Error-Tolerance and Error Management in Lightweight Description Logics

Rafael Peñaloza

KI - Künstliche Intelligenz
German Journal of Artificial Intelligence - Organ des Fachbereichs "Künstliche Intelligenz" der Gesellschaft für Informatik e.V.

ISSN 0933-1875
Volume 34
Number 4

Künstl Intell (2020) 34:491-500
DOI 10.1007/s13218-020-00684-5

KI - Künstliche Intelligenz (2020) 34:491–500
https://doi.org/10.1007/s13218-020-00684-5

TECHNICAL CONTRIBUTION

Error-Tolerance and Error Management in Lightweight Description Logics

Rafael Peñaloza[1]

Received: 31 October 2019 / Accepted: 17 July 2020 / Published online: 24 July 2020

Abstract

The construction and maintenance of ontologies is an error-prone task. As such, it is not uncommon to detect unwanted or erroneous consequences in large-scale ontologies which are already deployed in production. While waiting for a corrected version, these ontologies should still be available for use in a "safe" manner, which avoids the known errors. At the same time, the knowledge engineer in charge of producing the new version requires support to explore only the potentially problematic axioms, and reduce the number of exploration steps. In this paper, we explore the problem of deriving meaningful consequences from ontologies which contain known errors. Our work extends the ideas from inconsistency-tolerant reasoning to allow for arbitrary entailments as errors, and allows for any part of the ontology (be it the terminological elements or the facts) to be the causes of the error. Our study shows that, with a few exceptions, tasks related to this kind of reasoning are intractable in general, even for very inexpressive description logics.

Keywords Description logics · Error-tolerance · Reasoning

1 Introduction

Description logics (DLs) [4] are a family of knowledge representation formalisms, which have been successfully applied to build large ontologies modelling different application domains. Among the members of this large family, two subfamilies of *lightweight* DLs known as DL-Lite [2, 11] and $\mathcal{EL}$ [3, 6] are of particular interest due to the low complexity of their standard reasoning tasks. Unfortunately, building and maintaining large ontologies in these or other languages is error-prone, and one often encounters errors, even after a careful pre-publication verification step. In addition, well-maintained ontologies usually stick to specific production cycles; for example, Snomed CT [26] produces a new version every 6 months. In the meantime, it should still be possible to use this ontology, although applying a "safe" mode that tries to avoid the (potential) causes for the known error.

In order to tackle this goal, we follow the original ideas from inconsistency-tolerant query answering first developed in the database realm [7] and then studied also for DLs [8, 15, 19], except that instead of focusing on inconsistencies as the sole proxy of erroneous modelling, we allow arbitrary consequences to be considered erroneous. For example, $\mathcal{EL}$ ontologies are always consistent, but earlier versions of Snomed (which is modelled in this logic) wrongly entailed that every amputation of a finger is an amputation of a hand. Analogously to previous work, we consider error-tolerant consequences to be those which can avoid the errors, in three levels of generality: brave (if there is one way to avoid the error and entail the consequence), cautious (if any correction of the ontology entails the consequence) and the intersection of all repairs, where a repair refers to a maximal subset of the ontology which avoids the error. We study the complexity of reasoning with these three variants, and show that in most cases the problems become intractable.

At the end of the paper, we study the extra-logical problem of helping the knowledge engineer in finding the wrong axioms which caused the error in the first place. We suggest finding an axiom that divides the number of potential repairs in half according to its membership in them, but show that even this task is hard for very simple logics.

This paper collects, corrects, and improves results which have been previously presented at conferences [20, 22, 23].

✉ Rafael Peñaloza
rafael.penaloza@unimib.it

[1] Department of Computer Science, Systems and Communications, University of Milano-Bicocca, Milan, Italy

2 Preliminaries

For this paper we focus on the lightweight families of description logics, which are known as the DL-Lite and $\mathcal{EL}$ families, using a meaningful representative of each family.

Consider three mutually disjoint sets N_C, N_R, and N_I of *concept*-, *role*-, and *individual names*, respectively. The class of *$\mathcal{EL}$ concepts* is built by the grammar rule $C ::= A \mid \top \mid C \sqcap C \mid \exists r.C$, where $A \in N_C$ and $r \in N_R$. The classes of DL-Lite *concepts* and DL-Lite *roles* are defined through the grammar rules $B ::= A \mid \exists s \mid \top \mid \bot$ and $s ::= r \mid r^-$, where $A \in N_C$ and $r \in N_R$. An *$\mathcal{EL}$ TBox* is a finite set of *general concept inclusions* (GCIs) of the form $C \sqsubseteq D$, where C and D are $\mathcal{EL}$ concepts. A DL-Lite$_{\text{Horn}}$ *TBox* is a finite set of *Horn concept inclusions* (HCIs) of the form $B_1 \sqcap \cdots \sqcap B_n \sqsubseteq B$, where $n \geq 1$, and each B, B_i is a DL-Lite concept, and *role inclusions* (RIs) of the form $s_1 \sqsubseteq s_2$, where r_1 and r_2 are DL-Lite roles. An ($\mathcal{EL}$ or DL-Lite$_{\text{Horn}}$) *ABox* is a finite set of *assertions* of the form $A(a)$ (concept assertion) or $r(a, b)$ (role assertion), where $A \in N_C$, $r \in N_R$, and $a, b \in N_I$. A *knowledge base* (sometimes also called an *ontology*) is a pair $(\mathcal{T}, \mathcal{A})$ where $\mathcal{T}$ is a TBox and $\mathcal{A}$ is an ABox.

In the following, we will handle DL-Lite and $\mathcal{EL}$ cases simultaneously, and hence often avoid the prefix in the name, speaking of e.g. a TBox. If there are several elements, we are implicitly assuming that they all belong to the same logical language. We sometimes use the term *axiom* to refer to GCIs, HCIs, RIs, and assertions as a whole, when it is not relevant what kind of element of an ontology we are referring to. In that case, an ontology becomes simply a finite set of axioms.

As an important special case, we will also consider the sublogic $\mathcal{HL}$, in which a TBox is a finite set of HCIs formed using concept names exclusively. We note that $\mathcal{HL}$ is a notational variant of propositional Horn logic, and that it lies at the intersection of $\mathcal{EL}$ and DL-Lite. That is, every $\mathcal{HL}$ ontology is also an $\mathcal{EL}$ and a DL-Lite ontology.

The semantics of these logics, as all DLs, is based on first-order semantics, where concepts correspond to unary predicates, and roles are binary predicates. Formally, an *interpretation* is a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty set called the *domain*, and $\cdot^{\mathcal{I}}$ is the *interpretation function*, which maps every individual name $a \in N_I$ to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, every concept name $A \in N_C$ to a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, and every role name $r \in N_R$ to a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The interpretation function is extended to cover all other constructors of DL-Lite and $\mathcal{EL}$—and hence interpret arbitrary $\mathcal{EL}$ and DL-Lite concepts and roles—as follows:

- $(r^-)^{\mathcal{I}} := \{(y, x) \mid (x, y) \in r^{\mathcal{I}}\}$;
- $\top^{\mathcal{I}} := \Delta^{\mathcal{I}}$;
- $\bot^{\mathcal{I}} := \emptyset$;
- $(C \sqcap D)^{\mathcal{I}} := C^{\mathcal{I}} \cap D^{\mathcal{I}}$;
- $(\exists s.C)^{\mathcal{I}} := \{x \mid \exists y \in C^{\mathcal{I}}.(x, y) \in s^{\mathcal{I}}\}$; and
- $(\exists s)^{\mathcal{I}} := (\exists s.\top)^{\mathcal{I}}$.

