Practical Reasoning with Nominals in the $\mathcal{EL}$ Family of Description Logics

Yevgeny Kazakov
Institute of Artificial Intelligence
Ulm University, Germany

Markus Krötzsch and **František Simančík**
Department of Computer Science
University of Oxford, UK

Abstract

The $\mathcal{EL}$ family of description logics (DLs) has been designed to provide a restricted syntax for commonly used DL constructors with the goal to guarantee polynomial complexity of reasoning. Yet, polynomial complexity does not always mean that the underlying reasoning procedure is efficient in practice. In this paper we consider a simple DL $\mathcal{ELO}$ from the $\mathcal{EL}$ family that admits nominals, and argue that existing polynomial reasoning procedures for $\mathcal{ELO}$ can be impractical for many realistic ontologies. To solve the problem, we describe an optimization strategy in which the inference rules required for reasoning with nominals are avoided as much as possible. The optimized procedure is evaluated within the reasoner ELK and demonstrated to perform well in practice.

Introduction

Description logics (DLs) have been remarkably successful in many applications of knowledge representation and reasoning. Reasoning in DLs, however, often is of very high worst-case complexity, motivating the study of smaller logics that allow for polynomial time algorithms for major reasoning tasks. A prominent result of this research was the DL $\mathcal{EL}$ which is already expressive enough for the important medical ontology SNOMED CT. The theoretical advantage of polynomial complexity could also be exploited in practice, leading to dedicated reasoners that show excellent performance on SNOMED CT, including CEL (Baader, Lutz, and Suntisrivaraporn 2006), Snorocket (Lawley and Bousquet 2010), jCEL (Mendez, Ecke, and Turhan 2011), and ELK (Kazakov, Krötzsch, and Simančík 2011a).

Continued research strove to extend $\mathcal{EL}$ with additional features while preserving its low worst-case complexity. This led to the description logic $\mathcal{EL}^{++}$ (Baader, Brandt, and Lutz 2005), its extension with certain *range restrictions* (Baader, Brandt, and Lutz 2008), and ultimately to the OWL EL profile of the Web Ontology Language as standardized by the W3C (Motik et al. 27 October 2009).

A very interesting feature that $\mathcal{EL}^{++}$ and OWL EL add to $\mathcal{EL}$ are *nominals*, i.e., concepts that have exactly one element. In general, this can be used to express enumerations, e.g., expressions of the form

$$\mathit{TheBeatles} \equiv \{\mathit{john}\} \sqcup \{\mathit{paul}\} \sqcup \{\mathit{george}\} \sqcup \{\mathit{ringo}\}.$$

Since $\mathcal{EL}$-type logics do not feature unions ($\sqcup$), the use of nominals there is limited to singleton concepts. Yet, there are still a number of interesting applications in this case. For example, the medical ontology Galen defines *MalePatternBaldness* as a kind of *LossOfScalpHair* that occurs in male patients. Using nominals, this could be expressed as follows:

$$\begin{aligned}&\mathit{LossOfScalpHair} \sqcap\\ &\quad \exists \mathit{hasPhenotypicalSex}.\exists \mathit{hasAbsoluteState}.\{\mathit{maleSex}\}.\end{aligned}$$

The nominal $\{\mathit{maleSex}\}$ denotes a concept with a single element, and the definition thus asserts that the role *hasAbsoluteState* has exactly this single value for every instance of *MalePatternBaldness*. This is generally expressed with concept expressions of the form $\exists R.\{c\}$ for which the OWL standard even introduces a dedicated syntactic shortcut "ObjectHasValue."

In practice, however, nominals are hardly used in OWL EL ontologies. Even Galen models *maleSex* as an atomic concept, which seems unintuitive since there is only one male sex. A closer look reveals many other atomic concepts that are used as *values* for roles rather than as *classes* of objects, e.g., *blue*, *soluble*, and even *sixteen*.

What is the reason for this apparent lack of nominals in current ontologies? One possible explanation is that practical tool support for nominals in OWL EL is extremely limited. Amongst the currently available $\mathcal{EL}$ reasoners, Snorocket provides no support for nominals, CEL only supports ABox assertions, and the support for nominals in jCEL is incomplete. One could hope this to be a minor omission, given that reasoning is still known to be polynomial in the worst case. However, the implementation of algorithms that can handle nominals efficiently turned out to be challenging. A difficulty in this case is that, in the presence of nominals, mere non-emptiness of concepts can lead to new entailments, e.g., asserting that a particular concept has at least one instance may lead to a new subsumption between atomic concepts. This contrasts strongly to the case of $\mathcal{EL}$ without nominals, where non-emptiness of concepts (and, in fact, arbitrary ABox assertions) can never entail a new TBox fact.

To deal with this difficulty, algorithms must take non-emptiness of concepts into account during reasoning, e.g., by tracking whether non-emptiness of one concept implies non-emptiness of another. Baader et al. (2005) proposed to

	Syntax	Semantics
Concepts:		
atomic concept	A	$A^{\mathcal{I}}$
nominal	$\{a\}$	$\{a^{\mathcal{I}}\}$
top	$\top$	$\Delta^{\mathcal{I}}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists R.C$	$\{x \mid \exists y \in C^{\mathcal{I}} \colon \langle x, y\rangle \in R^{\mathcal{I}}\}$
Axioms:		
concept inclusion	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$

Table 1: Syntax and semantics of $\mathcal{ELO}$

do this by computing a *reachability relation* $\rightsquigarrow_R$. Unfortunately, this relation turns out to be prohibitively large in many practical cases: experiments in this paper show that it often exceeds the total amount of entailed atomic concept subsumptions by several orders of magnitude. Recent work indicates that such problems are not merely a deficiency of the particular algorithm, but that nominals represent a key challenge for consequence-based OWL EL reasoning procedures in general (Krötzsch 2011).

In this paper, we address this challenge with new reasoning procedures that we have developed in the context of our OWL EL reasoner ELK. Since the difficulties caused by adding nominals to $\mathcal{EL}$ are largely orthogonal to those caused by the remaining features of OWL EL, in order to keep the presentation as simple as possible, in this paper we focus on a very simple logic $\mathcal{ELO}$. Our results presented here can be applied to other logics from the $\mathcal{EL}$ family as well. Our main contributions are the following:

Reasoning Calculus We analyze reasoning with nominals and present a sound and complete consequence-based inferencing calculus for $\mathcal{ELO}$.

Optimization We optimize our algorithm to obtain a "pay-as-you-go" behavior that avoids the performance penalties of the general algorithm in cases where no interesting entailments can possibly follow from the use of nominals. We present three techniques: *axiom reuse*, use of *strongly connected components*, and *overestimation*.

Implementation and Evaluation Based on our implementation in ELK, we evaluate to what extent these optimizations improve performance in practical cases. We find that all three optimizations can lead to significant improvements for practical ontologies. Our experiments also show that the basic calculus without our modifications is infeasible in many cases.

Safe Use of Nominals Abstracting from the ideas underlying this optimization, we formulate syntactic conditions by which one can easily check whether nominals are used *safely* in the sense that they do not lead to additional entailments. Experiments show that many practical ontologies satisfy this criterion.

Preliminaries

The vocabulary of $\mathcal{ELO}$ consists of countably infinite sets of atomic *concepts*, (atomic) *roles*, and *individuals*. Complex concepts are defined recursively using the constructors in Table 1. Here we use the letters C and D for concepts, A for atomic concepts, R for roles, and a for individuals. An *ontology* is a finite set of concept inclusion axioms $C \sqsubseteq D$. A concept equivalence $C \equiv D$ is an abbreviation for the two concept inclusions $C \sqsubseteq D$ and $D \sqsubseteq C$.

$$\mathsf{R}_{\sqsubseteq}\ \frac{C \sqsubseteq D}{C \sqsubseteq E} : D \sqsubseteq E \in \mathcal{O}$$

$$\mathsf{R}_{\sqcap}^{-}\ \frac{C \sqsubseteq D_1 \sqcap D_2}{\begin{array}{c} C \sqsubseteq D_1 \\ C \sqsubseteq D_2 \end{array}}$$

$$\mathsf{R}_{\exists}^{-}\ \frac{C \sqsubseteq \exists R.D}{D \sqsubseteq D}$$

$$\mathsf{R}_{\top}^{+}\ \frac{C \sqsubseteq C}{C \sqsubseteq \top} : \top \text{ occurs in } \mathcal{O}$$

$$\mathsf{R}_{\sqcap}^{+}\ \frac{C \sqsubseteq D_1 \quad C \sqsubseteq D_2}{C \sqsubseteq D_1 \sqcap D_2} : D_1 \sqcap D_2 \text{ occurs in } \mathcal{O}$$

$$\mathsf{R}_{\exists}^{+}\ \frac{C \sqsubseteq \exists R.D \quad D \sqsubseteq E}{C \sqsubseteq \exists R.E} : \exists R.E \text{ occurs in } \mathcal{O}$$

Table 2: Inference rules for reasoning in $\mathcal{EL}$

$\mathcal{ELO}$ has Tarski-style semantics. An *interpretation* $\mathcal{I}$ consists of a non-empty set $\Delta^{\mathcal{I}}$ called the domain of $\mathcal{I}$ and an interpretation function $\cdot^{\mathcal{I}}$ that assigns to each A a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, to each R a binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and to each a an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. The interpretation function is extended to complex concepts as shown in Table 1.

