DL-Lite, Artificial Intelligence 174 (15) (2010) 1093–1141.

[15] C. Haase, C. Lutz, Complexity of subsumption in the $\mathcal{EL}$ family of description logics: Acyclic and cyclic TBoxes, in: Proceedings of the 18th European Conference on Artificial Intelligence, ECAI 2008, Vol. 178 of Frontiers in Artificial Intelligence and Applications, IOS Press, Amsterdam, Netherlands, 2008, pp. 25–29.

[16] B. Cuenca Grau, I. Horrocks, Y. Kazakov, U. Sattler, Just the right amount: extracting modules from ontologies, in: Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Alberta, Canada, 2007, pp. 717–726.

[17] U. Sattler, T. Schneider, M. Zakharyaschev, Which kind of module should I extract?, in: Proceedings of the 22nd International Workshop

on Description Logics, DL 2009, Vol. 477 of CEUR Workshop Proceedings, CEUR-WS.org, 2009.

[18] B. Konev, C. Lutz, D. Walther, F. Wolter, Semantic modularity and module extraction in description logics, in: Proceedings of the 8th European Conference on Artificial Intelligence, ECAI 2008, IOS Press, Amsterdam, Netherlands, 2008, pp. 55–59.

[19] F. Baader, D. Calvanes, D. McGuiness, D. Nardi, P. Patel-Schneider, The Description Logic Handbook: Theory, implementation and applications, Cambridge University Press, Cambridge, UK, 2003.

[20] F. Baader, S. Brandt, C. Lutz, Pushing the $\mathcal{EL}$ envelope further, in: Proceedings of the 4th Workshop on OWL: Experiences and Directions, OWLED 2008 DC, Washington, USA, 2008.

[21] F. Baader, S. Brandt, C. Lutz, Pushing the $\mathcal{EL}$ envelope, in: Proceedings of the 19th International Joint Conference on Artificial Intelligence, IJCAI 2005, Edinburgh, UK, 2005, pp. 364–369.

[22] O. Kutz, T. Mossakowski, A modular consistency proof for DOLCE, in: Proceedings of the 25th AAAI Conference on Artificial Intelligence, AAAI 2011, AAAI Press, Menlo Park, CA, USA, 2011, pp. 227–234.

[23] D. Tsarkov, I. Palmisano, Divide et impera: Metareasoning for large ontologies, in: Proceedings of OWL: Experiences and Directions Workshop, OWLED 2012, Vol. 849 of CEUR Workshop Proceedings, CEUR-WS.org, 2012.

[24] A. Armas Romero, B. Cuenca Grau, I. Horrocks, Modular combination of reasoners for ontology classification, in: Proceedings of the 2012 International Workshop on Description Logics, DL-2012, Vol. 846 of CEUR Workshop Proceedings, CEUR-WS.org, 2012.

[25] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati, DL-Lite: Tractable description logics for ontologies, in: Proceedings of the 20th National Conference on Artificial Intelligence, AAAI 2005, AAAI Press / The MIT Press, Menlo Park, CA, USA, 2005, pp. 602–607.

[26] A. Poggi, D. Lembo, D. Calvanese, G. De Giacomo, M. Lenzerini, R. Rosati, Linking data to ontologies, J. Data Semantics 10 (2008) 133–173.

[27] P. Bonatti, C. Lutz, F. Wolter, Description logics with circumscription, in: Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning, KR 2006, AAAI Press, Menlo Park, CA, USA, 2006, pp. 400–410.

[28] P. Bonatti, C. Lutz, F. Wolter, The complexity of circumscription in description logic, J. Artificial Intelligence Research (JAIR) 35 (2009) 717–773.

[29] B. Cuenca Grau, B. Motik, Y. Kazakov, Import-by-query: Ontology reasoning under access limitations, in: Proceedings of the 21st International Joint Conference on Artificial Intelligence, IJCAI 2009, AAAI Press, Menlo Park, CA, USA, 2009, pp. 727–732.

[30] B. Cuenca Grau, B. Motik, Pushing the limits of reasoning over ontologies with hidden content, in: Proceedings of the Twelfth International Conference on Principles of Knowledge Representation and Reasoning, KR 2010, AAAI Press, Menlo Park, CA, USA, 2010, pp. 214–224.

[31] F. Wolter, M. Zakharyaschev, Modal decision problems, in: J. van Benthem, P. Blackburn, F. Wolter (Eds.), Handbook of Modal Logic, Elsevier, Amsterdam, Netherlands, 2006.

[32] C. Lutz, R. Piro, F. Wolter, Description logic TBoxes: Model-theoretic characterizations and rewritability, in: Proceedings of the 22nd International Joint Conference on Artificial Intelligence, IJCAI 2011, AAAI Press, Menlo Park, CA, USA, 2011, pp. 983–988.

[33] C. Peschiera, L. Pulina, A. Tacchella, U. Bubeck, O. Kullmann, I. Lynce, The seventh QBF solvers evaluation (QBFEVAL'10), in: Proceedings of Theory and Applications of Satisfiability Testing, SAT 2010, Vol. 6175 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2010, pp. 237–250.

[34] H. Kleine Büning, M. Karpinski, A. Flögel, Resolution for quantified Boolean formulas, Information and Computation 117 (1) (1995) 12–18.

[35] R. Kontchakov, L. Pulina, U. Sattler, T. Schneider, P. Selmer, F. Wolter, M. Zakharyaschev, Minimal module extraction from DL-Lite ontologies using QBF solvers, in: Proceedings of the 21st International Joint Conference on Artificial Intelligence, IJCAI 2009, AAAI Press, Menlo Park, CA, USA, 2009, pp. 836–841.

[36] M. Horridge, S. Bechhofer, The OWL API: A java API for OWL ontologies, Semantic Web 2 (1) (2011) 11–21.

[37] C. del Vescovo, P. Klinov, B. Parsia, U. Sattler, T. Schneider, D. Tsarkov, Empirical study of logic-based modules: Cheap is cheerful, Tech. rep., University of Manchester (2013).

[38] B. Suntisrivaraporn, Module extraction and incremental classification: A pragmatic approach for ontologies, in: Proceedings of the 5th European Semantic Web Conference, ESWC 2008, Vol. 5021 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2008, pp. 230–244.

[39] H. Stuckenschmidt, C. Parent, S. Spaccapietra (Eds.), Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization, Vol. 5445 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2009.

[40] C. Lutz, D. Walther, F. Wolter, Conservative extensions in expressive description logics, in: Proceedings of the 20th International Joint Conference on Artificial Intelligence, IJCAI 2007, AAAI Press, Menlo Park, CA, USA, 2007, pp. 453–458.

[41] O. Kutz, T. Mossakowski, Conservativity in structured ontologies, in: Proceedings of the 18th European Conference on Artificial Intelligence, ECAI 2008, IOS Press, Amsterdam, Netherlands, 2008, pp. 89–93.

[42] P. Byers, D. Pitt, Conservative extensions: a cautionary note, Bulletin of the EATCS 41 (1990) 196–201.

[43] T. Maibaum, Conservative extensions, interpretations between theories and all that!, in: Proceedings of the 7th International Joint CAAP/FASE Conference on Theory and Practice of Software Development, TAPSOFT 1997, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 1997, pp. 40–66.

[44] P. Veloso, Yet another cautionary note on conservative extensions: a simple case with a computing flavour, Bulletin of the EATCS (1992) 188–193.

[45] P. Veloso, S. Veloso, Some remarks on conservative extensions. A Socratic dialog, Bulletin of the EATCS (1991) 189–198.

[46] J. Goguen, R. Burstall, Institutions: Abstract model theory for specification and programming, Journal of the ACM 39 (1) (1992) 95–146.

[47] S. Ghilardi, C. Lutz, F. Wolter, Did I damage my ontology? A case for conservative extensions in description logic, in: Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning, KR 2006), AAAI Press, Menlo Park, CA, USA, 2006, pp. 187–197.

[48] C. Lutz, F. Wolter, Deciding inseparability and conservative extensions in the description logic $\mathcal{EL}$, Journal of Symbolic Computing 45 (2) (2010) 194–228.

[49] B. Konev, R. Kontchakov, M. Ludwig, T. Schneider, F. Wolter, M. Zakharyaschev, Conjunctive query inseparability of OWL 2 QL TBoxes, in: Proceedings of the 25th Conference on Artificial Intelligence, AAAI 2011, AAAI Press, Menlo Park, CA, USA, 2011, pp. 221–226.

[50] B. Cuenca Grau, I. Horrocks, Y. Kazakov, U. Sattler, Extracting modules from ontologies: A logic-based approach, in: Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization, Springer, Berlin, Heidelberg, 2009, pp. 159–186.

[51] J. Seidenberg, A. L. Rector, Web ontology segmentation: analysis, classification and use, in: Proceedings of the 15th international conference on World Wide Web, WWW 2006, Edinburgh, UK, 2006, pp. 13–22.

[52] N. Fridman Noy, M. A. Musen, Traversing ontologies to extract views, in: Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization, Vol. 5445 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2009, pp. 245–260.

[53] P. Doran, V. Tamma, L. Iannone, Ontology module extraction for ontology reuse: An ontology engineering perspective, in: Proceedings of

the 16th ACM Conference on Information and Knowledge Management, CIKM 2007, ACM, 2007, pp. 61–70.

[54] I. Palmisano, V. Tamma, T. Payne, P. Doran, Task oriented evaluation of module extraction techniques, in: Proceedings of the 8th International Semantic Web Conference, ISWC 2009, Vol. 5823 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2009, pp. 130–145.

[55] K. Wang, Z. Wang, R. W. Topor, J. Z. Pan, G. Antoniou, Concept and role forgetting in ALC ontologies, in: Proceedings of the 8th International Semantic Web Conference, ISWC 2009, Vol. 5823 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2009, pp. 666–681.

[56] C. Lutz, F. Wolter, Foundations for uniform interpolation and forgetting in expressive description logics, in: Proceedings of the 22nd International Joint Conference on Artificial Intelligence, IJCAI 2011, AAAI Press, Menlo Park, CA, USA, 2011, pp. 989–995.

[57] N. Nikitina, S. Rudolph, ExpExpExplosion: Uniform interpolation in general EL terminologies, in: Proceedings of the 20th European Conference on Artificial Intelligence, ECAI 2012, IOS Press, Amsterdam, Netherlands, 2012, pp. 618–623.

[58] C. Lutz, I. Seylan, F. Wolter, An automata-theoretic approach to uniform interpolation and approximation in the description logic EL, in: Proceedings of the 13th International Conference on Principles of Knowledge Representation, KR 2012, AAAI Press, Menlo Park, CA, USA, 2012.

[59] B. Konev, D. Walther, F. Wolter, Forgetting and uniform interpolation in large-scale description logic terminologies, in: Proceedings of the 21st International Joint Conference on Artificial Intelligence, IJCAI 2009, AAAI Press, Menlo Park, CA, USA, 2009, pp. 830–835.

[60] Z. Wang, K. Wang, R. W. Topor, J. Z. Pan, Forgetting for knowledge bases in DL-Lite, Annals of Mathematics and Artificial Intelligence 58 (1-2) (2010) 117–151.

[61] N. Nikitina, B. Glimm, Hitting the sweetspot: Economic rewriting of knowledge bases, in: Proceedings of the 11th International Semantic

Web Conference, ISWC 2012, Vol. 7649 of Lecture Notes in Computer Science, 2012, pp. 394–409.

[62] H. Stuckenschmidt, M. Klein, Structure-based partitioning of large class hierarchies, in: Proceedings of the 3rd International Semantic Web Conference, ISWC 2004, Vol. 3298 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2004, pp. 289–303.