The interpretation $\mathcal{I}$ *satisfies* the GCI or HCI $C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$; the RI $s_1 \sqsubseteq s_2$ iff $s_1^{\mathcal{I}} \subseteq s_2^{\mathcal{I}}$; the concept assertion $A(a)$ iff $a^{\mathcal{I}} \in A^{\mathcal{I}}$; and the role assertion $r(a, b)$ iff $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$. $\mathcal{I}$ is a *model* of the TBox $\mathcal{T}$, the ABox $\mathcal{A}$ or the ontology $\mathcal{O}$ iff it satisfies all the axioms in $\mathcal{T}$, $\mathcal{A}$, or $\mathcal{O}$, respectively. We denote this as $\mathcal{I} \models \mathcal{T}, \mathcal{I} \models \mathcal{A}$, and $\mathcal{I} \models \mathcal{O}$, respectively. The ontology $\mathcal{O}$ is *consistent* iff there is a model of $\mathcal{O}$. Given an ontology $\mathcal{O}$ and an axiom α, we say that $\mathcal{O}$ *entails* α (denoted as $\mathcal{O} \models \alpha$) iff every model of $\mathcal{O}$ also satisfies α.

One of the main reasoning tasks in DLs is entailment checking; that is, deciding whether a given ontology entails an axiom. In $\mathcal{EL}$ and DL-Lite, entailments can be checked in polynomial time. In some cases, the axiom that is tested for entailment is not a wanted consequence, but rather an error that one tries to avoid. For example, if we want to check whether an ontology is consistent, we might test whether $\mathcal{O} \models \bot(a)$, which holds only in the case the $\mathcal{O}$ has no models. When this entailment holds, it is a signal of an error in the ontology. In these situations, if this unwanted entailment holds, then one may be interested in identifying the axioms that cause this consequence—in an attempt to place the blame —or a candidate sub-ontology which excludes it, giving rise to the following definitions.

Definition 1 *(justification, repair)* Let $\mathcal{O}$ be an ontology and α an axiom such that $\mathcal{O} \models \alpha$. A *justification* of α w.r.t. $\mathcal{O}$ is a sub-ontology $\mathcal{M} \subseteq \mathcal{O}$ such that $\mathcal{M} \models \alpha$ and for all $\mathcal{N} \subsetneq \mathcal{M}$, $\mathcal{N} \not\models \alpha$. A *repair* for α w.r.t. $\mathcal{O}$ is a sub-ontology $\mathcal{R} \subseteq \mathcal{O}$ such that $\mathcal{R} \not\models \alpha$ and for all $\mathcal{Q} \supsetneq \mathcal{R}$, $\mathcal{Q} \models \alpha$.

In words, a justification is a minimal (w.r.t. set inclusion) sub-ontology that entails the consequence, while a repair is a maximal (w.r.t. set inclusion) sub-ontology that avoids it. It is well known that there might exist exponentially many justifications or repairs for a given consequence. In the following, $\mathsf{Just}(\mathcal{O}, \alpha)$ and $\mathsf{Rep}(\mathcal{O}, \alpha)$ denote the sets of all justifications and repairs for α w.r.t. $\mathcal{O}$. If the specific ontology used is irrelevant, we often remove the first argument, and write simply e.g. $\mathsf{Just}(\alpha)$.

An interesting property of $\mathcal{HL}$ ontologies, which we will use throughout this paper, is that they can be represented as directed hypergraphs. Under this view, nodes represent concept names or individual names, and a hyperedge corresponds to an axiom in the ontology. Hence, entailment checking corresponds to the task of deciding reachability between nodes. More importantly, a justification is nothing more than a simple hyperpath. If we further restrict $\mathcal{HL}$ to disallow conjunctions on the left-hand side of axioms, then this representation collapses to classical graphs and the exploration of simple paths determines justifications.

The (hyper)graph representation can be extended to DL-Lite$_{\text{Horn}}$ ontologies by allowing nodes to represent also the complex concepts provided in this language; that is, $\exists s$ and $\bot$. However, reasoning becomes more complex from the need to handle role inclusions, inverse roles, and in particular, the special semantics of $\bot$ which represents a contradiction. For example, to derive $\mathcal{O} \models A \sqsubseteq B$ from a DL-Lite$_{\text{Horn}}$ ontology $\mathcal{O}$, it suffices to derive $\mathcal{O} \models A \sqsubseteq \bot$.

3 Error-Tolerant Reasoning

If an unwanted consequence or error is entailed by an ontology, then we know that this ontology must contain some errors. That means, in particular, that we cannot directly trust the consequences derived from it. Still, we do not want to throw the whole ontology away and start from scratch, or wait until a human expert has fixed all the issues to use it. Alternatively, we want to be able to derive some consequences that may still be trusted. Intuitively, these are consequences that can be derived using axioms which do not play a role in the error. This intuition gives rise to three main semantics.

Definition 2 *(error-tolerant semantics)* Let $\mathcal{O}$ be an ontology and α, β two axioms such that $\mathcal{O} \models \alpha$. We say that β is a *cautious* consequence of $\mathcal{O}$ w.r.t. α if for every $\mathcal{R} \in \mathsf{Rep}(\mathcal{O}, \alpha)$, $\mathcal{R} \models \beta$. It is a *brave* consequence of $\mathcal{O}$ w.r.t. α if there exists some $\mathcal{R} \in \mathsf{Rep}(\mathcal{O}, \alpha)$ such that $\mathcal{R} \models \beta$. It is an *intersection of all repairs* (IAR) consequence of $\mathcal{O}$ w.r.t. α if $\bigcap_{\mathcal{R} \in \mathsf{Rep}(\mathcal{O}, \alpha)} \mathcal{R} \models \beta$.[1]

Note that this definition considers the presence of *one* error only. While in practice one should expect to observe multiple errors, for the scope of this paper we focus on the simpler case and leave open the question of dealing with several errors. In any case, the notion of a repair, and by extension the error-tolerant semantics, can be easily generalised to the case with several errors in the obvious way: a repair is a maximal sub-ontology from which *none* of the known errors follows.

In Definition 2 we consider error-tolerant semantics based on the whole ontology. It is sometimes convenient to consider the more general cases where a portion of the ontology is fixed, and its axioms cannot be removed to form a repair. Thus, we can analogously define the notions of *ABox* error-tolerant semantics, where the TBox is fixed and repairs are defined only as subsets of the ABox, and dually *TBox* error-tolerant semantics where the ABox is fixed.

Note that there is a natural strength relationship between the three semantics from Definition 2: every IAR consequence is also a cautious consequence, and each cautious consequence is a brave consequence. It is easy to build examples showing that the converse implications do not hold in general, even for $\mathcal{HL}$.

Recall that classical entailments in these logics can be decided in polynomial time. Unfortunately, as we will see shortly, the same cannot be said about the error-tolerant semantics in general. But first, we consider a tractable case. For the following proof we use the notion of a directed hypergraph. While there are many ways to define directed hypergraphs, are interested in those having singleton heads[16].

