MORe: Modular Combination of OWL Reasoners for Ontology Classification

Ana Armas Romero, Bernardo Cuenca Grau, Ian Horrocks

Department of Computer Science. University of Oxford

Abstract. Classification is a fundamental reasoning task in ontology design, and there is currently a wide range of reasoners highly optimised for classification of OWL 2 ontologies. There are also several reasoners that are complete for restricted fragments of OWL 2 , such as the OWL 2 EL profile. These reasoners are much more efficient than fully-fledged OWL 2 reasoners, but they are not complete for ontologies containing (even if just a few) axioms outside the relevant fragment. In this paper, we propose a novel classification technique that combines an OWL 2 reasoner and an efficient reasoner for a given fragment in such a way that the bulk of the workload is assigned to the latter. Reasoners are combined in a black-box modular manner, and the specifics of their implementation (and even of their reasoning technique) are irrelevant to our approach.

1 Introduction

Classification—the problem of identifying the subsumption relationships between all pairs of classes in the input ontology—is a fundamental reasoning task in ontology design. For expressive ontology languages, however, the decision problems associated with classification have a very high worst-case complexity; in particular, subsumption with respect to an OWL 2 ontology is known to be a 2NEXPTIME-complete problem [15, 6].

In spite of these discouraging complexity results, highly optimised reasoners such as Pellet [21], FaCT++ [22], RacerPro [10] and HermiT [8] are able to classify many ontologies used in applications. The optimisations employed by these reasoners aim not only to improve performance on individual subsumption tests, but also to reduce the number of tests performed when classifying a given ontology—most OWL 2 reasoners use variants of the well-known Enhanced Traversal Algorithm to incrementally construct a compact representation of the subsumption relation, along with the *told subsumptions* optimisation, which provides an inexpensive way of identifying "obvious" subsumption relationships that hold in the input ontology [2, 11]. Identifying obvious non-subsumptions is also important, as most possible subsumption relationships do not hold, and has been addressed by optimisations such as *completely defined concepts*, which identifies a fragment of the ontology for which told subsumptions provide complete information, *model-merging*, and other related techniques that exploit the computations performed during individual class satisfiability tests [23, 11, 9, 8].

However, notwithstanding extensive and ongoing research into optimisation techniques, the classification of large ontologies—such as the SNOMED medical ontology—can still require a very large number of subsumption tests, and even if no individual test is very costly, the total amount of time required for classification can still be large. This (and other performance issues) has motivated a growing interest in so-called *lightweight description logics*: weaker logics that enjoy favourable computational properties. OWL 2 includes several *profiles* (language fragments) based on such lightweight DLs, including OWL 2 EL, a profile based on the $\mathcal{EL}^{++}$ DL for which most standard reasoning tasks can be performed in polynomial time [19]. Very efficient profile-specific reasoners have been developed for OWL 2 EL, including CEL [3] and ELK [16], which can classify ontologies as large as SNOMED in just a few seconds.

Unfortunately, a reasoner for profile $\mathcal{L}$ (an $\mathcal{L}$-reasoner) is only able to (completely) classify ontologies in the $\mathcal{L}$ profile ($\mathcal{L}$-ontologies), and restricting the ontology to a given profile may be undesirable or even infeasible in practice, with many existing ontologies falling outside all of the tractable profiles of OWL 2. In many cases, however, such ontologies contain only a relatively small number of axioms that are outside one of the tractable fragments. For example, of the 219,224 axioms in the latest version of the National Cancer Institute Ontology (NCI), only 65 are outside the OWL 2 EL profile. Using a suitable $\mathcal{L}$-reasoner to efficiently classify most classes (i.e., to find all their subsumers) in the signature (or vocabulary) of a given ontology might, therefore, lead to significant improvements in performance. Unfortunately, using an $\mathcal{L}$-reasoner in this way is far from simple, as even a single axiom that is outside $\mathcal{L}$ could affect every class in the ontology.

In this paper, we propose a novel technique where an efficient $\mathcal{L}$-reasoner and a fully fledged OWL 2 reasoner are combined in a modular way to classify an OWL 2 ontology. More precisely, given an OWL 2 ontology $\mathcal{O}$, and a fragment $\mathcal{L}$ of OWL 2 , our classification algorithm proceeds as follows:

1. It computes a signature $\Sigma^{\mathcal{L}} \subseteq \mathsf{Sig}(\mathcal{O})$ and an $\mathcal{L}$-ontology $\mathcal{M}^{\mathcal{L}} \subseteq \mathcal{O}$ such that the classes in $\Sigma^{\mathcal{L}}$ can be completely classified using only the axioms in $\mathcal{M}^{\mathcal{L}}$.
2. It computes an ontology $\mathcal{M}^{\overline{\mathcal{L}}} \subseteq \mathcal{O}$ such that the classes in $\mathcal{O} \setminus \Sigma^{\mathcal{L}}$ can be fully classified using only the axioms in $\mathcal{M}^{\overline{\mathcal{L}}}$.
3. It classifies $\mathcal{M}^{\overline{\mathcal{L}}}$ using a fully-fledged OWL 2 reasoner and then completes the classification of $\mathcal{O}$ by classifying $\mathcal{M}^{\mathcal{L}}$ using an $\mathcal{L}$-reasoner.

Step 1 involves two important technical challenges. First, $\Sigma^{\mathcal{L}}$ should be as large as possible; in particular, for ontologies with only a few non $\mathcal{L}$-axioms, it is reasonable to expect $\Sigma^{\mathcal{L}}$ to contain most of the ontology's signature. Second, $\mathcal{M}^{\mathcal{L}}$ must be *complete* for $\Sigma^{\mathcal{L}}$; i.e., if a subsumption relationship between two classes in $\Sigma^{\mathcal{L}}$ is entailed by $\mathcal{O}$, then it must also be entailed by $\mathcal{M}^{\mathcal{L}}$. Although techniques such as the completely defined concepts optimisation can be used to identify a fragment of the ontology that is complete for a certain signature, these techniques are very restricted—they are not applicable to all OWL 2 ontologies, and even when they are applicable they use a fixed fragment of OWL 2 that is much smaller than the OWL 2 EL profile. In contrast, we exploit module extraction [5, 4] to develop a technique that provides the following compelling features:

- It is general and flexible, as it is neither tied to any particular fragment or profile $\mathcal{L}$ of OWL 2, nor to any particular reasoner or reasoning technique.
- It is easy to implement, as reasoners are combined in a black-box manner, with no modification of their internals being required.
- It exhibits "pay-as-you-go" behaviour when an $\mathcal{L}$-ontology is extended with axioms outside $\mathcal{L}$: on the one hand, the use of an $\mathcal{L}$-reasoner is not precluded by the extension; on the other hand, performance degrades gracefully with the number of additional non $\mathcal{L}$-axioms.

We believe that our results are interesting from both a theoretical and a practical point of view. From a theoretical point of view, we show that given an OWL 2 ontology $\mathcal{O}$ that is not captured by any known tractable fragment of OWL 2, it is often possible to identify a large subset Σ of its signature such that all subsumers of classes in Σ w.r.t. $\mathcal{O}$ can be computed using a polynomial time classification algorithm. From a practical point of view, our experiments with a prototype implementation MORe,[1] which integrates the OWL 2 reasoner HermiT and the OWL 2 EL reasoner ELK, illustrate the potential of this approach for optimising classification and providing a modular reasoning system with robust pay-as-you-go performance.

2 Preliminaries

We assume basic familiarity with the W3C standard OWL 2 [6] and its EL profile [19]. When talking about *ontologies* and *axioms* we implicitly refer to OWL 2 ontologies and axioms, and when talking about OWL 2 we implicitly refer to OWL 2 under the *direct semantics* [20].

For compactness reasons, we adopt description logic notation rather than OWL syntax in examples and definitions; hence we also assume basic familiarity with the syntax and semantics of the DLs $\mathcal{SROIQ}$ [12] and $\mathcal{EL}^{++}$ [1], which provide the logical underpinning for OWL 2 and OWL 2 EL, respectively.