[63] B. Cuenca Grau, B. Parsia, E. Sirin, A. Kalyanpur, Modularity and web ontologies, in: Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning, KR 2006, AAAI Press, Menlo Park, CA, USA, 2006, pp. 198–209.

[64] H. Stuckenschmidt, A. Schlicht, Structure-based partitioning of large ontologies, in: Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization, Vol. 5445 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2009, pp. 187–210.

[65] B. Konev, C. Lutz, D. Ponomaryov, F. Wolter, Decomposing description logic ontologies, in: Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning, KR 2010, AAAI Press, Menlo Park, CA, USA, 2010, pp. 236–246.

[66] C. Del Vescovo, B. Parsia, U. Sattler, T. Schneider, The modular structure of an ontology: Atomic decomposition, in: Proceedings of the 22nd International Joint Conference on Artificial Intelligence, IJCAI 2011, AAAI Press, Menlo Park, CA, USA, 2011, pp. 2232–2237.

[67] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family, Journal of Automated Reasoning 39 (3) (2007) 385–429.

Appendix A. Proofs for Section 4

Lemma A.1. *Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be $\mathcal{ALCI}$-TBoxes and Σ a concept signature. Assume that $\mathcal{T}_1$ does not Σ-entail $\mathcal{T}_2$. Then there exists a finite model $\mathcal{I}$ of $\mathcal{T}_1$ such that there is no model $\mathcal{J}$ of $\mathcal{T}_2$ which coincides with $\mathcal{I}$ on Σ.*

Proof. Assume $\mathcal{T}_1$ does not Σ-entail $\mathcal{T}_2$. Let $\mathcal{I}$ be a model of $\mathcal{T}_1$ such that there is no model $\mathcal{J}$ of $\mathcal{T}_2$ which coincides with $\mathcal{I}$ on Σ. Using $\mathcal{I}$, we construct a finite model as required in the lemma using the standard filtration technique. Define an equivalence relation $\sim$ of $\Delta^{\mathcal{I}}$ by setting for all $d_1, d_2 \in \Delta^{\mathcal{I}}$:

- $d_1 \sim d_2$ if, and only if, $d_1 \in D^{\mathcal{I}} \Leftrightarrow d_2 \in D^{\mathcal{I}}$ for all $D \in \Sigma \cup \mathsf{sub}(\mathcal{T}_1)$, where $\mathsf{sub}(\mathcal{T}_1)$ is the set of subconcepts of concepts in $\mathcal{T}_1$.

Let $[d] = \{d' \in \Delta^{\mathcal{I}} \mid d \sim d'\}$, for $d \in \Delta^{\mathcal{I}}$. Now define $\mathcal{I}'$ as follows

- $\Delta^{\mathcal{I}'} = \{[d] \mid d \in \Delta^{\mathcal{I}}\}$;
- for all $A \in \mathsf{N_C}$, set $[d] \in A^{\mathcal{I}'}$ if, and only if, $d \in A^{\mathcal{I}}$;
- for all $r \in \mathsf{N_R}$, set $([d], [d']) \in r^{\mathcal{I}'}$ if, and only if, there are $e \in [d]$ and $e' \in [d']$ such that $(e, e') \in r^{\mathcal{I}}$.

One can show that $\mathcal{I}'$ is a model of $\mathcal{T}_1$. It remains to show that there does not exist any model $\mathcal{J}'$ of $\mathcal{T}_2$ with $\mathcal{J}'|_\Sigma = \mathcal{I}'|_\Sigma$. Suppose the contrary. Let $\mathcal{J}'$ be a model of $\mathcal{T}_2$ such that $\mathcal{J}'|_\Sigma = \mathcal{I}'|_\Sigma$. We derive a contradiction by defining a model $\mathcal{J}$ of $\mathcal{T}_2$ such that $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. In detail, let $A^{\mathcal{J}} = A^{\mathcal{I}}$ for all $A \in \Sigma$ and define

- for all $A \in \mathsf{N_C} \setminus \Sigma$: $d \in A^{\mathcal{J}}$ iff $[d] \in A^{\mathcal{J}'}$;
- for all $r \in \mathsf{N_R}$: $(d, d') \in r^{\mathcal{J}}$ iff $([d], [d']) \in r^{\mathcal{J}'}$.

$\mathcal{J}$ is well-defined since Σ does not contain any role names. By definition, $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. Moreover, since $\mathcal{J}'$ is a model of $\mathcal{T}_2$, $\mathcal{J}$ is a model of $\mathcal{T}_2$ as well. We have derived a contradiction. □

Theorem 10. *Given acyclic $\mathcal{ALC}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ and a concept signature Σ, it is* coNExp$^{\text{NP}}$*-hard to decide whether $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$. This is even true when $\mathcal{T}_1 \cup \mathcal{T}_2$ is acyclic.*

Proof. We have to prove the coNExp$^{\mathrm{NP}}$-lower bound under the condition that $\mathcal{T}_1 \cup \mathcal{T}_2$ is an acyclic $\mathcal{ALC}$-TBox. The proof is by reduction of the complement of the concept satisfiability problem w.r.t. singleton concept-circumscribed ABoxes which is NExp$^{\mathrm{NP}}$-hard (cf. [28] (Theorem 15)).

In detail, the problem we reduce is defined as follows: given an ABox $\mathcal{A} = \{C(a)\}$ with C an $\mathcal{ALC}$-concept, an $\mathcal{ALC}$-concept C_0 and sets M and F of minimised and fixed concept names, decide whether C_0 is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{A})$; i.e., whether there exists a model $\mathcal{I}$ of $\mathsf{Circ}_{M,F}(\mathcal{A})$ such that $C_0^{\mathcal{I}} \neq \emptyset$. Here $\mathcal{I}$ is a model of $\mathsf{Circ}_{M,F}(\mathcal{A})$ iff it is a model of $\mathcal{A}$ and there does not exist any model $\mathcal{J}$ of $\mathcal{A}$ such that $\mathcal{J} <_{M,F} \mathcal{I}$ (where we assume that the interpretation of the individual name a is fixed; i.e., $a^{\mathcal{I}} = a^{\mathcal{J}}$ whenever $\mathcal{J} <_{M,F} \mathcal{I}$).

Assume $\mathcal{A} = \{C(a)\}$, C_0, and sets M and F as above are given. We construct acyclic $\mathcal{ALC}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ with $\mathcal{T}_1 \cup \mathcal{T}_2$ acyclic and a signature Σ consisting of concept names only such that $\mathcal{T}_1$ Σ-entails $\mathcal{T}_2$ if, and only if, C_0 is not satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{A})$.

To construct $\mathcal{T}_1$ and $\mathcal{T}_2$, introduce for every $X \in M \cup F \cup \mathsf{sig}(C) \cup \mathsf{sig}(C_0)$ a fresh X' and denote by C' the resulting concept when every X is replaced by X'. Take fresh role names $\mathsf{aux}_1, \ldots, \mathsf{aux}_5$ and concept names A, B_0, B_1, B_2. We set

$$\mathcal{T}_1 = \{B_0 \sqsubseteq B_1 \sqcap (\exists \mathsf{aux}_1.(A \sqcap C) \sqcap \exists \mathsf{aux}_2.C_0), \quad B_2 \equiv \neg(\exists \mathsf{aux}_1.(A \sqcap C) \sqcap \exists \mathsf{aux}_2.C_0)\}.$$

$\mathcal{T}_2$ consists of

- $B_1 \sqsubseteq (\exists \mathsf{aux}_3.B_2) \sqcup (\exists \mathsf{aux}_4.(A \sqcap C') \sqcap \exists \mathsf{aux}_5. \bigsqcup_{B \in M}(B \sqcap \neg B'))$,
- $B' \sqsubseteq B$, for all $B \in M$, and
- $B' \equiv B$, for all $B \in F$.

Let

$$\Sigma = \{A, B_0, B_1, B_2\} \cup M \cup F.$$

Observe that $\mathcal{T}_1 \cup \mathcal{T}_2$ is acyclic. It remains to show the following

Claim. C_0 is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{A})$ if, and only if, $\mathcal{T}_1$ does not Σ-entail $\mathcal{T}_2$.

Proof of Claim. We start with the direction from left to right. Let $\mathcal{I}$ be a model of $\mathsf{Circ}_{M,F}(\mathcal{A})$ with $C_0^{\mathcal{I}} \neq \emptyset$. Since $\mathsf{aux}_1, \mathsf{aux}_2, A, B_0, B_1, B_2$ are fresh symbols that do not occur in C_0 nor C, we may assume that $\mathcal{I}$ is a model of $\mathcal{T}_1$ such that $B_0^{\mathcal{I}} = B_1^{\mathcal{I}} = \Delta^{\mathcal{I}}$ and $A^{\mathcal{I}} = \{a^{\mathcal{I}}\}$ and $B_2^{\mathcal{I}} = \emptyset$. We show that there does not exist any model $\mathcal{J}$ of $\mathcal{T}_2$ such that $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. For a proof by contradiction, assume there exists such an interpretation $\mathcal{J}$. We set $a^{\mathcal{J}} = a^{\mathcal{I}}$. Observe that $(\exists \mathsf{aux}_3.B_2)^{\mathcal{J}} = \emptyset$ since $B_2^{\mathcal{I}} = \emptyset$ and $B_2 \in \Sigma$. We also have $B_1^{\mathcal{J}} = \Delta^{\mathcal{I}}$ since $B_1^{\mathcal{I}} = \Delta^{\mathcal{I}}$ and $B_1 \in \Sigma$. Hence

$$(\exists \mathsf{aux}_4.(A \sqcap C') \sqcap \exists \mathsf{aux}_5. \bigsqcup_{B \in M} (B \sqcap \neg B'))^{\mathcal{J}} = \Delta^{\mathcal{I}}$$

As $A \in \Sigma$ and $A^{\mathcal{I}} = \{a^{\mathcal{I}}\}$, we have $a^{\mathcal{J}} \in C'^{\mathcal{J}}$ and there exists $B \in M$ with $B'^{\mathcal{J}} \subsetneq B^{\mathcal{I}}$. We show that this contradicts the condition that there does not exist any $\mathcal{J}' <_{M,F} \mathcal{J}$ that is a model of $\mathcal{A}$: such a $\mathcal{J}'$ is obtained from $\mathcal{J}$ by setting $X^{\mathcal{J}'} := X'^{\mathcal{J}}$ for all symbols X from $\mathsf{sig}(C) \cup \mathsf{sig}(C_0)$. Then $a^{\mathcal{J}'} \in C^{\mathcal{J}'}$ since $a^{\mathcal{J}} \in C'^{\mathcal{J}}$ and so $\mathcal{J}'$ is a model of $\mathcal{A}$. We have $\mathcal{J}' <_{M,F} \mathcal{J}$ since $B^{\mathcal{J}'} \subseteq B^{\mathcal{J}}$ for $B \in M$ (since $B' \sqsubseteq B \in \mathcal{T}_2$), $B^{\mathcal{J}'} = B^{\mathcal{J}}$ for $B \in F$ (since $B' \equiv B \in \mathcal{T}_2$) and we have that there exists $B \in M$ with $B^{\mathcal{J}'} \subsetneq B^{\mathcal{J}}$ since there exists $B \in M$ with $B'^{\mathcal{J}} \subsetneq B^{\mathcal{J}}$.

Conversely, assume that $\mathcal{T}_1 \not\models_\Sigma \mathcal{T}_2$. Take a model $\mathcal{I}$ of $\mathcal{T}_1$ such that there does not exist any model $\mathcal{J}$ of $\mathcal{T}_2$ with $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$.