Definition 3 *(directed hypergraph)* A *directed hypergraph* is a pair (V, E) where V is a set of *nodes* and E is a set of *directed hyperedges* of the form (W, v), where $W \subseteq V$ and $v \in V$.

A *hyperpath* from $X \subseteq V$ to $x \in V$ is a sequence $(W_0, v_0), \dots, (W_n, v_n)$ of hyperedges such that $v_n = x$ and for every $k, 0 \leq k \leq n$, $W_k \subseteq X \cup \{v_\ell \mid 0 \leq \ell < k\}$. This hyperpath is *simple* if for all $k, 0 \leq k \leq n$, $v_k \notin X \cup \{v_\ell \mid 0 \leq \ell < k\}$. A *subpath* of the hyperpath H is a subsequence of H.

In words, a directed hyperedge connects a set of nodes (the sources) with one node (the head). The notion of a hyperpath generalises the idea of a path in a graph, by using hyperedges; that is, all the sources W of a hyperedge (W, v) need to be reached before this hyperedge can be followed. We sometimes see hyperpaths as sets of hyperedges, and hence treat them as hypergraphs.

Theorem 4 *Brave entailment w.r.t.* DL-Lite$_{\text{Horn}}$ *ontologies can be decided in polynomial time.*

Proof Let $\mathcal{O}$ be a DL-Lite$_{\text{Horn}}$ ontology. We construct the directed hypergraph $\mathcal{H}_{\mathcal{O}} = (V, E)$ by setting V to be the set of all DL-Lite$_{\text{Horn}}$ individual names and concept names appearing in $\mathcal{O}$, together with the concepts $\top, \bot$, and $\exists s$, where s is a role name appearing in $\mathcal{O}$ or its inverse. The set E of hyperedges is defined by the axioms as follows:

[1] Originally, the name IAR stood for *intersection of ABox repairs* in the context of inconsistent-tolerant query answering. We have adapted the name to this slightly more general context to preserve the intuitive connection to existing work on inconsistency-tolerance.

$$\begin{aligned} E := & \left\{(W, v) \mid \bigsqcap_{B\in W} B \sqsubseteq v \in \mathcal{O}\right\} \cup \\ & \{(\{\exists s_1\}, \exists s_2) \mid s_1 \sqsubseteq s_2 \in \mathcal{O}\} \cup \\ & \{(\{a\}, A) \mid A(a) \in \mathcal{O}\} \cup \\ & \{(\{a\}, \exists r), (\{b\}, \exists r^-) \mid r(a,b) \in \mathcal{O}\}. \end{aligned}$$

It is easy to verify (see also[24]) that $\mathcal{O} \models B_1 \sqcap \cdots \sqcap B_n \sqsubseteq B$ iff there is a hyperpath from $\{B_1, \ldots, B_n\}$ to B in $\mathcal{H}_\mathcal{O}$, and $\mathcal{O} \models A(a)$ iff there is a hyperpath from $\{a\}$ to A in the same hypergraph. For the rest of the proof we consider as axioms only HCIs, but all other cases are treated analogously.

$B_1 \sqcap \cdots \sqcap B_n \sqsubseteq B$ is a brave consequence of $\mathcal{O}$ w.r.t. $B'_1 \sqcap \cdots \sqcap B'_m \sqsubseteq B'$ iff there is a hyperpath from $\{B_1, \ldots, B_n\}$ to B that does not contain as a subpath any hyperpath from $\{B'_1, \ldots, B'_m\}$ to B'. This can be verified in polynomial time (on the size of the ontology) through a generalisation of the usual reachability algorithm, which marks nodes with three potential values: *unreachable*, *reachable*, and *safely reachable* meaning that it is reachable through a path that does not contain the undesired subpath. These labels are forward propagated through the edges of the hypergraph, until all nodes have been adequatedly marked. □

This theorem extends the previous tractability result known for $\mathcal{HL}$[20] which is a sublogic of DL-Lite$_{\text{Horn}}$. It is also in contrast with the hardness result by[9], who show that under the ABox error-tolerant semantics, brave entailments in this same logic are NP-complete, even if limited to the special case of α being an inconsistency check and β an instance query (that is, a concept assertion). Note that this latter result does not contradict Theorem 4 since for the ABox semantics no axioms from the TBox may be removed. In terms of the proof of the theorem, this means that several hyperedges of $\mathcal{H}_\mathcal{O}$ are always present and the construction of a path avoiding some nodes does not suffice to guarantee the existence of an adequate repair.

If we consider $\mathcal{EL}$, even brave entailment checking is NP-hard, as stated next.

Theorem 5 *Brave entailment w.r.t $\mathcal{EL}$ ontologies is NP-complete.*

Proof The upper bound is obtained by considering the following non-deterministic algorithm: first guess a sub-ontology $\mathcal{M} \subseteq \mathcal{O}$ and then verify that $\mathcal{M} \models \beta$ and $\mathcal{M} \not\models \alpha$. If this is true, then there exists a repair of α, which extends $\mathcal{M}$, that entails β.

For the lower bound, we present a reduction from the *more minimal valuations* (MMV) problem for monotone Boolean formulas, which is known to be NP-hard[5, 14]: given a monotone Boolean formula φ and a set $\mathfrak{B}$ of minimal valuations satisfying φ, decide whether there exists a valuation satisfying φ which does not contain any valuation from $\mathfrak{B}$. The reduction is based on an idea previously used in the context of the enumeration of justifications in[24].

Let $\varphi, \mathfrak{B}$ be an instance of MMV, and let $\mathsf{sub}(\varphi)$ and $\mathsf{csub}(\varphi)$ denote the sets of all subformulas of φ and of all complex subformulas of φ, respectively. That is, csub excludes all propositional variables. For every $\psi \in \mathsf{sub}(\varphi)$, we introduce three concept names B_ψ, C_ψ, D_ψ, and two role names r_ψ, s_ψ. For every $\mathcal{V} \in \mathfrak{B}$, we similarly introduce $B_\mathcal{V}, C_\mathcal{V}, D_\mathcal{V}, r_\mathcal{V}$, and $s_\mathcal{V}$. In addition, we introduce the concept names A, E, D, and F. Each $\psi \in \mathsf{sub}(\varphi)$ defines a TBox $\mathcal{T}_\psi$ as follows: if ψ is the propositional variable p, then $\mathcal{T}_\psi := \{A \sqsubseteq B_p\}$; if $\psi = \psi_1 \wedge \psi_2$, then

$$\begin{aligned} \mathcal{T}_\psi := \{ & A \sqsubseteq \exists r_\psi.C_\psi, \quad C_\psi \sqsubseteq B_{\psi_1}, \quad C_\psi \sqsubseteq B_{\psi_2}, \\ & \exists r_\psi.B_\psi \sqsubseteq D_\psi, \quad B_{\psi_1} \sqcap B_{\psi_2} \sqsubseteq B_\psi \qquad \}, \end{aligned}$$

and if $\psi = \psi_1 \vee \psi_2$, then

$$\begin{aligned} \mathcal{T}_\psi := \{ & A \sqsubseteq \exists r_\psi.B_{\psi_1}, \quad A \sqsubseteq \exists s_\psi.B_{\psi_2}, \\ & \exists r_\psi.B_\psi \sqcap \exists s_\psi.B_\psi \sqsubseteq D_\psi, \\ & B_{\psi_1} \sqsubseteq B_\psi, \quad B_{\psi_2} \sqsubseteq B_\psi \qquad \}. \end{aligned}$$