We consider the standard notions of signature (or vocabulary), interpretations and models, entailment, satisfiability, and class subsumption. We denote with $\mathsf{Sig}(\mathcal{O})$ (respectively, $\mathsf{Sig}(\alpha)$) the signature of an ontology $\mathcal{O}$ (respectively, of an axiom α), and use the greek letters Σ and Γ to denote signatures. We denote with $\mathcal{L}$ a generic fragment of OWL 2—either one of its profiles or any other possible fragment for which we may have an efficient reasoner. Finally, given an OWL 2 ontology $\mathcal{O}$, we denote with $\mathcal{O}_\mathcal{L}$ the set of $\mathcal{L}$-axioms in $\mathcal{O}$.

2.1 Module Extraction

Intuitively, a module $\mathcal{M}$ for an ontology $\mathcal{O}$ w.r.t. a signature Σ is an ontology $\mathcal{M} \subseteq \mathcal{O}$ such that $\mathcal{M}$ and $\mathcal{O}$ entail the same axioms over Σ [5].

[1] Modular OWL Reasoner.

$$\underline{\mathsf{BursitisOrCellulitisOfKnee} \equiv \mathsf{BursitisOfKnee} \sqcup \mathsf{CellulitisOfKnee}} \tag{1}$$

$$\mathsf{BursitisOfKnee} \equiv \mathsf{Bursitis} \sqcap \exists \mathsf{hasLocation}.\mathsf{Knee} \tag{2}$$

$$\mathsf{BursitisOfJoint} \equiv \mathsf{Bursitis} \sqcap \exists \mathsf{hasLocation}.\mathsf{Joint} \tag{3}$$

$$\mathsf{Bursitis} \sqsubseteq \mathsf{Swelling} \tag{4}$$

$$\mathsf{Cellulitis} \sqsubseteq \mathsf{Swelling} \sqcap \exists \mathsf{hasOrigin}.\mathsf{Infection} \tag{5}$$

$$\mathsf{InfectiousDisease} \equiv \exists \mathsf{hasOrigin}.\mathsf{Infection} \tag{6}$$

$$\exists \mathsf{hasOrigin}.\top \sqsubseteq \mathsf{Disease} \tag{7}$$

$$\mathsf{Knee} \sqsubseteq \mathsf{Joint} \tag{8}$$

Fig. 1. The example ontology $\mathcal{O}^{\mathsf{ex}}$. Its only non OWL 2 EL axiom is underlined

This intuition is typically formalised using the notions of deductive and model-based *conservative extensions* [17, 5]. In this paper, we define modules in terms of the model-based notion of conservative extension.

Definition 1 (Model Conservative Extension). *Let $\mathcal{O}$ be an ontology and let $\Sigma \subseteq \mathsf{Sig}(\mathcal{O})$. We say that $\mathcal{O}$ is a* model conservative extension *of $\mathcal{M} \subseteq \mathcal{O}$ w.r.t. Σ if, for every model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of $\mathcal{M}$, there exists a model $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ of $\mathcal{O}$ such that $\Delta^{\mathcal{I}} = \Delta^{\mathcal{J}}$ and $X^{\mathcal{I}} = X^{\mathcal{J}}$ for every symbol $X \in \Sigma$.*

That is, $\mathcal{O}$ is a model conservative extension of $\mathcal{M}$ for Σ if every model of $\mathcal{M}$ can be extended to a model of $\mathcal{O}$ without changing either the interpretation domain, or the interpretation of the symbols in Σ.

Definition 2 (Module). *Let $\mathcal{O}$ be an ontology and let Σ be a signature. We say that $\mathcal{M} \subseteq \mathcal{O}$ is a module for $\mathcal{O}$ w.r.t. Σ if $\mathcal{O}$ is a model conservative extension of $\mathcal{M}$ w.r.t. Σ.*

In particular, if $\mathcal{M}$ is a module for $\mathcal{O}$ w.r.t. Σ, then the following condition holds: for each axiom α with $\mathsf{Sig}(\alpha) \subseteq \Sigma$, we have $\mathcal{M} \models \alpha$ iff $\mathcal{O} \models \alpha$.

Example 1. Consider the ontology $\mathcal{O}^{\mathsf{ex}}$, given in Figure 1, which we will use as a running example. Consider also the fragment $\mathcal{O}^{\mathsf{ex}}_1 = \{(1), (2), (3), (4), (8)\}$ of $\mathcal{O}^{\mathsf{ex}}$. Let $\mathcal{I}$ be an arbitrary model of $\mathcal{O}^{\mathsf{ex}}_1$. We can obtain a model of $\mathcal{O}^{\mathsf{ex}}$ by interpreting all symbols in $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_1)$ in the same way as $\mathcal{I}$, and all symbols outside $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_1)$ as the empty set. Thus, $\mathcal{O}^{\mathsf{ex}}$ is a model conservative extension of $\mathcal{O}^{\mathsf{ex}}_1$, and $\mathcal{O}^{\mathsf{ex}}_1$ is a module for $\mathcal{O}^{\mathsf{ex}}$ w.r.t. $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_1)$. As a result, $\mathcal{O}^{\mathsf{ex}}$ and $\mathcal{O}^{\mathsf{ex}}_1$ entail exactly the same axioms constructed using only symbols from $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_1)$. $\diamondsuit$

The problem of checking whether $\mathcal{M}$ is a module for $\mathcal{O}$ w.r.t. Σ is, however, already undecidable for fairly lightweight fragments of OWL 2, such as the OWL 2 EL profile [18]; therefore, approximations are needed in practice. The following sufficient condition for model conservativity is known to work well [5, 7].

Definition 3 ($\emptyset$-locality). *Let Σ be a signature. An interpretation $\mathcal{I}$ is $\emptyset$-local for Σ if for every class $A \notin \Sigma$ and every property $R \notin \Sigma$, we have $A^{\mathcal{I}} = R^{\mathcal{I}} = \emptyset$. An axiom α is $\emptyset$-local for Σ if $\mathcal{I} \models \alpha$ for each $\mathcal{I}$ that is $\emptyset$-local for Σ. An ontology $\mathcal{O}$ is $\emptyset$-local for Σ if every axiom in $\mathcal{O}$ is $\emptyset$-local for Σ.*

Example 2. It is easy to check that the ontology $\mathcal{O}^{\mathsf{ex}} \setminus \mathcal{O}^{\mathsf{ex}}_1$, consisting of axioms (5), (6) and (7), is $\emptyset$-local w.r.t. $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_1)$. For example, to see that axiom (5) is indeed $\emptyset$-local w.r.t. $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_1)$, consider any $\mathcal{I}$ that interprets all symbols in (5) other than those in $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_1)$ as the empty set. Thus, we have $\mathsf{Cellulitis}^{\mathcal{I}} = \mathsf{hasOrigin}^{\mathcal{I}} = \mathsf{Infection}^{\mathcal{I}} = \emptyset$. Clearly, both left and right hand sides of axiom (5) are interpreted as the empty set by $\mathcal{I}$ (see below) and hence $\mathcal{I}$ satisfies (5).

$$\overbrace{\mathsf{Cellulitis}}^{\emptyset} \sqsubseteq \underbrace{\mathsf{Swelling} \sqcap \exists\, \overbrace{\mathsf{hasOrigin}}^{\emptyset} . \overbrace{\mathsf{Infection}}^{\emptyset}}_{\emptyset} \qquad \diamondsuit$$

Checking $\emptyset$-locality for OWL 2 axioms is, however, a PSPACE-complete problem [5]. Since our goal is to optimise classification, checking $\emptyset$-locality might still be too costly. Instead, we will use $\bot$-locality—a well-known sufficient syntactic condition for $\emptyset$-locality which has been successfully used for both ontology reuse and reasoning problems [5, 14, 7, 4].