An interpretation $\mathcal{I}$ *satisfies* an axiom $C \sqsubseteq D$ (written $\mathcal{I} \models C \sqsubseteq D$) if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. If an interpretation $\mathcal{I}$ satisfies all axioms in an ontology $\mathcal{O}$, then $\mathcal{I}$ is a *model* of $\mathcal{O}$ (written $\mathcal{I} \models \mathcal{O}$). An axiom α is a *consequence* of an ontology $\mathcal{O}$ (written $\mathcal{O} \models \alpha$) if every model of $\mathcal{O}$ satisfies α. A concept C is *subsumed* by D w.r.t. $\mathcal{O}$ if $\mathcal{O} \models C \sqsubseteq D$. *Ontology classification* is the task that requires to compute all pairs $\langle A, B\rangle$ of atomic concepts such that $\mathcal{O} \models A \sqsubseteq B$.

The DL $\mathcal{EL}$ is $\mathcal{ELO}$ without nominals. Reasoning in $\mathcal{EL}$ can be performed using the inference rules in Table 2. These rules are closely related to the original completion rules for $\mathcal{EL}^{++}$ (Baader, Brandt, and Lutz 2005), but do not require the ontology to be normalized. Intuitively, the rules are distinguished to those introducing constructors ($\mathsf{R}_{\top}{}^{+}$, $\mathsf{R}_{\sqcap}{}^{+}$, $\mathsf{R}_{\exists}{}^{+}$), eliminating constructors ($\mathsf{R}_{\sqcap}{}^{-}$, $\mathsf{R}_{\exists}{}^{-}$), and using the axioms from the ontology ($\mathsf{R}_{\sqsubseteq}$). Note that the axioms in $\mathcal{O}$ are not used as premises of the inference rules, but as side conditions of $\mathsf{R}_{\sqsubseteq}$. The inference rules in Table 2 are sound in the sense that for every model $\mathcal{I}$ of $\mathcal{O}$, if $\mathcal{I}$ is a model of the premises, then $\mathcal{I}$ is a model of the conclusions. Furthermore, the rules are complete in the following sense:

Theorem 1 (Completeness for $\mathcal{EL}$)**.** *Let $\mathcal{O}$ be an $\mathcal{EL}$ ontology,* $\mathbf{S}$ *a set of axioms closed under the rules in Table 2, and G a concept such that $G \sqsubseteq G \in$* $\mathbf{S}$. *Then for each concept D occurring in $\mathcal{O}$ we have $\mathcal{O} \models G \sqsubseteq D$ implies $G \sqsubseteq D \in$* $\mathbf{S}$.

Theorem 1 follows from completeness of a more general procedure for $\mathcal{ELH}_{\mathcal{R}^+}$ (Kazakov, Krötzsch, and Simančík 2011a), of which the rules in Table 2 are obtained by restricting the language to $\mathcal{EL}$. Intuitively, the theorem says that in order to compute subsumptions between the *goal concept* G and concepts occurring in $\mathcal{O}$, it is sufficient to compute the conclusions of the inference rules from the initial axiom $G \sqsubseteq G$. Because this procedure is not well known, we demonstrate how Theorem 1 can be used for computing subsumption relations in $\mathcal{EL}$ ontologies.

Example 2. Consider $\mathcal{O}$ consisting of the following axioms:

$$A \sqsubseteq \exists R.B, \qquad (1)$$

$$B \sqsubseteq C, \qquad (2)$$

$$\exists R.(B \sqcap C) \sqsubseteq B. \qquad (3)$$

We prove that $\mathcal{O} \models A \sqsubseteq B$ by applying Theorem 1 for the goal concept $G = A$, i.e., by computing the conclusions of the initial axiom $A \sqsubseteq A$ using the rules in Table 2. We write $\mathbf{R}_{\mathbf{X}}(ax_1), \ldots, (ax_n)[:(ax)]$ to denote that an axiom is obtained by applying the rule $\mathbf{R}_{\mathbf{X}}$ to premises $(ax_1), \ldots, (ax_n)$ possibly using an axiom (ax) in $\mathcal{O}$ as a side condition.

$A \sqsubseteq A$	initial axiom	(4)
$A \sqsubseteq \exists R.B$	by $\mathbf{R}_{\sqsubseteq}$ (4): (1)	(5)
$B \sqsubseteq B$	by $\mathbf{R}_{\exists}^{-}$ (5)	(6)
$B \sqsubseteq C$	by $\mathbf{R}_{\sqsubseteq}$ (6): (2)	(7)
$B \sqsubseteq B \sqcap C$	by $\mathbf{R}_{\sqcap}^{+}$ (6), (7)	(8)
$A \sqsubseteq \exists R.(B \sqcap C)$	by $\mathbf{R}_{\exists}^{+}$ (5), (8)	(9)
$A \sqsubseteq B$	by $\mathbf{R}_{\sqsubseteq}$ (9): (3)	(10)

Since the inference rules in Table 2 are sound, from (10), we can conclude that $\mathcal{O} \models A \sqsubseteq B$. Furthermore, since the set $\mathbf{S}$ of axioms (4)–(10) is, in fact, closed under all inference rules in Table 2 and contains the initial axioms $A \sqsubseteq A$ and $B \sqsubseteq B$ for the goal concepts A and B, by Theorem 1, $\mathbf{S}$ contains all and only implied subsumptions between the concepts A and B and the concepts occurring in $\mathcal{O}$. In particular, we can conclude that $\mathcal{O} \not\models A \sqsubseteq C$ and $\mathcal{O} \not\models B \sqsubseteq A$.

As can be seen from Example 2, in order to classify an ontology, it is sufficient to apply Theorem 1 for all atomic concepts A occurring in the ontology as the goal concepts, i.e., to compute all conclusions of the inference rules in Table 2 from the axioms $A \sqsubseteq A$, where A is an atomic concept.

Note that the inferences (8) and (9) use the property that the concepts $B \sqcap C$ and $\exists R.(B \sqcap C)$ occur in $\mathcal{O}$ as side conditions of the rules $\mathbf{R}_{\sqcap}^{+}$ and $\mathbf{R}_{\exists}^{+}$. Even though these side conditions are not required for soundness or completeness, they prevent the rules from deriving unnecessary consequences. For example, from (4) and (10) it is possible to derive $A \sqsubseteq A \sqcap B$, but this axiom is irrelevant since $A \sqcap B$ does not occur in the ontology. Restricting the rules in this way makes the classification procedure polynomial in worst case. Indeed, it can be shown by induction that a consequence $C \sqsubseteq D$ is derived only if both C and D occur in the ontology. Therefore, the maximum number of derived subsumptions is quadratic in the size of the ontology.

Reasoning in $\mathcal{ELO}$

Extending the $\mathcal{EL}$ language with nominals—concepts that are interpreted by singleton sets—provides sufficient functionality for expressing several commonly used constructors and axioms in ontologies, such as concept assertions $a : C$, which can be written as $\{a\} \sqsubseteq C$, role assertions $R(a, b)$, which can be written as $\{a\} \sqsubseteq \exists R.\{b\}$, and OWL constructors such as "ObjectHasValue," which can be written as $\exists R.\{a\}$. However, nominals can also be used to express more sophisticated properties.

Consider the following axiom with a nominal:

$$A \sqsubseteq \exists R.(B \sqcap \{o\}). \qquad (11)$$

This axiom expresses the property that (i) every instance of A is R-connected to the individual o, and (ii) o is an instance of B if A has at least one instance. The property (ii) can be regarded as a *conditional axiom*—an axiom that holds only if some other property holds, e.g., concept A is non-empty.

It is possible to express not only conditional instance axioms, but also conditional subsumption axioms. For example, if we extend (11) with two concept definitions

$$C \equiv \exists S.\{o\} \quad \text{and} \quad D \equiv \exists S.B, \qquad (12)$$

then these axioms would imply that C is subsumed by D if A is non-empty. We will write such *conditional subsumptions* as $A\colon C \sqsubseteq D$ with the semantics $\mathcal{I} \models A\colon C \sqsubseteq D$ if $A^{\mathcal{I}} \neq \emptyset$ implies $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. Thus, (11) is equivalent to $A \sqsubseteq \exists R.\{o\}$ and $A\colon \{o\} \sqsubseteq B$. To distinguish from conditional subsumptions $A\colon C \sqsubseteq D$, we refer to ordinary subsumptions $C \sqsubseteq D$ as *definite subsumptions*. Note that the definite subsumption $C \sqsubseteq D$ implies a conditional subsumption $A\colon C \sqsubseteq D$ for every A, and is equivalent to the conditional subsumptions $C\colon C \sqsubseteq D$ and $\top\colon C \sqsubseteq D$.

It turns out that new definite subsumptions can be derived from conditional subsumptions. Therefore, conditional subsumptions cannot be ignored for classification.

Example 3. Consider $\mathcal{O}$ consisting of the following axioms:

$$A \sqsubseteq \exists R.(B \sqcap \{o\}), \qquad (13)$$

$$A \sqsubseteq \exists S.\{o\}, \qquad (14)$$

$$\exists S.B \sqsubseteq B. \qquad (15)$$

We prove that $\mathcal{O} \models A \sqsubseteq B$. Indeed, as has been shown, (13) implies $A\colon \{o\} \sqsubseteq B$. Therefore, from (14) we obtain $A\colon A \sqsubseteq \exists S.B$, which is equivalent to $A \sqsubseteq \exists S.B$, from which, using (15), we obtain $A \sqsubseteq B$.

The conditional subsumption $A\colon \{o\} \sqsubseteq B$ follows from (11) because non-emptiness of A implies non-emptiness of $B \sqcap \{o\}$, which, in turn, implies $\{o\} \sqsubseteq B$. The same effect can also be caused by axiom $A \sqsubseteq \exists S.\exists R.(B \sqcap \{o\})$, or even by a combination of several axioms, such as $A \sqsubseteq \exists S.\exists R.C$, $C \sqsubseteq D$, and $\exists R.D \sqsubseteq \exists R.(B \sqcap \{o\})$. Therefore, for computing conditional subsumptions, it is necessary to analyze implications between non-emptiness of concepts.