Following the same method, we construct for every $\mathcal{V} \in \mathfrak{B}$ the TBox

$$\begin{aligned} \mathcal{T}_\mathcal{V} := & \left\{A \sqsubseteq \exists r_\mathcal{V}.C_\mathcal{V},\ \exists r_\mathcal{V}.B_\mathcal{V} \sqsubseteq D_\mathcal{V},\ \bigsqcap_{p\in\mathcal{V}} B_p \sqsubseteq B_\mathcal{V}\right\} \cup \\ & \{C_\mathcal{V} \sqsubseteq B_p \mid p \in \mathcal{V}\}. \end{aligned}$$

Finally, we set

$$\begin{aligned} \mathcal{T}_\mathfrak{B} := & \{A \sqsubseteq \exists s_\mathcal{V}.B_\mathcal{V}, B_\mathcal{V} \sqsubseteq F \mid \mathcal{V} \in \mathfrak{B}\} \cup \\ & \left\{\bigsqcap_{\mathcal{V}\in\mathfrak{B}} \exists s_\mathcal{V}.F \sqsubseteq D\right\} \\ \mathcal{T} := & \bigcup_{\psi\in\mathsf{sub}(\varphi)} \mathcal{T}_\psi \cup \bigcup_{\mathcal{V}\in\mathfrak{B}} \mathcal{T}_\mathcal{V} \cup \mathcal{T}_\mathfrak{B} \cup \\ & \left\{\bigsqcap_{\psi\in\mathsf{csub}(\varphi)} D_\psi \sqcap \bigsqcap_{\mathcal{V}\in\mathfrak{B}} D_\mathcal{V} \sqcap D \sqcap B_\varphi \sqsubseteq E\right\}. \end{aligned}$$

Notice that, for every $\mathcal{T}' \subseteq \mathcal{T}$, if $\mathcal{T}' \models A \sqsubseteq E$, then also $\mathcal{T}' \models A \sqsubseteq D_\psi$ for all $\psi \in \mathsf{csub}(\varphi)$. It is easily seen, exploring the axioms in $\mathcal{T}_\psi$, that $\mathcal{T}' \models A \sqsubseteq D_\psi$ can only hold if $\mathcal{T}'$ contains $\mathcal{T}_\psi$. In particular, if $\psi = \psi_1 \wedge \psi_2$, then $B_{\psi_1} \sqcap B_{\psi_2} \sqsubseteq B_\psi \in \mathcal{T}'$ and if $\psi = \psi_1 \vee \psi_2$, then $\{B_{\psi_1} \sqsubseteq B_\psi, B_{\psi_2} \sqsubseteq B_\psi\} \subseteq \mathcal{T}'$. Similarly, it must hold that $\mathcal{T}' \models A \sqsubseteq D_\mathcal{V}$ for all $\mathcal{V} \in \mathfrak{B}$ and $\mathcal{T}' \models A \sqsubseteq D$ which means that for every $\mathcal{V} \in \mathfrak{B}$, $\sqcap_{p\in\mathcal{V}} B_p \sqsubseteq B_\mathcal{V} \in \mathcal{T}'$ and also $B_\mathcal{V} \sqsubseteq F \in \mathcal{T}'$.

Thus, a valuation $\mathcal{W}$ satisfies φ iff the TBox

$$\mathcal{S}_{\mathcal{W}} := \{A \sqsubseteq B_p \mid p \in \mathcal{W}\} \cup \bigcup_{\psi \in \mathsf{csub}(\varphi)} \mathcal{T}_\psi \cup \bigcup_{\mathcal{V} \in \mathfrak{V}} \mathcal{T}_{\mathcal{V}} \cup$$
$$\mathcal{T}_{\mathfrak{V}} \cup \left\{ \bigsqcap_{\psi \in \mathsf{csub}(\varphi)} D_\psi \sqcap \bigsqcap_{\mathcal{V} \in \mathfrak{V}} D_{\mathcal{V}} \sqcap D \sqcap B_\varphi \sqsubseteq E \right\}$$

entails $A \sqsubseteq E$. This valuation does not contain any $\mathcal{V} \in \mathfrak{V}$ iff $\mathcal{S}_{\mathcal{W}} \not\models A \sqsubseteq F$. Thus, $\varphi, \mathfrak{V}$ is a positive instance of MMV iff $A \sqsubseteq E$ is a brave consequence of $\mathcal{T}$ w.r.t. $A \sqsubseteq F$. □

The next step is to show that the two remaining semantics are intractable as well, in general. In this case, we show that the problems are coNP-complete and that hardness holds already for $\mathcal{HL}$, even if we disallow conjunctions on the left of HCIs.

Theorem 6 *Deciding cautious and IAR entailments w.r.t. $\mathcal{HL}$, DL-Lite$_{\text{Horn}}$, or $\mathcal{EL}$ ontologies is co NP-complete.*

Proof For the upper bounds, we exploit the fact that entailments in all these logics can be decided in polynomial time, similarly to the approach used to prove upper bounds for ABox repair semantics. If β is not cautiously entailed by $\mathcal{O}$ w.r.t. α, we can guess a subset $\mathcal{R} \subseteq \mathcal{O}$ and verify in polynomial time that $\mathcal{R}$ is a repair and that $\mathcal{R} \not\models \beta$. Similarly, if β is not IAR entailed by $\mathcal{O}$ w.r.t. α, we can guess a linear family of sets $\mathcal{R}_1, \ldots, \mathcal{R}_n$ with $n \leq |\mathcal{O}|$ and verify that each $\mathcal{R}_i$ is a repair and $\bigcap_{i=1}^n \mathcal{R}_i \not\models \beta$. Both conditions can be checked in polynomial time.

For the lower bound, we reduce the coNP-complete *no-path-through-node* (NPTN) problem: given a graph $G = (V, E)$ and nodes $s, t, m \in V$, decide if there is no simple path from s to t that passes through m. Given an instance of NPTN, we introduce a concept name A_v for every $v \in (V \setminus \{m\}) \cup \{m_1, m_2\}$, where $m_1, m_2 \notin V$, and construct the $\mathcal{HL}$ TBox

$$\begin{aligned}\mathcal{T} := \{&A_v \sqsubseteq A_w \mid (v,w) \in E, v, w \neq m\} \cup \\ \{&A_v \sqsubseteq A_{m_1} \mid (v,m) \in E, v \neq m\} \cup \\ \{&A_{m_2} \sqsubseteq A_v \mid (m,v) \in E, v \neq m\} \cup \\ \{&A_{m_1} \sqsubseteq A_{m_2}\}.\end{aligned}$$

There is no path from s to t passing through m iff every repair of $A_s \sqsubseteq A_t$ w.r.t. $\mathcal{O}$ contains $A_{m_1} \sqsubseteq A_{m_2}$. This holds iff $A_{m_1} \sqsubseteq A_{m_2}$ is both, a cautious and an IAR consequence of $\mathcal{O}$ w.r.t. $A_s \sqsubseteq A_t$. □

In the literature, the higher complexity observed for the error-tolerant semantics has been often attributed to the fact that the number of repairs may be exponential on the size of the ontology. While there is some truth in this argument, it is also incomplete; for example, it fails to explain why brave consequences in DL-Lite$_{\text{Horn}}$ remain polynomial, even though the number of repairs remains unchanged. As we will see next, the reasons for hardness are more subtle. In fact, we can guarantee the existence of $\mathcal{EL}$ ontologies having a sub-exponential number of repairs w.r.t. a given consequence, for which error-tolerant entailments are still intractable (unless P = NP).