The grammar defining $\bot$-locality for $\mathcal{SROIQ}$ can be found in the literature [4, 5]. It suffices to note that $\bot$-locality can be checked in polynomial time and that it implies $\emptyset$-locality. Furthermore, the following property holds [4, 5]:

Proposition 1. *If an axiom α is $\bot$-local w.r.t. a signature Σ, then α is $\bot$-local w.r.t. Σ' for any $\Sigma' \subseteq \Sigma$.*

We use $\bot$-locality to define the notion of $\bot$-module. The fact that $\bot$-locality implies $\emptyset$-locality ensures that, if $\mathcal{M}$ is a $\bot$-module for $\mathcal{O}$ w.r.t. Σ (as defined next), then it is a module for $\mathcal{O}$ w.r.t. Σ.

Definition 4 ($\bot$-module). *Let $\mathcal{O}$ be an ontology and let Σ be a signature. We say that $\mathcal{M} \subseteq \mathcal{O}$ is a $\bot$-module for $\mathcal{O}$ w.r.t. Σ if $\mathcal{O} \setminus \mathcal{M}$ is $\bot$-local for $\Sigma \cup \mathsf{Sig}(\mathcal{M})$.*

Example 3. It was pointed out in Example 2 that $\mathcal{O}^{\mathsf{ex}} \setminus \mathcal{O}^{\mathsf{ex}}_1$ is $\emptyset$-local w.r.t. $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_1)$. In particular, $\mathcal{O}^{\mathsf{ex}} \setminus \mathcal{O}^{\mathsf{ex}}_1$ is also $\bot$-local w.r.t $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_1)$, and therefore $\mathcal{O}^{\mathsf{ex}}_1$ is a $\bot$-module for $\mathcal{O}^{\mathsf{ex}}$ w.r.t $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_1)$. $\diamondsuit$

The algorithm for $\bot$-module extraction [7] is given in Algorithm 1. This algorithm computes the unique smallest $\bot$-module for a given $\mathcal{O}$ and Σ (the smallest subset $\mathcal{M} \subseteq \mathcal{O}$ s.t. $\mathcal{O} \setminus \mathcal{M}$ is $\bot$-local for $\Sigma \cup \mathsf{Sig}(\mathcal{M})$). We refer to such a smallest $\bot$-module as *the* $\bot$-module for $\mathcal{O}$ w.r.t. Σ, and denote it with $\mathcal{M}_{[\mathcal{O},\Sigma]}$.

Example 4. Suppose that we want to extract a $\bot$-module for $\mathcal{O}$ w.r.t. Γ, with

$$\Gamma = \{\mathsf{Knee}, \mathsf{Bursitis}, \mathsf{hasLocation}\}$$

It can be observed in Algorithm 1 that new symbols added to the module's signature in some iteration may cause more axioms to be added to the module

Algorithm 1 $\bot$-module$(\mathcal{O}, \Sigma)$

```
Input: an ontology $\mathcal{O}$ and a signature $\Sigma$
$\mathcal{M} := \emptyset$; $\mathcal{O}' := \mathcal{O}$
repeat
    $\mathsf{changed} := \mathsf{false}$
    for all $\alpha \in \mathcal{O}'$ do
        if $\alpha$ not $\bot$-local w.r.t $\Sigma \cup \mathsf{Sig}(\mathcal{M})$ then
            $\mathcal{M} := \mathcal{M} \cup \{\alpha\}$; $\mathcal{O}' := \mathcal{O}' \setminus \{\alpha\}$
            $\mathsf{changed} := \mathsf{true}$
until $\mathsf{changed} = \mathsf{false}$
return $\mathcal{M}$
```

in subsequent iterations. The algorithm stops once a fixpoint is reached and no more symbols need to be added to the module's signature.

On the first iteration, we would only add axioms (2), (4) and (8) to our module. Then, due to having added Joint and BursitisOfKnee to the module's signature, we would have to add axioms (1) and (3) as well. We would thus find that $\mathcal{M}_{[\mathcal{O}^{\mathsf{ex}},\Gamma]}$ coincides with $\mathcal{O}_1^{\mathsf{ex}}$ and its signature is precisely $\mathsf{Sig}(\mathcal{O}_1^{\mathsf{ex}})$. ♢

In addition to being modules as in Definition 2, $\bot$-modules enjoy a property that makes them especially well-suited for optimising classification [4].

Proposition 2. *Let $\mathcal{O}$ be an ontology, let A, B be classes in $\mathsf{Sig}(\mathcal{O}) \cup \{\top, \bot\}$, let $\Sigma \subseteq \mathsf{Sig}(\mathcal{O})$ with $A \in \Sigma$, and let $\mathcal{M} \subseteq \mathcal{O}$ be a $\bot$-module in $\mathcal{O}$ w.r.t. Σ. Then $\mathcal{O} \models A \sqsubseteq B$ iff $\mathcal{M} \models A \sqsubseteq B$.*

Example 5. Proposition 2 implies that $\mathcal{O} \not\models \mathsf{Bursitis} \sqsubseteq \mathsf{Cellulitis}$. Indeed, we have that $\mathsf{Bursitis} \in \Gamma$ but $\mathsf{Cellulitis} \notin \mathsf{Sig}(\mathcal{M}_{[\mathcal{O},\Gamma]})$; therefore, it must be the case that $\mathcal{M}_{[\mathcal{O},\Gamma]} \not\models \mathsf{Bursitis} \sqsubseteq \mathsf{Cellulitis}$, and thus $\mathcal{O} \not\models \mathsf{Bursitis} \sqsubseteq \mathsf{Cellulitis}$. ♢

3 Modular Classification of Ontologies

Consider an ontology $\mathcal{O}$ such that most of the axioms in it are expressed in some restricted fragment $\mathcal{L}$ of OWL 2. This is the case, considering $\mathcal{L}$ = OWL 2 EL, for our example ontology $\mathcal{O}^{\mathsf{ex}}$, whose $\mathcal{L}$-fragment $\mathcal{O}_{\mathcal{L}}^{\mathsf{ex}}$ contains all the axioms in $\mathcal{O}^{\mathsf{ex}}$ except axiom (1).

Our first goal is to identify a signature $\Sigma^{\mathcal{L}} \subseteq \mathsf{Sig}(\mathcal{O})$ such that the corresponding $\bot$-module $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]}$ is contained in the $\mathcal{L}$-fragment $\mathcal{O}_{\mathcal{L}}$ of $\mathcal{O}$. We call any such $\Sigma^{\mathcal{L}}$ an *$\mathcal{L}$-signature for $\mathcal{O}$*. Proposition 2 ensures that an $\mathcal{L}$-reasoner can then be used to determine all the subsumers of classes in $\Sigma^{\mathcal{L}}$. Section 3.1 addresses the problem of identifying as large an $\mathcal{L}$-signature as possible.

Once an $\mathcal{L}$-signature $\Sigma^{\mathcal{L}}$ has been identified, the use of a fully-fledged OWL 2 reasoner can be restricted to computing the subsumers of the classes in the complementary signature $\overline{\Sigma^{\mathcal{L}}} = \mathsf{Sig}(\mathcal{O}) \setminus \Sigma^{\mathcal{L}}$. The details of how our classification algorithm combines the use of both an $\mathcal{L}$-reasoner and an OWL 2 reasoner as *black boxes* are given in Section 3.2.

3.1 Computing an $\mathcal{L}$-signature

The definition of $\bot$-module suggests a simple "guess and check" algorithm for computing a (maximal) $\mathcal{L}$-signature for $\mathcal{O}$: consider all subsets $\Sigma \subseteq \mathsf{Sig}(\mathcal{O})$ in decreasing size order and, for each of them, check whether $\mathcal{M}_{[\mathcal{O},\Sigma]}$ is an $\mathcal{L}$-ontology. This could, however, be quite costly, and as our objective is to optimise classification we propose a goal directed algorithm. Although our algorithm is not guaranteed to compute a maximal $\mathcal{L}$-signature, it can be implemented very efficiently; furthermore, as shown in the evaluation section, it typically computes large $\mathcal{L}$-signatures, provided that $\mathcal{O}_{\mathcal{L}}$ is a large enough fragment of $\mathcal{O}$.