Observe first that $B_2^{\mathcal{I}} = \emptyset$: if this is not the case then we can satisfy the inclusions of $\mathcal{T}_2$ in a model $\mathcal{J}$ with $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$ by taking some $d \in B_2^{\mathcal{J}}$ and set $\mathsf{aux}_3^{\mathcal{J}} = \Delta^{\mathcal{I}} \times \{d\}$ and $B'^{\mathcal{J}} = B^{\mathcal{I}}$, for all $B \in M \cup F$. Then, from $B_2 \equiv \neg(\exists \mathsf{aux}_1.(A \sqcap C) \sqcap \exists \mathsf{aux}_2.C_0) \in \mathcal{T}_1$, it follows that $(\exists \mathsf{aux}_1.(A \sqcap C) \sqcap \exists \mathsf{aux}_2.C_0)^{\mathcal{I}} \neq \emptyset$. We may assume that $a^{\mathcal{I}} \in (A \sqcap C)^{\mathcal{I}}$. Then $\mathcal{I}$ is a model of $\mathcal{A}$ and $C_0^{\mathcal{I}} \neq \emptyset$. It remains to show that there does not exist a model $\mathcal{J}$ of $\mathcal{A}$ such that $\mathcal{J} <_{M,F} \mathcal{I}$. For a proof by contradiction assume that $\mathcal{J}$ is such a model of $\mathcal{A}$. We show that then there exists a model $\mathcal{J}'$ of $\mathcal{T}_2$ such that $\mathcal{J}'|_\Sigma = \mathcal{I}|_\Sigma$ and thus obtain a contradiction. Define $\mathcal{J}'$ by setting $X'^{\mathcal{J}'} := X^{\mathcal{J}}$, for all $X \in \mathsf{sig}(C) \cup \mathsf{sig}(C_0)$, $\mathsf{aux}_4^{\mathcal{J}'} := \Delta^{\mathcal{I}} \times \{a^{\mathcal{I}}\}$, and $\mathsf{aux}_5^{\mathcal{J}'} = \Delta^{\mathcal{J}} \times \{e\}$, where $e \in B^{\mathcal{J}} \setminus B^{\mathcal{I}}$ for some $B \in M$ that witnesses that $\mathcal{J} <_{M,F} \mathcal{I}$. It is readily checked that $\mathcal{J}'|_\Sigma \equiv_\Sigma \mathcal{I}|_\Sigma$. □

Appendix B. Proofs for Section 5

We start with some preliminaries. Let $\mathcal{I}$ be an interpretation and $D \subseteq \Delta^{\mathcal{I}}$ non-empty. Then the *subinterpretation $\mathcal{I}_D$ of $\mathcal{I}$ induced by D* is defined as follows:

- $\Delta^{\mathcal{I}_D} = D$;
- $r^{\mathcal{I}_D} = r^{\mathcal{I}} \cap (D \times D)$, for all $r \in \mathsf{N_R}$;
- $A^{\mathcal{I}_D} = A^{\mathcal{I}} \cap D$, for all $A \in \mathsf{N_C}$.

Let Σ be signature. Call a non-empty subset D of $\Delta^{\mathcal{I}}$ Σ-*closed* if for all $d, d' \in \Delta^{\mathcal{I}}$ and role names $r \in \Sigma$: if $d \in D$ and $(d, d') \in r^{\mathcal{I}}$, then $d' \in D$. (Note that the concept names in Σ are not relevant for being Σ-closed.) If D is Σ-closed, then $\mathcal{I}_D$ is called a Σ-*generated subinterpretation of* $\mathcal{I}$. The following lemma is straightforward.

Lemma B.1. *If $\mathcal{I}_D$ is a Σ-generated subinterpretation of $\mathcal{I}$, then $C^{\mathcal{I}_D} = C^{\mathcal{I}} \cap D$, for all $\mathcal{ALC}$-concepts C such that $\mathsf{sig}(C) \subseteq \Sigma$. In particular, if $\mathcal{I}$ is a model of an $\mathcal{ALC}$-TBox $\mathcal{T}$ with $\mathsf{sig}(\mathcal{T}) \subseteq \Sigma$, then $\mathcal{I}_D$ is a model of $\mathcal{T}$.*

Let $x \in \Delta^{\mathcal{I}}$. Then $\mathcal{I}_D$ is called the (x, Σ)-*generated subinterpretation of* $\mathcal{I}$ if D is the smallest Σ-closed subset of $\Delta^{\mathcal{I}}$ containing x. In this case we set $\mathcal{I}_x := \mathcal{I}_D$.

Theorem 15. *Given an $\mathcal{EL}$-TBox $\mathcal{T}$ and signature Σ, it is undecidable whether $\mathcal{T} \equiv_\Sigma \emptyset$.*

Proof. The proof is by reduction of the undecidable word problem for semigroups. We use $\vec{r}$ to denote a *composition* $\vec{r} = r_1 \circ r_2 \circ \cdots \circ r_k$ of roles names, where $k \geq 1$ and $r_i \in \mathsf{N_R}$ for all $i \in \{1, \ldots, k\}$. The semantics of role compositions is defined as usual. In detail, the *extension* $\vec{r}^{\mathcal{I}}$ of $\vec{r} = r_1 \circ \cdots \circ r_k$ in $\mathcal{I}$ is defined by

$$\vec{r}^{\mathcal{I}} = \{(d, d') \mid \exists y_0 \ldots y_k . (y_0 = d) \wedge \bigwedge_{i=1..k} (y_{i-1}, y_i) \in r_i^{\mathcal{I}} \wedge (y_k = d')\}.$$

For compositions $\vec{r}$ and $\vec{s}$, we define

- $\mathcal{I} \models \vec{r} \sqsubseteq \vec{s}$ iff $\vec{r}^{\mathcal{I}} \subseteq \vec{s}^{\mathcal{I}}$,
- $\mathcal{I} \models \vec{r} \equiv \vec{s}$ iff $\vec{r}^{\mathcal{I}} = \vec{s}^{\mathcal{I}}$,

and set

$$\{\vec{r_1} \equiv \vec{s}_1, \dots, \vec{r_n} \equiv \vec{s}_n\} \models_d \vec{r} \sqsubseteq \vec{s} \tag{B.1}$$

iff for all interpretations $\mathcal{I}$ such that for each $r \in \mathsf{N_R}$, the *domain* of $r^{\mathcal{I}}$ (that is, the set $\{d \in \Delta^{\mathcal{I}} \mid \exists d' : (d, d') \in r^{\mathcal{I}}\}$) coincides with $\Delta^{\mathcal{I}}$:

$$(\forall i \leq n : \quad \mathcal{I} \models \vec{r_i} \equiv s_i) \quad \Rightarrow \quad \mathcal{I} \models \vec{r} \sqsubseteq \vec{s}. \tag{B.2}$$

Note that $\{\vec{r_1} \equiv \vec{s}_1, \dots, \vec{r_n} \equiv \vec{s}_n\} \models_d \vec{r} \sqsubseteq \vec{s}$ holds if, and only if, (B.2) holds for all interpretations $\mathcal{I}$ in which each $r^{\mathcal{I}}$, $r \in \mathsf{N_R}$, is a function. Thus, in the following proof it will not make any difference whether we work with interpretations in which each role is interpreted as a function or with interpretations in which the domain of each role coincides with the domain of the interpretation. The problem whether (B.1) holds coincides with the word problem for semigroups and is undecidable.

In what follows $\exists \vec{r}.C$ stands for the $\mathcal{EL}$-concept $\exists r_1. \cdots \exists r_k.C$, where $\vec{r} = r_1 \circ \cdots \circ r_k$.

For the reduction assume that a problem instance $\{\vec{r_1} \equiv \vec{s}_1, \dots, \vec{r_n} \equiv \vec{s}_n\}$ and $\vec{r} \equiv \vec{s}$ is given. Let Σ be the set of all roles names that occur in the problem instance and a fresh concept name A. Additionally, take fresh concept names A_i, B_i, N_i, E, M and fresh role names u_0, u_i, for $i \in \{1, \dots, n\}$. Let $\mathcal{T}$ be the set consisting of all instances of the following inclusions: for all role names $v \in \Sigma$ and all $i \in \{1, \dots, n\}$,

$$\begin{align}
\exists \vec{r_i}.\top \sqcap \exists \vec{s_i}.\top &\sqsubseteq \exists u_i.(\exists \vec{r_i}.A_i \sqcap \exists \vec{s_i}.B_i); \tag{B.3}\\
A_i \sqcap B_i &\sqsubseteq N_i; \tag{B.4}\\
\exists v.N_i &\sqsubseteq N_i; \tag{B.5}\\
\top &\sqsubseteq \exists u_0.E; \tag{B.6}\\
E \sqcap \bigsqcap_{v \in \Sigma} \exists v.\top &\sqsubseteq M; \tag{B.7}\\
\exists v.M &\sqsubseteq M; \tag{B.8}\\
\exists \vec{r}.A \sqcap M \sqcap \bigsqcap_{1 \leq j \leq n} N_j &\sqsubseteq \exists \vec{s}.A. \tag{B.9}
\end{align}$$

We show the following

Claim. $\mathcal{T} \equiv_\Sigma \emptyset$ if, and only if, $(B.1)$ holds.

Assume first that $\mathcal{T} \not\equiv_\Sigma \emptyset$. We show that $(B.1)$ does not hold. Take an interpretation $\mathcal{I}$ such that there does not exist any model $\mathcal{J}$ of $\mathcal{T}$ with

$\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. We first show that $\mathcal{I}$ is a model of $\vec{r}_i \equiv \vec{s}_i$, for $1 \leq i \leq n$, such that for all $v \in \Sigma$ the domain of $v^\mathcal{I}$ coincides with $\Delta^\mathcal{I}$. Formally:

(a) $\mathcal{I} \models \top \sqsubseteq \exists v.\top$, for all role names $v \in \Sigma$;

(b) $\vec{r}_i^{\mathcal{I}} = \vec{s}_i^{\mathcal{I}}$, for all $i \in \{1, \ldots, n\}$.

We show both, (a) and (b), by contradiction. Suppose (a) does not hold. Let $\mathcal{J}$ be an interpretation that coincides with $\mathcal{I}$ on the symbols in Σ and that interprets the symbols in $\mathsf{sig}(\mathcal{T}) \setminus \Sigma$ as follows. Let x be a point without v-successor, for some role $v \in \Sigma$. Set $E^\mathcal{J} = \{x\}$, $M^\mathcal{J} = \emptyset$, and, for all $i \in \{1, \ldots, n\}$, set $A_i^\mathcal{J}, B_i^\mathcal{J}, N_i^\mathcal{J}$ to $\Delta^\mathcal{J}$ and $u_0^\mathcal{J}, u_i^\mathcal{J}$ to $\Delta^\mathcal{J} \times \Delta^\mathcal{J}$. It is readily checked that $\mathcal{J}$ satisfies all inclusions in (B.3) to (B.9) and, thus, $\mathcal{J}$ is a model of $\mathcal{T}$. We have derived a contradiction. Now consider (b). Suppose (b) does not hold, i.e., there is a pair $(x, y) \in \vec{r}_i^{\mathcal{I}}$ such that $(x, y) \notin \vec{s}_i^{\mathcal{I}}$, for some $i \in \{1, \ldots, n\}$. By (a), there is a point $z \in \Delta^\mathcal{I}$ such that $(x, z) \in \vec{s}_i^{\mathcal{I}}$. Obtain $\mathcal{J}$ from $\mathcal{I}$ by interpreting the symbols in $\mathsf{sig}(\mathcal{T}) \setminus \Sigma$ as follows. Set $u_i^\mathcal{J} = \{(x', x) \mid x' \in \Delta^\mathcal{J}\}$, $A_i^\mathcal{J} = \{y\}$, $B_i^\mathcal{J} = \{z\}$, and $N_i^\mathcal{J} = \emptyset$. Set $E^\mathcal{J}, M^\mathcal{J}, A_j^\mathcal{J}, B_j^\mathcal{J}, N_j^\mathcal{J}$ to $\Delta^\mathcal{J}$ and $u_0^\mathcal{J}, u_j^\mathcal{J}$ to $\Delta^\mathcal{J} \times \Delta^\mathcal{J}$, for all $j \in \{1, \ldots, n\}$ and $j \neq i$. It is readily checked that $\mathcal{J}$ satisfies all inclusions in (B.3) to (B.9). Thus, $\mathcal{J}$ is a model of $\mathcal{T}$. We have derived a contradiction.