Theorem 7 *Assuming* P ≠ NP, *there is no algorithm for deciding cautious or brave entailments w.r.t. an $\mathcal{EL}$ ontology $\mathcal{O}$ and unwanted consequence α that runs in polynomial time in the size of $\mathcal{O}$ and* $\mathsf{Rep}(\mathcal{O}, \alpha)$.

Proof Consider the NP-complete *more maximal falsifiers* (MMF) problem[21]: given a monotone Boolean formula φ and a set $\mathfrak{F}$ of maximal valuations falsifying φ, decide whether there exists a valuation $\mathcal{W}$ falsifying φ such that $\mathcal{W} \not\subseteq \mathcal{V}$ for all $\mathcal{V} \in \mathfrak{F}$. Given an instance $\varphi, \mathfrak{F}$ of MMF, let $\mathsf{sub}(\varphi)$ be the set of all subformulas of φ, and construct the TBoxes $\mathcal{T}_\psi$ for $\psi \in \mathsf{sub}(\varphi)$ as in the proof of Theorem 5. Construct then the TBoxes

$$\mathcal{T}_{\mathfrak{F}} := \left\{ \bigsqcap_{p \in \mathcal{V}} B_p \sqsubseteq F \mid \mathcal{V} \in \mathfrak{F} \right\}$$
$$\mathcal{T} := \bigcup_{\psi \in \mathsf{sub}(\varphi)} \mathcal{T}_\psi \cup \mathcal{T}_{\mathfrak{F}} \cup \left\{ \bigsqcap_{\psi \in \mathsf{sub}(\varphi)} D_\psi \sqcap B_\varphi \sqsubseteq E \right\}.$$

There are two kinds of repairs of $\mathcal{T}$ w.r.t. $A \sqsubseteq E$, those of the form $\mathcal{T} \setminus \{\alpha\}$ for $\alpha \in \bigcup_{\psi \in \mathsf{csub}(\varphi)} \mathcal{T}_\psi \cup \left\{\bigsqcap_{\psi \in \mathsf{csub}(\varphi)} D_\psi \sqcap B_\varphi \sqsubseteq E\right\}$, and those taking the form

$$\bigcup_{p \in \mathcal{V}} \mathcal{T}_p \cup \bigcup_{\psi \in \mathsf{csub}(\varphi)} \mathcal{T}_\psi \cup \mathcal{T}_{\mathfrak{F}} \cup \left\{ \bigsqcap_{\psi \in \mathsf{csub}(\varphi)} D_\psi \sqcap B_\varphi \sqsubseteq E \right\}.$$

for some maximal valuation $\mathcal{V}$. We can thus bound $|\mathsf{Rep}(\mathcal{T}, A \sqsubseteq E)| \leq 5|\mathsf{sub}(\varphi)| + n$, where n is the number of maximal valuations falsifying φ.

Assume by contradiction that there exists an algorithm that decides cautious entailments in polynomial time on $|\mathcal{T}|$ and $|\mathsf{Rep}(\mathcal{T}, A \sqsubseteq E)|$; that is, there exists an algorithm A with runtime bounded by some polynomial $p(t, r)$, where t is the size of the ontology and r the number of repairs, which decides cautious entailment. Using this algorithm, we can decide MMF as follows: run A on $\mathcal{T}$ for the cautious entailment $A \sqsubseteq F$ and stop after at most $p(|\mathcal{T}|, |\mathfrak{F}| + 5|\mathsf{sub}(\varphi)|)$ steps. If the answer is *yes*, then $\mathfrak{F}$ is the set of all falsifying valuations, and so there is no new one. If it answers *no*, or the execution of A did not terminate until this time bound, then there must be at least one more falsifying valuation. This means that A can be used to decide MMF in polynomial time, contradicting the fact that MMF is NP-complete.

The proof for brave entailments is analogous, but using a construction closer to that of Theorem 5. We leave the details as an exercise to the reader. □

All kinds of error-tolerant entailments can be decided in exponential time on the size of the ontology in all the logics that we consider here. In fact, one can simply enumerate all the $2^{|\mathcal{O}|}$ sub-ontologies of $\mathcal{O}$, and for each of them check in polynomial time on $|\mathcal{O}|$ that (i) it is a repair of α and (ii) whether it entails β. This means that whenever a consequence α has exponentially many repairs, brave and cautious entailments can always be decided in polynomial time on the size of $|\mathsf{Rep}(\mathcal{O}, \alpha)|$. Hence, the hardness from Theorem 7 must arise from a situation with less than exponentially many repairs.

Corollary 8 *It is intractable to decide cautious and brave consequences of $\mathcal{EL}$ ontologies, even if the number of repairs is sub-exponential.*

Obviously, we cannot have an analogue of Theorem 7 for DL-Lite$_{\text{Horn}}$, since brave entailments are already known to be decidable in polynomial time. However, if we restrict to ABox repairs (that is, where the TBox is fixed, and only assertions from the ABox can be removed to avoid an error) then hardness arises again. In the following theorem, we call *ABox-cautious* and *ABox-brave* the error-tolerant semantics obtained by restricting to ABox repairs only.

Theorem 9 *Assuming* P $\neq$ NP, *there is no algorithm for deciding ABox-cautious or ABox-brave entailments w.r.t. a* DL-Lite$_{\text{Horn}}$ *ontology $\mathcal{O}$ and unwanted consequence α that runs in polynomial time in the size of $\mathcal{O}$ and the number of ABox repairs.*

Proof The proof follows a similar idea as that of Theorem 7, but the ontology is slightly adapted to this case. Let $\varphi, \mathfrak{F}$ be an instance of MMF. For each $\psi \in \mathsf{sub}(\varphi)$ we create a concept name A_ψ, and for each $\psi \in \mathsf{csub}(\varphi)$ build the TBoxes

$$\mathcal{T}_\psi := \begin{cases} \{A_{\psi_1} \sqcap A_{\psi_2} \sqsubseteq A_\psi\} & \psi = \psi_1 \wedge \psi_2 \\ \{A_{\psi_1} \sqsubseteq A_\psi, A_{\psi_2} \sqsubseteq A_\psi\} & \psi = \psi_1 \vee \psi_2 \end{cases}$$

$$\mathcal{T} := \bigcup_{\psi \in \mathsf{csub}(\varphi)} \mathcal{T}_\psi \cup \{A_\varphi \sqsubseteq \bot\} \cup \left\{ \bigsqcap_{p \in \mathcal{V}} A_p \sqsubseteq B \mid \mathcal{V} \in \mathfrak{F} \right\},$$

and the ABox $\mathcal{A} := \{A_p(a) \mid p \text{ is a variable of } \varphi\}$. The ontology $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ has as many ABox repairs w.r.t. $\top \sqsubseteq \bot$ as there are maximal valuations falsifying φ, and $B(a)$ is an ABox-cautious entailment of $\mathcal{O}$ w.r.t. $\top \sqsubseteq \bot$ iff every valuation falsifying φ is contained in some $\mathcal{V} \in \mathfrak{F}$. If there was an algorithm that could decide cautious entailments in time $p(|\mathcal{O}|, |\mathsf{Rep}(\mathcal{O}, \top \sqsubseteq \bot)|)$, where p is a polynomial, then we can solve MMF by running this algorithm for time $p(|\mathcal{O}|, |\mathfrak{F}|)$.

