Semantic Modularity and Module Extraction in Description Logics

Boris Konev[1] and **Carsten Lutz**[2] and **Dirk Walther**[1] and **Frank Wolter**[1]

Abstract. The aim of this paper is to study semantic notions of modularity in description logic (DL) terminologies and reasoning problems that are relevant for modularity. We define two notions of a module whose independence is formalised in a model-theoretic way. Focusing mainly on the DLs $\mathcal{EL}$ and $\mathcal{ALC}$, we then develop algorithms for module extraction, for checking whether a part of a terminology is a module, and for a number of related problems. We also analyse the complexity of these problems, which ranges from tractable to undecidable. Finally, we provide an experimental evaluation of our module extraction algorithms based on the large-scale terminology SNOMED CT.

1 Introduction

The main use of ontologies in computer 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 meaning of terms built from the vocabulary and their relationships. To emphasise this usage, we speak of terminologies rather than of ontologies. Current applications lead to the development of large and comprehensive terminologies, as witnessed, e.g., by the Systematized Nomenclature of Medicine, Clinical Terms (SNOMED CT), which comprises ∼0.4 million terms and underlies the systematised medical vocabulary used in the health systems of the US, the UK, and other countries [13]. When working with terminologies of this size and complexity, often only a fragment of the defined vocabulary is of interest. For example, a terminology designer may want to reuse a part of a large terminology inside his own terminology without being forced to adopt it completely. If the terminology is deployed in an application, it is often also unwieldy to work with the whole terminology compared to working only with the part that is relevant for the application.

These observations illustrate the importance of the *module extraction problem* for terminologies, as studied, e.g., in [6, 2, 12, 4, 3]: given a relevant signature Σ and a terminology $\mathcal{T}$ that defines, among others, the terms from Σ, extract a minimal subset (*module*) $\mathcal{T}_0$ from $\mathcal{T}$ such that $\mathcal{T}_0$ can serve as a substitute for $\mathcal{T}$ w.r.t. Σ. What it means that $\mathcal{T}_0$ can serve as a substitute for $\mathcal{T}$ depends on the application at hand. In this paper, we aim at minimal modules $\mathcal{T}_0$ that induce the same *dependencies* between terms in Σ as the original terminology $\mathcal{T}$. We understand such dependencies in a model-theoretic way, identifying the dependencies between Σ-terms with the class of all Σ-reducts of models satisfying the terminology. Thus, two terminologies induce the same dependencies between terms in Σ if the classes of Σ-reducts of their models coincide. Applications for which the resulting type of module is appropriate include (a) importing, instead of the whole terminology, the module into another terminology; see also [6], (b) computing the classification of only the terms in the signature Σ, and (c) querying a database using the module instead of the whole terminology. The main advantage of our model-theoretic approach compared to entailment-based notions of dependencies [9, 11] is its robustness under changes to the language in which terminologies and queries are formulated.

The contribution of this paper is as follows. We introduce a model-theoretic notion of dependencies and explore the complexity of basic reasoning problems such as checking whether two terminologies induce the same dependencies and whether a terminology induces any dependencies at all. Considering terminologies formulated in the standard description logic (DL) $\mathcal{ALC}$, the lightweight DL $\mathcal{EL}$, and their extensions with inverse roles, we find that the complexity ranges from tractable via Π_2^p-complete and coNEXP$^{\text{NP}}$-complete to undecidable. Based on these notions of dependency, we introduce two notions of a module and develop algorithms for module extraction and checking whether a subset of a terminology is a module. The algorithms work on acyclic terminologies formulated in $\mathcal{ALCI}$ and $\mathcal{ELI}$. The module extraction algorithm for $\mathcal{ELI}$ has been implemented in a system called MEX, and we present experimental results comparing modules extracted from SNOMED CT by MEX with modules extracted using the $\bot$-local modules approach of [6].

Detailed proofs are provided in the appendices.

2 Preliminaries

In this paper, we consider the description logics $\mathcal{EL}$, $\mathcal{ELI}$, $\mathcal{ALC}$ and $\mathcal{ALCI}$. Let $\mathsf{N_C}$ and $\mathsf{N_R}$ be countably infinite and disjoint sets of *concept names* and *role names*, respectively. In $\mathcal{ALC}$, composite concepts are built up starting from the concept names in $\mathsf{N_C}$, and applying the concept constructors shown in the upper four rows of Figure 1. In the figure and in general, C and D denote concepts, and r denotes a role name. As usual, we use $\bot$ to abbreviate $\neg\top$, $\sqcup$, $\rightarrow$, and $\leftrightarrow$ for the usual Boolean connectives defined in terms of $\neg$ and $\sqcap$, and $\forall r.C$ for $\neg\exists r.\neg C$. $\mathcal{EL}$ is the fragment obtained from $\mathcal{ALC}$ by dropping negation. We obtain $\mathcal{ALCI}$ from $\mathcal{ALC}$ and $\mathcal{ELI}$ from $\mathcal{EL}$ by additionally allowing inverse roles inside existen-

[1] University of Liverpool, Liverpool, UK
[2] TU Dresden, Dresden, Germany

Name	Syntax	Semantics
top-concept	$\top$	$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$
negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists r.C$	$\{d \in \Delta^{\mathcal{I}} \mid \exists d'\ (d,d') \in r^{\mathcal{I}} \wedge d' \in C^{\mathcal{I}}\}$
inverse role	r^-	$\{(e,d) \mid (d,e) \in r^{\mathcal{I}}\}$

Figure 1. Syntax and semantics.

tial restrictions, as shown in the bottom-most line of Figure 1.

The semantics of concepts is defined by means of *interpretations* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where the interpretation *domain* $\Delta^{\mathcal{I}}$ is a non-empty set, and $\cdot^{\mathcal{I}}$ is a function mapping each concept name A to a subset $A^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$ and each role name $r^{\mathcal{I}}$ to a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The function $\cdot^{\mathcal{I}}$ is inductively expanded to composite concepts as shown in Figure 1.

A *general TBox* is a finite set of *axioms*, where an axiom can be either a *concept inclusion (CI)* $C \sqsubseteq D$ or a *concept equality (CE)* $C \equiv D$, with C and D concepts. If all concepts used in $\mathcal{T}$ belong to a description logic $\mathcal{L}$, then $\mathcal{T}$ is also called a general $\mathcal{L}$-TBox. An interpretation $\mathcal{I}$ *satisfies* a CI $C \sqsubseteq D$ (written $\mathcal{I} \models C \sqsubseteq D$) if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$; it *satisfies* a CE $C \equiv D$ (written $\mathcal{I} \models C \equiv D$) if $C^{\mathcal{I}} = D^{\mathcal{I}}$. $\mathcal{I}$ is a *model* of a general TBox $\mathcal{T}$ if it satisfies all axioms in $\mathcal{T}$. We write $\mathcal{T} \models C \sqsubseteq D$ ($\mathcal{T} \models C \equiv D$) if every model of $\mathcal{T}$ satisfies $C \sqsubseteq D$ ($C \equiv D$). A general TBox $\mathcal{T}$ is called a *terminology* if it satisfies the following conditions:

- all CEs are of the form $A \equiv C$ (*concept definition*) and all CIs are of the form $A \sqsubseteq C$ (*primitive concept definition*), where A is a concept name;
- no concept name occurs more than once on the left hand side of an axiom.

Define the relation $\prec_{\mathcal{T}} \subseteq \mathsf{N_C} \times (\mathsf{N_C} \cup \mathsf{N_R})$ by setting $A \prec_{\mathcal{T}} X$ iff there exists an axiom of the form $A \sqsubseteq C$ or $A \equiv C$ in $\mathcal{T}$ such that X occurs in C. Denote by $\prec_{\mathcal{T}}^*$ the transitive closure of $\prec_{\mathcal{T}}$ and set $\mathsf{depend}_{\mathcal{T}}(A) = \{X \mid A \prec_{\mathcal{T}}^* X\}$. Intuitively, $\mathsf{depend}_{\mathcal{T}}(A)$ consists of all symbols X which are used in the definition of A in $\mathcal{T}$. A terminology $\mathcal{T}$ is called *acyclic* if $A \notin \mathsf{depend}_{\mathcal{T}}(A)$ for any $A \in \mathsf{N_C}$. In an acyclic terminology $\mathcal{T}$, the set $\mathsf{Pr}(\mathcal{T})$ of *primitive* symbols in $\mathcal{T}$ consists of all role names and concept names that do not occur on the left hand side of an axiom of $\mathcal{T}$. The set $\mathsf{PPr}(\mathcal{T})$ of *pseudo-primitive* symbols in $\mathcal{T}$ consists of all symbols primitive in $\mathcal{T}$ and all A such that $A \sqsubseteq C \in \mathcal{T}$ for some C.

A *signature* Σ is a finite subset of $\mathsf{N_C} \cup \mathsf{N_R}$. The signature $\mathsf{sig}(C)$ ($\mathsf{sig}(\alpha)$, $\mathsf{sig}(\mathcal{T})$) of a concept C (axiom α, TBox $\mathcal{T}$) is the set of concept and role names which occur in C (α, $\mathcal{T}$, respectively). If $\mathsf{sig}(C) \subseteq \Sigma$, we call C a Σ-*concept* and similarly for axioms and TBoxes.

3 Semantic modularity

We introduce the fundamental notions underlying semantic dependencies and modules and give two definitions of a module. In the following, we say that two interpretations $\mathcal{I}$ and $\mathcal{J}$ *coincide on a signature* Σ, written $\mathcal{I}|_\Sigma = \mathcal{J}|_\Sigma$, iff $\Delta^{\mathcal{I}} = \Delta^{\mathcal{J}}$ and $X^{\mathcal{I}} = X^{\mathcal{J}}$ for all $X \in \Sigma$.

Definition 1. Let $\mathcal{T}_0$ and $\mathcal{T}_1$ be general TBoxes and Σ a signature.

- $\mathcal{T}_1$ is a *semantic Σ-consequence of* $\mathcal{T}_0$, written $\mathcal{T}_0 \models_\Sigma \mathcal{T}_1$, if for every model $\mathcal{I}_0$ of $\mathcal{T}_0$, there exists a model $\mathcal{I}_1$ of $\mathcal{T}_1$ with $\mathcal{I}_0|_\Sigma = \mathcal{I}_1|_\Sigma$;
- $\mathcal{T}_0$ and $\mathcal{T}_1$ are *semantically Σ-inseparable*, written $\mathcal{T}_0 \equiv_\Sigma \mathcal{T}_1$, if $\mathcal{T}_0 \models_\Sigma \mathcal{T}_1$ and $\mathcal{T}_1 \models_\Sigma \mathcal{T}_0$;
- $\mathcal{T}_1$ is a *model-conservative extension of* $\mathcal{T}_0$ w.r.t. Σ if $\mathcal{T}_1 \supseteq \mathcal{T}_0$ and $\mathcal{T}_0 \equiv_\Sigma \mathcal{T}_1$;
- $\mathcal{T}_0$ is a *semantic Σ-tautology* if $\mathcal{T}_0 \equiv_\Sigma \emptyset$.

Intuitively, two general TBoxes are semantically Σ-inseparable if they induce the same dependencies between Σ-concepts in a very strong sense: it can be shown that $\mathcal{T}_0 \equiv_\Sigma \mathcal{T}_1$ iff for every sentence φ of *second-order logic* which uses no symbols from $\mathsf{sig}(\mathcal{T}_0 \cup \mathcal{T}_1) \setminus \Sigma$, we have $\mathcal{T}_0 \models \varphi$ iff $\mathcal{T}_1 \models \varphi$. We give examples of typical applications of the notions introduced above.