To track implications between non-emptiness of concepts, we introduce a new type of axioms $C \rightsquigarrow D$ called *reachability axioms* with the semantics $\mathcal{I} \models C \rightsquigarrow D$ if $C^{\mathcal{I}} \neq \emptyset$ implies $D^{\mathcal{I}} \neq \emptyset$. Note that $C \rightsquigarrow D$ can be expressed using

$$\mathsf{R}_{\sqsubseteq}\ \frac{G\colon C \sqsubseteq D}{G\colon C \sqsubseteq E} : D \sqsubseteq E \in \mathcal{O}$$

$$\mathsf{R}_{\sqcap}^{-}\ \frac{G\colon C \sqsubseteq D_1 \sqcap D_2}{\begin{array}{c} G\colon C \sqsubseteq D_1 \\ G\colon C \sqsubseteq D_2 \end{array}}$$

$$\mathsf{R}_{\rightsquigarrow}^{+}\ \frac{G \rightsquigarrow C \quad G\colon C \sqsubseteq \exists R.D}{G \rightsquigarrow D}$$

$$\mathsf{R}_{\rightsquigarrow}^{-}\ \frac{G \rightsquigarrow D}{G\colon D \sqsubseteq D}$$

$$\mathsf{R}_{\top}^{+}\ \frac{G \rightsquigarrow C}{G\colon C \sqsubseteq \top} : \top \text{ occurs in } \mathcal{O}$$

$$\mathsf{R}_{\sqcap}^{+}\ \frac{G\colon C \sqsubseteq D_1 \quad G\colon C \sqsubseteq D_2}{G\colon C \sqsubseteq D_1 \sqcap D_2} : D_1 \sqcap D_2 \text{ occurs in } \mathcal{O}$$

$$\mathsf{R}_{\exists}^{+}\ \frac{G\colon C \sqsubseteq \exists R.D \quad G\colon D \sqsubseteq E}{G\colon C \sqsubseteq \exists R.E} : \exists R.E \text{ occurs in } \mathcal{O}$$

$$\mathsf{R}_{\{\}}\ \frac{G\colon C \sqsubseteq \{o\} \quad G\colon D \sqsubseteq \{o\} \quad G \rightsquigarrow C \quad G \rightsquigarrow D}{G\colon C \sqsubseteq D}$$

Table 3: Inference rules for reasoning in $\mathcal{ELO}$

the universal role U as $C \sqsubseteq \exists U.D$. The axiom $C \rightsquigarrow D$ is closely related to the relation $C \rightsquigarrow_R D$ used in the completion rules for $\mathcal{EL}^{++}$ (Baader, Brandt, and Lutz 2005).

We are now ready to explain the inference rules for reasoning in $\mathcal{ELO}$ listed in Table 3. The rules derive conditional subsumptions of the form $G\colon C \sqsubseteq D$ as well as reachability axioms $G \rightsquigarrow D$. Rules $\mathsf{R}_{\sqsubseteq}$, $\mathsf{R}_{\sqcap}^{-}$, $\mathsf{R}_{\sqcap}^{+}$, $\mathsf{R}_{\exists}^{+}$ are analogous to the corresponding rules in Table 2. Rule $\mathsf{R}_{\rightsquigarrow}^{+}$ uses positive existential restrictions to propagate reachability, which can be used in rules $\mathsf{R}_{\rightsquigarrow}^{-}$ and $\mathsf{R}_{\top}^{+}$ to derive the conclusions similar to those of rules $\mathsf{R}_{\exists}^{-}$ and $\mathsf{R}_{\top}^{+}$ in Table 2.

Rule $\mathsf{R}_{\{\}}$ is a new rule for reasoning with nominals. Intuitively, it says that if, under assumption that G is not empty, the concepts C and D are subsumed by the nominal $\{o\}$ and are not empty, then C is equivalent to D: note that the rule is symmetric w.r.t. C and D, so it will, in fact, derive two conclusions $G\colon C \sqsubseteq D$ and $G\colon D \sqsubseteq C$. Note also that the premise $G \rightsquigarrow C$ is not necessary for deriving the conclusion $G\colon C \sqsubseteq D$. The purpose of the additional premise is to avoid irrelevant consequences, similar to the side conditions of the rules $\mathsf{R}_{\sqcap}^{+}$ and $\mathsf{R}_{\exists}^{+}$. It is easy to see that all rules in Table 3 are sound, that is, for every model $\mathcal{I}$ of $\mathcal{O}$, if $\mathcal{I}$ satisfies all premises, then $\mathcal{I}$ satisfies all conclusions. The analogue of Theorem 1 is formulated for $\mathcal{ELO}$ as follows:

Theorem 4 (Completeness for $\mathcal{ELO}$)**.** *Let $\mathcal{O}$ be an $\mathcal{ELO}$ ontology,* $\mathbf{S}$ *a set of axioms closed under the rules in Table 3, and G a concept such that $G \rightsquigarrow G \in$* $\mathbf{S}$ *and $G \rightsquigarrow \{o\} \in$* $\mathbf{S}$ *for every nominal $\{o\}$. Then for each concept D occurring in $\mathcal{O}$ we have $\mathcal{O} \models G \sqsubseteq D$ implies $G\colon G \sqsubseteq D \in$* $\mathbf{S}$.

Proof (sketch). We will construct a model $\mathcal{I} = \mathcal{I}(G)$ of $\mathcal{O}$ such that for every D occurring in $\mathcal{O}$, if $G\colon G \sqsubseteq D \notin \mathbf{S}$ then $\mathcal{I} \not\models G \sqsubseteq D$.

For every concept D, let us define a set of concepts

$$[D] := \{C \mid G \rightsquigarrow C \text{ and } G\colon C \sqsubseteq D \in \mathbf{S}\}. \tag{16}$$

Intuitively, $[D]$ represents the set of concepts reachable from G that are derived sub-concepts of D under the non-emptiness assumption for G.

Let us define the interpretation $\mathcal{I} = \mathcal{I}(G)$ as follows:

$$\Delta^{\mathcal{I}} = \{x_{[D]} \mid G \rightsquigarrow D \in \mathbf{S}\}, \tag{17}$$

$$A^{\mathcal{I}} = \{x_{[D]} \in \Delta^{\mathcal{I}} \mid [D] \subseteq [A]\}, \tag{18}$$

$$R^{\mathcal{I}} = \{\langle x_{[D]}, x_{[E]}\rangle \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid [D] \subseteq [\exists R.E]\}, \tag{19}$$

$$a^{\mathcal{I}} = x_{[\{a\}]}, \tag{20}$$

where $x_{[D]}$ is a distinguished element for each set $[D]$. Note that it is possible that $[D_1] = [D_2]$ for different D_1 and D_2, in which case we shall also have $x_{[D_1]} = x_{[D_2]}$. Note that $x_{[\{a\}]} \in \Delta^{\mathcal{I}}$ since $G \rightsquigarrow \{a\} \in \mathbf{S}$ by our assumption, so $a^{\mathcal{I}}$ is well-defined for every a. Since $\mathbf{S}$ is closed under the rule $\mathsf{R}_{\rightsquigarrow}^{-}$, by (17) and (16) we have

$$x_{[D]} \in \Delta^{\mathcal{I}} \quad \text{implies} \quad D \in [D]. \tag{21}$$

The following properties (22) and (23) can be proved by structural induction on D using the fact that S is closed under the inference rules in Table 3. Full details can be found in the appendix in the accompanying technical report (Kazakov, Krötzsch, and Simančík 2011b).

For every concept D we have

$$D^{\mathcal{I}} \supseteq \{x_{[C]} \in \Delta^{\mathcal{I}} \mid [C] \subseteq [D]\}. \tag{22}$$

In addition, if D occurs in $\mathcal{O}$, we have

$$D^{\mathcal{I}} \subseteq \{x_{[C]} \in \Delta^{\mathcal{I}} \mid [C] \subseteq [D]\}. \tag{23}$$

To prove that $\mathcal{I}$ is a model of $\mathcal{O}$, take any axiom $D \sqsubseteq E \in \mathcal{O}$. Since D and E occur in $\mathcal{O}$, by (22) and (23), we have

$$D^{\mathcal{I}} = \{x_{[C]} \in \Delta^{\mathcal{I}} \mid [C] \subseteq [D]\}, \tag{24}$$

$$E^{\mathcal{I}} = \{x_{[C]} \in \Delta^{\mathcal{I}} \mid [C] \subseteq [E]\}. \tag{25}$$

Therefore, it is sufficient to show that $[D] \subseteq [E]$. Assume that $C \in [D]$. We will prove that $C \in [E]$.

Since $C \in [D]$, by (16), we have $G \rightsquigarrow C$ and $G\colon C \sqsubseteq D \in \mathbf{S}$. Since $D \sqsubseteq E \in \mathcal{O}$ and $\mathbf{S}$ is closed under $\mathsf{R}_{\sqsubseteq}$, we have $G\colon C \sqsubseteq E \in \mathbf{S}$. Therefore, since $G \rightsquigarrow C$, by (16), $C \in [E]$, which was required to be shown.

Finally, it remains to prove that $\mathcal{I} \not\models G \sqsubseteq D$ if D occurs in $\mathcal{O}$ and $G\colon G \sqsubseteq D \notin \mathbf{S}$. Since D occurs in $\mathcal{O}$ (but not necessarily G), by (22) and (23), we have

$$G^{\mathcal{I}} \supseteq \{x_{[C]} \in \Delta^{\mathcal{I}} \mid [C] \subseteq [G]\}, \tag{26}$$

$$D^{\mathcal{I}} = \{x_{[C]} \in \Delta^{\mathcal{I}} \mid [C] \subseteq [D]\}. \tag{27}$$

Since, by assumption of the theorem, $G \rightsquigarrow G \in \mathbf{S}$, by (17) $x_{[G]} \in \Delta^{\mathcal{I}}$, and, since $[G] \subseteq [G]$, by (26), $x_{[G]} \in G^{\mathcal{I}}$.