We start by pointing out that every $\mathcal{L}$-signature $\Sigma^{\mathcal{L}}$ *must* satisfy the property ($\star$) below. If ($\star$) does not hold, then $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]}$ will contain some non $\mathcal{L}$-axiom.

$$\text{Property } (\star): \qquad \mathcal{O} \setminus \mathcal{O}_{\mathcal{L}} \text{ is } \bot\text{-local w.r.t. } \Sigma^{\mathcal{L}}$$

Example 6. Consider again our example ontology $\mathcal{O}^{\mathsf{ex}}$ and let $\mathcal{L}$ be OWL 2 EL. As already mentioned, the $\mathcal{L}$-fragment $\mathcal{O}^{\mathsf{ex}}_{\mathcal{L}}$ of $\mathcal{O}^{\mathsf{ex}}$ contains all axioms in $\mathcal{O}^{\mathsf{ex}}$ except for (1). One may think that the signature of $\mathcal{O}^{\mathsf{ex}}_{\mathcal{L}}$ is an $\mathcal{L}$-signature, which makes the computation of a maximal $\mathcal{L}$-signature trivial; this is, however, not the case. Note that the signature of $\mathcal{O}^{\mathsf{ex}}_{\mathcal{L}}$, namely

$$\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_{\mathcal{L}}) = \mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}) \setminus \{\mathsf{BursitisOrCellulitisOfKnee}, \mathsf{CellulitisOfKnee}\}$$

is not an $\mathcal{L}$-signature for $\mathcal{O}^{\mathsf{ex}}$; indeed, axiom (1) is not $\bot$-local w.r.t $\mathsf{Sig}(\mathcal{O}^{\mathsf{ex}}_{\mathcal{L}})$. In contrast, we have that axiom (1) is $\bot$-local w.r.t.

$$\begin{aligned}\Gamma' = \{&\mathsf{Bursitis}, \mathsf{hasLocation}, \mathsf{Joint}, \mathsf{BursitisOfJoint}, \mathsf{Swelling}, \mathsf{Infection},\\ &\mathsf{InfectiousDisease}, \mathsf{Disease}, \mathsf{hasOrigin}\}\end{aligned}$$

Furthermore, $\mathcal{M}_{[\mathcal{O}^{\mathsf{ex}},\Gamma']}$ consists of axioms (3), (4) (5) (6) (7), which are all within the OWL 2 EL fragment; hence, Γ' is an $\mathcal{L}$-signature for $\mathcal{O}^{\mathsf{ex}}$. ♢

Although Example 6 might suggest that property ($\star$) is also a *sufficient* condition for $\Sigma^{\mathcal{L}}$ to be an $\mathcal{L}$-signature in $\mathcal{O}$, this is unfortunately not the case.

Example 7. Consider again the signature Γ from Example 4. Clearly, axiom (1) (the only non $\mathcal{L}$-axiom in $\mathcal{O}^{\mathsf{ex}}$) is $\bot$-local w.r.t Γ and hence ($\star$) holds for Γ. Note, however, that Γ is not an $\mathcal{L}$-signature for $\mathcal{O}^{\mathsf{ex}}$ since, as already discussed, axiom (1) is contained in $\mathcal{M}_{[\mathcal{O}^{\mathsf{ex}},\Gamma]}$. One way to address this problem is to reduce Γ to $\Gamma \setminus \{\mathsf{Knee}\}$. The corresponding $\bot$-module only contains axiom (4), which implies that such reduced signature is indeed an $\mathcal{L}$-signature for $\mathcal{O}^{\mathsf{ex}}$. ♢

Example 7 suggests an algorithm for computing an $\mathcal{L}$-signature for $\mathcal{O}$, which can be intuitively described as follows.

1. Reduce $\Sigma_0 = \mathsf{Sig}(\mathcal{O})$ to a subset Σ_1 of Σ_0 such that $\mathcal{S}_0 = \mathcal{O} \setminus \mathcal{O}_{\mathcal{L}}$ is $\bot$-local w.r.t. Σ_1 (thus satisfying $(\star)$).
2. Compute the set $\mathcal{S}_1$ of axioms in $\mathcal{M}_{[\mathcal{O},\Sigma_1]}$ containing symbols not in Σ_1.
3. Reduce Σ_1 to a subset Σ_2 of Σ_1 such that $\mathcal{S}_1$ is $\bot$-local w.r.t. Σ_2.
4. Repeat Steps [2-4] until the set of axioms computed in Step 2 is empty.

The sequence $(\Sigma_i)_{i\geq 0}$ will eventually converge to a fixpoint (as we will shortly prove), and this fixpoint will be guaranteed to be an $\mathcal{L}$-signature, $\Sigma^{\mathcal{L}}$. We next explain the intuition behind our algorithm with an example.

Example 8. Consider once more our example ontology $\mathcal{O}^{\mathsf{ex}}$. As already mentioned, the only non OWL 2 EL axiom is (1), so we start with

$$\begin{aligned}\Sigma_0 &= \mathsf{Sig}(\mathcal{O}^{\mathsf{ex}})\\ \mathcal{S}_0 &= \{\mathsf{BursitisOrCellulitisOfKnee} \equiv \mathsf{BursitisOfKnee} \sqcup \mathsf{CellulitisOfKnee}\}\end{aligned}$$

The only way to make axiom (1) $\bot$-local is by removing BursitisOrCellulitisOfKnee, BursitisOfKnee and CellulitisOfKnee from Σ_0. So we take

$$\Sigma_1 = \Sigma_0 \setminus \{\mathsf{BursitisOrCellulitisOfKnee}, \mathsf{BursitisOfKnee}, \mathsf{CellulitisOfKnee}\}$$

Next, we compute $\mathcal{M}_{[\mathcal{O}^{\mathsf{ex}},\Sigma_1]}$ using Algorithm 1. This module contains axiom (2), which mentions BursitisOfKnee (not in Σ_1). Because this class is in the module's signature, the module needs to contain axiom (1) as well. This gives us

$$\begin{aligned}\mathcal{S}_1 = \{&\mathsf{BursitisOrCellulitisOfKnee} \equiv \mathsf{BursitisOfKnee} \sqcup \mathsf{CellulitisOfKnee},\\ &\mathsf{BursitisOfKnee} \equiv \mathsf{Bursitis} \sqcap \exists \mathsf{hasLocation}.\mathsf{Knee}\}\end{aligned}$$

We have seen that, unless axiom (2) in $\mathcal{S}_1$ is outside the module, axiom (1) cannot be outside the module either. Thus, we need to make sure that axiom (2) is $\bot$-local. For this, we can take $\Sigma_2 = \Sigma_1 \setminus \{\mathsf{Knee}\}$.

At this point, we need not worry about axiom (1) anymore; it was already local w.r.t. Σ_1, so, by Proposition 1, it will be $\bot$-local w.r.t. any subset of Σ_1.

Next, we compute $\mathcal{M}_{[\mathcal{O}^{\mathsf{ex}},\Sigma_2]}$ and find that it contains all axioms in $\mathcal{O}^{\mathsf{ex}}$ except for axioms (1), (2) and (8). This implies that all symbols in $\mathcal{M}_{[\mathcal{O}^{\mathsf{ex}},\Sigma_2]}$ are in Σ_2 and hence $\mathcal{S}_3 = \emptyset$. The algorithm then terminates and returns $\Sigma^{\mathcal{L}} = \Sigma_2$.