For brave entailments, we reduce MMV. Given an instance $\varphi, \mathfrak{B}$ of MMV, construct $\mathcal{T}_\psi$ and $\mathcal{A}$ as in the previous part of the proof and define

$$\mathcal{T} := \bigcup_{\psi \in \mathsf{csub}(\varphi)} \mathcal{T}_\psi \cup \left\{ \bigsqcap_{p \in \mathcal{V}} A_p \sqsubseteq \bot \mid \mathcal{V} \in \mathfrak{B} \right\}.$$

Then, $A_\varphi(a)$ is a brave consequence of $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ w.r.t. $\top \sqsubseteq \bot$ iff there is a valuation satisfying φ that does not contain any $\mathcal{V} \in \mathfrak{B}$. Using the same argument from the case of cautious consequences, this shows that brave entailments cannot be decided in polynomial time on the number of ABox repairs. □

4 IAR Repairs

For the hardness results presented at the end of the previous section, we did not consider the IAR semantics. In this section we show that, despite the complexity of the problem in general, some practical approaches can still be implemented for DL-Lite$_{\text{Horn}}$. To achieve this, we exploit the duality between repairs and justifications, and results on enumeration complexity.

It was previously shown that the simple hyperpaths of a directed hypergraph can be enumerated with polynomial delay[24]; that is, through a method that requires only polynomial time (on the size of the hypergraph) between the output of successive answers[17]. This fact was used to prove that all justifications for a DL-Lite$_{\text{Horn}}$ TBox (when the ABox is empty) can be enumerated in polynomial delay, using the reduction to hypergraphs sketched before. The result (and its proof) trivially extends to general ontologies by including the hyperedges that represent assertions from the ABox.

Proposition 10 *All the justifications for an axiom α w.r.t. the* DL-Lite$_{\text{Horn}}$ *ontology $\mathcal{O}$ can be enumerated with polynomial delay.*

From the duality between justifications and repairs, we know that the union of all justifications and the intersection of all repairs complement each other. In other words, to compute the intersection of all repairs, as a step to deduce IAR entailments, it suffices to remove from the ontology the union of all justifications. From our complexity results (Theorem 6), it follows immediately that the latter task—finding the union of all justifications—is also intractable. Still, we can devise an *anytime* algorithm, which iteratively computes one justification at a time—over-approximating the intersection of all repairs—and stop when either the consequence does not follow, or no more justifications are available. This approach is described in Algorithm 1,

Algorithm 1: Anytime algorithm for IAR entailments w.r.t. DL-Lite$_{\mathrm{Horn}}$ ontologies.

```
Data: DL-Lite$_{\mathrm{Horn}}$ ontology $\mathcal{O}$, axioms $\alpha, \beta$
Result: Is $\beta$ IAR-entailed by $\mathcal{O}$ w.r.t. $\alpha$?
$\mathcal{U} \leftarrow \mathcal{O}$
while MORE-JUSTIFICATIONS ($\alpha$) do
    $\mathcal{N} \leftarrow$ NEXT-JUSTIFICATION
    $\mathcal{U} \leftarrow \mathcal{U} \setminus \mathcal{N}$
    if $\mathcal{U} \not\models \beta$ then
        Return false
Return true
```

where MORE-JUSTIFICATIONS is a Boolean function that verifies whether there are still more justifications for α w.r.t. $\mathcal{O}$ that have not yet been enumerated, and NEXT-JUSTIFICATION in that case outputs the next justification in the enumeration.

Note that Algorithm 1 stops as soon as it is obvious that β cannot be entailed by the intersection of all repairs: at each iteration of the **while** loop, the set $\mathcal{U}$ monotonically decreases, hence further iterations would only remove more consequences, but never adds new ones. When the loop finishes, we know that we have enumerated all justifications, and hence $\mathcal{U}$ is exactly the intersection of all repairs, which guarantees the correctness of the algorithm. An important property of this algorithm is that the order of the enumeration can be manipulated to try to add justifications with previously unseen axioms first, so that the set $\mathcal{U}$ shrinks as fast as possible. However, one can only guarantee that the IAR entailment holds after *all* justifications have been found.

The practical benefit of Algorithm 1 resides not only in its anytime nature, but also in the fact that it deals with the enumeration of justifications, rather than repairs. Indeed, although in theory an entailment may also have exponentially many justifications, it has been empirically verified that in human-developed ontologies the number of justifications, and their size, tends to be small[18, 27, 29]. In contrast, the number of repairs does grow exponentially in well-maintained ontologies[20].

As an alternative to over-approximating the intersection of all repairs, one can try to under-approximate it. One way to do this is to use modularisation techniques to efficiently compute a so-called *justification-preserving module*. In essence, these modules are sub-ontologies that contain the union of all justifications. Different techniques balancing the computation time and the quality of the approximation have been proposed[12, 13, 25, 28], but in general the methods based on a syntactic analysis of the ontology tend to behave better.

As a final remark on this aspect, we note that the computation of a justification preserving module $\mathcal{M}$ is also useful to improve the efficiency of Algorithm 1: during the **while** loop, rather than computing the justification w.r.t. the original ontology $\mathcal{O}$, one can restrict to the axioms in the module $\mathcal{M}$. This allows the execution to avoid paths that will not lead to a justification, reducing the time and space required during runtime.

5 Correcting Errors

So far in this paper, we have considered the problem of dealing automatically with ontologies that are known to contain some errors by trying to avoid the causes of these errors during reasoning. Beyond this, there is not much that can be achieved in a fully automated manner. Indeed, from the purely logical point of view, all possible repairs are equal in the sense that they all remove the undesired consequence. From the knowledge representation point of view, however, we expect only one of them to be *correct* in the sense that their axioms all represent truths from the domain being modelled. Note that this is true even if multiple errors occur: there is one maximal sub-ontology that avoids all the known errors; and this is the one we are interested in finding.

Following approaches from belief revision, considering the postulate of minimal change, one could propose to focus on repairs of maximum cardinality. Alternatively, one could associate a degree of trust or preference to each axiom, and focus on the most trusted or preferred repairs. These, and other similar solutions that have been proposed, not only still suffer from the problem of a multiplicity of solutions (in the worst case, still exponentially many), but in addition cannot guarantee that the *correct* repair is among those selected; e.g., the correct repair might in fact be one with minimal cardinality. The issue is that correctness is an *extra-logical* property, which does not depend on the shape or interrelation of axioms, but rather on the domain that the ontology is modelling. In fact, the only way to know whether an axiom—and by extension, an ontology—is a correct representation of the domain knowledge is to ask a domain expert.

The process of consulting with a domain expert is the most expensive part of the process of error resolution in an ontology. Not only are these experts a limited resource, but they need to understand what the axioms say before they can make a determination on their correctness. For that reason, one would like to provide the expert with as few questions as possible in order to find the repair that resolves the error. To achieve this goal, one potential idea is to find an axiom β, called *cut axiom*, that partitions the space of repairs into two halves according to whether they contain β or not.