Assume, to the contrary that $\mathcal{I} \models G \sqsubseteq D$. Then $x_{[G]} \in G^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ thus, by (27), $[G] \subseteq [D]$. Since $x_{[G]} \in \Delta^{\mathcal{I}}$, by (21), $G \in [G]$. Therefore $G \in [D]$, and by (16), $G\colon G \sqsubseteq D \in \mathbf{S}$. This contradicts to the assumption $G\colon G \sqsubseteq D \notin \mathbf{S}$. Therefore, $\mathcal{I} \not\models G \sqsubseteq D$.

Since $\mathcal{I}$ is a model of $\mathcal{O}$, it follows that $\mathcal{O} \not\models G \sqsubseteq D$. □

Example 5. Let us compute the entailed super-concepts of A for ontology $\mathcal{O}$ consisting of axioms (13)–(15) using Theorem 4. By the theorem, it is sufficient to compute the conclusions of the inference rules in Table 3 for the goal $G = A$, i.e., from the axioms $A \rightsquigarrow A$ and $A \rightsquigarrow \{o\}$ in our case.

$$\begin{array}{rll}
A \rightsquigarrow A & \text{initial axiom} & (28)\\
A \rightsquigarrow \{o\} & \text{initial axiom} & (29)\\
A\colon A \sqsubseteq A & \text{by } \mathbf{R}^-_{\rightsquigarrow}\ (28) & (30)\\
A\colon \{o\} \sqsubseteq \{o\} & \text{by } \mathbf{R}^-_{\rightsquigarrow}\ (29) & (31)\\
A\colon A \sqsubseteq \exists R.(B \sqcap \{o\}) & \text{by } \mathbf{R}_{\sqsubseteq}\ (30)\text{: } (13) & (32)\\
A\colon A \sqsubseteq \exists S.\{o\} & \text{by } \mathbf{R}_{\sqsubseteq}\ (30)\text{: } (14) & (33)\\
A \rightsquigarrow B \sqcap \{o\} & \text{by } \mathbf{R}^+_{\rightsquigarrow}\ (28), (32) & (34)\\
A\colon B \sqcap \{o\} \sqsubseteq B \sqcap \{o\} & \text{by } \mathbf{R}^-_{\rightsquigarrow}\ (34) & (35)\\
A\colon B \sqcap \{o\} \sqsubseteq B & \text{by } \mathbf{R}^-_{\sqcap}\ (35) & (36)\\
A\colon B \sqcap \{o\} \sqsubseteq \{o\} & \text{by } \mathbf{R}^-_{\sqcap}\ (35) & (37)\\
A\colon \{o\} \sqsubseteq B \sqcap \{o\} & \text{by } \mathbf{R}_{\{\}}\ (31), (37), (29), (34) & (38)\\
A\colon \{o\} \sqsubseteq B & \text{by } \mathbf{R}^-_{\sqcap}\ (38) & (39)\\
A\colon A \sqsubseteq \exists S.B & \text{by } \mathbf{R}^+_{\exists}\ (33), (39) & (40)\\
A\colon A \sqsubseteq B & \text{by } \mathbf{R}_{\sqsubseteq}\ (40)\text{: } (15) & (41)
\end{array}$$

Since axioms (28) and (29) are satisfied in every model and the inference rules are sound, all computed axioms are entailed by $\mathcal{O}$. Therefore, from (30), (40), and (41), we obtain $\mathcal{O} \models A \sqsubseteq A$, $\mathcal{O} \models A \sqsubseteq \exists S.B$, and $\mathcal{O} \models A \sqsubseteq B$. Since the computed set of axioms (28)–(41) is closed under the rules in Table 3, by Theorem 4, we conclude that A, $\exists S.B$, and B are the only entailed super-concepts of A occurring in $\mathcal{O}$.

In order to classify an $\mathcal{ELO}$ ontology $\mathcal{O}$, it is sufficient to apply Theorem 4 for every atomic concept in $\mathcal{O}$ as the goal, i.e., to compute the closure under the rules in Table 3 of the axioms $A \rightsquigarrow A$ and $A \rightsquigarrow \{o\}$ for every atomic concept A and nominal $\{o\}$ occurring in $\mathcal{O}$. It is easy to see that only axioms of the form $A \rightsquigarrow C$ and $A\colon C \sqsubseteq D$ with A, C, and D occurring in $\mathcal{O}$, can be derived by the inference rules. Therefore, the number of derived axioms is at most cubic in the size of $\mathcal{O}$.

Remark 6. The original procedure for $\mathcal{EL}^{++}$ (Baader, Brandt, and Lutz 2005) was formulated with much simpler rules for reasoning with nominals. In particular, the rules derive only definite subsumptions, like those in Table 2, and the analogue of $\mathbf{R}_{\{\}}$ was formulated as follows:

$$\frac{C \sqsubseteq \{o\} \quad D \sqsubseteq \{o\} \quad C \rightsquigarrow D}{C \sqsubseteq D} \qquad (42)$$

Although rule (42) is sound, this procedure is not complete for nominals. In particular, it is not possible to prove the subsumption $A \sqsubseteq B$ in Example 5. It was recently argued that under quite general assumptions, every complete deterministic rule-based procedure for $\mathcal{ELO}$ must derive at least cubically many axioms (Krötzsch 2011). Therefore, any procedure deriving just definite subsumptions $C \sqsubseteq D$ and reachability axioms $C \rightsquigarrow D$ (with C and D occurring in the ontology) would be incomplete.

Axiom Reuse

Although the classification procedure based on the rules in Table 3 is tractable, a direct implementation of this procedure would be impractical. For example, if an ontology contains a large number of atomic concepts and nominals, then already the number of initialization axioms $A \rightsquigarrow \{a\}$ can be quadratic. Algorithms that are quadratic in a typical case, rather than in the worst case, are usually considered to be impractical. Even when the ontology contains a small number of nominals, or no nominals at all, the procedure can be impractical due to a large number of conclusions produced.

To demonstrate the problem, consider the ontology $\mathcal{O}$ in Example 5 extended with one additional axiom

$$C \sqsubseteq \exists R.A. \qquad (43)$$

In order to classify this ontology, we have to compute, in particular, the conclusions for the goals A and C under the inference rules in Table 3. As demonstrated in Example 5, for A we obtain the conclusions (28)–(41). Similarly, for C we derive:

$$\begin{array}{rll}
C \rightsquigarrow C & \text{initial axiom} & (44)\\
C \rightsquigarrow \{o\} & \text{initial axiom} & (45)\\
C\colon C \sqsubseteq C & \text{by } \mathbf{R}^-_{\rightsquigarrow}\ (44) & (46)\\
C\colon C \sqsubseteq \exists R.A & \text{by } \mathbf{R}_{\sqsubseteq}\ (46)\text{: } (43) & (47)\\
C \rightsquigarrow A & \text{by } \mathbf{R}^+_{\rightsquigarrow}\ (44), (47) & (48)
\end{array}$$

It is easy to see that for every axiom $A \rightsquigarrow D$ and $A\colon D \sqsubseteq E$ in (28)–(41), we would also derive $C \rightsquigarrow D$ and $C\colon D \sqsubseteq E$ because of (48). That is, whenever one goal G_1 is reachable from another goal G_2 (i.e., $G_2 \rightsquigarrow G_1$ is derivable), the inferences computed for G_1 would always have to be repeated for G_2 as well. Essentially, the conclusions computed for a goal G_1 can never be reused for another goal G_2. This is the case even when the ontology contains no nominals.

The procedure for $\mathcal{EL}$, on the other hand, does not have this drawback. To compare the two procedures, let us compute the conclusions produced by the rules in Table 2 for the goals A and C (although the result will be incomplete in our case). For A, we obtain the following conclusions:

$$\begin{array}{rll}
A \sqsubseteq A & \text{initial axiom} & (49)\\
A \sqsubseteq \exists R.(B \sqcap \{o\}) & \text{by } \mathbf{R}_{\sqsubseteq}\ (49)\text{: } (13) & (50)\\
A \sqsubseteq \exists S.\{o\} & \text{by } \mathbf{R}_{\sqsubseteq}\ (49)\text{: } (14) & (51)\\
B \sqcap \{o\} \sqsubseteq B \sqcap \{o\} & \text{by } \mathbf{R}^-_{\exists}\ (50) & (52)\\
\{o\} \sqsubseteq \{o\} & \text{by } \mathbf{R}^-_{\exists}\ (51) & (53)\\
B \sqcap \{o\} \sqsubseteq B & \text{by } \mathbf{R}^-_{\sqcap}\ (52) & (54)\\
B \sqcap \{o\} \sqsubseteq \{o\} & \text{by } \mathbf{R}^-_{\sqcap}\ (52) & (55)
\end{array}$$

For C, in addition, we obtain the following conclusions:

$$\begin{array}{rll}
C \sqsubseteq C & \text{initial axiom} & (56)\\
C \sqsubseteq \exists R.A & \text{by } \mathbf{R}_{\sqsubseteq}\ (56)\text{: } (43) & (57)
\end{array}$$

Note that rule $\mathbf{R}^-_{\exists}$ can be applied to (57), but unlike (48), it produces an axiom (49), which has been already derived.

Therefore, all further inferences for C are not necessary because all conclusions have already been computed for A.

The ability to share derived consequences for different goals is one of the distinguished properties of the $\mathcal{EL}$-style reasoning procedures, which makes them able to classify complex ontologies, such as Galen, that could not be classified using conventional tableau procedures (Kazakov 2009). It is, therefore, essential to retain this property for $\mathcal{ELO}$.