Example 2. Semantic Σ-inseparability of TBoxes $\mathcal{T}_0$ and $\mathcal{T}_1$ implies that

(∗) $\mathcal{T}_0 \cup \mathcal{T} \models C \sqsubseteq D$ iff $\mathcal{T}_1 \cup \mathcal{T} \models C \sqsubseteq D$, for all TBoxes $\mathcal{T}$ and CIs $C \sqsubseteq D$ with $\mathcal{T}$, C, D formulated in any standard description logic and not using symbols from $\mathsf{sig}(\mathcal{T}_0 \cup \mathcal{T}_1) \setminus \Sigma$.

Assume, for example, that $\mathcal{T}$ is a terminology describing terms related to hospital administration which uses a set Σ of medical terms from $\mathcal{T}_1 =$ SNOMED CT. If the designer of $\mathcal{T}$ knows that $\mathcal{T}_1$ and another medical terminology $\mathcal{T}_0$ are semantically Σ-inseparable, then it does not make any difference whether he imports $\mathcal{T}_1$ or $\mathcal{T}_0$ into $\mathcal{T}$. If $\mathcal{T}_0$ is much smaller than $\mathcal{T}_1$, the latter might be preferable. Observe that the quantification over $\mathcal{T}$ in (∗) ensures that this property does not break when $\mathcal{T}$ evolves.

Example 3. It follows from (∗) that for any semantic Σ-tautology $\mathcal{T}_0$, the following holds: for all TBoxes $\mathcal{T}$ and CIs $C \sqsubseteq D$ such that $\mathcal{T}$, C, and D use no symbols from $\mathsf{sig}(\mathcal{T}_0 \cup \mathcal{T}_1) \setminus \Sigma$, $\mathcal{T} \models C \sqsubseteq D$ iff $\mathcal{T}_0 \cup \mathcal{T} \models C \sqsubseteq D$. Thus, one can import into $\mathcal{T}_0$ any such $\mathcal{T}$ without changing the dependencies that $\mathcal{T}$ induces between terms in Σ. If $\mathcal{T}_0$ is a terminology for hospital administration and a semantic Σ-tautology for a set Σ of medical terms defined in SNOMED CT, then one can import SNOMED CT into $\mathcal{T}_0$ without corrupting the meaning of medical terms defined in SNOMED CT. This application is discussed in detail in [6] under the name of *safety* for a signature Σ.

To illustrate the difference between entailment-based notions of inseparability as in [9, 11] and semantic Σ-inseparability consider the following example. Let

$$\Sigma = \{A, B\} \qquad \text{and} \qquad \mathcal{T}_1 = \{A \sqsubseteq \exists r.B\}.$$

Observe that, in models $\mathcal{I}$ of $\mathcal{T}_1$, $A^{\mathcal{I}} \neq \emptyset$ implies $B^{\mathcal{I}} \neq \emptyset$. Thus, $\mathcal{T}_1$ is not a semantic Σ-tautology. However, this dependency between A and B cannot be expressed in terms of a CI, and $\mathcal{T}_1$ entails the same Σ-CIs as the empty TBox in all of the DLs introduced in Section 2. Slightly more complex examples show that even property (∗) above is not equivalent to semantic Σ-inseparability. The exact relation between semantic

Σ-inseparability and entailment-based notions of conservative extensions has been investigated in detail in [5, 6, 9, 11].

Throughout this paper, we consider two kinds of modules.

Definition 4. Let $\mathcal{T}_0 \subseteq \mathcal{T}_1$ be general TBoxes and $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_0)$. Then $\mathcal{T}_0$ is a

- *weak semantic* Σ*-module* of $\mathcal{T}_1$ iff $\mathcal{T}_1$ is semantically Σ-inseparable from $\mathcal{T}_0$;
- *strong semantic* Σ*-module* of $\mathcal{T}_1$ iff $\mathcal{T}_1 \setminus \mathcal{T}_0$ is a semantic Σ-tautology.

The requirement that $\mathcal{T}_0$ only contains Σ-symbols reflects the idea that modules should be self-contained: if an ontology $\mathcal{T}$ induces a dependency between symbols occurring in $\mathcal{T}_0$, then this dependency is induced by $\mathcal{T}_0$ already. Notions of a module in which this is not the case are of interest as well and are considered, e.g., in [2].

Lemma 5. *Every strong semantic module is a weak semantic module. The converse fails for acyclic $\mathcal{EL}$-terminologies.*

Proof. The first part is obvious. For the second part, let $\mathcal{T}_0 = \{A \equiv \top\}$, $\mathcal{T}_1 = \mathcal{T}_0 \cup \{B \sqsubseteq A\}$, and $\Sigma = \{A, B\}$. Then $\mathcal{T}_0$ is a weak semantic module of $\mathcal{T}_1$, but not a strong semantic module. □

Intuitively, the difference between weak and strong modules is that strong modules *additionally* require the ontology without the module to not induce any dependencies between symbols in Σ.

4 Deciding semantic Σ-consequence

It has been observed in [6, 11, 9] that semantic notions of entailment and inseparability as given in Definition 1 tend to be computationally difficult. Indeed, we can prove a strong undecidability result for deciding semantic Σ-tautologies using a reduction of the validity of a bimodal formula on a frame.

Theorem 6. *Given an acyclic $\mathcal{ALC}$-terminology $\mathcal{T}$, it is undecidable whether $\mathcal{T}$ is a semantic Σ-tautology. This even holds for acyclic $\mathcal{ALC}$-terminologies of the form $\{A \sqsubseteq C\}$ and for $\Sigma = \{A, r_1, r_2\}$.*

By definition of modules, Theorem 6 implies that it is not possible to decide, given acyclic $\mathcal{ALC}$-terminologies $\mathcal{T}_1$ and $\mathcal{T}_0 \subseteq \mathcal{T}_1$ and a signature Σ, whether $\mathcal{T}_0$ is a weak/strong semantic Σ-module in $\mathcal{T}_1$. Thus, Theorem 6 and related results explain why the notions of modularity from Definition 4 have not yet found practical applications. Instead, applications use notions of a module based on locality [6] or deductive versions of inseparability [2], or notions of a module that are not logic-based [12, 4, 3]. One aim of this paper is to challenge this approach by identifying relevant cases in which reasoning about modules as defined in Section 3 is decidable, and sometimes even tractable. A first observation is that avoiding roles in Σ improves the situation.

Theorem 7. *Let $\mathcal{L}$ be $\mathcal{ALC}$ or $\mathcal{ALCI}$. Given general $\mathcal{L}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_0$ and a signature Σ with $\mathsf{sig}(\mathcal{T}_i) \cap \Sigma \subseteq \mathsf{N_C}$ for $i = 0, 1$, it is*

(1) coNExp$^{\text{NP}}$-complete to decide whether $\mathcal{T}_0 \equiv_\Sigma \mathcal{T}_1$; if Σ is fixed, then this problem is coNP$^{\text{NExp}}$-complete;

(2) Π_2^p-complete to decide whether $\mathcal{T}_0$ is a semantic Σ-tautology. The lower bound applies already to acyclic TBoxes.

Observe that deciding semantic Σ-tautologies under the restrictions given in Theorem 7 is easier than deciding satisfiability and subsumption in $\mathcal{ALC}$ w.r.t. acyclic TBoxes, which is PSpace-complete [10].

We remark that Theorem 7 is also of interest when analysing merged TBoxes, as it implies decidability of the following problem: given general $\mathcal{ALC}$-TBoxes $\mathcal{T}_0$ and $\mathcal{T}_1$ such that the set of symbols Σ shared by $\mathcal{T}_0$ and $\mathcal{T}_1$ contains only concept names, decide whether the union $\mathcal{T}_0 \cup \mathcal{T}_1$ is semantically Σ-inseparable from $\mathcal{T}_0$ and $\mathcal{T}_1$.

5 Deciding semantic modules

Theorem 7 suggests that controlling the role names in Σ can help to overcome undecidability of semantic modules. We identify a syntactic restriction that is inspired by this observation and recovers decidability of semantic modules for acyclic terminologies formulated in $\mathcal{ALC}$ and $\mathcal{ALCI}$. It also provides the basis for showing that, in $\mathcal{EL}$ and $\mathcal{ELI}$, we can decide semantic modules for acyclic terminologies without any further restrictions.

Definition 8. Let $\mathcal{T}$ be an acyclic terminology and $\Sigma, \Sigma_1, \Sigma_2$ signatures. $\mathcal{T}$ *contains a syntactic* (Σ_1, Σ_2)*-dependency* if there exists a concept name $A \in \Sigma_1$ such that $\mathsf{depend}_\mathcal{T}(A) \cap \Sigma_2 \neq \emptyset$. A syntactic (Σ, Σ)-dependency is called a *syntactic Σ-dependency*.

The notion of a syntactic (Σ_1, Σ_2)-dependency generalises the notion of acyclicity ($A \notin \mathsf{depend}_\mathcal{T}(A)$) to pairs of sets of symbols. Syntactic Σ-dependencies give rise to a natural case in which semantic modules in $\mathcal{ALCI}$ are decidable.

Theorem 9. *Let $\mathcal{L}$ be $\mathcal{ALC}$ or $\mathcal{ALCI}$. For acyclic $\mathcal{L}$-terminologies $\mathcal{T}_1 \supseteq \mathcal{T}_0$ and signature $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_0)$ such that $\mathcal{T}_1 \setminus \mathcal{T}_0$ contains no syntactic $(\Sigma, \Sigma \cap \mathsf{N_R})$-dependencies, it is*

(1) decidable in coNExp$^{\text{NP}}$ whether $\mathcal{T}_0$ is a weak semantic Σ-module of $\mathcal{T}_1$; this problem is coNExpTime-hard;

(2) Π_2^p-complete to decide whether $\mathcal{T}_0$ is a strong semantic Σ-module of $\mathcal{T}_1$.

We conjecture that the problem in Point (1) is actually coNExp$^{\text{NP}}$-complete. It is natural to consider also a stronger syntactic condition, namely to disallow *any* Σ-dependency instead of only $(\Sigma, \Sigma \cap \mathsf{N_R})$-dependencies. In this case, the notions of strong and weak semantic modules coincide and deciding modules is only Π_2^p-complete for acyclic $\mathcal{ALC}$- and $\mathcal{ALCI}$-terminologies.

Theorem 10. *Let $\mathcal{L}$ be $\mathcal{ALC}$ or $\mathcal{ALCI}$. For acyclic $\mathcal{L}$-terminologies $\mathcal{T}_1 \supseteq \mathcal{T}_0$ and a signature $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_0)$ such that $\mathcal{T}_1 \setminus \mathcal{T}_0$ contains no syntactic Σ-dependencies, the following are equivalent:*

- *$\mathcal{T}_0$ is a strong semantic Σ-module of $\mathcal{T}_1$;*
- *$\mathcal{T}_0$ is a weak semantic Σ-module of $\mathcal{T}_1$;*
- *for all $P \subseteq \Sigma \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus \mathsf{Pr}(\mathcal{T}_1))$, the following concept is satisfiable in a model of $\mathcal{T}_1 \setminus \mathcal{T}_0$ of cardinality 1:*

$$C_P = \bigsqcap_{A \in P} A \sqcap \bigsqcap_{A \in \Sigma \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus (\mathsf{Pr}(\mathcal{T}_1) \cup P))} \neg A.$$

It is Π_2^p-complete to decide whether $\mathcal{T}_0$ is a weak/strong semantic module of $\mathcal{T}_1$.

Output "not module" if any of the two conditions applies, and "module" otherwise:

1. there exists $A \in \Sigma \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus \mathsf{Pr}(\mathcal{T}_1))$ such that $\mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap \Sigma \neq \emptyset$;
2. there exists $A \in \Sigma \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus \mathsf{Pr}(\mathcal{T}_1))$ such that $A \equiv C \in \mathcal{T}_1$ for some C and

$$\bigcup_{B \in \Sigma \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus (\mathsf{Pr}(\mathcal{T}_1) \cup \{A\}))} \mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(B) \supseteq \mathsf{depend}^{\equiv}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap \mathsf{PPr}(\mathcal{T}_1 \setminus \mathcal{T}_0)$$

Figure 2. Module checking in $\mathcal{ELI}$

We now consider $\mathcal{EL}$ and $\mathcal{ELI}$. Theorems 6 and 9 show that, in the case of acyclic $\mathcal{ALCI}$-TBoxes, $(\Sigma, \Sigma \cap \mathsf{N_R})$-dependencies are the culprit for undecidability of semantic Σ-tautologies. In $\mathcal{EL}$ and $\mathcal{ELI}$, the situation is rather different. Here, dealing with Σ-dependencies (and thus also $(\Sigma, \Sigma \cap \mathsf{N_R})$-dependencies) is trivial.