We now show that (B.1) does not hold. Since we have (a) and (b) already, it remains to show that $\mathcal{I} \not\models \vec{r} \sqsubseteq \vec{s}$. Obtain $\mathcal{J}$ from $\mathcal{I}$ by interpreting the symbols in $\mathsf{sig}(\mathcal{T}) \setminus \Sigma$ as follows. Set $E^\mathcal{J}, M^\mathcal{J}, A_i^\mathcal{J}, B_i^\mathcal{J}, N_i^\mathcal{J}$ to $\Delta^\mathcal{J}$ and $u_0^\mathcal{J}, u_i^\mathcal{J}$ to $\Delta^\mathcal{J} \times \Delta^\mathcal{J}$, for all $i \in \{1, \ldots, n\}$. It is readily verified that $\mathcal{J}$ satisfies the inclusions in (B.3) to (B.8). As there does not exist any model $\mathcal{J}'$ of $\mathcal{T}$ such that $\mathcal{J}' \equiv_\Sigma \mathcal{I}$, (B.9) does not hold in $\mathcal{J}$. That is, there is a point $x \in \Delta^\mathcal{J}$ such that $x \in (\exists \vec{r}.A)^\mathcal{J}$ but $x \notin (\exists \vec{s}.A)^\mathcal{J}$. Hence, $\mathcal{J} \not\models \vec{r} \sqsubseteq \vec{s}$.

Conversely, suppose $\mathcal{T} \equiv_\Sigma \emptyset$. We show (B.1) by contradiction. Suppose (B.1) does not hold. We may assume that there is an interpretation $\mathcal{I}$ in which all role names in Σ are interpreted as functions such that $\mathcal{I} \models \vec{r}_i \equiv \vec{s}_i$, for all $i \in \{1, \ldots, n\}$, but $\mathcal{I} \not\models \vec{r} \sqsubseteq \vec{s}$. There is a pair $(x, y) \in \vec{r}^{\mathcal{I}}$ such that $(x, y) \notin \vec{s}^{\mathcal{I}}$.

We may assume that $A^\mathcal{I} = \{y\}$ and, therefore, $x \in (\exists \vec{r}.A)^\mathcal{I}$. Let $\mathcal{I}_x$ be the (x, Σ)-generated subinterpretation of $\mathcal{I}$. Since $\mathcal{T} \equiv_\Sigma \emptyset$, there exists a model $\mathcal{J}$ of $\mathcal{T}$ that coincides with $\mathcal{I}_x$ on Σ. By (B.6) we have $E^\mathcal{J} \neq \emptyset$. Therefore, by (B.7), $M^\mathcal{J} \neq \emptyset$. Every point in $M^\mathcal{J}$ is reachable from x by following a Σ-path: this is the case since $\mathcal{J}$ coincides with $\mathcal{I}_x$ on Σ and $\mathcal{I}_x$ is an (x, Σ)-generated subinterpretation of $\mathcal{I}$. Hence $x \in M^{\mathcal{J}}$ due to (B.8). Let $i \in \{1, \ldots, n\}$. We have that $(\exists \vec{r_i}.\top)^{\mathcal{J}} = (\exists \vec{s_i}.\top)^{\mathcal{J}} = \Delta^{\mathcal{J}}$. By (B.3) there exists a u_i-successor d_i of x from which, by the fact that $\mathcal{J} \models \vec{r_i} \equiv \vec{s_i}$ and the functionality of all $v \in \Sigma$ in $\mathcal{J}$, the role paths $\vec{r_i}$ and $\vec{s_i}$ lead to a common point d_i' with $d_i' \in (A_i \sqcap B_i)^{\mathcal{J}}$. By (B.4), $d_i' \in N_i^{\mathcal{J}}$. As d_i' is reachable from x, we obtain $x \in N_i^{\mathcal{J}}$ due to (B.5). Thus, we have $x \in (\exists \vec{r}.A \sqcap M \sqcap \bigsqcap_{i \leq n} N_i)^{\mathcal{J}}$. Inclusion (B.9) yields that $x \in (\exists \vec{s}.A)^{\mathcal{J}}$. But then $(x, y) \in \vec{s}^{\mathcal{J}}$ and we have derived a contradiction. □

Theorem 16. *Given $\mathcal{EL}$- or $\mathcal{ELI}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ and concept signatures Σ, it is* coNExp$^{\text{NP}}$*-complete to decide $\mathcal{T}_1 \equiv_\Sigma \mathcal{T}_2$.*

Proof. Containment in coNExp$^{\text{NP}}$ follows from Theorem 9, which shows that the problem is in coNExp$^{\text{NP}}$ even for $\mathcal{ALCI}$-TBoxes.

Hardness is (essentially) proved by reduction of Σ-entailment for $\mathcal{ALC}$-TBoxes and concept signatures Σ, which is coNExp$^{\text{NP}}$-hard by Theorem 10.

More precisely, for $\mathcal{ALC}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$ and a set Σ of concept names, we construct $\mathcal{EL}$-TBoxes $\mathcal{T}_1', \mathcal{T}_2'$ and a signature Σ' consisting of concept names only the following two statements are equivalent:

(a) for every model $\mathcal{I}$ of $\mathcal{T}_1$ of cardinality ≥ 2 there exists a model $\mathcal{J}$ of $\mathcal{T}_2$ such that $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$;

(b) $\mathcal{T}_1' \models_{\Sigma'} \mathcal{T}_2'$.

One can easily show that the proof of Theorem 10 shows that deciding (a) is coNExp$^{\text{NP}}$-hard.

Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be $\mathcal{ALC}$-TBoxes and Σ a set of concept names. We may assume w.l.o.g. that $\Sigma = \mathsf{sig}(\mathcal{T}_1) \cap \mathsf{sig}(\mathcal{T}_2)$ (see proof of Lemma 3) and that both, $\mathcal{T}_1$ and $\mathcal{T}_2$, consist of inclusions of the form

$$\top \sqsubseteq C, \quad A \sqsubseteq C, \quad C \sqsubseteq A,$$

where A is a concept name and C is of the form $\neg B$, $B_1 \sqcap B_2$, or $\exists r.B$, where B, B_1, B_2 are concept names.

We define the $\mathcal{EL}$-TBoxes $\mathcal{T}_1'$ and $\mathcal{T}_2'$. To simulate negation in $\mathcal{EL}$ we introduce for every concept name $B \in \mathsf{sig}(\mathcal{T}_1) \cup \mathsf{sig}(\mathcal{T}_2)$ a fresh concept name $\overline{B}$ that stands for $\neg B$. Denote for $C_1 \sqsubseteq C_2 \in \mathcal{T}_1 \cup \mathcal{T}_2$ by $C_1' \sqsubseteq C_2'$ the resulting

$\mathcal{EL}$-inclusion when all $\neg B$ in $C_1 \sqsubseteq C_2$ are replaced by $\overline{B}$. To enforce the intended behaviour of $\overline{B}$, we require, in addition, fresh concept names V and $\overline{V}$, fresh role names e and f, and for every concept name $B \in \mathsf{sig}(\mathcal{T}_1) \cup \mathsf{sig}(\mathcal{T}_2)$, a fresh role name s_B. To define $\mathcal{T}_i'$, first replace every $C_1 \sqsubseteq C_2 \in \mathcal{T}_i$ by the two inclusions

$$C_1' \sqcap \exists e.V \sqsubseteq C_2', \quad C_1' \sqcap \exists e.\overline{V} \sqsubseteq C_2' \tag{†}$$

and denote the resulting $\mathcal{EL}$-TBoxes by $\mathcal{C}_i$, $i = 1, 2$. Intuitively, we want $\overline{V}$ to behave similarly to $\neg V$ in intended interpretations. Moreover, we want that if $\overline{V}$ behaves similarly to $\neg V$, then all $\overline{B}$ behave similarly to $\neg B$. Note that in an interpretation $\mathcal{I}$ with $(V \sqcup \overline{V})^{\mathcal{I}} \neq \Delta^{\mathcal{I}}$ one can satisfy the inclusions in $\mathcal{C}_i$ in a trivial way by setting $e^{\mathcal{I}} = \Delta^{\mathcal{I}} \times \{d\}$ for some $d \in \Delta^{\mathcal{I}} \setminus (V \sqcup \overline{V})^{\mathcal{I}}$. We use this property to enforce that in intended interpretations $\mathcal{I}$, $(\overline{V} \sqcup V)^{\mathcal{I}} = \Delta^{\mathcal{I}}$. In addition, in the intended interpretations we want that $(V \sqcap \overline{V})^{\mathcal{I}} = \emptyset$. Since one cannot enforce this in $\mathcal{EL}$, we instead enforce that the subinterpretation induced by $(V \sqcap \overline{V})^{\mathcal{I}}$ does not interfere with the subinterpretation induced by $\Delta^{\mathcal{I}} \setminus (V \sqcap \overline{V})^{\mathcal{I}}$.

To achieve all this, we consider the following set $\mathcal{C}$ of $\mathcal{EL}$-inclusions: for all concept names $B \in \mathsf{sig}(\mathcal{T}_1 \cup \mathcal{T}_2)$ and role names $r \in \mathsf{sig}(\mathcal{T}_1) \cup \mathsf{sig}(\mathcal{T}_2) \cup \{f\} \cup \{s_B \mid B \in (\mathsf{sig}(\mathcal{T}_1) \cup \mathsf{sig}(\mathcal{T}_2)) \cap \mathsf{N_C}\}$,

$$B \equiv \exists s_B.V; \tag{B.10}$$

$$\overline{B} \equiv \exists s_B.\overline{V}; \tag{B.11}$$

$$\top \sqsubseteq \exists s_B.\top; \tag{B.12}$$

$$\top \sqsubseteq \exists e.\top; \tag{B.13}$$

$$V \sqsubseteq \exists f.\overline{V}, \quad \overline{V} \sqsubseteq \exists f.V; \tag{B.14}$$

$$B \sqcap \overline{B} \equiv V \sqcap \overline{V}; \tag{B.15}$$

$$\exists r.(V \sqcap \overline{V}) \sqsubseteq V \sqcap \overline{V}; \tag{B.16}$$

Let $\mathcal{T}_i' = \mathcal{C} \cup \mathcal{C}_i$, for $i = 1, 2$ and let

$$\Sigma' = \Sigma \cup \{V, \overline{V}\} \cup \{\overline{B} \mid B \in \Sigma\}.$$

Note that there are no role names in Σ'.

Claim 1. If not (a), then not (b).