Recall that any definite subsumption $D \sqsubseteq E$ is stronger than the conditional subsumption $G\colon D \sqsubseteq E$. Therefore, the conditional subsumptions (30)–(33), (35)–(37), (46), and (47) become redundant once definite subsumptions (49)–(57) are derived. Unlike conditional subsumptions, definite subsumptions can be shared among different goals.

This observation suggests our first optimization. We compute the classification in two stages. In Stage 1, the $\mathcal{EL}$ rules are applied to compute definite subsumptions. In Stage 2, we apply a modified version of the $\mathcal{ELO}$ rules that can use any definite subsumption $D \sqsubseteq E$ as if it were $G\colon D \sqsubseteq E$ for arbitrary goal G. Stage 2 does, however, not consider rules where *all* premises are of a form obtained in Stage 1, as these would clearly be redundant. Moreover, it is not necessary to store conclusions $G\colon D \sqsubseteq E$ for which $D \sqsubseteq E$ was already derived.

Using this optimization, it is possible to share derived axioms among several goals. The optimized procedure exhibits a so-called "pay-as-you-go" behavior w.r.t. nominals: if there are no nominals in the ontology, no conditional subsumption will be derived, and the procedure will work almost exactly as for $\mathcal{EL}$, apart from deriving reachability axioms. Even if the ontology contains a small number of axioms with nominals, the number of derived conditional subsumptions is likely to be small as well.

We can further extend this approach to also address the problem caused by a large number of nominals. As explained above, the number of initialization axioms of the form $G \rightsquigarrow \{o\}$ in Stage 2 can be very large. To reduce this number, we extend Stage 1 to also apply the rules of Table 3 to the initial axioms $\top \rightsquigarrow \top$ and $\top \rightsquigarrow \{o\}$ for every nominal $\{o\}$. Note, that $\mathcal{I} \models \top \rightsquigarrow C$ iff $C^{\mathcal{I}} \neq \emptyset$ and $\mathcal{I} \models \top\colon C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, so, this approach essentially produces further definite non-emptiness and subsumption axioms. The additional axioms $\top \rightsquigarrow C$ and $\top\colon C \sqsubseteq D$ can serve the same purpose as the definite subsumptions computed by the $\mathcal{EL}$ rules, i.e., they can implicitly represent the corresponding reachability axioms $G \rightsquigarrow C$ and conditional subsumptions $G\colon C \sqsubseteq D$ for every goal G. In particular, it is not necessary to create the axioms of the form $G \rightsquigarrow \{o\}$ since stronger axioms $\top \rightsquigarrow \{o\}$ are provided by Stage 1.

Pruning of Reachability Axioms

Reusing axioms can significantly reduce the number of derived conditional subsumptions, but the number of reachability axioms $G \rightsquigarrow C$ that are computed in Stage 2 can still be very large. This is particularly problematic for ontologies containing many cyclic axioms. For example, ontologies Galen and FMA use cyclic axioms to express partonomy relationships between anatomic structures, such as "the myocardium is a muscle that is a part of the heart":

$$\mathit{Myocardium} \equiv \mathit{Muscle} \sqcap \exists \mathit{isPartOf}.\mathit{Heart}, \tag{58}$$

$$\mathit{Heart} \sqsubseteq \exists \mathit{hasPart}.\mathit{Myocardium}. \tag{59}$$

From (58), we can derive $\mathit{Myocardium} \rightsquigarrow \mathit{Heart}$, and from (59), we can derive $\mathit{Heart} \rightsquigarrow \mathit{Myocardium}$. Because of the large number of axioms, such as (58) and (59), and the fact that from every anatomic structure one can, in theory, reach any other anatomic structure through a chain of "*isPartOf*" or "*hasPart*" relations, there are almost quadratically many reachability axioms $C \rightsquigarrow D$ in Galen and FMA.

Cyclic existential axioms, such as (58) and (59) are likely to result in cyclic reachability relations. A large component of mutually reachable concepts can easily cause a quadratic blowup in the number of reachability axioms. On the other hand, all reachable concepts and conditional subsumptions for elements of the same component are the same because all concepts in such component are non-empty if one of them is.

This observation suggests our second optimization. After completing Stage 1, we build a directed graph containing an edge $\langle C, D\rangle$ for each derived axiom $C \sqsubseteq \exists R.D$, and compute all strongly components in this graph in linear time (Tarjan 1972). For each two concepts C and D in a strongly connected component, we have $\mathcal{O} \models C \rightsquigarrow D$ and $\mathcal{O} \models D \rightsquigarrow C$. Therefore, we can choose one representative of each component as the goal for Stage 2; the computed reachability axioms and conditional subsumptions can then be reused for all other elements of the same component.

This strategy can be optimized even further by recording, for every derived (conditional) subsumption $C \sqsubseteq \exists R.D$ and $G\colon C \sqsubseteq \exists R.D$, a *(conditional) connection* $C' \rightarrow D'$ and $G\colon C' \rightarrow D'$ between representatives C' and D' of the components for C and D. These connections can be used instead of the original subsumptions in rule $\mathbf{R}^{+}_{\rightsquigarrow}$. This way, we reduce the number of applications of this rule because there could be many existential axioms $C \sqsubseteq \exists R.D$ and $G\colon C \sqsubseteq \exists R.D$ with the same representatives C' and D'.

Optimized Reasoning with Overestimation

For Galen, computing reachability axioms is not necessary since this ontology does not contain any nominals. But even in ontologies containing nominals, computing reachability for a goal concept G is necessary only if for some concept D, subsumption $G \sqsubseteq D$ is not derived by the $\mathcal{EL}$ rules, but $G\colon G \sqsubseteq D$ can be derived by the $\mathcal{ELO}$ rules. But how can we check if $G\colon G \sqsubseteq D$ can be derived by the $\mathcal{ELO}$ rules without actually computing the reachability axioms for G?

The main idea behind our third optimization is to *overestimate* the entailed subsumption relations in $\mathcal{ELO}$. We will call such axioms *potential subsumptions* and denote them by $?\colon C \sqsubseteq D$. The inference rules for deriving potential subsumptions are presented in Table 4. All rules but $\mathbf{R}_{\{\}}$ are identical to the $\mathcal{EL}$ rules in Table 2, except that they operate with potential subsumptions instead of definite subsumptions. Clearly, rule $\mathbf{R}_{\{\}}$, if it were formulated for definite subsumptions, would be unsound. This rule can be seen as a weakened version of rule $\mathbf{R}_{\{\}}$ in Table 3 if we delete all reachability axioms in the premises and replace conditional subsumptions with the respective potential subsumptions.

$$\mathsf{R}_{\sqsubseteq}\ \frac{?\colon C \sqsubseteq D}{?\colon C \sqsubseteq E} : D \sqsubseteq E \in \mathcal{O}$$

$$\mathsf{R}_{\sqcap}^{-}\ \frac{?\colon C \sqsubseteq D_1 \sqcap D_2}{\begin{array}{l}?\colon C \sqsubseteq D_1\\ ?\colon C \sqsubseteq D_2\end{array}}$$

$$\mathsf{R}_{\exists}^{-}\ \frac{?\colon C \sqsubseteq \exists R.D}{?\colon D \sqsubseteq D}$$

$$\mathsf{R}_{\top}^{+}\ \frac{?\colon C \sqsubseteq C}{?\colon C \sqsubseteq \top} : \top \text{ occurs in } \mathcal{O}$$

$$\mathsf{R}_{\sqcap}^{+}\ \frac{?\colon C \sqsubseteq D_1 \quad ?\colon C \sqsubseteq D_2}{?\colon C \sqsubseteq D_1 \sqcap D_2} : D_1 \sqcap D_2 \text{ occurs in } \mathcal{O}$$

$$\mathsf{R}_{\exists}^{+}\ \frac{?\colon C \sqsubseteq \exists R.D \quad ?\colon D \sqsubseteq E}{?\colon C \sqsubseteq \exists R.E} : \exists R.E \text{ occurs in } \mathcal{O}$$

$$\mathsf{R}_{\{\}}\ \frac{?\colon C \sqsubseteq \{o\} \quad ?\colon D \sqsubseteq \{o\}}{?\colon C \sqsubseteq D}$$

Table 4: The overestimation inference rules for $\mathcal{ELO}$

The main purpose of the rules in Table 4 is to provide an efficient way of checking if the axioms derived by the $\mathcal{EL}$ rules are already all subsumptions entailed in $\mathcal{ELO}$: if the definite subsumptions derived by the underestimation rules in Table 2 coincide with the potential subsumptions derived by the overestimation rules in Table 4, we know that all the relevant entailed subsumptions are computed. The correctness of this method follows from the following theorem:

Theorem 7 (Overestimation). *Let $\mathcal{O}$ be an $\mathcal{ELO}$ ontology, $\mathbf{S}$ a set of axioms closed under the rules in Table 4, and G a concept such that $?\colon G \sqsubseteq G \in \mathbf{S}$, and $?\colon \{o\} \sqsubseteq \{o\} \in \mathbf{S}$ for every nominal $\{o\}$. Then for each concept D occurring in $\mathcal{O}$, we have $\mathcal{O} \models G \sqsubseteq D$ implies $?\colon G \sqsubseteq D \in \mathbf{S}$.*

Proof. Given a set $\mathbf{S}$ and a concept G satisfying the condition of the theorem, define

$$\begin{aligned}\mathbf{S}' := {} & \{G \rightsquigarrow C \mid ?\colon C \sqsubseteq C \in \mathbf{S}\} \cup \\ & \{G\colon C \sqsubseteq D \mid ?\colon C \sqsubseteq D \in \mathbf{S}\}.\end{aligned} \tag{60}$$

We prove that $\mathbf{S}'$ satisfies the condition of Theorem 4, from which it follows that $\mathcal{O} \models G \sqsubseteq D$ implies $G\colon G \sqsubseteq D \in \mathbf{S}'$, which by (60) implies $?\colon G \sqsubseteq D \in \mathbf{S}$.