Lemma 11. *Let $\mathcal{L}$ be $\mathcal{EL}$ or $\mathcal{ELI}$. If $\mathcal{T}$ is an acyclic $\mathcal{L}$-terminology that contains a syntactic Σ-dependency, then $\mathcal{T}$ is not a semantic Σ-tautology.*

Proof. In any model $\mathcal{I}$ of an acyclic $\mathcal{ELI}$-terminology $\mathcal{T}$, from $X \in \mathsf{depend}_{\mathcal{T}}(A)$ and $X^{\mathcal{I}} = \emptyset$ it follows that $A^{\mathcal{I}} = \emptyset$. The lemma follows immediately. □

Based on Lemma 11, we show that in $\mathcal{EL}$ and $\mathcal{ELI}$, modules can be decided and extracted in polytime. In what follows, we work with acyclic $\mathcal{EL}$-terminologies $\mathcal{T}$ that *contain no trivial axioms*, i.e., no axiom in $\mathcal{T}$ is of the form $A \equiv \top$ (nor $A \equiv \top \sqcap \top$, etc.). In acyclic $\mathcal{EL}$-terminologies, any such A can be eliminated by replacing it with $\top$. Thus, it is harmless to disregard trivial axioms.

Theorem 12. *Let $\mathcal{L}$ be $\mathcal{EL}$ or $\mathcal{ELI}$. For acyclic $\mathcal{L}$-terminologies $\mathcal{T}_1 \supseteq \mathcal{T}_0$ containing no trivial axioms and signature $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_0)$, the following are equivalent:*

- *$\mathcal{T}_0$ is a strong semantic Σ-module of $\mathcal{T}_1$;*
- *$\mathcal{T}_0$ is a weak semantic Σ-module of $\mathcal{T}_1$.*

It is decidable in polytime whether $\mathcal{T}_0$ is a weak/strong semantic module of $\mathcal{T}_1$.

The polytime bound of Theorem 12 is established by the algorithm in Figure 2, which takes as input acyclic $\mathcal{ELI}$-terminologies $\mathcal{T}_1 \supseteq \mathcal{T}_0$ and a signature $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_0)$. In the formulation of the algorithm, we use the following notation. A concept name $A \in \mathsf{N_C}$ *directly $\equiv$-depends on* $X \in \mathsf{N_C} \cup \mathsf{N_R}$, in symbols $A \prec^{\equiv}_{\mathcal{T}} X$, iff there exists $A \equiv C \in \mathcal{T}$ such that X occurs in C. Then, $\mathsf{depend}^{\equiv}_{\mathcal{T}}(A)$ denotes the set of all X such that (A, X) is in the transitive closure of $\prec^{\equiv}_{\mathcal{T}}$. The algorithm takes as input acyclic $\mathcal{ELI}$-terminologies $\mathcal{T}_1 \supseteq \mathcal{T}_0$ and a signature $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_0)$.

Theorem 12 yields the interesting result that, for acyclic $\mathcal{ELI}$-terminologies, checking semantic modules is computationally simpler than subsumption, which is PSPACE-complete [7].

6 Module extraction

Based on the results given in the previous section, we devise algorithms for extracting modules from acyclic $\mathcal{ALCI}$- and $\mathcal{ELI}$-terminologies. We start with $\mathcal{ALCI}$, for which an extraction algorithm is given in Figure 3. It takes as input an acyclic $\mathcal{ALCI}$-terminology $\mathcal{T}_1$ and a signature Σ, and it outputs a module $\mathcal{T}_0$ as described by the following theorem.

Initialise: $\mathcal{T}_0 = \emptyset$.
Apply Rules 1 and 2 exhaustively, preferring Rule 1
Output $\mathcal{T}_0$.

1. if $A \in \Sigma \cup \mathsf{sig}(\mathcal{T}_0)$, $\alpha \in \mathcal{T}_1 \setminus \mathcal{T}_0$ has A on the left hand side, and $\mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap (\Sigma \cup \mathsf{sig}(\mathcal{T}_0)) \neq \emptyset$, set $\mathcal{T}_0 := \mathcal{T}_0 \cup \{\alpha\}$,
2. if $\alpha \in \mathcal{T}_1 \setminus \mathcal{T}_0$ with A on the left-hand side and there is a minimal subset $Q \subseteq (\Sigma \cup \mathsf{sig}(\mathcal{T}_0)) \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus \mathsf{Pr}(\mathcal{T}_1))$ such that $A \in Q$ and for some $P \subseteq Q$, the concept

$$C_{P,Q} = \bigsqcap_{A \in P} A \sqcap \bigsqcap_{A \in Q \setminus P} \neg A$$

is not satisfiable in a one-point model of $\mathcal{T}_1 \setminus \mathcal{T}_0$, then set $\mathcal{T}_0 := \mathcal{T}_0 \cup \{\alpha\}$.

Figure 3. Module extraction in $\mathcal{ALCI}$

Initialise: $\mathcal{T}_0 = \emptyset$.
Apply Rules 1 and 2 exhaustively, preferring Rule 1
Output $\mathcal{T}_0$.

1. if $A \in \Sigma \cup \mathsf{sig}(\mathcal{T}_0)$, $\alpha \in \mathcal{T}_1 \setminus \mathcal{T}_0$ has A on the left hand side, and $\mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap (\Sigma \cup \mathsf{sig}(\mathcal{T}_0)) \neq \emptyset$, set $\mathcal{T}_0 := \mathcal{T}_0 \cup \{\alpha\}$.
2. if $A \in \Sigma \cup \mathsf{sig}(\mathcal{T}_0)$, $A \equiv C \in \mathcal{T}_1 \setminus \mathcal{T}_0$, and

$$\bigcup_{B \in (\Sigma \cup \mathsf{sig}(\mathcal{T}_0)) \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus (\mathsf{Pr}(\mathcal{T}_1) \cup \{A\}))} \mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(B) \supseteq \mathsf{depend}^{\equiv}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap \mathsf{PPr}(\mathcal{T}_1 \setminus \mathcal{T}_0),$$

set $\mathcal{T}_0 := \mathcal{T}_0 \cup \{A \equiv C\}$.

Figure 4. Module extraction in $\mathcal{ELI}$

Theorem 13. *Let $\mathcal{T}_1$ be an acyclic $\mathcal{ALCI}$-terminology and Σ a signature. The output $\mathcal{T}_0$ of the algorithm in Figure 3 is the unique smallest strong (equivalently, weak) semantic $\Sigma \cup \mathsf{sig}(\mathcal{T}_0)$-module of $\mathcal{T}_1$ such that $\mathcal{T}_1 \setminus \mathcal{T}_0$ contains no syntactic Σ-dependencies.*

The condition that $\mathcal{T}_1 \setminus \mathcal{T}_0$ contains no syntactic Σ-dependencies is essential. Without it, we would have smaller modules, but cannot extract them automatically. The latter follows from Theorem 6 and the fact that, without the mentioned condition, the smallest semantic Σ-module of a terminology $\mathcal{T}_1$ is empty iff $\mathcal{T}_1$ is a semantic Σ-tautology. Also observe that the module $\mathcal{T}_0$ extracted by the algorithm is not necessarily formulated in Σ, but may contain additional symbols. This is clearly unavoidable even in simple cases, e.g. when extracting a semantic $\{A, B\}$-module from the terminology $\{A \sqsubseteq B \sqcap B'\}$. If implemented carefully, the check whether Rule 2 is applicable to a given axiom $\alpha \in \mathcal{T}_1 \setminus \mathcal{T}_0$ can be done in Σ^p_2 (and is also hard for Σ^p_2). Apart from this, the algorithm runs in polynomial time.

The algorithm for module extraction in $\mathcal{ALCI}$ first checks for syntactic Σ-dependencies and then applies (a variation of) module checking. When extracting modules from $\mathcal{ELI}$-

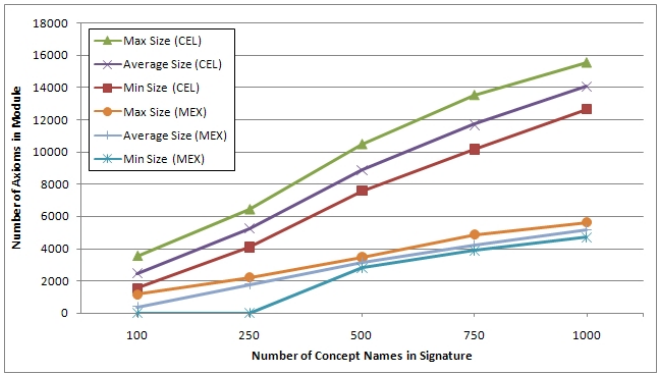


Figure 5. Sizes of $\bot$-local modules and semantic modules

terminologies, we apply the same strategy. In contrast to $\mathcal{ALCI}$, we know from Lemma 11 that if $\mathcal{T}_0 \subseteq \mathcal{T}_1$ is such that $\mathcal{T}_1 \setminus \mathcal{T}_0$ contains a Σ-dependency, then $\mathcal{T}_0$ is not a weak (equivalently, strong) Σ-module of $\mathcal{T}_1$. It follows that we do not need the additional condition on modules from Theorem 13, i.e., that $\mathcal{T}_1 \setminus \mathcal{T}_0$ contains no Σ-dependency. The extraction algorithm for $\mathcal{ELI}$ is given in Figure 4. It takes as input an acyclic $\mathcal{ELI}$-terminology $\mathcal{T}_1$ and a signature Σ, and it outputs a semantic module $\mathcal{T}_0$ as described by the following theorem.

Theorem 14. *Let $\mathcal{T}_1$ be an acyclic $\mathcal{ELI}$-terminology containing no trivial axioms and Σ a signature. The output $\mathcal{T}_0$ of the algorithm in Figure 4 is the unique smallest strong (equivalently, weak) semantic $\Sigma \cup \mathsf{sig}(\mathcal{T})$-module of $\mathcal{T}_1$.*

Example 15. Consider again the scenario described in Example 2, but now suppose that $\mathcal{T}_0$ is the output of the algorithm of Figure 4 applied to SNOMED CT and Σ. Then it does not make any difference whether the user imports $\mathcal{T}_0$ or SNOMED CT into $\mathcal{T}$. Experimental results in the next section show that $\mathcal{T}_0$ is often much smaller than SNOMED CT.

7 Experiments with MEX

To evaluate our approach to module extraction, we have carried out a number of experiments on the medical terminology SNOMED CT, an acyclic $\mathcal{EL}$-terminology that additionally comprises role inclusion statements of the form $r \sqsubseteq s$ (*role hierarchies*) and $r \circ s \sqsubseteq r$ (*right identities*). A variation of the algorithm in Figure 4 that addresses this case is presented in the technical report accompanying this paper. It was implemented in the system MEX, which is written in OCaml.

The main aim of our experiments is to compare the size of modules extracted by MEX with the size of minimal $\bot$-local modules as introduced in [6]. For $\mathcal{EL}$, $\bot$-local modules coincide with the modules extracted by the extraction feature of the CEL reasoner [14], which is used in the experiments below. We have used the SNOMED CT version of February 2005, which comprises 379 691 axioms. Experiments are based on randomly selected signatures of size between 100 and 1 000 symbols and were carried out for 1 000 different signatures of each size. Figure 5 shows the maximal, minimal, and average module sizes depending on the size of the signature. Note that, in every case, the largest semantic module is smaller than the smallest $\bot$-local module.