Note that $\Sigma^{\mathcal{L}}$ is indeed an $\mathcal{L}$-signature since the module $\mathcal{M}_{[\mathcal{O}^{\mathsf{ex}},\Sigma_2]}$ does not contain axiom (1) and hence is an OWL 2 EL ontology. ♢

Note that there can be many ways to perform the signature reduction required in Steps 1 and 4. In Example 8, for instance, we could have taken $\Sigma_2 = \Sigma_1 \setminus \{\mathsf{Bursitis}\}$ or $\Sigma_2 = \Sigma_1 \setminus \{\mathsf{hasLocation}\}$, or even any subset thereof. Making reasonable choices requires good heuristics. In our implementation our choices are guided so that as many properties as possible are kept within $\Sigma^{\mathcal{L}}$. Indeed, ontologies typically contain many more classes than properties, and each property typically occurs in a larger number of axioms; thus, having a property outside $\Sigma^{\mathcal{L}}$ is likely to cause many other symbols to be left outside $\Sigma^{\mathcal{L}}$. The following example illustrates how different choices lead to rather different $\mathcal{L}$-signatures.

Algorithm 2 $\mathcal{L}$-$\mathsf{signature}(\mathcal{O})$

```
Input: an ontology $\mathcal{O}$
$\Sigma^{\mathcal{L}} := \mathsf{Sig}(\mathcal{O})$ ; $\mathcal{S} := \mathcal{O} \setminus \mathcal{O}_{\mathcal{L}}$
$\mathsf{canReduce} :=$ true
while $\mathcal{S} \neq \emptyset$ and $\mathsf{canReduce}$ do
    $\Sigma^{\mathcal{L}} := \mathsf{reduce}(\Sigma^{\mathcal{L}}, \mathcal{S})$
    if $\Sigma^{\mathcal{L}} = \emptyset$ then $\mathsf{canReduce} :=$ false
    else $\mathcal{S} := \{\alpha \in \mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \mid \mathsf{Sig}(\alpha) \nsubseteq \Sigma^{\mathcal{L}}\}$
return $\Sigma^{\mathcal{L}}$
```

Example 9. As already mentioned, in Example 8 we could have alternatively chosen to take $\Sigma_2 = \Sigma_1 \setminus \{\mathsf{hasLocation}\}$. This would have caused $\mathcal{M}_{[\mathcal{O},\Sigma_2]} = \mathcal{O}^{\mathsf{ex}}$ again, and we would have obtained

$$\mathcal{S}_3 = \mathcal{S}_2 \cup \{\mathsf{BursitisOfJoint} \equiv \mathsf{Bursitis} \sqcap \exists \mathsf{hasLocation}.\mathsf{Joint}\}$$

It would now be enough to take $\Sigma_3 = \Sigma_2 \setminus \{\mathsf{BursitisOfJoint}\}$, and we would have $\mathcal{S}_4 = \emptyset$. In this case we would get $\Sigma^{\mathcal{L}} = \Sigma_3$, which is smaller (and hence less appealing) than the $\Sigma^{\mathcal{L}}$ obtained in Example 8. ♢

These signature reductions satisfy the same properties that make them "acceptable". We can characterise these acceptable reductions as given next.

Definition 5. *Given an ontology $\mathcal{O}$, a signature reduction is a function*

$$\mathsf{reduce} : \mathcal{P}(\mathsf{Sig}(\mathcal{O})) \times \mathcal{P}(\mathcal{O}) \rightarrow \mathcal{P}(\mathsf{Sig}(\mathcal{O}))$$

that, given $\Sigma \in \mathcal{P}(\mathsf{Sig}(\mathcal{O}))$ and $\mathcal{S} \in \mathcal{P}(\mathcal{O})$ not $\bot$-local w.r.t. Σ, returns

1. *Σ if $\mathcal{S} = \emptyset$.*
2. *$\Sigma' \subset \Sigma$ s.t. each axiom in $\mathcal{S}$ is $\bot$-local w.r.t. Σ' if $\mathcal{S} \neq \emptyset$ and Σ' exists.*
3. *$\emptyset$ otherwise.*

Note that the Cases 1 and 3 correspond to the extreme situations when $\mathcal{S} = \emptyset$ or when there is no satisfying way of reducing Σ. Case 2 constitutes the essence of the reduction, namely to compute a strict subset of the signature that makes the given set of axioms $\bot$-local.

Given a particular reduce function, Algorithm 2 accepts an ontology $\mathcal{O}$ and returns a signature $\Sigma^{\mathcal{L}} \subseteq \mathsf{Sig}(\mathcal{O})$. Theorem 1 guarantees the termination of Algorithm 2 as well as its correctness: any non-empty signature returned by the algorithm is an $\mathcal{L}$-signature for $\mathcal{O}$.

Theorem 1. *Let $\mathcal{O}$ be an ontology and let reduce be a signature reduction function. Furthermore, let $\mathcal{S}_i$, Σ_i ($i \geq 0$) be defined by the following construction:*

$$(i = 0): \quad \Sigma_0 = \mathsf{Sig}(\mathcal{O}) \qquad \mathcal{S}_0 = \mathcal{O} \setminus \mathcal{O}_{\mathcal{L}}$$

$$(i \geq 1): \quad \Sigma_i = \mathsf{reduce}(\Sigma_{i-1}, \mathcal{S}_{i-1}) \quad \mathcal{S}_i = \{\alpha \in \mathcal{M}_{[\mathcal{O},\Sigma_i]} \mid \mathsf{Sig}(\alpha) \nsubseteq \Sigma_i\}$$

Finally, let $\Sigma^{\mathcal{L}} := \bigcap_{i \geq 0} \Sigma_i$. *Then, the following properties hold:*

1. *There exists* $k < |\mathsf{Sig}(\mathcal{O})|$ *such that either* $\Sigma_k = \emptyset$ *or* $\mathcal{S}_k = \emptyset$.
2. *Either* $\Sigma^{\mathcal{L}} = \emptyset$ *or* $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \subseteq \mathcal{O}_{\mathcal{L}}$.

Proof. We first show Claim 1. Suppose $\Sigma_i \neq \emptyset$ for each $i \geq 0$. A straightforward inductive argument would show that $\Sigma_j \subseteq \Sigma_i$ for each $j > i \geq 0$. Furthermore, $\Sigma_0 = \mathsf{Sig}(\mathcal{O})$, so it cannot be the case that $\Sigma_j \subset \Sigma_i$ for each $0 \leq i < j \leq |\mathsf{Sig}(\mathcal{O})|$. Therefore, there must be some $k < |\mathsf{Sig}(\mathcal{O})|$ such that $\Sigma_{k+1} = \Sigma_k$; by the definition of reduce, this implies that $\mathcal{S}_k = \emptyset$.

We finally show Claim 2. Suppose $\Sigma^{\mathcal{L}} \neq \emptyset$. It is enough to prove that each $\alpha \in \mathcal{O} \setminus \mathcal{O}_{\mathcal{L}}$ is $\bot$-local w.r.t. $\Sigma^{\mathcal{L}} \cup \mathsf{Sig}(\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]})$.

First, we are going to see that $\mathsf{Sig}(\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]}) \subseteq \Sigma^{\mathcal{L}}$. According to Claim 1, there exists $k < |\mathsf{Sig}(\mathcal{O})|$ such that $\mathcal{S}_k = \emptyset$. This implies that, for each axiom $\alpha \in \mathcal{M}_{[\mathcal{O},\Sigma_k]}$, we have $\mathsf{Sig}(\alpha) \subseteq \Sigma_k$. It is easy to see that $\mathcal{S}_k = \emptyset$ also implies that $\Sigma_j = \Sigma_k$ for each $j > k$. Together with the fact that $\Sigma_j \subseteq \Sigma_i$ for each $j > i \geq 0$, this implies $\Sigma^{\mathcal{L}} = \bigcap_{i \geq 0} \Sigma_i = \Sigma_k$. But then for each $\alpha \in \mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} = \mathcal{M}_{[\mathcal{O},\Sigma_k]}$ we have $\mathsf{Sig}(\alpha) \subseteq \Sigma_k = \Sigma^{\mathcal{L}}$, and so $\mathsf{Sig}(\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]}) \subseteq \Sigma^{\mathcal{L}}$.