Indeed, since $?\colon G \sqsubseteq G \in \mathbf{S}$, by (60), $G \rightsquigarrow G \in \mathbf{S}'$, and for every nominal $\{o\}$, since $?\colon \{o\} \sqsubseteq \{o\} \in \mathbf{S}$, by (60), $G \rightsquigarrow \{o\} \in \mathbf{S}'$. Furthermore, $\mathbf{S}'$ is closed under the inference rules in Table 3. For all rules except for $\mathsf{R}_{\rightsquigarrow}^{+}$ and $\mathsf{R}_{\rightsquigarrow}^{-}$ this follows from the fact that $\mathbf{S}$ is closed under the corresponding rules in Table 4. For rule $\mathsf{R}_{\rightsquigarrow}^{+}$, if $G \rightsquigarrow C \in \mathbf{S}'$ and $G\colon C \sqsubseteq \exists R.D \in \mathbf{S}'$, then, by (60), $?\colon C \sqsubseteq \exists R.D \in \mathbf{S}$; then, since $\mathbf{S}$ is closed under $\mathsf{R}_{\exists}^{-}$ in Table 4, $?\colon D \sqsubseteq D \in \mathbf{S}$, so, by (60), $G \rightsquigarrow D \in \mathbf{S}'$. For rule $\mathsf{R}_{\rightsquigarrow}^{-}$, if $G \rightsquigarrow D \in \mathbf{S}'$, then, by (60), $?\colon D \sqsubseteq D \in \mathbf{S}$, so, again by (60), $G\colon D \sqsubseteq D \in \mathbf{S}'$. □

The optimized reasoning procedure for $\mathcal{ELO}$ can now be described as follows. Given an $\mathcal{ELO}$ ontology $\mathcal{O}$ and a goal concept G, the procedure works in two stages. Stage 1 is an extension of the first stage in the axiom reusing algorithm above, i.e., it applies the $\mathcal{EL}$ rules in Table 2 (with initial axiom $G \sqsubseteq G$) and the $\mathcal{ELO}$ rules in Table 3 for the goal $\top$ (with initial axioms $\top \rightsquigarrow \top$ and $\top \rightsquigarrow \{o\}$ for every nominal $\{o\}$). In addition, Stage 1 applies the rules in Table 4 using the initial axioms $?\colon G \sqsubseteq G$ and $?\colon \{o\} \sqsubseteq \{o\}$ for every nominal $\{o\}$. After that, we check if there is an (atomic) concept D such that a potential subsumption $?\colon G \sqsubseteq D$ is derived, but the corresponding definite subsumption $G \sqsubseteq D$ (or, possibly, $\top\colon G \sqsubseteq D$) is not derived. If no such D exists, we know that we have computed all entailed (atomic) superconcepts of G occurring in $\mathcal{O}$. Indeed, if $G \sqsubseteq D$ is derived, then $\mathcal{O} \models G \sqsubseteq D$. Conversely, if $\mathcal{O} \models G \sqsubseteq D$, then by Theorem 7, $?\colon G \sqsubseteq D$ is derived, in which case we know that $G \sqsubseteq D$ is derived as well.

If we have found some D such that $?\colon G \sqsubseteq D$ is derived but $G \sqsubseteq D$ is not derived, then Stage 2 is necessary for G in order to determine whether $\mathcal{O} \models G \sqsubseteq D$. In this case, we apply the $\mathcal{ELO}$ rules in Table 3 for the initial axiom $G \rightsquigarrow G$, reusing the definite axioms from Stage 1 as before. $\mathcal{O} \models G \sqsubseteq D$ holds exactly if $G\colon G \sqsubseteq D$ is derived.

In practice, we do not compute the overestimation axioms independently from the definite axioms. Instead, in the same way as for the conditional subsumptions, we reuse every definite subsumption $C \sqsubseteq D$ and $\top\colon C \sqsubseteq D$ as potential subsumption $?\colon C \sqsubseteq D$, and apply the rules accordingly.

Example 8. Let us demonstrate how to compute the entailed super-concepts of A for ontology $\mathcal{O}$ in Example 5 using our optimized procedure. By applying the $\mathcal{EL}$ rules for the goal $G = A$, we derive definite subsumptions (49)–(55). Applying the $\mathcal{ELO}$ rules to the goal $G = \top$, we derive two reachability axioms and no new subsumptions:

$$\top \rightsquigarrow \top \qquad \text{initial axiom} \tag{61}$$

$$\top \rightsquigarrow \{o\} \qquad \text{initial axiom} \tag{62}$$

By reusing definite subsumptions (49)–(55) as potential subsumptions, we additionally derive the following potential non-definite subsumptions for A using the rules in Table 4:

$$?\colon \{o\} \sqsubseteq B \sqcap \{o\} \qquad \text{by } \mathsf{R}_{\{\}} \text{ (53), (55)} \tag{63}$$

$$?\colon \{o\} \sqsubseteq B \qquad \text{by } \mathsf{R}_{\sqcap}^{-} \text{ (63)} \tag{64}$$

$$?\colon A \sqsubseteq \exists S.B \qquad \text{by } \mathsf{R}_{\exists}^{+} \text{ (51), (64)} \tag{65}$$

$$?\colon A \sqsubseteq B \qquad \text{by } \mathsf{R}_{\sqsubseteq} \text{ (65): (15)} \tag{66}$$

Note that the first potential non-definite subsumption can only be derived by the rule $\mathsf{R}_{\{\}}$. Since $?\colon A \sqsubseteq B$ has been derived, but $A \sqsubseteq B$ has not been derived, we have to apply Stage 2 for A. To this end, we derive the following reachability axioms and conditional non-definite subsumptions using the rules in Table 3, again, reusing definite reachability and subsumption axioms as conditional ones for A:

$$A \rightsquigarrow A \qquad \text{initial axiom} \tag{67}$$

$$A \rightsquigarrow B \sqcap \{o\} \qquad \text{by } \mathsf{R}_{\rightsquigarrow}^{+} \text{ (67), (50)} \tag{68}$$

	SNOMED	Galen	FMA
atomic concepts	315,491	23,136	41,646
concepts	544,055	50,259	82,036
roles	58	950	86
nominals	0	0	85
axioms	430,844	36,547	116,111

Table 5: Ontology metrics

$$A\colon \{o\} \sqsubseteq B \sqcap \{o\} \qquad \text{by } \mathsf{R}_{\{\}} \text{ (53), (55), (62), (68)} \tag{69}$$

$$A\colon \{o\} \sqsubseteq B \qquad \text{by } \mathsf{R}^{-}_{\sqcap} \text{ (69)} \tag{70}$$

$$A\colon A \sqsubseteq \exists S.B \qquad \text{by } \mathsf{R}^{+}_{\exists} \text{ (51), (70)} \tag{71}$$

$$A\colon A \sqsubseteq B \qquad \text{by } \mathsf{R}_{\sqsubseteq} \text{ (71): (15)} \tag{72}$$

Since the computed set of axioms is closed under the $\mathcal{ELO}$ rules, from (49), (71), and (72), we conclude that A, $\exists S.B$, and B are the only super-concepts of A occurring in $\mathcal{O}$.

Let us now look what happens if we, additionally, have an axiom (43) in $\mathcal{O}$, and are required to compute super-concepts of C. As we have demonstrated, (56) and (57) are the only additional definite subsumptions derived by the $\mathcal{EL}$ rules for C. The rules in Table 4 will not derive any new potential subsumptions since rule $\mathsf{R}_{\{\}}$ is not applicable to (56) or (57). Therefore, Stage 2 is not necessary for C. Thus, C and $\exists R.A$ are the only super-concepts of C occurring in $\mathcal{O}$.

As demonstrated in Example 8, the use of the overestimation rules in Table 4 in conjunction with underestimation rules in Table 2 provides an effective filter that can prevent deriving many conditional subsumptions and reachability axioms. Of course, this filter is not perfect, and it may well happen that a potential subsumption is derived that is not confirmed in Stage 2.

Experimental Results

We have implemented the two stage classification procedure from the previous section in our OWL EL reasoner ELK, and conducted a series of experiments on realistic ontologies to analyze the performance improvement given by each optimization. The implementation in ELK covers additional features that are not the focus of this paper, in particular transitive roles and role hierarchies (Kazakov, Krötzsch, and Simančík 2011a). This improves our coverage of realistic test ontologies without affecting the validity of our experiments. Since no other $\mathcal{EL}$ reasoner supports nominals fully, we do not compare the performance of ELK against other reasoners here. All experiments were performed on a laptop with Intel Core i7-2630QM 2GHz quad core CPU and 6GB of RAM running Java 1.6 under Microsoft Windows 7.

None of the existing ontologies that are commonly used for testing $\mathcal{EL}$ reasoners, including SNOMED CT, Galen, FMA-lite, and GO, contain nominals. In order to be able to experiment with at least one large ontology that contains nominals explicitly, we considered FMA-Constitutional, the largest ontology containing nominals that was used in the evaluation of the HermiT reasoner (Motik, Shearer, and Horrocks 2009), and reduced it to $\mathcal{ELO}$ by discarding all axioms with unsupported features. This way we obtained an ontology that contains 85 nominals occurring in 6,455 axioms.