Let $\mathcal{I}_1$ be a model of $\mathcal{T}_1$ of cardinality at least 2 such that there does not exist a model $\mathcal{I}_2$ of $\mathcal{T}_2$ with $\mathcal{I}_1|_\Sigma = \mathcal{I}_2|_\Sigma$. Define $\mathcal{I}_1'$ as $\mathcal{I}_1$ with the following modifications: let $V^{\mathcal{I}_1'}$ and $\overline{V}^{\mathcal{I}_1'}$ form a partition of $\Delta^{\mathcal{I}_1}$ (this is possible since $\Delta^{\mathcal{I}_1}$ has at least two elements) and

- set $\overline{B}^{\mathcal{I}_1'} = \Delta^{\mathcal{I}_1} \setminus B^{\mathcal{I}_1}$;
- fix $d_1 \in V^{\mathcal{I}_1}$ and $d_2 \in \overline{V}^{\mathcal{I}_1}$ and set

$$s_B^{\mathcal{I}_1'} = \{(d, d_1) \mid d \in B^{\mathcal{I}_1}\} \cup \{(d, d_2) \mid d \in \Delta^{\mathcal{I}_1} \setminus B^{\mathcal{I}_1}\};$$

- set $e^{\mathcal{I}_1'} = f^{\mathcal{I}_1'} = \Delta^{\mathcal{I}_1} \times \Delta^{\mathcal{I}_1}$.

It is readily seen that $\mathcal{I}_1'$ is a model of $\mathcal{T}_1'$. To show not (b) it is sufficient to prove that there does not exist a model $\mathcal{J}$ of $\mathcal{T}_2'$ such that $\mathcal{J}|_{\Sigma'} = \mathcal{I}_1'|_{\Sigma'}$. For a proof by contradiction suppose there exists such a $\mathcal{J}$. We show that $\mathcal{J}$ is a model of $\mathcal{T}_2$. From this we obtain, since $\mathcal{I}_1'|_\Sigma = \mathcal{I}_1|_\Sigma$ and $\Sigma \subseteq \Sigma'$, that $\mathcal{J}$ is a model of $\mathcal{T}_2$ such that $\mathcal{J}|_\Sigma = \mathcal{I}_1|_\Sigma$. Thus we have derived a contradiction. To show that $\mathcal{J}$ is a model of $\mathcal{T}_2$ it is sufficient to prove

- $\overline{B}^{\mathcal{J}} = (\neg B)^{\mathcal{J}}$ for all concept names $B \in \mathsf{sig}(\mathcal{T}_2)$ and
- $(\exists e.V)^{\mathcal{J}} \cup (\exists e.\overline{V})^{\mathcal{J}} = \Delta^{\mathcal{J}}$

since then the inclusions in $\mathcal{T}_2$ follow from the inclusions in $\mathcal{C}_2$. For Point 1, since $\mathcal{J}$ coincides with $\mathcal{I}_1'$ on Σ', we have $\overline{B}^{\mathcal{J}} = (\neg B)^{\mathcal{J}}$, for all $B \in \Sigma$. Let $B \in \mathsf{sig}(\mathcal{T}_2) \setminus \Sigma'$. As $V, \overline{V} \in \Sigma'$, their interpretation in $\mathcal{J}$ is the same as in $\mathcal{I}_1'$, i.e., they form a partition of $\Delta^{\mathcal{J}}$. The concept inclusion in (B.12) yields that every node has an s_B-successor, which is either in $V^{\mathcal{J}}$ or in $\overline{V}^{\mathcal{J}}$. Then, due to the axioms in (B.10) and (B.11), we have that $B^{\mathcal{J}} \cup \overline{B}^{\mathcal{J}} = \Delta^{\mathcal{J}}$. Moreover, $B^{\mathcal{J}}$ and $\overline{B}^{\mathcal{J}}$ are disjoint by disjointness of $V^{\mathcal{J}}$ and $\overline{V}^{\mathcal{J}}$ together with the fact that $(B \sqcap \overline{B})^{\mathcal{J}} = (V \sqcap \overline{V})^{\mathcal{J}}$ due to (B.15). It follows that $\overline{B}^{\mathcal{J}} = (\neg B)^{\mathcal{J}}$, as required.

Point 2 follows from (B.13) and the fact that $V^{\mathcal{J}}$ and $\overline{V}^{\mathcal{J}}$ form a partition of $\Delta^{\mathcal{J}}$.

Claim 2. If not (b), then not (a).

Let $\mathcal{I}_1$ be a model of $\mathcal{T}_1'$ such that there does not exist a model $\mathcal{I}_2$ of $\mathcal{T}_2'$ with $\mathcal{I}_1|_{\Sigma'} = \mathcal{I}_2|_{\Sigma'}$. We first show that:

(i) $V^{\mathcal{I}_1} \cup \overline{V}^{\mathcal{I}_1} = \Delta^{\mathcal{I}_1}$,

(ii) $V^{\mathcal{I}_1} \neq \emptyset$ and $\overline{V}^{\mathcal{I}_1} \neq \emptyset$,

(iii) $V^{\mathcal{I}_1} \neq \overline{V}^{\mathcal{I}_1}$, and

(iv) $\mathcal{I}_1$ contains two elements d, d' with $d \in V^{\mathcal{I}_1} \setminus \overline{V}^{\mathcal{I}_1}$ and $d' \in \overline{V}^{\mathcal{I}_1} \setminus V^{\mathcal{I}_1}$.

For (i), suppose $V^{\mathcal{I}_1} \cup \overline{V}^{\mathcal{I}_1} \neq \Delta^{\mathcal{I}_1}$. As the role name e is not in Σ', we can choose an interpretation $\mathcal{I}_2$ with $\mathcal{I}_1|_{\Sigma'} = \mathcal{I}_2|_{\Sigma'}$ such that all $e^{\mathcal{I}_2} = \Delta^{\mathcal{I}_2} \times \{d\}$ for some $d \notin V^{\mathcal{I}_2} \cup \overline{V}^{\mathcal{I}_2}$. Then $(\exists e.V)^{\mathcal{I}_2} = (\exists e.\overline{V})^{\mathcal{I}_2} = \emptyset$ and, thus, $\mathcal{I}_2$ trivially satisfies $\mathcal{C}_2$. Moreover, since only the interpretation of e has changed and $\mathcal{I}_1$ is a model of $\mathcal{C}$, $\mathcal{I}_2$ is a model of $\mathcal{C}$ as well since (B.13) is still true and is the only axiom in $\mathcal{C}$ containing e. Hence, $\mathcal{I}_2$ is a model of $\mathcal{T}_2'$; a contradiction. Consider (ii). Given (i), at least one of $V^{\mathcal{I}_1}$ and $\overline{V}^{\mathcal{I}_1}$ is non-empty. But then, the axioms in (B.14) yield that both $V^{\mathcal{I}_1}$ and $\overline{V}^{\mathcal{I}_1}$ are non-empty. To prove (iii), assume that it does not hold; i.e., $V^{\mathcal{I}_1} = \overline{V}^{\mathcal{I}_1}$. By (i), we have that $V^{\mathcal{I}_1} = \overline{V}^{\mathcal{I}_1} = \Delta^{\mathcal{I}_1}$. Take the interpretation $\mathcal{I}_2$ with $\Delta^{\mathcal{I}_2} = \Delta^{\mathcal{I}_1}$ and

- $A^{\mathcal{I}_2} = \Delta^{\mathcal{I}_2}$ for all $A \in \mathsf{N_C}$;
- $r^{\mathcal{I}_2} = \Delta^{\mathcal{I}_2} \times \Delta^{\mathcal{I}_2}$, for all $r \in \mathsf{N_R}$.

Note that $\mathcal{I}_2$ satisfies *every* $\mathcal{EL}$-inclusion. In particular, $\mathcal{I}_2$ is a model of $\mathcal{T}_2'$. Moreover, $\mathcal{I}_1|_{\Sigma'} = \mathcal{I}_2|_{\Sigma'}$ since by (B.15), all concept names in Σ' are interpreted as the full domain $\Delta^{\mathcal{I}_2}$. We have derived a contradiction. Finally, to show (iv), assume that (iv) does not hold. By (i) and (iii), we can assume w.l.o.g. that $V^{\mathcal{I}_1} \subsetneq \overline{V}^{\mathcal{I}_1} = \Delta^{\mathcal{I}_1}$. Let $d \in \overline{V}^{\mathcal{I}_1} \setminus V^{\mathcal{I}_1}$. By (B.14), all points in $\overline{V}^{\mathcal{I}_1}$ have an f-successor in $V^{\mathcal{I}_1}$. Thus, there exists $d' \in V^{\mathcal{I}_1}$ such that $(d, d') \in f^{\mathcal{I}_1}$. But then $d' \in (V \sqcap \overline{V})^{\mathcal{I}_1}$. By (B.16), $d \in \overline{V}^{\mathcal{I}_1} \cap V^{\mathcal{I}_1}$; a contradiction.

Note that items (i) to (iv) do not imply that $V^{\mathcal{I}_1}$ and $\overline{V}^{\mathcal{I}_1}$ forms a partition of $\Delta^{\mathcal{I}_1}$, i.e., we can have $V^{\mathcal{I}_1} \cap \overline{V}^{\mathcal{I}_1} \neq \emptyset$. Set $\Delta_1 = V^{\mathcal{I}_1} \cap \overline{V}^{\mathcal{I}_1}$ and $\Delta_2 = \Delta^{\mathcal{I}_1} \setminus \Delta_1$. Note $V^{\mathcal{I}_1}$ and $\overline{V}^{\mathcal{I}_1}$ form a partition of Δ_2. By (iv), we have that $|\Delta_2| \geq 2$. Due to (B.16), the set Δ_2 is Γ-closed, where $\Gamma = \mathsf{sig}(\mathcal{T}_1) \cup \mathsf{sig}(\mathcal{T}_2) \cup \{f\} \cup \{s_B \mid B \in (\mathsf{sig}(\mathcal{T}_1) \cup \mathsf{sig}(\mathcal{T}_2)) \cap \mathsf{N_C}\}$. Since we can re-define $e^{\mathcal{I}_1}$ without changing any of the properties of $\mathcal{I}_1$ established above by setting

$$e^{\mathcal{I}_1} = \Delta^{\mathcal{I}_1} \times \Delta_2,$$

we can assume that Δ_2 is $\Gamma \cup \{e\}$-closed. It follows from Lemma B.1 that the subinterpretation of $\mathcal{I}_1$ induced by Δ_2 satisfies $\mathcal{T}_1'$. We denote this interpretation by $\mathcal{J}_1$.

We now show that $\mathcal{J}_1$ is a model of $\mathcal{T}_1$. To this end, it is sufficient to show

- $\overline{B}^{\mathcal{J}} = (\neg B)^{\mathcal{J}}$ for all concept names $B \in \mathsf{sig}(\mathcal{T}_1)$ and
- $(\exists e.V)^{\mathcal{J}} \cup (\exists e.\overline{V})^{\mathcal{J}} = \Delta^{\mathcal{J}}$

since then the inclusions in $\mathcal{T}_1$ follow from the inclusions in $\mathcal{C}_1$. For Point 1, as $V^{\mathcal{J}_1}$ and $\overline{V}^{\mathcal{J}_1}$ cover Δ_2, the concept inclusions in (B.10) to (B.12) yield $B^{\mathcal{J}_1} \cup \overline{B}^{\mathcal{J}_1} = \Delta_2$. As $V^{\mathcal{J}_1}$ and $\overline{V}^{\mathcal{J}_1}$ are disjoint, we obtain disjointness of $B^{\mathcal{J}_1}$ and $\overline{B}^{\mathcal{J}_1}$ by (B.15). Point 1 follows. Point 2 follows from (B.13) and the fact that $V^{\mathcal{J}_1}$ and $\overline{V}^{\mathcal{J}_1}$ cover Δ_2.

If there does not exist a model $\mathcal{J}_2$ of $\mathcal{T}_2$ such that $\mathcal{J}_2|_{\Sigma} = \mathcal{J}_1|_{\Sigma}$ then (a) does not hold and we are done. Assume there exists such a model $\mathcal{J}_2$. Based on $\mathcal{J}_2$ we construct a model $\mathcal{I}_2$ of $\mathcal{T}_2'$ such that $\mathcal{I}_2|_{\Sigma'} = \mathcal{I}_1|_{\Sigma'}$, thus contradicting the assumption that no such interpretation $\mathcal{I}_2$ exists.