Now we can just prove that each $\alpha \in \mathcal{O} \setminus \mathcal{O}_{\mathcal{L}}$ is $\bot$-local w.r.t. $\Sigma^{\mathcal{L}}$. Because $\Sigma^{\mathcal{L}} = \bigcap_{i \geq 0} \Sigma_i \neq \emptyset$, in particular it must be the case that $\Sigma_0 \neq \emptyset$. By definition of reduce, either $\mathcal{O} \setminus \mathcal{O}_{\mathcal{L}} = \emptyset$, in which case it is immediate that $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \subseteq \mathcal{O}_{\mathcal{L}}$, or every axiom in $\mathcal{S}_0 = \mathcal{O} \setminus \mathcal{O}_{\mathcal{L}}$ is $\bot$-local w.r.t. $\Sigma_1 = \mathsf{reduce}(\Sigma_0, \mathcal{S}_0)$. Then, by Proposition 1, each $\alpha \in \mathcal{O} \setminus \mathcal{O}_{\mathcal{L}}$ is $\bot$-local w.r.t. $\Sigma^{\mathcal{L}} \subseteq \Sigma_1$. □

3.2 Black-box Modular Classification

Having identified a (hopefully large) $\mathcal{L}$-signature for our ontology $\mathcal{O}$, we can next proceed to classify the ontology in a modular way.

As already mentioned, we can fully classify the classes in $\Sigma^{\mathcal{L}}$ using only an $\mathcal{L}$-reasoner. This is a consequence of Proposition 2 and the fact that $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \subseteq \mathcal{O}_{\mathcal{L}}$.

To classify the classes in $\overline{\Sigma^{\mathcal{L}}} = \mathsf{Sig}(\mathcal{O}) \setminus \Sigma^{\mathcal{L}}$, however, a fully fledged OWL 2 reasoner is still required. By Proposition 2, the OWL 2 reasoner does not need to consider all the axioms in $\mathcal{O}$, but only those in the relevant module $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}$.

Once the OWL 2 reasoner has computed the classification of $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}$, we can express the classification result as simple subsumption axioms of the form $A \sqsubseteq B$. These axioms, together with $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]}$, can be given to the $\mathcal{L}$-reasoner, which will use the resulting ontology to compute a complete classification of $\mathcal{O}$.

Algorithm 3 describes the entire classification process for a given fragment $\mathcal{L}$ of OWL 2 and a particular signature reduction function reduce. The function $\mathcal{L}$-signature is as given in Algorithm 2. The function classification returns the classification of the given ontology (computed using an OWL 2 reasoner) as a set of axioms of the form $A \sqsubseteq B$. The function $\mathsf{classification}_{\mathcal{L}}$ returns the classification of the given ontology as computed by an $\mathcal{L}$-reasoner.

```
Algorithm 3 $\mathcal{L}$-ModularClassification($\mathcal{O}$)
Input: an OWL 2 ontology $\mathcal{O}$
$\Sigma^{\mathcal{L}} := \mathcal{L}\text{-}\mathsf{signature}(\mathcal{O})$        ▷ See Algorithm 2
$\mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}} := \mathsf{classification}(\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]})$        ▷ using the OWL 2 reasoner
$\mathcal{H} := \mathsf{classification}_{\mathcal{L}}(\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}})$        ▷ using the $\mathcal{L}$-reasoner
return $\mathcal{H}$
```

Example 10. Recall the $\mathcal{L}$-signature $\Sigma^{\mathcal{L}}$ for $\mathcal{O}^{\mathsf{ex}}$ computed in Example 8:

$$\begin{aligned}\Sigma^{\mathcal{L}} = \{&\mathsf{Bursitis}, \mathsf{hasLocation}, \mathsf{BursitisOfJoint}, \mathsf{Joint}, \mathsf{Swelling}, \mathsf{Cellulitis},\\ &\mathsf{hasOrigin}, \mathsf{Infection}, \mathsf{InfectiousDisease}, \mathsf{Disease}\}\end{aligned}$$

The complementary signature is

$$\overline{\Sigma^{\mathcal{L}}} = \{\mathsf{BursitisOrCellulitisOfKnee}, \mathsf{BursitisOfKnee}, \mathsf{CellulitisOfKnee}, \mathsf{Knee}\}$$

and the relevant $\bot$-module is $\mathcal{M}_{[\mathcal{O}^{\mathsf{ex}},\overline{\Sigma^{\mathcal{L}}}]} = \{(1), (2), (3), (4), (8)\}$. The classification of this module leads to the following subsumptions $\mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}$:

$$\begin{aligned}\mathsf{CellulitisOfKnee} &\sqsubseteq \mathsf{BursitisOrCellulitisOfKnee} & \mathsf{BursitisOfJoint} &\sqsubseteq \mathsf{Bursitis}\\ \mathsf{BursitisOfKnee} &\sqsubseteq \mathsf{BursitisOrCellulitisOfKnee} & \mathsf{Bursitis} &\sqsubseteq \mathsf{Swelling}\\ \mathsf{BursitisOfKnee} &\sqsubseteq \mathsf{BursitisOfJoint} & \mathsf{Knee} &\sqsubseteq \mathsf{Joint}\\ \mathsf{BursitisOfKnee} &\sqsubseteq \mathsf{Bursitis} & &\end{aligned}$$

We can now use the $\mathcal{L}$-reasoner to classify $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}$, thus obtaining all the remaining subsumption relationships that hold in $\mathcal{O}^{\mathsf{ex}}$:

$$\begin{aligned}\mathsf{Cellulitis} &\sqsubseteq \mathsf{Swelling} & \mathsf{Cellulitis} &\sqsubseteq \mathsf{InfectiousDisease}\\ \mathsf{InfectiousDisease} &\sqsubseteq \mathsf{Disease} & \mathsf{Cellulitis} &\sqsubseteq \mathsf{Disease}\end{aligned}$$

♢

The following Theorem establishes the correctness of Algorithm 3.

Theorem 2. *Let $\mathcal{O}$ be an ontology, let* reduce *be a signature reduction function and let $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]}$, $\mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}$ be as computed by Algorithm 3. Then, for any two classes $A \in \mathsf{Sig}(\mathcal{O})$ and $B \in \mathsf{Sig}(\mathcal{O}) \cup \{\top, \bot\}$, we have that*

$$\mathcal{O} \models A \sqsubseteq B \textit{ iff } (\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}) \models A \sqsubseteq B$$

Proof. Let $A \in \mathsf{Sig}(\mathcal{O})$ and $B \in \mathsf{Sig}(\mathcal{O}) \cup \{\top, \bot\}$. We consider two cases.

- Case 1: $A \in \Sigma^{\mathcal{L}}$. Then, by Proposition 2, we have that $\mathcal{O} \models A \sqsubseteq B$ iff $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \models A \sqsubseteq B$. Also, because $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \subseteq \mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}$, by monotonicity we have that $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \models A \sqsubseteq B$ implies $(\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}) \models A \sqsubseteq B$. It remains to show that

$$(\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}) \models A \sqsubseteq B \text{ implies } \mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \models A \sqsubseteq B \qquad (9)$$

Because $\mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}$ encodes the classification of $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}$, we have that

$$(\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}) \models A \sqsubseteq B \text{ implies } (\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}) \models A \sqsubseteq B \quad (10)$$

Now, it is immediate that $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]} \subseteq \mathcal{O}$; thus, because $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]}$ is a module for $\mathcal{O}$ w.r.t. $\Sigma^{\mathcal{L}}$, it must also be a module for $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}$ w.r.t. $\Sigma^{\mathcal{L}}$. So, again by Proposition 2, we have

$$(\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}) \models A \sqsubseteq B \text{ implies } \mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \models A \sqsubseteq B \quad (11)$$

Now, (10) and (11) imply (9), as required.