Definition 11 *(cut axiom)* Let $\mathcal{O}$ be an ontology and α an unwanted consequence. For an axiom $\beta \in \mathcal{O}$, we define the sets

$$\mathfrak{R}^+_\beta := \{\mathcal{R} \in \mathsf{Rep}(\mathcal{O},\alpha) \mid \beta \in \mathcal{R}\},$$
$$\mathfrak{R}^-_\beta := \{\mathcal{R} \in \mathsf{Rep}(\mathcal{O},\alpha) \mid \beta \notin \mathcal{R}\}.$$

The axiom β is called a *cut axiom* iff for every $\gamma \in \mathcal{O}$ it holds that $|\mathfrak{R}^+_\beta| - |\mathfrak{R}^-_\beta| \leq |\mathfrak{R}^+_\gamma| - |\mathfrak{R}^-_\gamma|$.

Note that this definition allows a flexibility in that the set of repairs might not be partitioned in half, but the cut axiom gets as close to it as possible. The idea behind the cut axiom is that, by verifying its correctness, we can immediately cut the search space (almost) in half. Specifically, if α is correct, then we know that the right repair is among $\mathcal{R}^+_\alpha$, and if it is wrong, we should focus only on $\mathcal{R}^-_\alpha$. Hence, the first question is how to compute such an axiom. Unfortunately, it turns out that deciding whether an axiom is a cut axiom is coNP-hard already for the very simple sublogic of $\mathcal{HL}$ which disallows conjunctions.

Theorem 12 *Let $\mathcal{O}$ be an $\mathcal{HL}$ ontology, $\beta \in \mathcal{O}$ an axiom, and α an unwanted consequence of $\mathcal{O}$. Deciding whether β is a cut axiom is co NP-hard.*

Proof We prove this by a reduction from the coNP-complete *repair without edge* (RWE) problem[22]: given a graph $G = (V, E)$, nodes $s, t \in V$, and an edge $(v, w) \in E$, decide if there is a maximal subgraph $G' = (V, F)$ of G such that t is not reachable from s and $(v, w) \notin F$.

Let $n := |E|$. Note that there can exist at most 2^{n-1} maximal subgraphs of the form we seek that contain (v, w) and at most 2^{n-1} that do not contain this edge. Assuming w.l.o.g. that there are at least two maximal subgraphs, we construct an ontology $\mathcal{M}$, which simulates an extension of G obtained by adding $2n$ new vertices $z_1, \dots, z_{2n}$, and the edges

$$E' := \{v \to z_i, z_i \to w \mid 1 \leq i \leq 2n\}.$$

Formally, for every $u \in V \cup \{z_1, \dots, z_{2n}\}$, we create a concept name A_u, and construct the TBox $\mathcal{M} := \{A_u \sqsubseteq A_{u'} \mid (u, u') \in E \cup E'\}$. Clearly, the size of $\mathcal{M}$ is linear on the size of G. For a subgraph (V, F) of G, we define $\mathcal{M}_F$ to be the sub-ontology of $\mathcal{M}$ restricted to edges appearing in F.

This ontology has the following property. For every maximal subgraph (V, F) of G where t is not reachable from s, (i) if $(v, w) \in F$, then $\mathcal{M}_F$ is the only repair of $A_s \sqsubseteq A_t$ w.r.t. $\mathcal{M}$ that contains all edges from F; that is, there is a one-to-one correspondence between the maximal subgraphs of G that remove reachability and the repairs of $\mathcal{M}$ that contain $A_v \sqsubseteq A_w$; and (ii) if $(v, w) \notin F$, there exist 2^{2n} different repairs of $A_s \sqsubseteq A_t$ w.r.t. $\mathcal{M}$ that contain all edges in F; in particular, exactly 2^{2n-1} of them contain the axiom $A_v \to A_{z_1}$.

In particular, if G has m maximal subgraphs that contain (v, w) and ℓ that do not contain this edge, then $\mathcal{M}$ will have $m + \ell \cdot 2^{2n-1}$ repairs containing $A_v \sqsubseteq A_{z_1}$ and $\ell \cdot 2^{2n-1}$ repairs not containing the axiom. Moreover, every other edge of G will appear in at most 2^{n-1} repairs of $\mathcal{M}$, and all the edges in E' will be in exactly the same number of repairs as $A_v \sqsubseteq A_{z_1}$. Thus, $A_v \sqsubseteq A_{z_1}$ is a cut axiom w.r.t. $\mathcal{M}$ iff $\ell \geq 1$; that is, iff there is at least one maximal subgraph of G avoiding the paths from s to t which does not contain (v, w). □

In summary, this theorem tells us that it is not possible to efficiently construct a decision tree about the axioms proposed to the domain expert for analysis, which minimises the overall number of questions needed to guarantee that a repair is obtained. Still, as explained already, the most expensive resources are exactly those of the domain expert. The decision tree could, in fact, be constructed in advance, as a preprocessing step, or even in parallel as the expert is understanding and verifying the first proposed culprits.

6 Conclusions

We have studied the problem of dealing with and managing errors in lightweight description logic ontologies. For the former problem, we extend the idea of inconsistency-tolerant reasoning—defining different kinds of semantics depending on the use of the repairs—to deal with arbitrary errors that may, or may not, be connected to inconsistency. Analysing the complexity of three error-tolerant semantics, we have shown that in most cases this kind of reasoning becomes intractable, although we identified a few tractable cases and provided effective algorithms for handling them. Interestingly, we have shown that the cause for intractability is more subtle than just the number of potential repairs as previously argued.

For the second problem, we proposed to partition the space of all repairs as closely as possible in halves, to help a knowledge engineer (KE) to find the *correct* repair through a binary search-like process. While a decision plan can be constructed offline before being presented to the KE, just deciding whether an axiom produces an adequate partition of the repairs is NP-hard.

One issue that was not considered in this paper is the multiplicity of errors. Indeed, it is likely that more than one unwanted consequence is detected between two consecutive versions of an ontology. We note that the notions of justification and repair can be easily extended to consider several consequences, and all the hardness results still apply to the more general situation. Whether the tractable cases remain so is still to be verified.

As future work we want to pursue two different goals. On the one hand, we plan to extend our study to more complex entailments (e.g., conjunctive queries), while searching for conditions to regain tractability. On the other hand, we will develop methods for dealing efficiently with these error-tolerant reasoning tasks, despite their computational complexity. One potential approach to achieve this is to exploit the properties of very efficient SAT solvers. We note that SAT-based techniques have already shown promising results in the areas of axiom pinpointing and inconsistent query answering[1, 10, 27].

Acknowledgements Open access funding provided by Università degli Studi di Milano - Bicocca within the CRUI-CARE Agreement. The author wants to thank an anonymous reviewer for spotting several embarrasing mistakes in a previous version of this paper. Their thorough reading and insistence really helped improve it.