Additionally to generating small modules, MEX is rather efficient regarding runtime and memory consumption. We have carried out the experiments on a PC with Intel® Core™ 2 CPU at 2.13 GHz and with 3 GB of RAM. For all signature sizes in Figure 5, the average time of module extraction was 4.1 seconds and at most 124.7 MB of memory were consumed. This performance does not significantly decrease with large signature sizes: the average time and space consumed by MEX when extracting a module for 10 000 symbols in 5 seconds and 121.7 MByte. For 100 000 symbols, it is merely 9.6 seconds and 134.6 MByte.

8 Discussion

We have presented semantic notions of a module in a DL terminology and algorithms for checking and extracting such modules. Our experiments show that, at least in lightweight DLs of the $\mathcal{EL}$ family, highly efficient practical implementations of our algorithms are possible. We are optimistic that also the extraction algorithm for $\mathcal{ALCI}$ can be implemented in a reasonably efficient way.

9 Acknowledgements

The authors were supported by EPSRC grant EP/E065279/1.

REFERENCES

[1] F. Baader and C. Lutz and B. Suntisrivaraporn, 'CEL—A Polynomial-time Reasoner for Life Science Ontologies', in *Proc. of IJCAR'06*, pp. 287–291, (2006).
[2] A. Borgida, 'On importing knowdledge from DL ontologies: some intuitions and problems', in *Proc. of DL-07*, (2007).
[3] P. Doran, V. Tamma, and L.Iannone, 'Ontology module extraction for ontology reuse:an ontology engineering perspective', in *Proc. of CIKM-07*, (2007).
[4] J. Gennari et al., 'The evolution of protégé: an environment for knowledge-based systems development', *Int. J. Hum.-Comput. Stud.*, **58**(1), 89–123, (2003).
[5] B. Cuenca Grau, I. Horrocks, Y. Kazakov, and U. Sattler., 'A logical framework for modularity of ontologies', in *Proc of IJCAI'07*. AAAI Press, (2007).
[6] B. Cuenca Grau, I. Horrocks, Y. Kazakov, and U. Sattler, 'Just the right amount: extracting modules from ontologies', in *Proc. of WWW-07*, pp. 717–726, (2007).
[7] C. Haase and C. Lutz, 'Complexity of subsumption in the $\mathcal{EL}$ family of description logics: Acyclic and cyclic TBoxes', in *Proc. of ECAI-08*. (2008).
[8] C. Lutz, D. Walther, and F. Wolter, 'Conservative extensions in expressive description logics', in *Proc. of IJCAI-07*. AAAI Press, (2007).
[9] C. Lutz, 'Complexity of terminological reasoning revisited', in *Proc. of LPAR'99*, number 1705 in LNAI, pp. 181–200. Springer, (1999).
[10] C. Lutz and F. Wolter, 'Conservative extensions in the lightweight description logic $\mathcal{EL}$', in *Proc. of CADE-2007*. Springer, (2007).
[11] J. Seidenberg and A.L. Rector, 'Web ontology segmentation: analysis, classification and use', in *Proc. of WWW-06*, pp. 13–22, (2006).
[12] K.A. Spackman, 'Managing clinical terminology hierarchies using algorithmic calculation of subsumption: Experience with SNOMED-RT', *JAMIA*, (2000).
[13] B. Suntisrivaraporn, 'Module Extraction and Incremental Classification: A Pragmatic Approach for $\mathcal{EL}^+$ Ontologies', in *Proc. of ESWC-2008*. Springer, (2008).

A Proof of Theorem 6

Theorem 6. Given an acyclic $\mathcal{ALC}$-terminology $\mathcal{T}$, it is undecidable whether $\mathcal{T}$ is a Σ-tautology for $\Sigma = \{A, r_1, r_2\}$. This even holds for $\mathcal{ALC}$-terminologies of the form $\{A \sqsubseteq C\}$.

Proof. The proof is by reduction of the universal consistency problem in modal logic. To define this problem in terms of description logic, let r_1 and r_2 be fixed role names. 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 all $i \in \{1, 2\}$. An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ is *based on* a frame $\mathcal{F}$ iff $\Delta^{\mathcal{I}} = \Delta^{\mathcal{F}}$ and $r_i^{\mathcal{I}} = r^{\mathcal{F}}$ for all $i \in \{1, 2\}$. We say that an $\mathcal{ALC}$-concept C is *valid* on $\mathcal{F}$ and write $\mathcal{F} \models C$ iff $C^{\mathcal{I}} = \Delta^{\mathcal{I}}$ for every interpretation $\mathcal{I}$ based on $\mathcal{F}$. A concept C is *universally consistent* if there exist a frame $\mathcal{F}$ such that C is valid on $\mathcal{F}$. The *universal consistency problem* in modal logic is now defined as follows: Given an $\mathcal{ALC}$-concept C formulated in the signature $\mathsf{N_C} \cup \{r_1, r_2\}$, decide whether C is universally consistent. By a result of Thomason, this problem is undecidable [**?**].

For the reduction, let C be an $\mathcal{ALC}$-concept formulated in the signature $\mathsf{N_C} \cup \{r_1, r_2\}$. Fix an $A \in \mathsf{N_C}$ that does not occur in C. We claim that C is universally consistent iff $\mathcal{T} = \{A \sqsubseteq \exists u.\neg C\}$ is *not* a Σ-tautology, where $\Sigma = \{A, r_1, r_2\}$ and $u \in \mathsf{N_R}$ is a role name.

Suppose first that C is universally consistent. Then there is a frame $\mathcal{F}$ on which C is valid. Let $\mathcal{I}$ be an interpretation based on $\mathcal{F}$ that interprets only the symbols in Σ and satisfies $A^{\mathcal{I}} \neq \emptyset$. To show that $\mathcal{T}$ is not a Σ-tautology, it suffices to show that any way to extend $\mathcal{I}$ to an interpretation $\mathcal{I}'$ by interpreting the role name u and the concept names in C does not result in a model of $\mathcal{T}$. Let $\mathcal{I}'$ be such an extension. Since $\mathcal{I}$ is based in $\mathcal{F}$, we have $C^{\mathcal{I}'} = \Delta^{\mathcal{I}'}$. Together with $A^{\mathcal{I}'} \neq \emptyset$, it follows that $\mathcal{I}'$ is not a model of $\mathcal{T}$.

Conversely, suppose that $\mathcal{T}$ is not a Σ-tautology. Then there is an interpretation $\mathcal{I}$ that interprets only the symbols in Σ and cannot be extended to a model $\mathcal{I}'$ of $\mathcal{T}$ by interpreting the role name u and the concept names in C. Let $\mathcal{F}$ be the frame which $\mathcal{I}$ is based on. To show that C is universally consistent, it suffices to show that C is valid on $\mathcal{F}$. Let $\mathcal{J}$ be an interpretation based on $\mathcal{F}$ and assume, by contradiction, that $C^{\mathcal{J}} \neq \Delta^{\mathcal{J}}$. Then the interpretation $\mathcal{J}'$, which is obtained from $\mathcal{J}$ by setting $A^{\mathcal{J}'} = A^{\mathcal{I}}$ and $u^{\mathcal{J}'} := \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, extends $\mathcal{I}$ and is a model of $\mathcal{T}$; contradiction. □

B Proof of Theorem 7

Theorem 7. Let $\mathcal{L}$ be $\mathcal{ALC}$ or $\mathcal{ALCI}$. Given general $\mathcal{L}$-TBoxes $\mathcal{T}_1$ and $\mathcal{T}_0$ and a signature Σ with $\mathsf{sig}(\mathcal{T}_i) \cap \Sigma \subseteq \mathsf{N_C}$ for $i = 0, 1$, it is

(1) coNExp$^{\text{NP}}$-complete to decide whether $\mathcal{T}_0 \equiv_\Sigma \mathcal{T}_1$; if Σ is fixed, then this problem is coNP$^{\text{NExp}}$-complete;

(2) Π_2^p-complete to decide whether $\mathcal{T}_0$ is a Σ-tautology. The lower bound applies already to acyclic TBoxes.

Proof. To prove (1), we show that the problem whether $\mathcal{T}_0$ Σ-entails $\mathcal{T}_1$ (with $\mathsf{sig}(\mathcal{T}_1) \cap \Sigma \subseteq \mathsf{N_C}$) is polynomially equivalent to the satisfiability problem for simple concept circumscribed TBoxes which has been shown to be NExp$^{\text{NP}}$-complete for unbounded numbers of fixed and minimized concept names, and NP$^{\text{NExp}}$-complete if the number of fixed and minimized concept names is fixed in advance [**?**]. The result for Σ-inseparability can then be derived in a straighforward way.

A *simple concept circumscribed TBox* $\mathsf{Circ}_{M,F}(\mathcal{T})$ consists of finite sets M and F of concept names and a TBox $\mathcal{T}$. To define the semantics of simple concept circumscribed TBoxes, we define a preference relation $<_{M,F}$ on interpretations by setting $\mathcal{I} <_{M,F} \mathcal{I}'$ iff $\Delta^{\mathcal{I}} = \Delta^{\mathcal{I}'}$, $A^{\mathcal{I}} = A^{\mathcal{I}'}$ for all $A \in F$, $A^{\mathcal{I}} \subseteq A^{\mathcal{I}'}$ for all $A \in M$, and there exists $A \in M$ such that $A^{\mathcal{I}} \subset A^{\mathcal{I}'}$. An interpretation $\mathcal{I}$ *satisfies* $\mathsf{Circ}_{M,F}(\mathcal{T})$ if $\mathcal{I}$ is a model of $\mathcal{T}$ and there does not exist a model $\mathcal{I}'$ of $\mathcal{T}$ with $\mathcal{I}' <_{M,F} \mathcal{I}$. A concept C is *satisfiable w.r.t.* $\mathsf{Circ}_{M,F}(\mathcal{T})$ if there exists an interpretation $\mathcal{I}$ satisfying $\mathsf{Circ}_{M,F}(\mathcal{T})$ with $C^{\mathcal{I}} \neq \emptyset$. It has been shown in [**?**] that, for the description logics $\mathcal{ALC}$ and $\mathcal{ALCI}$, satisfiability of concepts w.r.t. simple concept circumscribed TBoxes is NExp$^{\text{NP}}$-complete for unbounded M, F and NP$^{\text{NExp}}$-complete, for bounded M, F.

We now prove polynomial equivalence of the respective problems. First, we reduce circumscription to Σ-entailment. Assume that an $\mathcal{ALCI}$-TBox $\mathcal{T}$, mutually disjoint finite sets of concept names M and F, and an $\mathcal{ALCI}$-concept C are given. Let

$$\mathcal{T}_0 = \mathcal{T} \cup \{\top \sqsubseteq \exists \mathsf{aux}.C\},$$

where aux is a new role name, and $\Sigma = M \cup F$. Introduce, for each $X \in \mathsf{sig}(\mathcal{T}) \cup M \cup F$ a copy X' of X and denote by $\mathcal{T}'$ the TBox resulting from $\mathcal{T}$ when every occurence of X is replaced by X'. Take a fresh role name aux' and define $\mathcal{T}_1$ by taking the union of $\mathcal{T} \cup \mathcal{T}'$ and

- $A' \sqsubseteq A$, 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 $\mathcal{T}_0$ does not Σ-entail $\mathcal{T}_1$.

Proof. Assume that C is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$. Let $\mathcal{I}$ satisfy $\mathsf{Circ}_{M,F}(\mathcal{T})$ and $C^{\mathcal{I}} \neq \emptyset$. By setting $\mathsf{aux}^{\mathcal{I}} = \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, we may assume that $\mathcal{I}$ is a model of $\mathcal{T}_0$. We prove that there is no model of $\mathcal{T}_1$ which coincides with $\mathcal{I}$ on Σ. Suppose the contrary and let $\mathcal{I}'$ be such a model. Consider the interpretation $\mathcal{I}''$ resulting from $\mathcal{I}'$ by setting $X^{\mathcal{I}''} = X'^{\mathcal{I}'}$, for all $X \in \mathsf{sig}(\mathcal{T}) \cup M \cup F$. Then $\mathcal{I}'' <_{M,F} \mathcal{I}$ and $\mathcal{I}''$ is a model of $\mathcal{T}$. So we have a contradiction to the assumption that $\mathcal{I}$ satisfies $\mathsf{Circ}_{M,F}(\mathcal{T})$.