	SNOMED	Galen	FMA
Stage 1:			
rules	22,082,002	2,043,182	1,527,174
axioms	14,091,757	1,447,049	1,343,746
runtime	7.6 s	1.2 s	1.3 s
Stage 2:			
rules	24,716,789	969,212,770	> 5 billion
axioms	5,780,349	38,042,481	> 400 million
runtime	4.5 s	118.2 s	> 25 min

Table 6: Experiments for axiom reuse

Our basic ontology test suite consists of SNOMED CT,[1] an OWL EL version of Galen,[2] and FMA-Constitutional reduced to $\mathcal{ELO}$. Table 5 contains some statistics about these ontologies. The reason for including ontologies without nominals was to evaluate the effect of computing reachability axioms on the performance of the algorithm without the overestimation optimization. For experiments with overestimation, we constructed further ontologies by introducing nominals into Galen and SNOMED CT as described below.

Axiom Reuse

Our first series of experiments evaluates the performance of the basic classification algorithm in Table 3 with the axiom reuse optimization, but without overestimation.

The results are shown in Table 6. For each of the two stages, we measure the number of rule applications, the number of derived axioms, and the running time. Different rule applications may lead to the same inferences, hence the number of rules is always above the number of derived axioms. Rule applications require significant computational effort, whether or not the inference is actually redundant or not, hence their number is often a better measure of performance than the number of unique axioms. In all cases, the only rule applied in Stage 2 was rule $\mathsf{R}^{+}_{\rightsquigarrow}$ from Table 3, and thus all newly derived axioms are reachability statements. This is is clear for SNOMED CT and Galen due to the absence of nominals, while it is an interesting observation for FMA. For the case of FMA, Stage 2 ran out of memory after 25 minutes, and the reported number of rules and axioms reflects the state at that time.

The results show that, for SNOMED CT, materializing reachability in Stage 2 requires similar amount of computation effort as applying the $\mathcal{EL}$ rules in the first stage. This is so since the reachability relation is acyclic in this ontology. This contrasts sharply to what happens for Galen and FMA, where reachability is highly cyclic and the second stage can require up to four orders of magnitude more inferences than the first stage. This confirms our hypothesis that axiom reuse alone does not provide reliable performance even in cases where nominals are not leading to new conclusions.

[1] from http://ihtsdo.org/ (needs registration)

[2] from http://condor-reasoner.googlecode.com/

	SNOMED	Galen	FMA
largest comp.	1	2,691	15,855
#components	315,491	19,957	25,203
#singletons	315,491	19,789	25,047
$\mathsf{R}^{+}_{\rightsquigarrow}$ rules	22,638,567	16,381,638	272,000,623
$\rightsquigarrow$ axioms	5,780,349	3,552,962	9,141,307

Table 7: Experiments for pruning reachability axioms

	Galen-n1	Galen-n2
nominals	739	1,113
potential subsumptions	1,407	54,424
confirmed subsumptions	357	129
goals for Stage 2	62	1,397
goals with new subsumptions	56	73
Stage 1:		
rules	2,105,091	3,114,416
axioms	1,460,923	1,814,528
runtime	1.7 s	1.9 s
Stage 2:		
rules	61,891	8,887,440
axioms	40,950	5,483,853
runtime	0.2 s	9.6 s

Table 8: Experiments for overestimation with axiom reuse

Pruning of Reachability Axioms

In this experiment, we evaluate the potential for optimizing Stage 2 using components of mutually reachable concepts, as explained in the corresponding section. Statistics about the strongly connected components obtained from Stage 1 are shown in Table 7. Although both Galen and FMA contain one very large component, the majority of concepts are still found in singleton components. We observed that, for both Galen and FMA, the size of the second largest component already drops under 20. Due to the large number of components, the number of goals for which Stage 2 is required is not reduced significantly in any of the cases.

The second part of Table 7 shows the effort of computing reachability axioms between representatives of the computed components. The result can be compared to Stage 2 in Table 6, which also computed nothing but reachability axioms. Although there is a significant reduction of effort for Galen and FMA, the numbers are still significantly larger than those of Stage 1. Note that, in our case, the number of components cannot be reduced any further since Stage 2 does not produce any new subsumptions, and therefore all reachability components are computed exactly after Stage 1.

Reasoning with Overestimation

In this experiment, we evaluate the benefits of using the overestimation rules in Table 4 to reduce the number of inferences in Stage 2. Stage 1 is as described in the corresponding section: we reuse definite axioms computed by the $\mathcal{EL}$ rules in Table 2 and $\mathcal{ELO}$ rules in Table 3 for the goal $\top$ when computing potential axioms using the rules in Table 4.

As long as all potential subsumptions are definite, the algorithm does not perform any computations beyond the basic $\mathcal{EL}$ approach. This happens for SNOMED CT and Galen (which do not have nominals), but also for FMA. The data for Stage 1 is thus as in Table 6, and Stage 2 is not needed.

To obtain more interesting results, we tried to construct realistic test ontologies by introducing nominals into SNOMED CT and Galen. Both ontologies contain several hundreds of concepts that are used as values for roles rather than as classes of objects, e.g., *maleSex*, *blue*, and even *sixteen*. These are good candidates for concepts that should perhaps have been modeled as nominals.

An online tutorial at OpenGALEN.org explains that, in Galen, all such “value types” are subsumed by the built-in concept *SymbolicValueType*, and, as a convention to distinguish them from the rest of the ontology, their names start with a lower case letter (OpenGalen.org 2011). In SNOMED CT, the concept *QualifierValue* plays a similar role to that of *SymbolicValueType* in Galen (Rogers 2011).

Based on the hints in the OpenGALEN tutorial, we thus constructed two variants of the Galen ontology. For *Galen-n1*, we identified all atomic sub-concepts of *SymbolicValueType* that do not have other atomic sub-concepts, i.e., which are *leaf concepts*. This yielded 739 concepts that we replaced by nominals. For *Galen-n2*, we replaced all atomic concepts with names starting in lower case by nominals. This produced a different set of 1,113 nominals, including 244 that were not leaf concepts. The ontology *SNOMED-n* was constructed from SNOMED CT by replacing all leaf atomic sub-concepts of *QualifierValue* by nominals. This produced 7,379 nominals.

The experiments showed that SNOMED-n does not require Stage 2 to be run, with Stage 1 leading to similar numbers as in Table 6. In Table 8 we thus only report the results for Galen-n1 and Galen-n2. The number of potential subsumptions refers to the subsumptions that are potential but not definite. The figures show that Galen-n2 is more challenging than Galen-n1. Indeed, non-leaf nominals can cause difficulties to our algorithm since the overestimation rule $\mathsf{R}_{\{\}}$ alone will derive quadratically many potential equivalences between all atomic sub-concepts of a nominal. Nonetheless, the overestimation technique is still able to detect that the second stage is needed only for 1,397 concepts, which is significantly less than the total number of 23,136 atomic concepts that are considered in Stage 2 of the basic axiom reuse algorithm. This reduction translates into significant performance gains in Stage 2.

When we inspected the axioms that were confirmed in the second stage for Galen-n1, we found many undesired subsumptions such as $\mathit{Adult} \sqsubseteq \mathit{Baby}$ and $\mathit{RetiredPerson} \sqsubseteq \mathit{Embryo}$. Further tests showed that all additional subsumptions produced in Stage 2 were due to the nominal status of the single concept *AgeState*. Not considering *AgeState* as a nominal leads to an ontology for which Stage 2 was not needed. This shows that even a single modeling error can have wide-reaching consequences. Similarly, the additional conclusions obtained in Stage 2 for Galen-n2 did rarely correspond to desirable subsumptions. Even the $\mathcal{ELO}$ rules applied in Stage 1 for $\top$ inferred many nominals to be equal (yielding a total of 8,432 equivalence axioms between nominals). In this case, however, no small set of concepts appears to be responsible for the additional conclusions.

Clearly, neither variant of Galen leads to a correct ontological model. In fact, the additional conclusions in Stage 2 indicate inappropriate use of nominals in almost all cases (see the discussion of *safe* uses of nominals below). However, ontology reasoners are a primary tool for detecting modeling errors at design time, and they must therefore yield reliable performance in such cases. The two variants of Galen provide interesting realistic "stress tests" that simulate a varying number of plausible modeling errors. Our results confirm that ELK can handle this challenge.

Safe use of nominals

We have observed in our experiments that for a large number of tested ontologies all entailed subsumptions are already computed by the $\mathcal{EL}$ rules in the first stage of our procedure. It would be interesting to explain this effect and define a fragment of $\mathcal{ELO}$ for which it is always the case.

Notice, from Example 3, that for deriving the subsumption $A \sqsubseteq B$ it is essential that nominal $\{o\}$ occurs in a conjunction of (13). We have not observed this to happen very often in our tested ontologies; the existing nominals mainly occur under existential restrictions, such as in axiom (14).

We say that an $\mathcal{ELO}$ concept C is *safe (for nominals)*, if every nominal $\{o\}$ occurs in C only in the form $\exists R.\{o\}$. In other words, safe concepts can be defined by the grammar

$$C_s = A \mid \exists R.\{o\} \mid \top \mid C_s \sqcap C_s \mid \exists R.C_s. \qquad (73)$$

Safe concepts are essentially $\mathcal{EL}$ concepts extended with the OWL 2 ObjectHasValue constructor. To capture concept assertions $\{a\} \sqsubseteq C$ and role assertions $\{a\} \sqsubseteq \exists R.\{b\}$, we also allow (non-safe) nominals $\{a\}$ to appear on the left-hand-side of concept inclusions. We say that an $\mathcal{ELO}$ concept C is *negatively safe (for nominals)* (short *n-safe*) if C is either a nominal or a safe concept. We demonstrate that the $\mathcal{EL}$ procedure is already sufficient for $\mathcal{ELO}$ ontologies containing axioms $C \sqsubseteq D$ where C is n-safe and D is safe:

Theorem 9. *Let $\mathcal{O}$ be an $\mathcal{ELO}$ ontology containing only axioms $C \sqsubseteq D$ such that C is n-safe and D is safe. Let G be an n-safe concept, and $\mathbf{S}$ a set of axioms closed under the rules in Table 2 such that $G \sqsubseteq G \in \mathbf{S}$ and $\{o\} \sqsubseteq \{o\} \in \mathbf{S}$ for every nominal $\{o\}$. Then for every concept D occurring in $\mathcal{O}$, if $\mathcal{O} \models G \sqsubseteq D$, then $G \sqsubseteq D \in \mathbf{S}$.*

Proof. Let $\mathbf{S}'$ be the set of axioms derivable from $?\colon G \sqsubseteq G$ and $?\colon \{o\} \sqsubseteq \{o\}$ for every nominal $\{o\}$ using the rules Table 4. We claim that for every $?\colon C \sqsubseteq D \in \mathbf{S}'$, either D is safe or $C = D = \{o\}$ for some nominal $\{o\}$. This is proved by induction over the application of rules in Table 4:

The base case for the initial axioms $?\colon G \sqsubseteq G$ and $?\colon \{o\} \sqsubseteq \{o\}$ holds trivially because G is safe.