We start by constructing an interpretation $\mathcal{J}_2'$ with domain $\Delta^{\mathcal{J}_2}$: all symbols not in $\mathsf{sig}(\mathcal{T}_2) \setminus \Sigma$ as well as the symbols in $\{\overline{B} \mid B \in \mathsf{sig}(\mathcal{T}_2) \setminus \Sigma)\} \cup \{s_B \mid B \in \mathsf{sig}(\mathcal{T}_2) \setminus \Sigma\}$ are interpreted in the same way as in $\mathcal{J}_1$. For all concept names $B \in \mathsf{sig}(\mathcal{T}_2) \setminus \Sigma$ let

- $B^{\mathcal{J}_2'} := B^{\mathcal{J}_2}$;
- $\overline{B}^{\mathcal{J}_2'} := \Delta^{\mathcal{J}_2'} \setminus B^{\mathcal{J}_2'}$;
- fix $d_1 \in V^{\mathcal{J}_2}$ and $d_2 \in \overline{V}^{\mathcal{J}_2}$ and set

$$s_B^{\mathcal{J}_2'} := \{(d, d_1) \mid d \in B^{\mathcal{J}_2}\} \cup \{(d, d_2) \mid d \in \Delta^{\mathcal{J}_2} \setminus B^{\mathcal{J}_2}\}.$$

By definition, $\mathcal{J}_2'$ is a model of $\mathcal{T}_2'$ such that $\mathcal{J}_1|_{\Sigma'} = \mathcal{J}_2'|_{\Sigma'}$.

Define an interpretation $\mathcal{J}_2''$ with domain $\Delta^{\mathcal{J}''} = \Delta_1$ by setting

- $A^{\mathcal{I}_2} = \Delta^{\mathcal{I}_2}$ for all $A \in \mathsf{N_C}$;
- $r^{\mathcal{I}_2} = \Delta^{\mathcal{I}_2} \times \Delta^{\mathcal{I}_2}$, for all $r \in \mathsf{N_R}$.

As before, $\mathcal{J}_2''$ satisfies *every* $\mathcal{EL}$-inclusion and is, therefore, a model of $\mathcal{T}_2'$. Let $\mathcal{K}$ be the subinterpretation of $\mathcal{I}_1$ induced by Δ_1. Then $\mathcal{K}|_{\Sigma'} = \mathcal{J}_2''|_{\Sigma'}$ since $V^{\mathcal{I}_1} \cap \overline{V}^{\mathcal{I}_1} = \Delta_1$ and, by the inclusions (B.15), $A^{\mathcal{I}_1} \supseteq \Delta_1$ for all $A \in \Sigma'$.

Define $\mathcal{I}_2$ as the (disjoint) union of $\mathcal{J}_2'$ and $\mathcal{J}_2''$. Then $\mathcal{I}_2$ is a model of $\mathcal{T}_2'$ since both $\mathcal{J}_2'$ and $\mathcal{J}_2''$ are models of $\mathcal{T}_2'$ and $\mathcal{EL}$ is preserved under disjoint unions. Moreover, $\mathcal{I}_2|_{\Sigma'} = \mathcal{I}_1|_{\Sigma'}$, as required. □

Appendix C. Proofs for Section 6

We supply the remaining proof step for Theorem 33.

Lemma C.1. *Let $\mathcal{T}$ be an acyclic $\mathcal{ALCI}$-TBox and Σ a signature. For every interpretation $\mathcal{I}$, the following conditions are equivalent:*

(a) there exists a model $\mathcal{J}$ of $\mathcal{T}$ such that $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$;

(b) there exists a model $\mathcal{J}$ of $\mathsf{Lhs}_\Sigma(\mathcal{T})$ such that $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$.

Proof. We prove "$(b) \Rightarrow (a)$". Let $\mathcal{J}$ be a model of $\mathsf{Lhs}_\Sigma(\mathcal{T})$ such that $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. Let $\Sigma' = \mathsf{sig}(\mathcal{T}) \setminus \mathsf{sig}(\mathsf{Lhs}_\Sigma(\mathcal{T}))$. Obtain an interpretation $\mathcal{J}'$ from $\mathcal{J}$ by setting $\Delta^{\mathcal{J}'} = \Delta^{\mathcal{J}}$ and

- $X^{\mathcal{J}'} = X^{\mathcal{J}}$ for all $X \in (\mathsf{N_C} \cup \mathsf{N_R}) \setminus \Sigma'$;
- $r^{\mathcal{J}'} = \emptyset$, for all r role names $r \in \Sigma'$;
- for concept names $A \in \Sigma'$ the definition of $A^{\mathcal{J}'}$ is by induction on the definitorial depth of A: set $A^{\mathcal{J}'} = \emptyset$, for all $A \in \Sigma'$ with $d_{\mathcal{T}}(A) = 0$. Assume $B^{\mathcal{J}'}$ has been defined for all B with $d_{\mathcal{T}}(B) = n$. Let $A \in \Sigma'$ with $d_{\mathcal{T}}(A) = n + 1$. If $A \notin \mathsf{Def}(\mathcal{T})$, set $A^{\mathcal{J}'} = \emptyset$; otherwise there is a unique concept definition $A \equiv C \in \mathcal{T}$ such that $B^{\mathcal{J}'}$ is defined for all $B \in \mathsf{sig}(C)$. Set $A^{\mathcal{J}'} = C^{\mathcal{J}'}$.

Observe that $\mathcal{J}'|_\Sigma = \mathcal{I}|_\Sigma$ since $\Sigma \cap \Sigma' = \emptyset$ and $\mathcal{J}|_\Sigma = \mathcal{I}|_\Sigma$. We show that $\mathcal{J}'$ is a model of $\mathcal{T}$. Since $\mathcal{J}$ coincides with $\mathcal{J}'$ on $\mathsf{sig}(\mathsf{Lhs}_\Sigma(\mathcal{T}))$, $\mathcal{J}'$ is a model of $\mathsf{Lhs}_\Sigma(\mathcal{T})$. Now let $A \bowtie C \in \mathcal{T} \setminus \mathsf{Lhs}_\Sigma(\mathcal{T})$. By definition of $\mathsf{Lhs}_\Sigma(\mathcal{T})$, we have $A \in \mathsf{sig}(\mathcal{T}) \setminus \mathsf{sig}(\mathsf{Lhs}_\Sigma(\mathcal{T}))$. We distinguish two cases: First, let $A \bowtie C$ be of the form $A \sqsubseteq C$. Then $A^{\mathcal{J}'} = \emptyset$ and so $\mathcal{J}'$ satisfies $A \sqsubseteq C$. Second, let $A \bowtie C$ be of the form $A \equiv C$. Then $A^{\mathcal{J}'} = C^{\mathcal{J}'}$ and so $\mathcal{J}'$ satisfies $A \equiv C$, as required. □

Appendix D. Proofs for Section 7

The following lemma is a straightforward generalisation of a claim from the proof of Theorem 69 in [14].

Lemma D.1. *Let $\mathcal{T}$ be an acyclic $\mathcal{ALCI}$-TBox, Σ a signature, and $\mathcal{M} \subseteq \mathcal{T}$ such that $\mathcal{T} \setminus \mathcal{M}$ does not contain any direct $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency and $\mathcal{T} \setminus \mathcal{M} \equiv_{(\Sigma \cup \mathsf{sig}(\mathcal{M}))} \emptyset$. Suppose Σ' is such that $\Sigma \subseteq \Sigma' \subseteq (\Sigma \cup \mathsf{sig}(\mathcal{M}))$ and let $\mathcal{W} \subseteq \mathcal{T}$ be a minimal set such that either $\mathcal{W}$ contains a direct Σ'-dependency or $\mathcal{W} \not\equiv_{\Sigma'} \emptyset$. Then $\mathcal{W} \subseteq \mathcal{M}$.*

Proof. Suppose the lemma does not hold, i.e., $\mathcal{W} \not\subseteq \mathcal{M}$. Let $\mathcal{X} = \mathcal{M} \cap \mathcal{W}$. Then $\mathcal{X}$ does not contain direct Σ'-dependencies and $\mathcal{X} \equiv_{\Sigma'} \emptyset$ (for otherwise $\mathcal{X}$ is a proper subset of $\mathcal{W}$ such that either $\mathcal{X}$ contains a direct Σ'-dependency or $\mathcal{X} \not\equiv_{\Sigma'} \emptyset$, contrary to the minimality of $\mathcal{W}$). Notice that $(\mathcal{T} \setminus \mathcal{M}) \cup \mathcal{X}$ is an acyclic TBox.

First we prove that $\mathcal{W}$ does not have any direct Σ'-dependency. We show this by demonstrating that $(\mathcal{T} \setminus \mathcal{M}) \cup \mathcal{X} \supseteq \mathcal{W}$ does not have any direct Σ'-dependency. If that is not the case, let $\{A, B\} \subseteq \Sigma'$ be such that $A \prec^{+}_{(\mathcal{T}\setminus\mathcal{M})\cup\mathcal{X}} B$. Let $A = Y_1, \ldots, Y_n = B$ be such that $Y_i \bowtie C_i \in (\mathcal{T} \setminus \mathcal{M}) \cup \mathcal{X}$, for some C_i, and Y_{i+1} occurs in C_i, for all $1 \leq i < n$. Clearly, as $(\mathcal{T} \setminus \mathcal{M}) \cup \mathcal{X}$ is an acyclic TBox, every such $Y_i \bowtie C_i$ either occurs in $\mathcal{T} \setminus \mathcal{M}$ or in $\mathcal{X}$. As $A \not\prec^{+}_{(\mathcal{T}\setminus\mathcal{M})} B$ and $A \not\prec^{+}_{\mathcal{X}} B$, either there exists $X \in \mathsf{sig}(\mathcal{X})$ such that $A \prec^{+}_{\mathcal{T}\setminus\mathcal{M}} X$ or there exists $Y \in \mathsf{sig}(\mathcal{X})$ such that $Y \prec^{+}_{\mathcal{T}\setminus\mathcal{M}} B$ or there exists $\{X, Y\} \subseteq \mathsf{sig}(\mathcal{X})$ such that $X \prec^{+}_{\mathcal{T}\setminus\mathcal{M}} Y$. In either case, as $(\Sigma' \cup \mathsf{sig}(\mathcal{X})) \subseteq (\Sigma \cup \mathsf{sig}(\mathcal{M}))$, $\mathcal{T} \setminus \mathcal{M}$ contains a direct $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency contradicting the assumptions of the lemma.

Now we show that $\mathcal{W} \equiv_{\Sigma'} \emptyset$, which together with the fact that $\mathcal{W}$ does not contain any direct Σ'-dependency, contradicts the assumptions of the lemma. As $\mathcal{T} \setminus \mathcal{M} \equiv_{(\Sigma \cup \mathsf{sig}(\mathcal{M}))} \emptyset$, by Theorem 5 (robustness under replacement) we obtain $(\mathcal{T} \setminus \mathcal{M}) \cup \mathcal{X} \equiv_{(\Sigma \cup \mathsf{sig}(\mathcal{M}))} \mathcal{X}$. Using $\Sigma' \subseteq (\Sigma \cup \mathsf{sig}(\mathcal{M}))$ and the monotonicity property of inseparability we conclude from $\mathcal{X} \equiv_{\Sigma'} \emptyset$ that $(\mathcal{T} \setminus \mathcal{M}) \cup \mathcal{X} \equiv_{\Sigma'} \emptyset$. As $\emptyset \subseteq \mathcal{W} \subseteq (\mathcal{T} \setminus \mathcal{M}) \cup \mathcal{X}$, we obtain $\mathcal{W} \equiv_{\Sigma'} \emptyset$, as required. □

For the convenience of the reader, we explicitly present in Figure D.6 the modification of the algorithm given in Figure 2 and re-state Theorem 36.