Conversely, assume that $\mathcal{T}_0$ does not Σ-entail $\mathcal{T}_1$. Take a model $\mathcal{I}$ of $\mathcal{T}_0$ such that there is no model $\mathcal{I}'$ of $\mathcal{T}_1$ which coincides with $\mathcal{I}$ on Σ. Clearly $C^{\mathcal{I}} \neq \emptyset$. Moreover, it is readily checked that there exists no model $\mathcal{I}'$ of $\mathcal{T}$ with $\mathcal{I}' <_{M,F} \mathcal{I}$. Therefore, C is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$.

We now polynomially reduce Σ-entailment to circumscription. Suppose $\mathcal{T}_0$, $\mathcal{T}_1$, and Σ are given and $\Sigma \cap \mathsf{sig}(\mathcal{T}_1) \subseteq \mathsf{N_C}$. We may assume that $\mathcal{T}_1 = \{C \equiv \top\}$, for some concept C. Set $M = \{A\}$, for a fresh concept name A, $F = \Sigma$, and $\mathcal{T} = \mathcal{T}_0 \cup \{A \equiv \neg C\}$.

Claim. A is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$ iff $\mathcal{T}_0$ does not Σ-entail $\mathcal{T}_1$.

Proof. Assume A is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$. Let $\mathcal{I}$ satisfy $\mathsf{Circ}_{M,F}(\mathcal{T})$ and $A^{\mathcal{I}} \neq \emptyset$. Then $\mathcal{I}$ satisfies $\mathcal{T}_0$. Suppose there exists an interpretation $\mathcal{I}'$ which coincides with $\mathcal{I}$ on Σ and that is a model of $\mathcal{T}_1$. Then $C^{\mathcal{I}'} = \Delta^{\mathcal{I}'}$ and, therefore, $A^{\mathcal{I}'} =$

$\emptyset \subset A^{\mathcal{I}}$. But then $\mathcal{I}' <_{M,F} \mathcal{I}$ and $\mathcal{I}'$ is a model of $\mathcal{T}$; a contradiction.

Conversely, assume that $\mathcal{T}_0$ does not Σ-entail $\mathcal{T}_1$. Let $\mathcal{I}$ be a model of $\mathcal{T}_0$ such that there is no model $\mathcal{I}'$ of $\mathcal{T}_1$ which coincides with $\mathcal{I}$ on Σ. A straighforward filtration argument (similar to the filtration argument in [**?**]) shows that we may assume that $\mathcal{I}$ is finite.

Let $\mathcal{K}$ be the set of interpretations which coincide with $\mathcal{I}$ on Σ and that are models of $\mathcal{T}_0$. As none of them is a model of $\mathcal{T}_1$, we have $(\neg C)^{\mathcal{J}} \neq \emptyset$ for all $\mathcal{J} \in \mathcal{K}$. As $\mathcal{I}$ is finite, there exists $\mathcal{I}' \in \mathcal{K}$ such that there does not exist $\mathcal{J} \in \mathcal{K}$ with $(\neg C)^{\mathcal{J}} \subset (\neg C)^{\mathcal{I}'}$. We may assume that $A^{\mathcal{I}'} = (\neg C)^{\mathcal{I}'}$. Then $\mathcal{I}'$ is a model of $\mathcal{T}$ and $A^{\mathcal{I}'} \neq \emptyset$. Moreover, by construction, there exists no model $\mathcal{I}''$ of $\mathcal{T}$ with $\mathcal{I}'' <_{M,F} \mathcal{I}'$. Hence, A is satisfiable w.r.t. $\mathsf{Circ}_{M,F}(\mathcal{T})$.

(2) For the upper bound, we first show that the following are equivalent for any $\mathcal{ALCI}$-TBox $\mathcal{T}$:

1. $\mathcal{T}$ is a Σ-tautology;
2. for every one-point interpretaion $\mathcal{I}$, there exists a model $\mathcal{I}'$ of $\mathcal{T}$ with $\mathcal{I}|_\Sigma = \mathcal{I}'|_\Sigma$.

The direction from Point 1 to Point 2 is trivial. Conversely, assume that Point 2 is satisfied and let $\mathcal{I}$ be an interpretation that interprets only the symbols in Σ. We have to show that $\mathcal{I}$ can be extended to a model of $\mathcal{T}$ by interpreting the non-Σ symbols. For every $d \in \Delta^{\mathcal{I}}$, let $\mathcal{I}_d$ be the restriction of $\mathcal{I}$ to domain $\{d\}$. By Point 2, for every $d \in \Delta^{\mathcal{I}}$ there exists a model $\mathcal{I}'_d$ of $\mathcal{T}$ with $\mathcal{I}_d|_\Sigma = \mathcal{I}'_d|_\Sigma$. Define the interpretation $\mathcal{I}' = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}'})$ by setting

$$X^{\mathcal{I}'} = \bigcup_{d \in \Delta^{\mathcal{I}}} X^{\mathcal{I}'_d}$$

for every $X \in \mathsf{sig}(\mathcal{T})$. Clearly, $\mathcal{I}' \models \mathcal{T}$. It follows from $\mathsf{sig}(\mathcal{T}) \cap \Sigma \subseteq \mathsf{N_C}$ that $\mathcal{I}$ and $\mathcal{I}'$ coincide on Σ.

To show that it is in Π^p_2 to decide whether $\mathcal{T}$ is a Σ-tautology, we show that it is in Σ^p_2 to decide whether it is not. First note that it is clearly in co-NP to decide, given a one-point interpretation $\mathcal{I}$, whether there is no model $\mathcal{I}'$ of $\mathcal{T}$ with $\mathcal{I}|_\Sigma = \mathcal{I}'|_\Sigma$. To check whether $\mathcal{T}$ is *not* a Σ-tautology, we may thus guess a one-point-interpretation $\mathcal{I}$ and then use the mentioned co-NP problem as an oracle. This is a Σ^p_2-algorithm.

For the lower bound, we reduce validity of quantified boolean formulas (QBFs) of the form $\psi = \forall p_1, \dots, p_n \exists q_1, \dots, q_m \varphi$, which is Π^p_2-complete. For the reduction, we use the propositional variables in ψ as concept names, and an additional concept name A. Then, ψ is valid iff the TBox $\{A \sqsubseteq \varphi\}$ is a Σ-tautology for $\Sigma = \{p_1, \dots, p_n, A\}$. Observe that no role names are needed for the reduction. □

C Proof of Theorem 9

Theorem 9. Let $\mathcal{L}$ be $\mathcal{ALC}$ or $\mathcal{ALCI}$. For acyclic $\mathcal{L}$-terminologies $\mathcal{T}_1 \supseteq \mathcal{T}_0$ and a signature $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_0)$ such that $\mathcal{T}_1 \setminus \mathcal{T}_0$ contains no syntactic $(\Sigma, \Sigma \cap \mathsf{N_R})$-dependencies, it is

(1) decidable in coNExp$^{\text{NP}}$ whether $\mathcal{T}_0$ is a weak semantic Σ-module of $\mathcal{T}_1$; this problem is co-NExpTime-hard;

(2) Π^p_2-complete to decide whether $\mathcal{T}_0$ is a strong semantic Σ-module of $\mathcal{T}_1$.

Proof. We only prove the co-NExpTime-hardness result. The upper bound in (1) can be proved similarly to the upper bound in Theorem 7. (2) can be proved similarly to the proof of Point 2 of Theorem 7.

We use a reduction from the following variant of the domino problem:

Definition 16 (Domino System)**.** A *domino system* $\mathfrak{D}$ is a triple (T, H, V), where $T = \{0, 1, \dots, k-1\}$, $k \geq 0$, is a finite set of *tile types* and $H, V \subseteq T \times T$ represent the horizontal and vertical matching conditions. Let $\mathfrak{D}$ be a domino system and $c = c_0, \dots, c_{n-1}$ an *initial condition*, i.e. an n-tuple of tile types. A mapping $\tau : \{0, \dots, 2^{n+1}-1\} \times \{0, \dots, 2^{n+1}-1\} \to T$ is a *solution* for $\mathfrak{D}$ and c iff for all $x, y < 2^{n+1}$, the following holds:

- if $\tau(x, y) = t$ and $\tau(x \oplus_{2^{n+1}} 1, y) = t'$, then $(t, t') \in H$
- if $\tau(x, y) = t$ and $\tau(x, y \oplus_{2^{n+1}} 1) = t'$, then $(t, t') \in V$
- $\tau(i, 0) = c_i$ for $i < n$.

where $\oplus_i$ denotes addition modulo i.

Given a domino system $\mathfrak{D}$ and an initial condition c, we describe TBoxes $\mathcal{T}_0$ and $\mathcal{T}_1$ such that there is a solution for $\mathfrak{D}$ and c iff there is a model $\mathcal{I}_0$ of $\mathcal{T}_0$ for which there is no model $\mathcal{I}_1$ of $\mathcal{T}_1$ with $\mathcal{I}_0|_\Sigma = \mathcal{I}_1|_\Sigma$. $\mathcal{T}_0$ contains a single axiom $A \sqsubseteq C$, where C is a conjunction with the following conjuncts:

- Provide an "interface" to $\mathcal{T}_1$:
$$B$$
- Generate a tree of depth $2n$ whose leaves are labeled with all possible combinations of the concept names $X_1, \dots, X_{2n}$:
$$L_0 \sqcap \prod_{i<2n} \forall r^i.\big(\exists r.(X_{i+1} \sqcap L_{i+1}) \sqcap \exists r.(\neg X_{i+1} \sqcap L_{i+1})\big) \sqcap \prod_{i<2n} \forall r^i. \prod_{j<i} \big((X_j \to \forall r.X_j) \sqcap (\neg X_j \to \forall r.\neg X_j)\big)$$
We use $\forall r^i$ to abbreviate the i-fold nesting of $\forall r$. Notice that each level i of the tree is labeled with a concept name L_i. From now on, we use $Y_1, \dots, Y_n$ as synonyms for $X_{n+1}, \dots, X_{2n}$. Thus, the leaves of the tree can be thought of as the elements of the torus, with horizontal position described in binary by $X_1, \dots, X_n$ and vertical position described in binary by $Y_1, \dots, Y_n$.
- Cardinality of models will play an important role. The following ensures that no nodes in the tree collapse. Thus, every model of $\mathcal{T}_0$ that contains an instance of A has at least $2^{2n+1} - 1$ nodes:
$$\prod_{0 \leq i < j \leq 2n} \forall r^j.\neg L_i$$
- We additionally say that each leaf has an r_x-successor representing its horizontal neighbor, and an r_y-successor representing its vertical neighbor. It is important that these nodes *are* allowed to coincide with tree nodes:
$$\forall r^{2n}.\Big(\exists r_x.L_{2n} \sqcap \exists r_y.L_{2n} \sqcap \mathsf{no\text{-}change\text{-}of}(Y_1, \dots, Y_n)\text{-}\mathsf{along}(r_x) \sqcap \mathsf{increment}(X_1, \dots, X_n)\text{-}\mathsf{along}(r_x) \sqcap \mathsf{no\text{-}change\text{-}of}(X_1, \dots, X_n)\text{-}\mathsf{along}(r_y) \sqcap \mathsf{increment}(Y_1, \dots, Y_n)\text{-}\mathsf{along}(r_y)\Big)$$

where the two predicates handling the counters are defined as follows:

$\mathsf{no\text{-}change\text{-}of}(Z_1,\dots,Z_n)\text{-}\mathsf{along}(s) :=$

$$\bigsqcap_{1\le i\le n}\big((Z_i \to \forall s.Z_i)\ \sqcap\ (\neg Z_i \to \forall s.\neg Z_i)\big)$$

$\mathsf{increment}(Z_1,\dots,Z_n)\text{-}\mathsf{along}(s) :=$

$$\bigsqcap_{1\le i\le n} Z_i\ \sqcup \bigsqcup_{1\le k\le n}\Big[\bigsqcap_{1\le i<k} Z_i\ \sqcap \neg Z_k \to \Big(\forall s.(\bigsqcap_{1\le i<k}\neg Z_i\ \sqcap Z_k)\ \sqcap \bigsqcap_{k<i\le n}\big((Z_i\to\forall s.Z_i)\ \sqcap (\neg Z_i\to\forall s.\neg Z_i)\big)\Big)\Big]$$

- We say that, regarding the roles r_x and r_y, the matching conditions are satisfied:

$$\forall r^{2n}.\Big(\bigsqcup_{(t,t')\in H}(t\sqcap\forall r_x.t')\ \sqcap \bigsqcup_{(t,t')\in V}(t\sqcap\forall r_y.t')\Big)$$

- Each leaf satisfies exactly one tile type:

$$\forall r^{2n}.\bigsqcup_{t\in T}\big(t\sqcap\bigsqcap_{t'\in T\setminus\{t\}}\neg t'\big)$$

- An additional conjunct ensures that the initial condition is met. Easy.