- Case 2: $A \in \overline{\Sigma^{\mathcal{L}}} = \mathsf{Sig}(\mathcal{O}) \setminus \Sigma^{\mathcal{L}}$. Then, by Proposition 2, $\mathcal{O} \models A \sqsubseteq B$ iff $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]} \models A \sqsubseteq B$. Because $\mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}$ represents the classification of $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}$, we have $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]} \models A \sqsubseteq B$ iff $\mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}} \models A \sqsubseteq B$. It remains to show that

$$\mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}} \models A \sqsubseteq B \text{ iff } \mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}} \models A \sqsubseteq B \quad (12)$$

Left to right implication holds directly by monotonicity, so let us assume $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}} \models A \sqsubseteq B$. Since $\mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}}$ is the classification of $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}$, we have $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]} \models A \sqsubseteq B$. Now, $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}$ is a $\bot$-module for $\mathcal{M}_{[\mathcal{O},\Sigma^{\mathcal{L}}]} \cup \mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]} \subseteq \mathcal{O}$ w.r.t. $\overline{\Sigma^{\mathcal{L}}}$ since it is a $\bot$-module for $\mathcal{O}$ w.r.t. $\overline{\Sigma^{\mathcal{L}}}$. By Proposition 2, $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]} \models A \sqsubseteq B$, which implies $\mathcal{H}_{\overline{\Sigma^{\mathcal{L}}}} \models A \sqsubseteq B$. □

4 Implementation and Experiments

We have implemented our modular reasoner MORe[2] in Java using the OWL API.[3] Our implementation of a signature reduction function reduce (see Section 3.1) is based on the $\bot$-locality module extractor described in [14].[4] Our system currently integrates ELK, which acts as an OWL 2 EL reasoner, and HermiT, which plays the role of a fully-fledged OWL 2 reasoner.

In the implementation of the signature reduction, symbols required to make a set of axioms $\bot$-local are selected greedily axiom by axiom. As discussed in Examples 8 and 9, when selecting symbols it is often a good idea to implement heuristics that try to keep as many properties as possible within $\Sigma^{\mathcal{L}}$.

Evaluation on BioPortal ontologies. We have compared classification times obtained by MORe and HermiT over a set of large bio-medical ontologies available from BioPortal.[5] Results are summarised in the upper part of Table 1. The Gene Ontology (GO) and Gazetteer are OWL 2 EL ontologies; therefore, MORe delegates all the work to ELK, with the consequent performance improvement.

[2] http : //www.cs.ox.ac.uk/isg/tools/MORe/

[3] http : //owlapi.sourceforge.net/

[4] http : //www.cs.ox.ac.uk/isg/tools/ModuleExtractor

[5] http : //bioportal.bioontology.org/

Table 1. MORe vs HermiT. Comparison on BioPortal ontologies and on mapped ontologies. Tables show number of axioms outside OWL 2 EL, relative size of $\Sigma^{\mathcal{L}}$ and $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}$, and classification times using HermiT and MORe. For MORe we specify the performance gain w.r.t. HermiT alone and the time taken by HermiT and ELK.

Ontology	$\|\mathcal{O} \setminus \mathcal{O}_{\mathcal{L}}\|$	$\|\Sigma^{\mathcal{L}}\|$	$\|\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}\|$	**Classif. time** (seconds)			
				HermiT	MORe		
					total	HermiT	ELK
GO	0	100%	0%	7.1	2.2 (↓69.0%)	0	0.1
Gazeteer	0	100%	0%	838.1	28.2 (↓96.6%)	0	15.6
NCI	65	94.9%	15.4%	84.1	28.6 (↓66.0%)	15.8	3.3
Protein	12	98.1%	6.6%	11.4	2.9 (↓74.6%)	0.4	0.9
Biomodels	22,079	45.2%	66.4%	741.4	575.6 (↓22.4%)	540.1	2.6
cellCycle	1	$> 99.9\%$	$< 0.1\%$	–	13.9 (–)	<0.1	4.9
NCI+CHEBI	65	95.6%	10.3%	116.6	34.0 (↓70.8%)	16.3	4.1
NCI+GO	65	96.7%	10.4%	110.0	37.6 (↓65.8%)	17.6	3.2
NCI+Mouse	65	96.0%	13.3%	93.7	31.0 (↓66.9%)	16.6	2.6

For the latest version of NCI and for Protein, which contain only a small number of axioms outside OWL 2 EL, the obtained $\Sigma^{\mathcal{L}}$ contains most of $\mathsf{Sig}(\mathcal{O})$, and hence MORe significantly outperforms HermiT. Biomodels, however, contains a large number of axioms outside OWL 2 EL, thus the size of $\Sigma^{\mathcal{L}}$ is proportionally much smaller and MORe must assign a higher workload to HermiT, which results in a more modest performance gain. Finally, the Cell Cycle ontology is an extreme case: an ontology that is *almost* OWL 2 EL and can be classified efficiently with MORe, while HermiT alone runs out of memory.

Evaluation on ontologies integrated via mappings. We have used the ontology matching tool LogMap [13] to integrate the latest version of NCI with other widely used ontologies. Results are summarised in the lower part of Table 1. We can observe that MORe consistently outperforms HermiT by 65-70%.

Evaluation on versions of NCI. We have compared MORe with HermiT on 10 versions of NCI.[6] Unsurprisingly, there have been significant variations in 10 years of development; for example, a 2003 version was entirely in OWL 2 EL, a version in 2009 contained more than $4,000$ axioms outside OWL 2 EL, and the current version only contains 65. Figure 2 summarises our results; in all cases, MORe outperforms HermiT.

Extensions of SNOMED. We have manually extended SNOMED (v. Jan 2010), which is fully expressed in OWL 2 EL, with a few disjunctive axioms suggested by domain experts who are involved in SNOMED's development. All these axioms share the same structure; for example,

$$\mathsf{Sprain\_of\_ankle\_OR\_foot} \equiv \mathsf{Sprain\_of\_ankle} \sqcup \mathsf{Sprain\_of\_foot}$$

[6] See http : //ncit.nci.nih.gov/. We consider the latest version in each year.

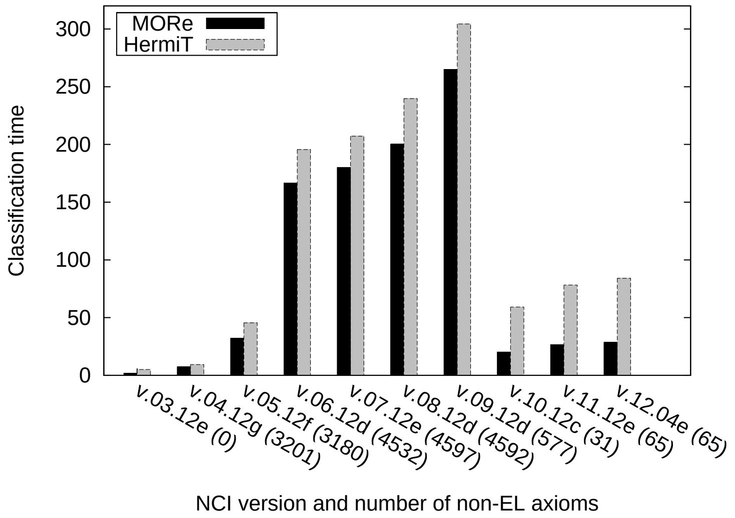

Fig. 2. Classification times (seconds) for MORe and HermiT on NCI. The X axis indicates the version and the number of axioms outside OWL 2 EL (in parenthesis).

introduces a new class that is fully defined as the set of all sprains that affect either the ankle or the foot (or both). In total, 20 such axioms were added to SNOMED, one by one.

Table 2 presents the results obtained for these extended ontologies. Each of them is identified, in the first column, by the number of disjunctive axioms that it contains. The second and third columns indicate the relative sizes of the computed $\mathcal{L}$-signature and the resulting $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}$. The last two columns give the classification times obtained using MORe and HermiT. In most cases classification times are improved by between one and two orders of magnitude.