Theorem 36. *Let $\mathcal{T}$ be an acyclic $\mathcal{ALCI}$-TBox and Σ a signature. Then the algorithm given in Figure D.6 computes the unique minimal depleting*

```
Input: TBox $\mathcal{T}$ and $\Sigma$.
$\mathcal{M} := \emptyset$;
$\mathcal{W} := \emptyset$;
while $(\mathcal{T} \setminus \mathcal{M}) \neq \mathcal{W}$ do
    choose $\alpha \in (\mathcal{T} \setminus \mathcal{M}) \setminus \mathcal{W}$
    $\mathcal{W} := \mathcal{W} \cup \{\alpha\}$;
    if $\mathcal{W} \not\equiv_{(\Sigma \cup \mathsf{sig}(\mathcal{M}))} \emptyset$ or $\mathcal{W}$ contains a direct $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency
    then
        $\mathcal{M} := \mathcal{M} \cup \{\alpha\}$;
        $\mathcal{W} := \emptyset$
    endif
end while
output $\mathcal{M}$
```

Figure D.6: The modified black box algorithm

Σ-module $\mathcal{M}$ such that $\mathcal{T} \setminus \mathcal{M}$ does not have any direct $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-*dependencies.*

Proof. By Theorem 34, we can check effectively (in fact in Σ_2^p) whether $\mathcal{W}$ has a $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency or $\mathcal{W} \equiv_{(\Sigma \cup \mathsf{sig}(\mathcal{M}))} \emptyset$. Thus, we indeed have an effective procedure. It partitions $\mathcal{T}$ into two sets $\mathcal{M}$ and $\mathcal{W}$ with $\mathcal{W}$ having no direct $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency and $\mathcal{W} \equiv_{(\Sigma \cup \mathsf{sig}(\mathcal{M}))} \emptyset$. So, $\mathcal{M}$ is a depleting Σ-module of $\mathcal{T}$ such that $\mathcal{T} \setminus \mathcal{M}$ does not have any direct $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency.

It remains to show that we obtain the unique minimal $\mathcal{M}$ with this property. To this end, let $\mathcal{M}_0 \subseteq \mathcal{T}$ be such that $\mathcal{T} \setminus \mathcal{M}_0$ does not have direct any $(\Sigma \cup \mathsf{sig}(\mathcal{M}_0))$-dependency and $\mathcal{T} \setminus \mathcal{M}_0 \equiv_{\Sigma \cup \mathsf{sig}(\mathcal{M}_0)} \emptyset$. We prove by induction on the number of loop iterations that at every loop iteration the set $\mathcal{M}$ is contained in $\mathcal{M}_0$.

Initially, $\mathcal{M} = \emptyset \subseteq \mathcal{M}_0$. Now let $\mathcal{W}$ and α from the while loop of the algorithm be such that $\mathcal{W}$ does not have any direct $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency and $\mathcal{W} \equiv_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \emptyset$ but either $(\mathcal{W} \cup \{\alpha\})$ contains a direct $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency or $(\mathcal{W} \cup \{\alpha\}) \not\equiv_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \emptyset$. By induction hypothesis, $\mathsf{sig}(\mathcal{M}) \subseteq \mathsf{sig}(\mathcal{M}_0)$. There must exist a minimal $\mathcal{W}' \subseteq (\mathcal{W} \cup \{\alpha\})$ such that either $\mathcal{W}'$ contains a direct $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency or $\mathcal{W}' \not\equiv_{\Sigma \cup \mathsf{sig}(\mathcal{M})} \emptyset$. It should be clear that $\alpha \in \mathcal{W}'$. But then we can apply Lemma D.1 (with $\Sigma' = \Sigma \cup \mathsf{sig}(\mathcal{M})$,

```
Input: acyclic $\mathcal{ELI}$-TBox $\mathcal{T}$ and signature $\Sigma$.
Initialise: $\mathcal{M} = \emptyset$.
Apply the rule (Loc) exhaustively.
Output: $\mathcal{M}$.

  (Loc) if $A \sqsubseteq C \in \mathcal{T} \setminus \mathcal{M}$ is non-$x$-local w.r.t. $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$ then
        set $\mathcal{M} := \mathcal{M} \cup \{A \sqsubseteq C\}$.
```

Figure E.7: x-module extraction in $\mathcal{ELI}$

$\mathcal{M} = \mathcal{M}_0$, and $\mathcal{W}' = \mathcal{W}$) and conclude $\mathcal{W}' \subseteq \mathcal{M}_0$. Hence $\alpha \in \mathcal{M}_0$, as required. □

Appendix E. Proofs for Section 8

In this section, we first show Proposition 38 and then develop an algorithm that extracts depleting modules from acyclic $\mathcal{ELI}$-TBoxes with role inclusions.

We first define locality-based module extraction specialised to acyclic $\mathcal{ELI}$-TBoxes without concept definitions. Let Σ be a signature. A primitive concept inclusion $A \sqsubseteq C$ is called

- *non-$\bot$-local w.r.t.* Σ if $A \in \Sigma$;
- *non-$\top$-local w.r.t.* Σ if $\mathsf{sig}(C) \cap \Sigma \neq \emptyset$.

Let $\mathcal{M} \subseteq \mathcal{T}$, where $\mathcal{T}$ contains no concept definitions. For $x \in \{\top, \bot\}$, $\mathcal{M}$ is the *x-module of $\mathcal{T}$ w.r.t.* Σ if it is the output of the algorithm from Figure E.7. Consider the sequence $\mathcal{M}_0 \supseteq \mathcal{M}_1 \supseteq \mathcal{M}_2 \supseteq \cdots$ where $\mathcal{M}_0 = \mathcal{T}$ and $\mathcal{M}_{2i+1}$ is the $\bot$-module of $\mathcal{M}_{2i}$ w.r.t. Σ for all $i \geq 0$ and $\mathcal{M}_{2i+2}$ is the $\top$-module of $\mathcal{M}_{2i+1}$ w.r.t. Σ for all $i \geq 0$. Then $\mathcal{M}$ is the *STAR-module of $\mathcal{T}$ w.r.t.* Σ if $\mathcal{M} = \mathcal{M}_i$ for the minimal $i > 0$ such that $\mathcal{M}_{i+1} = \mathcal{M}_i$. It is straightforward to check that in the case of acyclic $\mathcal{ELI}$-TBoxes containing no concept definitions the definition of STAR-module given here coincides with the definition given in [17].

$\bot$-modules can be characterised in terms of syntactic dependencies. The following lemma can be proved by induction on the number of rule application in the computation of $\mathcal{M}$ by the $\bot$-module extraction algorithm in Figure E.7.

Lemma E.1. *Let $\mathcal{T}$ be an acyclic $\mathcal{ELI}$-TBox containing no concept definitions and Σ a signature. Let $\mathcal{M}$ be the $\bot$-module of $\mathcal{T}$ w.r.t Σ. Then $A \sqsubseteq C \in \mathcal{M}$ if, and only if, $A \in \Sigma$ or there exists a concept name $B \in \Sigma$ such that $A \in \mathsf{depend}_{\mathcal{T}}(B)$.*

We also note the following properties of minimal depleting Σ-modules.

Lemma E.2. *Let $\mathcal{T}$ be an acyclic $\mathcal{ELI}$-TBox containing no concept definitions and Σ a signature. Let $\mathcal{M}$ be the minimal depleting Σ-module of $\mathcal{T}$.*

1. *Let $\{A_0 \sqsubseteq C_0, \ldots, A_n \sqsubseteq C_n\} \subseteq \mathcal{T}$ such that $A_0 \in \Sigma$, $\mathsf{sig}(C_n) \cap \Sigma \neq \emptyset$, and $A_{i+1} \in \mathsf{sig}(C_i)$ for all $i < n$. Then $A_i \sqsubseteq C_i \in \mathcal{M}$, for all $i \leq n$.*
2. *The minimal depleting $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-module of $\mathcal{T}$ coincides with $\mathcal{M}$.*

We are in the position to prove Proposition 38.

Proposition 38. *Let $\mathcal{T}$ be an acyclic $\mathcal{ELI}$-TBox containing no concept definitions. Then the STAR-module of $\mathcal{T}$ w.r.t. Σ coincides with the minimal depleting Σ-module of $\mathcal{T}$, for every signature Σ.*

Proof. Let $\mathcal{M}_{min}$ be the minimal depleting Σ-module of $\mathcal{T}$ and let $\mathcal{M}^*$ be the STAR-module of $\mathcal{T}$ w.r.t. Σ. We have $\mathcal{M}_{min} \subseteq \mathcal{M}^*$ since STAR-modules are depleting modules.

To prove the inclusion $\mathcal{M}^* \subseteq \mathcal{M}_{min}$, note that the $\top$-module of $\mathcal{M}^*$ w.r.t. Σ (as well as $\bot$-module of $\mathcal{M}^*$ w.r.t. Σ) coincide with $\mathcal{M}^*$ itself. Thus, the $\top$-module extraction algorithm in Figure E.7 applied to $\mathcal{M}^*$ and Σ outputs $\mathcal{M}^*$ itself. Let $\emptyset = \mathcal{M}^*_0 \subsetneq \mathcal{M}^*_1 \subsetneq \cdots \subsetneq \mathcal{M}^*_n = \mathcal{M}^*$, for some $n \geq 0$, be the steps in the computation of the $\top$-module of $\mathcal{M}^*$ by the $\top$-module extraction algorithm in Figure E.7; that is, $\mathcal{M}^*_{i+1}$, for $0 \leq i < n$, is obtained from $\mathcal{M}^*_i$ by adding one primitive concept inclusion $A_{i+1} \sqsubseteq C_{i+1} \in \mathcal{M}^* \setminus \mathcal{M}^*_i$ such that $A_{i+1} \sqsubseteq C_{i+1}$ is non-$\top$-local w.r.t. $(\Sigma \cup \mathsf{sig}(\mathcal{M}^*_i))$. We prove the following claim by induction on i for every $i < n$,

$$A_{i+1} \sqsubseteq C_{i+1} \in \mathcal{M}_{min}. \tag{$*$}$$

For $i = 0$ note that $\mathcal{M}^*_0 = \emptyset$. Hence $A_1 \sqsubseteq C_1$ is non-$\top$-local w.r.t. Σ, that is, $\mathsf{sig}(C_1) \cap \Sigma \neq \emptyset$. As $\mathcal{M}^*$ itself is the $\bot$-module of $\mathcal{M}^*$ w.r.t. Σ, by Lemma E.1, either $A_1 \in \Sigma$ or there exists $B \in \Sigma \cap \mathsf{N_C}$ such that

$A_1 \in \mathsf{depend}_{\mathcal{M}^*}(B)$. As $\mathcal{M}^* \subseteq \mathcal{T}$ we have $A_1 \in \Sigma$ or $A_1 \in \mathsf{depend}_{\mathcal{T}}(B)$. By Lemma E.2 (1) we obtain $A_1 \sqsubseteq C_1 \in \mathcal{M}_{min}$.