The TBox $\mathcal{T}_1$ extends $\mathcal{T}_0$ with a single axiom $B\sqsubseteq A\to C$, where C is a concept that contains only symbols that are not in Σ and ensures that models of $\mathcal{T}_1$ that contain an instance of B are of size at least 2^{2n+1}:

$$L'_0\sqcap\bigsqcap_{i\le 2n}\forall s^i.\big(\exists s.(X'_{i+1}\sqcap L'_{i+1})\sqcap\exists s.(\neg X'_{i+1}\sqcap L'_{i+1})\big)\sqcap \bigsqcap_{i\le 2n}\forall s^i.\bigsqcap_{j<i}\big(X'_j\to\forall s.X'_j\ \sqcap\neg X'_j\to\forall s.\neg X'_j\big)$$

Observe that $\mathcal{T}_1\setminus\mathcal{T}_0$ contains a Σ-dependency, but no $(\Sigma,\Sigma\cap\mathsf{N_R})$-dependency (actually, it does not contain any roles from Σ).

As for the correctness, distinguish three kinds of models $\mathcal{I}$ of $\mathcal{T}_0$:

1. If $A^{\mathcal{I}}=\emptyset$, then $\mathcal{I}$ can be extended to a model of $\mathcal{T}_1$ by interpreting the non-Σ symbols randomly.
2. $A^{\mathcal{I}}\neq\emptyset$ and $|\Delta^{\mathcal{I}}|\ge 2^{2n+1}$. Then $|\Delta^{\mathcal{I}}\setminus A^{\mathcal{I}}|\ge 2^{2n+1}-2$. We can extend $\mathcal{I}$ to a model of $\mathcal{T}_1$ by making L'_0 true at all elements of $A^{\mathcal{I}}$ and interpreting all remaining non-Σ-symbols in the elements of $\Delta^{\mathcal{I}}\setminus A^{\mathcal{I}}$.
3. $A^{\mathcal{I}}\neq\emptyset$ and $|\Delta^{\mathcal{I}}|\le 2^{2n+1}-1$. By definition of $\mathcal{T}_0$, we have $|\Delta^{\mathcal{I}}|=2^{2n+1}-1$. Then the L_{2n} elements form a torus w.r.t. the roles r_x and r_y. Such a model cannot be extended to a model of $\mathcal{T}_1$ because we have an instance of A, which is also a symbol of B, but we don't have enough room to interpret the non-Σ-symbols such that $B\sqsubseteq A\to C$ is satisfied.

It remains to note that, by definition of $\mathcal{T}_0$, a model of $\mathcal{T}_0$ of the third kind exists iff there is a solution for $\mathcal{D}$ and c.

□

D Proof of Theorem 10

Theorem 10. Let $\mathcal{L}$ be $\mathcal{ALC}$ or $\mathcal{ALCI}$. For acyclic $\mathcal{L}$-terminologies $\mathcal{T}_1\supseteq\mathcal{T}_0$ and a signature $\Sigma\supseteq\mathsf{sig}(\mathcal{T}_0)$ such that $\mathcal{T}_1\setminus\mathcal{T}_0$ contains no syntactic Σ-dependencies, the following are equivalent:

- $\mathcal{T}_0$ is a strong semantic Σ-module of $\mathcal{T}_1$;
- $\mathcal{T}_0$ is a weak semantic Σ-module of $\mathcal{T}_1$;
- for all $P\subseteq\Sigma\cap(\mathsf{Pr}(\mathcal{T}_0)\setminus\mathsf{Pr}(\mathcal{T}_1))$, the following concept is satisfiable in a model of $\mathcal{T}_1\setminus\mathcal{T}_0$ of cardinality 1:

$$C_P=\bigsqcap_{A\in P}A\sqcap\bigsqcap_{A\in\Sigma\cap(\mathsf{Pr}(\mathcal{T}_0)\setminus(\mathsf{Pr}(\mathcal{T}_1)\cup P))}\neg A.$$

It is Π^p_2-complete to decide whether $\mathcal{T}_0$ is a weak/strong semantic module of $\mathcal{T}_1$.

Proof. The implication from Point 1 to Point 2 is stated in Lemma 5.

Point 2 implies Point 3. Assume there exists a set $P\subseteq\Sigma\cap(\mathsf{Pr}(\mathcal{T}_0)\setminus\mathsf{Pr}(\mathcal{T}_1))$ such that C_P is not satisfiable in any one-point interpretation of $\mathcal{T}_1\setminus\mathcal{T}$. All concept names in C_P are primitive in $\mathcal{T}_0$, and so there exists a one-point interpretation satisfying $\mathcal{T}_0$ and C_P. Thus, we have found an interpretation $\mathcal{I}$ satisfying $\mathcal{T}_0$ for which there does not exist an interpretation $\mathcal{I}'$ satisfying $\mathcal{T}_1$ which coincides with $\mathcal{I}$ on Σ.

Point 3 implies Point 1. Assume Point 3 holds. Let $\mathcal{I}$ be an interpretation. For any $d\in\Delta^{\mathcal{I}}$ let

$$P_d=\{A\in\Sigma\cap(\mathsf{Pr}(\mathcal{T}_0)\setminus(\mathsf{Pr}(\mathcal{T}_1))\mid d\in A^{\mathcal{I}}\}.$$

By Point 3, for each d, there exists a one-point interpretation $\mathcal{I}_d$ satisfying $\mathcal{T}_1\setminus\mathcal{T}$ and C_{P_d}. If a concept name A from Σ is not primitive in $\mathcal{T}_1\setminus\mathcal{T}_0$, then it is contained in $\Sigma\cap(\mathsf{Pr}(\mathcal{T}_0)\setminus\mathsf{Pr}(\mathcal{T}_1))$ and $\mathsf{depend}_{\mathcal{T}_1\setminus\mathcal{T}_0}(A)\cap\Sigma=\emptyset$. Hence, we can construct an interpretation $\mathcal{I}'$ satisfying $\mathcal{T}_1\setminus\mathcal{T}_0$ which coincides with $\mathcal{I}$ on Σ such that

- for all $d\in\Delta^{\mathcal{I}}$ and all concept names $B_0\in\bigcup_{B\in\Sigma\cap(\mathsf{Pr}(\mathcal{T}_0)\setminus\mathsf{Pr}(\mathcal{T}_1))}\mathsf{depend}_{\mathcal{T}_1\setminus\mathcal{T}_0}(B)$:

$$d\in B_0^{\mathcal{I}'}\text{ iff }d\in B_0^{\mathcal{I}_d}$$

- for all $(d,d')\in\Delta^{\mathcal{I}}\times\Delta^{\mathcal{I}}$ and $r\notin\Sigma$:

$$(d,d')\in r^{\mathcal{I}'}\text{ iff }d=d'\text{ and }(d,d')\in r^{\mathcal{I}_d}.$$

The Π^p_2 upper bound can be shown as follows: to check whether there exists C_P which not satisfiable in a one-point interpretation satisfying $\mathcal{T}_1\setminus\mathcal{T}_0$ guess a set P and a one-point interpretation for the symbols in C_P, check in linear time whether C_P is satisfied and call an NP-oracle to check that no extension of the interpretation of the symbols in C_P satisfies $\mathcal{T}_1\setminus\mathcal{T}_0$.

For the lower bound, it is sufficient to show that it is Π^p_2-hard to decide, given propositional formulas $\varphi_1,\dots,\varphi_n$, whether (†) for every $P\subseteq\{1,\dots,n\}$ the formula

$$\bigwedge_{i\in P}\varphi_i\wedge\bigwedge_{i\notin P}\neg\varphi_i$$

is satisfiable. But let $p_1, \dots, p_{n-1}$ be propositional variables, φ a propositional formula, and p a fresh propositional variable. Then $p_1, \dots, p_{n-1}, \varphi \wedge p$ satisfies (†) iff the formula

$$\forall p_1 \cdots \forall p_{n-1} \exists q_1 \cdots \exists q_m \varphi,$$

($q_1, \dots, q_m$ the variables from φ which are not p_i's) is valid. □

E Proof of Theorem 12

Theorem 12. Let $\mathcal{L}$ be $\mathcal{EL}$ or $\mathcal{ELI}$. For acyclic $\mathcal{L}$-terminologies $\mathcal{T}_1 \supseteq \mathcal{T}_0$ containing no trivial axioms and signature $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_0)$, the following are equivalent:

- $\mathcal{T}_0$ is a strong semantic Σ-module of $\mathcal{T}_1$;
- $\mathcal{T}_0$ is a weak semantic Σ-module of $\mathcal{T}_1$.

It is decidable in polytime whether $\mathcal{T}_0$ is a weak/strong semantic module of $\mathcal{T}_1$.

Proof. By Theorem 10 and Lemma 11, the first step of the algorithm in Figure 2 is correct and it is sufficient to show that under the condition $\mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap \Sigma = \emptyset$ for all $A \in \Sigma \cap \mathsf{Pr}(\mathcal{T}_1) \setminus \mathsf{Pr}(\mathcal{T}_0)$ the following holds:

Claim. Some concept C_P from Point 3 of Lemma 10 is not satisfiable in a one-point interpretation satisfying $\mathcal{T}_1 \setminus \mathcal{T}_0$ iff there exists $A \in \Sigma \cap \mathsf{Pr}(\mathcal{T}_1) \setminus \mathsf{Pr}(\mathcal{T}_0)$ such that $A \equiv C \in \mathcal{T}_1 \setminus \mathcal{T}_0$ for some C and the set-inclusion of Figure 2 holds.

First, consider such an A for which $A \equiv C \in \mathcal{T}_1 \setminus \mathcal{T}_0$ for some C and the set-inclusion of Figure 2 holds. Then the concept C_P for $P = (\Sigma \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus \mathsf{Pr}(\mathcal{T}_1)) \setminus \{A\}$ is not satisfiable in a one-point interpretation satisfying $\mathcal{T}_1 \setminus \mathcal{T}_0$.