We can observe, however, that axioms 15, 16, 18, and 19 have a significant effect on the size of $\Sigma^{\mathcal{L}}$ and, consequently, on the size of $\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}$ and the total classification time. This is due to the classes involved in these particular axioms, which force our algorithm to keep very general classes outside $\Sigma^{\mathcal{L}}$; for example, in one of these cases, the class Liquid Substance is removed from the successive approximations to $\Sigma^{\mathcal{L}}$ at some point during its computation; by Proposition 2, all classes representing some kind of liquid substance—and therefore subsumed by the class Liquid Substance—must be left outside $\Sigma^{\mathcal{L}}$ too, which leads to a significantly smaller $\mathcal{L}$-signature. It is part of our future work plan to improve

Table 2. Extensions of SNOMED

$\#\sqcup$	$\lvert\Sigma^{\mathcal{L}}\rvert$	$\lvert\mathcal{M}_{[\mathcal{O},\overline{\Sigma^{\mathcal{L}}}]}\rvert$	Classif. time	
			HermiT	MORe
1	99.98%	0.10%	1,788.5	25.3
2	99.94%	0.24%	1,959.2	29.0
3	99.88%	0.52%	1,872.8	29.3
4	99.86%	0.61%	1,933.2	30.9
5	99.86%	0.63%	1,898.6	31.6
6	99.86%	0.63%	1,920.2	31.0
7	99.86%	0.64%	1,884.8	31.8
8	99.85%	0.65%	1,868.2	31.3
9	99.85%	0.66%	1,937.2	31.9
10	99.79%	1.00%	1,863.7	32.8
11	99.79%	1.00%	1,922.5	30.6
12	99.79%	1.01%	1,912.4	30.6
13	99.78%	1.02%	1,864.0	30.5
14	99.76%	1.91%	1,890.5	33.0
15	98.76%	3.19%	1,925.9	42.9
16	97.10%	9.79%	1,930.2	138.5
17	97.08%	9.89%	1,927.9	134.7
18	96.27%	13.50%	1,881.4	269.8
19	94.10%	17.65%	1,847.4	401.1
20	94.02%	17.78%	1,904.0	410.8

the heuristics we use in order to avoid, when possible, leaving out of $\Sigma^{\mathcal{L}}$ classes that are likely to be high up in the class subsumption hierarchy.

5 Conclusion and Future Work

In this paper, we have proposed a technique for classifying an OWL 2 ontology $\mathcal{O}$ by exploiting a reasoner for one of its profiles. Our technique allows us to show that the subsumers of many classes in $\mathcal{O}$ can be completely determined using only the fragment-specific reasoner. Our technique is general and flexible, it exhibits pay-as-you-go behaviour, and it is relatively easy to implement. Although the implementation in our reasoner MORe is still prototypical, our preliminary experiments show the potential of our approach in practice.

There are also many interesting possibilities for future work:

- Our heuristics for computing an $\mathcal{L}$-signature are rather basic, and there is plenty of room for improvement. For example, it might be possible to explore modular decomposition techniques to compute larger $\mathcal{L}$-signatures [24].
- $\bot$-modules provide very strong preservation guarantees (they preserve not just atomic subsumptions, but even *models*). It would be interesting to devise techniques for extracting modules that are more “permissive”, in the sense that they only provide preservation guarantees for atomic subsumptions.
- Our technique could also be applied to a different notion of locality, as long as it satisfied a result analogous to Proposition 2.
- We could explore ontology rewriting techniques that complement module extraction. By rewriting $\mathcal{O}$ into an $\mathcal{L}$-ontology $\mathcal{O}'$ such that $\mathcal{O}' \models \mathcal{O}$, and classifying $\mathcal{O}'$, we can obtain an “upper bound” on the classification of $\mathcal{O}$.

Acknowledgements. This work was supported by the Royal Society, the EU FP7 project Optique and the EPSRC projects Score!, ExODA and LogMap.

References

1. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ envelope. In: IJCAI (2005)
2. Baader, F., Franconi, E., Hollunder, B., Nebel, B., Profitlich, H.: An empirical analysis of optimization techniques for terminological representation systems. Applied Intelligence 4(2), 109–132 (1994)
3. Baader, F., Lutz, C., Suntisrivaraporn, B.: CEL - a polynomial-time reasoner for life science ontologies. In: Proc. of IJCAR. pp. 287–291 (2006)
4. Cuenca Grau, B., Halaschek-Wiener, C., Kazakov, Y., Suntisrivaraporn, B.: Incremental classification of description logics ontologies. JAR 44(4), 337–369 (2010)
5. Cuenca Grau, B., Horrocks, I., Kazakov, Y., Sattler, U.: Modular reuse of ontologies: Theory and practice. JAIR 31, 273–318 (2008)
6. Cuenca Grau, B., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P.F., Sattler, U.: OWL 2: The next step for OWL. J. Web Semantics (JWS) 6(4), 309–322 (2008)
7. Cuenca Grau, B., Horrocks, I., Kazakov, Y., Sattler, U.: Just the right amount: extracting modules from ontologies. In: WWW. pp. 717–726 (2007)
8. Glimm, B., Horrocks, I., Motik, B., Shearer, R., Stoilos, G.: A novel approach to ontology classification. J. of Web Semantics 10(1) (2011)
9. Haarslev, V., Möller, R.: High performance reasoning with very large knowledge bases: A practical case study. In: Proc. IJCAI. pp. 161–168 (2001)
10. Haarslev, V., Möller, R.: Racer system description. In: IJCAR. pp. 701–705 (2001)
11. Horrocks, I.: Implementation and optimisation techniques. In: The Description Logic Handbook: Theory, Implementation, and Applications, pp. 306–346 (2003)
12. Horrocks, I., Kutz, O., Sattler, U.: The even more irresistible $\mathcal{SROIQ}$. In: Proc. of KR. pp. 57–67 (2006)
13. Jiménez-Ruiz, E., Cuenca Grau, B.: Logmap: Logic-based and scalable ontology matching. In: Proc. of ISWC (2011)
14. Jimenez-Ruiz, E., Cuenca Grau, B., Schneider, T., Sattler, U., Berlanga, R.: Safe and economic re-use of ontologies: a logic-based methodology and tool support. In: Proc. of ESWC (2008)
15. Kazakov, Y.: $\mathcal{RIQ}$ and $\mathcal{SROIQ}$ are harder than $\mathcal{SHOIQ}$. In: Proc. of KR. pp. 274–284 (2008)
16. Kazakov, Y., Krötzsch, M., Simančík, F.: Concurrent classification of $\mathcal{EL}$ ontologies. In: Proc. of ISWC. vol. 7032 (2011)
17. Lutz, C., Walther, D., Wolter, F.: Conservative extensions in expressive description logics. In: Proc. of IJCAI. pp. 453–458 (2007)
18. Lutz, C., Wolter, F.: Conservative extensions in the lightweight description logic $\mathcal{EL}$. In: Proc. of CADE-21. vol. 4603 (2007)
19. Motik, B., Cuenca Grau, B., Horrocks, I., Wu, Z., Fokoue, A., Lutz, C.: OWL 2 web ontology language profiles. W3C Recommendation (2009)
20. OWL 2 Web Ontology Language Direct Semantics. W3C Recommendation (27 October 2009), http://www.w3.org/TR/owl2-direct-semantics/
21. Sirin, E., Parsia, B., Cuenca Grau, B., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL DL reasoner. J. of Web Semantics 5(2), 51–53 (2007)
22. Tsarkov, D., Horrocks, I.: FaCT++ description logic reasoner: System description. In: Proc. of IJCAR. vol. 4130, pp. 292–297 (2006)
23. Tsarkov, D., Horrocks, I., Patel-Schneider, P.: Optimizing terminological reasoning for expressive description logics. JAR 39(3), 277–316 (2007)
24. Vescovo, C.D., Parsia, B., Sattler, U., Schneider, T.: The modular structure of an ontology: Atomic decomposition. In: Proc. of IJCAI. pp. 2232–2237 (2011)