References

1. Arif MF, Mencía C, Ignatiev A, Manthey N, Peñaloza R, Marques-Silva J (2016) BEACON: an efficient SAT-based tool for debugging *EL*+ ontologies. In: Creignou N, Berre DL (eds) Proceedings of the 19th international conference on theory and applications of satisfiability testing (SAT 2016), Springer, New York, LNCS, vol 9710, pp 521–530. https://doi.org/10.1007/978-3-319-40970-2_32
2. Artale A, Calvanese D, Kontchakov R, Zakharyaschev M (2009) The dl-lite family and relations. J Artif Intell Res 36:1–69. https://doi.org/10.1613/jair.2820
3. Baader F, Brandt S, Lutz C (2005) Pushing the *EL* envelope. In: Kaelbling L, Saffiotti A (eds) In: Proceedings of the 19th international joint conference on artificial intelligence (IJCAI'05), Professional Book Center, pp 364–369
4. Baader F, Calvanese D, McGuinness D, Nardi D, Patel-Schneider P (eds) (2007a) The description logic handbook: theory, implementation, and applications, 2nd edn. Cambridge University Press, Cambridge
5. Baader F, Peñaloza R, Suntisrivaraporn B (2007b) Pinpointing in the description logic EL+. In: Hertzberg J, Beetz M, Englert R (eds) Proceedings of the 30th annual German conference on AI (KI'07), Springer, New York, LNCS, vol 4667, pp 52–67. https://doi.org/10.1007/978-3-540-74565-5_7
6. Baader F, Brandt S, Lutz C (2008) Pushing the *EL* envelope further. In: Clark KG, Patel-Schneider PF (eds) Proceedings of OWLED 2008 Workshop on OWL: experiences and directions, CEUR-WS.org, CEUR Workshop Proceedings, vol 496
7. Bertossi LE, Hunter A, Schaub T (eds) (2005) Inconsistency Tolerance, LNCS, vol 3300. Springer. https://doi.org/10.1007/b104925
8. Bienvenu M, Bourgaux C (2016) Inconsistency-tolerant querying of description logic knowledge bases. In: Pan JZ, Calvanese D, Eiter T, Horrocks I, Kifer M, Lin F, Zhao Y (eds) 12th International reasoning web summer school: logical foundation of knowledge graph construction and query answering, Tutorial Lectures, Springer, LNCS, vol 9885, pp 156–202. https://doi.org/10.1007/978-3-319-49493-7_5
9. Bienvenu M, Rosati R (2013) Tractable approximations of consistent query answering for robust ontology-based data access. In: Rossi F (ed) Proceedings of the 23rd international joint conference on artificial intelligence (IJCAI'13), AAAI Press/IJCAI, pp 775–781
10. Bienvenu M, Bourgaux C, Goasdoué F (2019) Computing and explaining query answers over inconsistent dl-lite knowledge bases. J Artif Intell Res 64:563–644. https://doi.org/10.1613/jair.1.11395
11. Calvanese D, De Giacomo G, Lembo D, Lenzerini M, Rosati R (2007) Tractable reasoning and efficient query answering in description logics: the DL-Lite family. J Autom Reason 39(3):385–429
12. Cuenca Grau B, Horrocks I, Kazakov Y, Sattler U (2007) Just the right amount: extracting modules from ontologies. In: Williamson CL, Zurko ME, Patel-Schneider PF, Shenoy PJ (eds) Proceedings of the 16th international conference on World Wide Web, WWW 2007, ACM, pp 717–726. https://doi.org/10.1145/1242572.1242669
13. Cuenca Grau B, Horrocks I, Kazakov Y, Sattler U (2008) Modular reuse of ontologies: theory and practice. J Artif Intell Res 31:273–318. https://doi.org/10.1613/jair.2375
14. Eiter T, Gottlob G (1995) Identifying the minimal transversals of a hypergraph and related problems. SIAM J Comput 24(6):1278–1304. https://doi.org/10.1137/S0097539793250299
15. Eiter T, Lukasiewicz T, Predoiu L (2016) Generalized consistent query answering under existential rules. In: Baral C, Delgrande JP, Wolter F (eds) Proceedings of the fifteenth international conference on principles of knowledge representation and reasoning (KR 2016), AAAI Press, pp 359–368
16. Gallo G, Longo G, Pallottino S, Nguyen S (1993) Directed hypergraphs and applications. Discr Appl Math 42(2):177–201. https://doi.org/10.1016/0166-218X(93)90045-P. http://www.sciencedirect.com/science/article/pii/0166218X9390045P
17. Johnson DS, Papadimitriou CH, Yannakakis M (1988) On generating all maximal independent sets. Inf Process Lett 27(3):119–123. https://doi.org/10.1016/0020-0190(88)90065-8
18. Kazakov Y, Skocovský P (2018) Enumerating justifications using resolution. In: Galmiche D, Schulz S, Sebastiani R (eds) Proceedings of the 9th international joint conference on automated reasoning (IJCAR 2018), Springer, New York, LNCS, vol 10900, pp 609–626. https://doi.org/10.1007/978-3-319-94205-6_40
19. Lembo D, Lenzerini M, Rosati R, Ruzzi M, Savo DF (2010) Inconsistency-tolerant semantics for description logics. In: Hitzler P, Lukasiewicz T (eds) Proceedings of the 4th international conference on web reasoning and rule systems (RR 2010), Springer, New York, LNCS, vol 6333, pp 103–117. https://doi.org/10.1007/978-3-642-15918-3_9
20. Ludwig M, Peñaloza R (2014) Error-tolerant reasoning in the description logic EL. Spring LNCS 8761:107–121. https://doi.org/10.1007/978-3-319-11558-0_8
21. Peñaloza Nyssen R (2009) Axiom pinpointing in description logics and beyond. PhD thesis, Technische Universität Dresden, Germany. http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-24743

22. Peñaloza R (2017) Inconsistency-tolerant instance checking in tractable description logics. Spring LNCS 10364:215–229. https://doi.org/10.1007/978-3-319-61252-2_15
23. Peñaloza R (2019) Making decisions with knowledge base repairs. In: Torra V, Narukawa Y, Pasi G, Viviani M (eds) Proceedings of the 16th international conference on modeling decisions for artificial intelligence (MDAI 2019), Springer, LNCS, vol 11676, pp 259–271. https://doi.org/10.1007/978-3-030-26773-5_23
24. Peñaloza R, Sertkaya B (2017) Understanding the complexity of axiom pinpointing in lightweight description logics. Artif Intell 250:80–104. https://doi.org/10.1016/j.artint.2017.06.002
25. Peñaloza R, Mencía C, Ignatiev A, Marques-Silva J (2017) Lean kernels in description logics. In: Blomqvist E, Maynard D, Gangemi A, Hoekstra R, Hitzler P, Hartig O (eds) Proceeding of the 14th Semantic Web Conference (ESWC 2017), LNCS, vol 10249, pp 518–533. https://doi.org/10.1007/978-3-319-58068-5_32
26. Price C, Spackman K (2000) Snomed clinical terms. Br J Healthcare Comput Inf Manag 17(3):27–31
27. Sebastiani R, Vescovi M (2009) Axiom pinpointing in lightweight description logics via Horn-SAT encoding and conflict analysis. Spring LNCS 5663:84–99. https://doi.org/10.1007/978-3-642-02959-2_6. https://doi.org/10.1007/978-3-642-02959-2
28. Suntisrivaraporn B (2008) Module extraction and incremental classification: A pragmatic approach for ontologies. In: Bechhofer S, Hauswirth M, Hoffmann J, Koubarakis M (eds) Proceedings of the 5th European Semantic Web Conference (ESWC 2008), Springer, LNCS, vol 5021, pp 230–244. https://doi.org/10.1007/978-3-540-68234-9_19
29. Suntisrivaraporn B (2009) Polynomial time reasoning support for design and maintenance of large-scale biomedical ontologies. PhD thesis, Technische Universität Dresden, Germany. http://hsss.slub-dresden.de/deds-access/hsss.urlmapping.MappingServlet?id=1233830966436-5928