Conversely, suppose there does not exist an $A \in \Sigma \cap \mathsf{Pr}(\mathcal{T}_1) \setminus \mathsf{Pr}(\mathcal{T}_0)$ such that $A \equiv C \in \mathcal{T}_1 \setminus \mathcal{T}_0$ for some C and the set-inclusion of Figure 2 holds. Consider a set $P \subseteq \Sigma \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus \mathsf{Pr}(\mathcal{T}_1))$. Then one can construct a one-point interpretation $\mathcal{I}$ satisfying $\mathcal{T}_1 \setminus \mathcal{T}_0$ and the following constraints:

- for all concept names $B_0 \in P \cup \bigcup_{B \in P} \mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(B)$, $B_0^{\mathcal{I}} = \Delta^{\mathcal{I}}$;
- for all role names $r \in \bigcup_{B \in P} \mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(B)$, $r^{\mathcal{I}} = \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$;
- for all remaining role names: $r^{\mathcal{I}} = \emptyset$;
- for all remaining concept names $B \in \mathsf{PPr}(\mathcal{T}_1 \setminus \mathcal{T}_0)$: $B^{\mathcal{I}} = \emptyset$.

It is readily checked that $\mathcal{I}$ satisfies C_P. □

F Proof of Theorem 13

Before we supply the proof of this result, we observe the following

Lemma 17. *Let $\mathcal{T}_1 \supseteq \mathcal{T}_0$ be acyclic terminologies and $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_0)$. For every $A \in \Sigma \cap \mathsf{Pr}(\mathcal{T}_0)$, $\mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap \Sigma \neq \emptyset$ iff $\mathsf{depend}_{\mathcal{T}_1}(A) \cap \Sigma \neq \emptyset$.*

Proof. The direction from left to right is trivial. Conversely, assume $X \in \mathsf{depend}_{\mathcal{T}_1}(A) \cap \Sigma$ with $X \notin \mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A)$. As A is primitive in $\mathcal{T}_0$ and $\Sigma \supseteq \mathsf{sig}(\mathcal{T}_0)$ we find $Y \in \Sigma$ with $A \prec^*_{\mathcal{T}_1 \setminus \mathcal{T}_0} Y \prec^*_{\mathcal{T}_1} X$. But then $Y \in \mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap \Sigma$. □

Theorem 13. Let $\mathcal{T}_1$ be an acyclic $\mathcal{ALCI}$-terminology and Σ a signature. The output $\mathcal{T}_0$ of the algorithm in Figure 3 is the unique smallest strong (equivalently, weak) semantic $\Sigma \cup \mathsf{sig}(\mathcal{T}_0)$-module of $\mathcal{T}_1$ such that $\mathcal{T}_1 \setminus \mathcal{T}_0$ contains no syntactic Σ-dependencies.

Proof. It follows from Theorem 10 that the output $\mathcal{T}$ of the algorithm in Figure 3 is a strong (equivalently, weak) semantic module of $\mathcal{T}_1$ w.r.t. $\Sigma \cup \mathsf{sig}(\mathcal{T})$ such that $\mathcal{T}_1 \setminus \mathcal{T}$ contains no syntactic $\Sigma \cup \mathsf{sig}(\mathcal{T})$-dependencies. Thus, it remains to show minimality. It is sufficient to check that any strong semantic module $\mathcal{T}'$ of $\mathcal{T}_1$ w.r.t. $\Sigma \cup \mathsf{sig}(\mathcal{T})$ such that $\mathcal{T}_1 \setminus \mathcal{T}'$ does not contain syntactic $\Sigma \cup \mathsf{sig}(\mathcal{T}')$-dependencies is contained in $\mathcal{T}$.

Let $\mathcal{R}$ be a run of the algorithm in Figure 3. Suppose $\mathcal{T}_0 \subseteq \mathcal{T}'$ has been computed in that run. It is sufficient to show that an application of Rule 1 or Rule 2 yields an output $\mathcal{T}_0'$ which is included in $\mathcal{T}'$.

Suppose Rule 1 is applied to $\mathcal{T}_0$ and A. Then $A \in \Sigma \cup \mathsf{sig}(\mathcal{T}')$ and an $\alpha \in \mathcal{T}_1$ has A on the left hand side. Assume $\alpha \notin \mathcal{T}'$. Then $A \in (\Sigma \cup \mathsf{sig}(\mathcal{T}')) \cap (\mathsf{Pr}(\mathcal{T}') \setminus \mathsf{Pr}(\mathcal{T}_1))$. By Lemma 17, $\mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap (\Sigma \cup \mathsf{sig}(\mathcal{T}_0)) \neq \emptyset$ implies $\mathsf{depend}_{\mathcal{T}_1}(A) \cap (\Sigma \cup \mathsf{sig}(\mathcal{T}_0)) \neq \emptyset$ and this implies $\mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}'}(A) \cap (\Sigma \cup \mathsf{sig}(\mathcal{T}')) \neq \emptyset$ which contradicts the assumption that $\mathcal{T}_1 \setminus \mathcal{T}'$ does not contain syntactic $\Sigma \cup \mathsf{sig}(\mathcal{T}')$-dependencies.

Suppose Rule 2 is applied to $\mathcal{T}_0$ with sets P and Q. Q is a minimal subset of $(\Sigma \cup \mathsf{sig}(\mathcal{T}_0)) \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus \mathsf{Pr}(\mathcal{T}_1))$ such that there exists $P \subseteq Q$ such that

$$C_{P,Q} = \prod_{A \in P} A \sqcap \prod_{A \in Q \setminus P} \neg A$$

is not satisfiable in a one point-interpretation satisfying $\mathcal{T}_1 \setminus \mathcal{T}$. Let

$$\mathcal{B} = \{\alpha \in \mathcal{T}_1 \mid \alpha \text{ has an } A \in Q \text{ on the left hand side } \}.$$

Assume that $\mathcal{B} \not\subseteq \mathcal{T}'$. By Theorem 10, $\mathcal{B}' = \mathcal{B} \setminus \mathcal{T}'$ is nonempty. Let Q' be the set of concept names with definitions in $\mathcal{B}'$. Let $C'_{P,Q}$ result from $C_{P,Q}$ by taking only the conjuncts from Q'. By minimality of Q, there exists a one-point interpretation $\mathcal{I}$ satisfying $\mathcal{T}_1$ and $C'_{P,Q}$. Then $\mathcal{I}$ satisfies $\mathcal{T}'$. As the symbols in $Q \setminus Q'$ are primitive in $\mathcal{T}'$, we can change the interpretation of symbols in $Q \setminus Q'$ in such a way that we obtain a one-point interpretation $\mathcal{I}_0$ satisfying $\mathcal{T}'$ and $C_{P,Q}$. But then there does not exist an interpretation $\mathcal{I}_0'$ which coincides with $\mathcal{I}_0$ on $\Sigma \cup \mathsf{sig}(\mathcal{T}')$ and satisfies $\mathcal{T}_1$. Thus, $\mathcal{T}'$ is not a weak semantic module of $\mathcal{T}_1$ w.r.t. $\Sigma \cup \mathsf{sig}(\mathcal{T}')$ such that $\mathcal{T}_1 \setminus \mathcal{T}'$ contains no syntactic $\Sigma \cup \mathsf{sig}(\mathcal{T}')$-dependencies. □

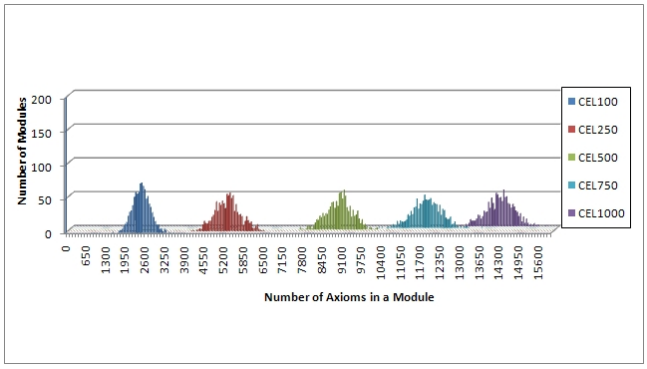
G Proof of Theorem 14

Theorem 14. Let $\mathcal{T}_1$ be an acyclic $\mathcal{ELI}$-terminology containing no trivial axioms and Σ a signature. The output $\mathcal{T}_0$ of the algorithm in Figure 4 is the unique smallest strong (equivalently, weak) semantic $\Sigma \cup \mathsf{sig}(\mathcal{T})$-module of $\mathcal{T}_1$.

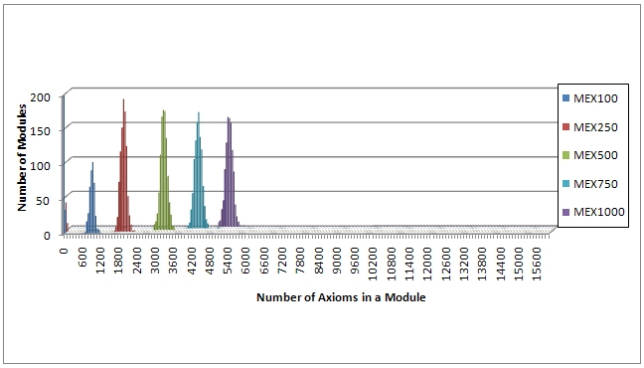
Proof. A straighforward combination of the proofs of Theorem 12 and Theorem 13. □

H Additional experimental results

In this section, we follow up Section 7 and provide some additional information about the distribution of the module sizes of the definition-closed modules extracted using CEL and the semantic modules extracted by MEX. Figure 6 shows the frequency distribution of the definition-closed and the semantic 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 sizes of 1 000 extracted SNOMED CT modules for randomly selected signatures of a certain size. For instance, the histogram labelled with CEL100 in Figure 6(a) shows the distribution of the size of 1 000 definition-closed modules for the signature size 100 extracted from SM-05. For the sake of comparison, the axes in both figures have the same scaling resulting in the histogram MEX100 being capped at 200 for empty semantic modules. The missing value of MEX100 for empty modules is 547.



(a) Definition-closed modules



(b) Semantic modules

Figure 6. Frequency distribution of the size of extracted modules

To facilitate the comparison of the module sizes, consider Table 1. It presents the average module size together with the standard deviation of definition-closed and semantic modules for random input signatures of various sizes. Recall that the standard deviation indicates how much the module sizes vary from the average. Notice that, for small signature sizes, the standard deviation of semantic module sizes is relatively high. The reason is that MEX extracts many small or even empty semantic modules for small signature sizes. For instance, 547 of 1 000 extracted modules where empty for signature size 100. Intuitively, the reason for an empty module is that SNOMED CT does not imply any subsumptions between concepts formulated in the chosen signature. When only considering the semantic modules for signature size 100 that contain more than 10 axioms, the average module size becomes 889.15 and the standard deviation decreases to 125.63; see the last column of Table 1.

I Roles

SNOMED CT is formulated in $\mathcal{EL}$ extended with additional role box statements of the following form: *role hierarchies* $r \sqsubseteq s$ and *right-identities* $r \circ s \sqsubseteq r$, where s and r are role names. To account for this, we have restricted the algorithm shown in Figure 4 to $\mathcal{EL}$, and then extended it to deal with role inclusions of the two mentioned forms, and additionally with *left identities* $s \circ r \sqsubseteq r$. An interpretion *satisfies* $r \sqsubseteq s$ if $r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$, and similarly for left and right identities. Other semantic notions are defined in the obvious way. Observe that left and right identities declare a role as transitive when $r = s$. In the following, we allow *role inclusions* of all three kinds to occur in $\mathcal{EL}$-terminologies.