For the induction step, assume that $(*)$ is proved for all $i < j$, for some $j < n$. Then $\mathcal{M}^*_j \subseteq \mathcal{M}_{min}$. Similar to the argument above, either $A_{j+1} \in \Sigma$ or there exists $B \in \Sigma \cap \mathsf{N_C}$ such that $A_{j+1} \in \mathsf{depend}_{\mathcal{T}}(B)$. As $A_{j+1} \sqsubseteq C_{j+1}$ is non-$\top$-local w.r.t. $(\Sigma \cup \mathsf{sig}(\mathcal{M}^*_j))$, we have $\mathsf{sig}(C_{j+1}) \cap (\Sigma \cup \mathsf{sig}(\mathcal{M}^*_j)) \neq \emptyset$. But then $\mathsf{sig}(C_{j+1}) \cap (\Sigma \cup \mathsf{sig}(\mathcal{M}_{min})) \neq \emptyset$ and, by Lemma E.2 (1), $A_{j+1} \sqsubseteq C_{j+1}$ belongs to the minimal depleting $(\Sigma \cup \mathsf{sig}(\mathcal{M}_{min}))$-module of $\mathcal{T}$. Hence, by Lemma E.2 (2), $A_{j+1} \sqsubseteq C_{j+1} \in \mathcal{M}_{min}$, as required. □

We present a module extraction algorithm for acyclic $\mathcal{ELI}$-TBoxes with role inclusions. The union $\mathcal{C}$ of an acyclic $\mathcal{ELI}$-TBox $\mathcal{T}$ and a set $\mathcal{R}$ of role inclusions of the form $r \sqsubseteq s$, where r is a role name and s is a role (a role name or an inverse role), is called an *acyclic $\mathcal{ELI}$-constraint box* (CBox). An interpretation $\mathcal{I}$ *satisfies* $r \sqsubseteq s$ if $r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$, and $\mathcal{I}$ satisfies $\mathcal{C}$ if $\mathcal{I}$ satisfies $\mathcal{T}$ and every role inclusion from $\mathcal{C}$. Other semantic notions and notions relevant for modularity introduced above (signature of a role inclusion, depleting and self-contained module, etc.) are extended to role inclusions and CBoxes in the obvious way.

In particular, for an acyclic $\mathcal{ELI}$-CBox $\mathcal{C} = \mathcal{T} \cup \mathcal{R}$, signature Σ, and $A \in \Sigma$,

- set $\mathsf{depend}_{\mathcal{C}}(A) = \mathsf{depend}_{\mathcal{T}}(A) \cup \{r \in \mathsf{N_R} \mid \exists r' \in \mathsf{depend}_{\mathcal{T}}(A) \text{ such that } \mathcal{C} \models r' \sqsubseteq r \text{ or } \mathcal{C} \models r' \sqsubseteq r^-\}$;

- say that A *has a direct Σ-dependency in $\mathcal{C}$* if $\mathsf{depend}_{\mathcal{C}}(A) \cap \Sigma \neq \emptyset$;

- say that A *has an indirect Σ-dependency in $\mathcal{C}$* if there are $A_1, \ldots, A_n \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma$ and $A \in \mathsf{Def}(\mathcal{T}) \cap \Sigma$ such that $A \notin \{A_1, \ldots, A_n\}$ and

$$\mathsf{depend}^{\equiv}_{\mathcal{T}}(A) \setminus \mathsf{Def}(\mathcal{T}) \subseteq \bigcup_{1 \leq i \leq n} \mathsf{depend}_{\mathcal{C}}(A_i);$$

- and say that $\mathcal{C}$ *contains a Boolean Σ-constraint* if there is a $P \subseteq \mathsf{Lhs}(\mathcal{T}) \cap \Sigma$ such that the concept

$$C_P = \bigsqcap_{A \in P} A \sqcap \bigsqcap_{A \in (\mathsf{Lhs}(\mathcal{T}) \cap \Sigma) \setminus P} \neg A$$

is not satisfiable in a one-point model of $\mathcal{C}$.

The relationship between direct and indirect Σ-dependencies and Boolean Σ-constraints established in Lemma 23 carries over to CBoxes.

Lemma E.3 (CBox version of Lemma 23)**.** *Let $\mathcal{C} = \mathcal{T} \cup \mathcal{R}$ be an acyclic $\mathcal{ELI}$-CBox and Σ be a signature. If $\mathcal{C}$ does not contain neither a direct nor an indirect Σ-dependency, then $\mathcal{C}$ does not contain any Boolean Σ-constraints.*

Proof. For every $P \subseteq \mathsf{Lhs}(\mathcal{T}) \cap \Sigma$ we define a one-point interpretation $\mathcal{I}$ in the same way as in the proof of Lemma 23 with the only exception that for all $r \in \mathsf{N_R}$, we set

$$r^{\mathcal{I}} = \begin{cases} \{(d,d)\} & : \quad r \in \bigcup_{B \in P} \mathsf{depend}_{\mathcal{C}}(B) \\ \emptyset & : \quad \text{otherwise} \end{cases}$$

(The difference is that we consider $\mathsf{depend}_{\mathcal{C}}$ rather than $\mathsf{depend}_{\mathcal{T}}$.) Exactly the same argument as in the proof of Lemma 23 shows that $\mathcal{I}$ is a model of $\mathcal{T}$ and the concept C_P is satisfied in $\mathcal{I}$.

To show that $\mathcal{I}$ is a model of $\mathcal{R}$ let $r \sqsubseteq s \in \mathcal{R}$ and suppose that $(d,d) \in r^{\mathcal{I}}$. Then, by definition of $\mathcal{I}$, $r \in \bigcup_{B \in P} \mathsf{depend}_{\mathcal{C}}(B)$. But then $s \in \bigcup_{B \in P} \mathsf{depend}_{\mathcal{C}}(B)$, if s is a role name, and $s^{-} \in \bigcup_{B \in P} \mathsf{depend}_{\mathcal{C}}(B)$, if s is an inverse role. In any case, $(d,d) \in s^{\mathcal{I}}$, as required. □

Notice that in contrast to our results for acyclic $\mathcal{ELI}$-TBoxes the absence of direct Σ-dependencies and Boolean Σ-constraints does not imply safety of acyclic $\mathcal{ELI}$-CBoxes.

Example E.4. *Consider $\mathcal{C} = \{r \sqsubseteq s, A \equiv \exists s.X\}$ and $\Sigma = \{A, r\}$. Clearly $\mathcal{C}$ does not contain neither a direct nor an indirect Σ-dependency. On the other hand, for a two-point interpretation $\mathcal{I}$ such that $\Delta^{\mathcal{I}} = \{d_1, d_2\}$, $A^{\mathcal{I}} = \{d_1\}$, $r^{\mathcal{I}} = \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, it is easy to see that there exists no model $\mathcal{J}$ of $\mathcal{C}$ such that $\mathcal{J}|_{\Sigma} = \mathcal{I}|_{\Sigma}$: to satisfy the CBox, $s^{\mathcal{J}}$ has to be $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ but then one cannot have both $d_1 \in (\exists s.X)^{\mathcal{J}}$ and $d_2 \notin (\exists s.X)^{\mathcal{J}}$.*

Thus, without any additional conditions on the modules the module extraction algorithm developed above for acyclic TBoxes is not sound for acyclic CBoxes. We now show that being $\bot$-*local* for Σ-role inclusions [12] is a condition on modules that facilitates a sound extension of our module extraction algorithm.

Definition E.5 ($\bot$-locality for Σ)**.** Let $\mathcal{R}$ be a set of role inclusions and Σ a signature. We say that $\mathcal{R}$ is $\bot$-local for Σ if, and only if, there exists no $r \sqsubseteq r' \in \mathcal{R}$ with $r \in \Sigma$.

Input: Acyclic $\mathcal{ELI}$-CBox $\mathcal{C} = \mathcal{T} \cup \mathcal{R}$ and signature Σ.
Initialise: $\mathcal{M} = \emptyset$.
Apply the Rules 1, 2 and 3 exhaustively, preferring (R1) to (R2) and (R2) to (R3)
Output: $\mathcal{M}$.

(R1) if $r \in (\Sigma \cup \mathsf{sig}(\mathcal{M}))$ and $r \sqsubseteq r' \in \mathcal{C} \setminus \mathcal{M}$, then set $\mathcal{M} := \mathcal{M} \cup \{r \sqsubseteq r'\}$.

(R2) if $A \in (\Sigma \cup \mathsf{sig}(\mathcal{M}))$ has a direct $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency in $\mathcal{C} \setminus \mathcal{M}$ then set $\mathcal{M} := \mathcal{M} \cup \{A \bowtie C\}$ for $A \bowtie C \in \mathcal{C} \setminus \mathcal{M}$.

(R3) if $A \in (\Sigma \cup \mathsf{sig}(\mathcal{M}))$ has an indirect $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency in $\mathcal{T} \setminus \mathcal{M}$ then set $\mathcal{M} := \mathcal{M} \cup \{A \equiv C\}$ for $A \equiv C \in \mathcal{T} \setminus \mathcal{M}$.

Figure E.8: Module extraction for $\mathcal{ELI}$-CBoxes

Lemma E.6 (CBox version of Lemma 24)**.** *Let $\mathcal{C} = \mathcal{T} \cup \mathcal{R}$ be an acyclic $\mathcal{ELI}$-CBox and Σ a signature such that $\mathcal{R}$ is $\bot$-local for Σ. If $\mathcal{C}$ contains no direct Σ-dependency and no Boolean Σ-constraints then $\mathcal{C} \equiv_\Sigma \emptyset$.*

Proof. Let $\mathcal{I}$ be an arbitrary interpretation. For each $d \in \Delta^\mathcal{I}$, let

$$P_d = \{A \in \mathsf{Lhs}(\mathcal{T}) \cap \Sigma \mid d \in A^\mathcal{I}\}.$$

Since $\mathcal{C}$ does not contain any Boolean Σ-constraints, for each $d \in \Delta^\mathcal{I}$ there is a one-point model $\mathcal{I}_d$ of $\mathcal{C}$ that satisfies C_{P_d}. Let $\mathcal{J}$ be defined as in the proof of Lemma 24. Then $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$ and $\mathcal{J}$ is a model of $\mathcal{T}$. Recall that for all $r \in \mathsf{N_R}$

$$r^\mathcal{J} = \begin{cases} r^\mathcal{I} & \text{if } r \in \Sigma \\ \bigcup_{d \in \Delta^\mathcal{I}} r^{\mathcal{I}_d} & \text{otherwise} \end{cases}$$

To prove that $\mathcal{J}$ is a model of $\mathcal{R}$ let $r \sqsubseteq s \in \mathcal{R}$. As $\mathcal{R}$ is $\bot$-local for Σ, $r \notin \Sigma$. But then $r^\mathcal{J} = \bigcup_{d \in \Delta^\mathcal{I}} r^{\mathcal{I}_d}$. As every $\mathcal{I}_d$ is a model of $\mathcal{C}$, we have $\mathcal{J} \models r \sqsubseteq s$, as required. □

We obtain the main result of this section.

Theorem E.7 (Soundness of module extraction algorithm)**.** *For any acyclic $\mathcal{ELI}$-CBox $\mathcal{C}$ and signature Σ the output of the algorithm in Figure E.8 is a depleting Σ-module of $\mathcal{C}$.*

Proof. Let $\mathcal{M}$ be the output of the algorithm in Figure E.8. Then $\mathcal{C} \setminus \mathcal{M}$ does not contain neither a direct nor an indirect $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$-dependency and the role inclusions of $\mathcal{C} \setminus \mathcal{M}$ are $\bot$-local for $(\Sigma \cup \mathsf{sig}(\mathcal{M}))$. By Lemmas E.3 and E.6, $\mathcal{M}$ is a depleting Σ-module of $\mathcal{T}$. □

Notice that if $\mathcal{C}$ does not contain any role inclusions, the output of the algorithm in Figure E.8 coincides with the output of the algorithm in Figure 3.