Rule $\mathsf{R}_{\sqsubseteq}$ derives only $?\colon C \sqsubseteq E$ such that $D \sqsubseteq E \in \mathcal{O}$. Therefore E is safe by assumption of the theorem.

For rule $\mathsf{R}^{-}_{\sqcap}$, by induction hypothesis applied to the premise $?\colon C \sqsubseteq D_1 \sqcap D_2$, we have that $D_1 \sqcap D_2$ is safe (since it is not a nominal). Then both D_1 and D_2 are safe, so the claim holds for the conclusions $?\colon C \sqsubseteq D_1$ and $?\colon C \sqsubseteq D_2$.

For rule $\mathsf{R}^{-}_{\exists}$, by induction hypothesis applied to the premise $?\colon C \sqsubseteq \exists R.D$, we have that $\exists R.D$ is safe (because it is not a nominal). Then either D is a nominal, or D is safe. Therefore, the claim holds for the conclusion $?\colon D \sqsubseteq D$.

For rules $\mathsf{R}^{+}_{\top}$, $\mathsf{R}^{+}_{\sqcap}$, and $\mathsf{R}^{+}_{\exists}$, the claim holds for the produced conclusion $?\colon C \sqsubseteq D$, because D occurs in $\mathcal{O}$ and is not a nominal, so it has to be safe.

For rule $\mathsf{R}_{\{\}}$, by induction hypothesis applied to the premises $?\colon C \sqsubseteq \{a\}$ and $?\colon D \sqsubseteq \{a\}$, we have $C = D = \{a\}$, so the claim holds for the conclusion $?\colon \{a\} \sqsubseteq \{a\}$.

From the last case it follows, in particular, that $\mathbf{S}'$ contains only axioms derivable from the initial axioms without using rule $\mathsf{R}_{\{\}}$. Since the remaining rules in Table 4 correspond to the rules in Table 2, $\mathbf{S}$ closed under the rules in Table 2, $G \sqsubseteq G \in \mathbf{S}$, and $\{o\} \sqsubseteq \{o\} \in \mathbf{S}$ for every nominal $\{o\}$, it follows that if $?\colon C \sqsubseteq D \in \mathbf{S}'$ then $C \sqsubseteq D \in \mathbf{S}$.

Finally, assume that $\mathcal{O} \models G \sqsubseteq D$ for some D occurring in $\mathcal{O}$. Then, by Theorem 7, $?\colon G \sqsubseteq D \in \mathbf{S}'$. Therefore, $G \sqsubseteq D \in \mathbf{S}$, which was required to show. □

Example 10. We show that n-safety of G in Theorem 9 is essential. Let $G = \{o\} \sqcap \exists R.(A \sqcap \{o\})$, $\mathcal{O} = \{A \sqsubseteq A\}$,

$$\mathbf{S} = \{G \sqsubseteq G,\ \{o\} \sqsubseteq \{o\},\ G \sqsubseteq \{o\},\ G \sqsubseteq \exists R.(A \sqcap \{o\}),$$
$$A \sqcap \{o\} \sqsubseteq A \sqcap \{o\},\ A \sqcap \{o\} \sqsubseteq A,\ A \sqcap \{o\} \sqsubseteq \{o\}\}.$$

Then $\mathbf{S}$ is closed under the rules in Table 2, $\mathcal{O} \models G \sqsubseteq A$, but $G \sqsubseteq A \notin \mathbf{S}$.

In spite of their rareness in practice, there are interesting non-safe uses of nominals in $\mathcal{ELO}$. One can state, e.g., that Alice is the only female child of Bob and Mary: $\{\mathit{alice}\} \equiv \exists \mathit{hasSex}.\{\mathit{female}\} \sqcap \exists \mathit{isChildOf}.\{\mathit{bob}\} \sqcap \exists \mathit{isChildOf}.\{\mathit{mary}\}$.

Conclusions and Outlook

This work is part of a bigger research agenda to develop efficient algorithms and implementations for all of OWL EL. The present paper complements our previous work on reasoning with role compositions (Kazakov, Krötzsch, and Simančík 2011c). Together, these contributions handle the two features of OWL EL that have been argued to be most difficult to implement efficiently (Krötzsch 2011).

Both features are now supported by the free and open source reasoner ELK, which uses concurrent computation strategies for highest performance (Kazakov, Krötzsch, and Simančík 2011a). Support for nominals is currently implemented using the overestimation optimization together with axiom reuse. The additional optimization based on computing of reachability components did not show any further improvements in our experiments. Future work on ELK will focus on the remaining features of OWL EL, e.g., datatype support and local reflexivity (Self). Although we do not expect the same difficulties for these features, an efficient implementation is still needed. Indeed, for practitioners, the availability of tools like ELK plays a key role in the decision for or against the use of new features, which ultimately determines the overall success of KR languages like OWL EL.

Acknowledgments

This work was supported by the EU FP7 project SEALS and by the EPSRC projects ConDOR, ExODA and LogMap.

References

Baader, F.; Brandt, S.; and Lutz, C. 2005. Pushing the $\mathcal{EL}$ envelope. In Kaelbling, L., and Saffiotti, A., eds., *Proc. 19th Int. Joint Conf. on Artificial Intelligence (IJCAI'05)*, 364–369. Professional Book Center.

Baader, F.; Brandt, S.; and Lutz, C. 2008. Pushing the $\mathcal{EL}$ envelope further. In Clark, K. G., and Patel-Schneider, P. F., eds., *Proc. OWLED 2008 DC Workshop on OWL: Experiences and Directions*, volume 496 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Baader, F.; Lutz, C.; and Suntisrivaraporn, B. 2006. CEL—a polynomial-time reasoner for life science ontologies. In Furbach, U., and Shankar, N., eds., *Proc. 3rd Int. Joint Conf. on Automated Reasoning (IJCAR'06)*, volume 4130 of *LNCS*, 287–291. Springer.

Kazakov, Y.; Krötzsch, M.; and Simančík, F. 2011a. Concurrent classification of $\mathcal{EL}$ ontologies. In Aroyo, L.; Welty, C.; Alani, H.; Taylor, J.; Bernstein, A.; Kagal, L.; Noy, N.; and Blomqvist, E., eds., *Proceedings of the 10th International Semantic Web Conference (ISWC'11)*, volume 7032 of *LNCS*. Springer.

Kazakov, Y.; Krötzsch, M.; and Simančík, F. 2011b. Practical reasoning with nominals in the $\mathcal{EL}$ family of description logics. Technical report, University of Oxford. available from http://code.google.com/p/elk-reasoner/wiki/Publications.

Kazakov, Y.; Krötzsch, M.; and Simančík, F. 2011c. Unchain my $\mathcal{EL}$ reasoner. In *Proceedings of the 23rd International Workshop on Description Logics (DL'10)*, volume 745 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Kazakov, Y. 2009. Consequence-driven reasoning for Horn $\mathcal{SHIQ}$ ontologies. In Boutilier, C., ed., *Proc. 21st Int. Joint Conf. on Artificial Intelligence (IJCAI'09)*, 2040–2045. IJCAI.

Krötzsch, M. 2011. Efficient rule-based inferencing for OWL EL. In Walsh, T., ed., *Proc. 22nd Int. Joint Conf. on Artificial Intelligence (IJCAI'11)*, 2668–2673. AAAI Press/IJCAI.

Lawley, M. J., and Bousquet, C. 2010. Fast classification in Protégé: Snorocket as an OWL 2 EL reasoner. In Taylor, K.; Meyer, T.; and Orgun, M., eds., *Proc. 6th Australasian Ontology Workshop (IAOA'10)*, volume 122 of *Conferences in Research and Practice in Information Technology*, 45–49. Australian Computer Society Inc.

Mendez, J.; Ecke, A.; and Turhan, A.-Y. 2011. Implementing completion-based inferences for the $\mathcal{EL}$-family. In Rosati, R.; Rudolph, S.; and Zakharyaschev, M., eds., *Proceedings of the international Description Logics workshop*, volume 745. CEUR.

Motik, B.; Cuenca Grau, B.; Horrocks, I.; Wu, Z.; Fokoue, A.; and Lutz, C., eds. 27 October 2009. *OWL 2 Web Ontology Language: Profiles*. W3C Recommendation. Available at http://www.w3.org/TR/owl2-profiles/.

Motik, B.; Shearer, R.; and Horrocks, I. 2009. Hypertableau reasoning for description logics. *J. of Artificial Intelligence Research* 36:165–228.

OpenGalen.org. 2011. Value types. http://www.opengalen.org/tutorials/grail/tutorial327.html, accessed Dec 9, 2011.

Rogers, J. 2011. Principal Terminology Specialist, NHS Connecting for Health, UK Terminology Centre. Personal communication, November 2011.

Tarjan, R. 1972. Depth-first search and linear graph algorithms. *SIAM Journal on Computing* 1(2):146–160.