An *$\mathcal{EL}$-constraint box* (CBox) $\mathcal{C}$ is the union of an acyclic $\mathcal{EL}$-terminology $\mathcal{T}$ and a set $\mathcal{R}$ of role inclusions. The notions relevant for modularity introduced above (signature of a role inclusion, weak/strong semantic module, etc.) are extended to role inclusions and CBoxes in the obvious way. Observe that for CBoxes $\mathcal{C}$ and signatures Σ, there does not always exist a smallest weak semantic module $\mathcal{C}_0$ of $\mathcal{C}$ w.r.t. $\Sigma \cup \mathsf{sig}(\mathcal{C}_0)$: let $\mathcal{C} = \{r \sqsubseteq s_1, s_1 \sqsubseteq r', r \sqsubseteq s_2, s_2 \sqsubseteq r'\}$ and $\Sigma = \{r, r'\}$. Clearly, $\mathcal{C}$ is not semantically safe w.r.t. Σ. For $\mathcal{C}_1 = \{r \sqsubseteq s_1, s_1 \sqsubseteq r'\}$ and $\mathcal{C}_2 = \{r \sqsubseteq s_2, s_2 \sqsubseteq r'\}$, both $\mathcal{C}_i$ are minimal weak semantic modules of $\mathcal{C}$ w.r.t. $\Sigma \cup \mathsf{sig}(\mathcal{C}_i)$. This example also shows that weak and strong semantic modules are distinct notions for CBoxes: $\mathcal{C}_i$ is not a strong semantic module of $\mathcal{C}$ w.r.t. $\Sigma \cup \{s_i\}$ because $\mathcal{C} \setminus \mathcal{C}_i \models r \sqsubseteq r'$. It would be of great interest to explore the possiblity of sound and complete algorithms for checking weak and strong semantic modules for CBoxes and design extraction algorithms.[3] In this paper, however, we confine ourselves to an investigation of modularity under additional syntactic constraints.

Definition 18. Let $\mathcal{C}$ be an $\mathcal{EL}$-CBox and Σ a signature. $\mathcal{C}$ contains a *syntactic Σ-role-dependency* if

- there exists $r \in \Sigma$ with $r \sqsubseteq r' \in \mathcal{C}$; or
- there exists $s \in \Sigma$ such that $r \circ s \sqsubseteq r \in \mathcal{C}$ or $s \circ r \sqsubseteq r \in \mathcal{C}$.

The following lemma states that role inclusions alone can be dealt with in a straighfoward manner once one considers the case without syntactic Σ-role-dependencies.

Lemma 19. *Let $\mathcal{C}$ be an $\mathcal{EL}$-CBox consisting of role inclusions only and Σ a signature such that $\mathcal{C}$ contains no syntactic Σ-role-dependencies. Then $\mathcal{C}$ is semantically safe for Σ.*

More precisely, every interpretation $\mathcal{I}$ in which all $r \in \mathsf{sig}(\mathcal{C}) \setminus \Sigma$ are interpreted as subsets of $\mathsf{id}^{\mathcal{I}} = \{(d, d) \mid d \in \Delta^{\mathcal{I}}\}$ with

[3] Observe that role boxes consisting of transitivity axioms $r \circ r \sqsubseteq r$ only do not cause any problems. They can be handled with the algorithms developed so far.

Signature size	Size of definition-closed modules		Size of semantic modules		Size of semantic modules of Size > 10	
	Average	Standard deviation	Average	Standard deviation	Average	Standard deviation
100	2 462.17	293.49	370.10	447.08	889.15	125.63
250	5 253.21	419.08	1 774.53	434.66	1 875.80	98.04
500	8 872.74	441.92	3 138.25	110.84	3 138.25	110.84
750	11 691.71	478.83	4 210.94	121.40	4 210.94	121.40
1 000	14 053.48	462.09	5 167.07	122.76	5 167.07	122.76

Table 1. Average and std. deviation definition-closed and semantic modules

- *if* $r' \sqsubseteq r \in \mathcal{C}$ *and* $r \in \Sigma$, $r' \notin \Sigma$, *then* $r'^{\mathcal{I}} = \emptyset$;
- *if* $r' \sqsubseteq r \in \mathcal{C}$ *and* $r', r \notin \Sigma$, *then* $r'^{\mathcal{I}} \subseteq r^{\mathcal{I}}$

satisfies $\mathcal{C}$.

Proof. Clearly, all implications between role names of $\mathcal{C}$ are satisfied. The left and right-identities $r \circ s \sqsubseteq r$ and $s \circ r \sqsubseteq r$ of $\mathcal{C}$ are satisfied because all role names in $\mathsf{sig}(\mathcal{C}) \setminus \Sigma$ are interpreted as subsets of $\mathsf{id}^{\mathcal{I}}$. □

Using this result one can readily extend to CBoxes the previous proofs on checking and extracting weak and strong modules for acyclic terminologies without role boxes. We require the following sets. For a signature Σ and CBox $\mathcal{C} = \mathcal{T} \cup \mathcal{R}$, set

$$\mathsf{depend}_{\mathcal{C}}(A) = \mathsf{depend}_{\mathcal{T}}(A) \cup \{r \mid \exists r' \in \mathsf{depend}_{\mathcal{T}}(A)\ \mathcal{C} \models r' \sqsubseteq r\}$$

and

$$\Sigma^{\downarrow \mathcal{C}} = \{r \in \mathsf{N_R} \mid \exists r' \in \Sigma\ \mathcal{C} \models r \sqsubseteq r'\}.$$

Theorem 20. *Let* $\mathcal{C}_1 \supseteq \mathcal{C}_0$ *be* $\mathcal{EL}$*-CBoxes not containing trivial axioms and* $\Sigma \supseteq \mathsf{sig}(\mathcal{C}_0)$ *such that* $\mathcal{C}_1 \setminus \mathcal{C}_0$ *contains not syntactic* Σ*-role-dependencies. Then the following conditions are equivalent:*

- $\mathcal{C}_0$ *is a strong semantic module of* $\mathcal{C}_1$ *w.r.t.* Σ;
- $\mathcal{C}_0$ *is a weak semantic module of* $\mathcal{C}_1$ *w.r.t.* Σ;
- *The algorithm in Figure 7 outputs 'module'.*

The algorithm in Figure 7 runs in polynomial time.

Proof. The implication from Point 1 to Point 2 is a straightforward extension of Lemma 5.

Point 2 implies Point 3. Assume first that the algorithm of Figure 7 outputs 'not module' in Step 1. Take $A \in \Sigma \cap \mathsf{Pr}(\mathcal{T}_0) \setminus \mathsf{Pr}(\mathcal{T}_1)$ and $X \in \mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap (\Sigma \cup \Sigma^{\downarrow \mathcal{C}_1})$.

As A is primitive in $\mathcal{T}_0$, there exists an interpretation $\mathcal{I}$ satisfying $\mathcal{C}_0$ such that $A^{\mathcal{I}} = \Delta^{\mathcal{I}}$, $r^{\mathcal{I}} = \emptyset$, for all roles r, and $B^{\mathcal{I}} = \emptyset$ for all $B \in \Sigma$ with $B \neq A$. Then there does not exist an interpretation $\mathcal{I}'$ which coincides with $\mathcal{I}$ on Σ satisfying $\mathcal{C}_1$.

The remaining steps of the proof are a straighforward combination of Lemma 19 and the arguments used for $\mathcal{ELI}$. □

Input: $\mathcal{EL}$-CBoxes $\mathcal{C}_1 = \mathcal{T}_1 \cup \mathcal{R}_1$ and $\mathcal{C}_0 = \mathcal{T}_0 \cup \mathcal{R}_0$ with $\mathcal{C}_1 \supseteq \mathcal{C}_0$ and $\Sigma \supseteq \mathsf{sig}(\mathcal{C}_0)$ such that $\mathcal{C}_1 \setminus \mathcal{C}_0$ contains no syntactic Σ-role-dependencies.

- If $A \in \Sigma \cap \mathsf{Pr}(\mathcal{T}_0) \setminus \mathsf{Pr}(\mathcal{T}_1)$ and $\mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap (\Sigma \cup \Sigma^{\downarrow \mathcal{C}_1}) \neq \emptyset$, output 'not module' and stop. Otherwise:
- If $A \in \Sigma \cap \mathsf{Pr}(\mathcal{T}_0) \setminus \mathsf{Pr}(\mathcal{T}_1)$, $A \equiv C \in \mathcal{T}_1$, and

$$\bigcup_{\substack{B \in \Sigma \cap (\mathsf{Pr}(\mathcal{T}_0) \setminus \\ (\mathsf{Pr}(\mathcal{T}_1) \cup \{A\}))}} \mathsf{depend}_{\mathcal{C}_1 \setminus \mathcal{C}_0}(B) \supseteq \mathsf{depend}^{\equiv}_{\mathcal{T}_1 \setminus \mathcal{T}_0}(A) \cap \mathsf{PPr}(\mathcal{T}_1 \setminus \mathcal{T}_0),$$

output 'not module' and stop. Otherwise output 'module'.

Figure 7. Checking modules for $\mathcal{EL}$ CBoxes

Theorem 21. *Let* $\mathcal{C}_1$ *be an* $\mathcal{EL}$*-CBox not containing trivial axioms and* Σ *a signature. The output* $\mathcal{C}$ *of the algorithm in Figure 8 is the smallest strong/weak semantic module of* $\mathcal{C}_1$ *w.r.t.* $\Sigma \cup \mathsf{sig}(\mathcal{C})$ *such that* $\mathcal{C}_1 \setminus \mathcal{C}$ *contains no syntactic* Σ*-role-dependencies.*

Input: $\mathcal{EL}$-CBox $\mathcal{C}_1 = \mathcal{T}_1 \cup \mathcal{R}_1$ and signature Σ.

Output: smallest strong (equivalently, weak) semantic module $\mathcal{C}$ of $\mathcal{C}_1$ w.r.t. $\Sigma \cup \mathsf{sig}(\mathcal{C})$ such that $\mathcal{C}_1 \setminus \mathcal{C}$ contains no $\Sigma \cup \mathsf{sig}(\mathcal{C})$-role-dependencies.

- Initialize: $\mathcal{C} = \mathcal{T} \cup \mathcal{R} = \emptyset$ (in the rules below, an update of $\mathcal{C}$ implies the obvious update for $\mathcal{T}$ and $\mathcal{R}$),
- Apply the rules 1 to 4 below exhaustively. Output $\mathcal{C}$ if no rule is applicable.

1. If $r \in \Sigma \cup \mathsf{sig}(\mathcal{C})$ and $r \sqsubseteq r' \in \mathcal{C}_1 \setminus \mathcal{C}$, then set $\mathcal{C} := \mathcal{C} \cup \{r \sqsubseteq r'\}$.
2. If $s \in \Sigma \cup \mathsf{sig}(\mathcal{C})$ such that $r \circ s \in \mathcal{C}_1 \setminus \mathcal{C}$ or $s \circ r \sqsubseteq r \in \mathcal{C}_1 \setminus \mathcal{C}$, then set $\mathcal{C} := \mathcal{C} \cup \{r \circ s \sqsubseteq r\}$ ($\mathcal{C} := \mathcal{C} \cup \{s \circ r \sqsubseteq r\}$, respectively).
3. If $A \in \Sigma \cup \mathsf{sig}(\mathcal{T})$, $\alpha \in \mathcal{T}_1 \setminus \mathcal{T}$ has A on the left hand side, and $\mathsf{depend}_{\mathcal{T}_1 \setminus \mathcal{T}}(A) \cap (\Sigma \cup \mathsf{sig}(\mathcal{T}) \cup \Sigma^{\downarrow \mathcal{C}_1}) \neq \emptyset$, set $\mathcal{C} := \mathcal{C} \cup \{\alpha\}$.
4. If $A \in \Sigma \cup \mathsf{sig}(\mathcal{T})$, $A \equiv C \in \mathcal{T}_1 \setminus \mathcal{T}$, and

$$\bigcup_{\substack{B \in (\Sigma \cup \mathsf{sig}(\mathcal{T})) \cap \\ (\mathsf{Pr}(\mathcal{T}_0) \setminus (\mathsf{Pr}(\mathcal{T}_1) \cup \{A\}))}} \mathsf{depend}_{\mathcal{C}_1 \setminus \mathcal{C}}(B) \supseteq \mathsf{depend}^{\equiv}_{\mathcal{T}_1 \setminus \mathcal{T}}(A) \cap \mathsf{PPr}(\mathcal{T}_1 \setminus \mathcal{T}),$$

   set $\mathcal{C} := \mathcal{C} \cup \{A \equiv C\}$.

Figure 8. Computing modules for $\mathcal{EL}$-CBoxes