Technische Universität Dresden
Institute for Theoretical Computer Science
Chair for Automata Theory

LTCS–Report

Error-Tolerant Reasoning in the Description Logic $\mathcal{EL}$

Michel Ludwig Rafael Peñaloza

LTCS-Report 14–11

Postal Address:
Lehrstuhl für Automatentheorie
Institut für Theoretische Informatik
TU Dresden
01062 Dresden

http://lat.inf.tu-dresden.de

Visiting Address:
Nöthnitzer Str. 46
Dresden

Error-Tolerant Reasoning in the Description Logic $\mathcal{EL}$*

Michel Ludwig
Rafael Peñaloza
Theoretical Computer Science, TU Dresden, Germany
Center for Advancing Electronics Dresden
{michel,penaloza}@tcs.inf.tu-dresden.de

Abstract

Developing and maintaining ontologies is an expensive and error-prone task. After an error is detected, users may have to wait for a long time before a corrected version of the ontology is available. In the meantime, one might still want to derive meaningful knowledge from the ontology, while avoiding the known errors. We study error-tolerant reasoning tasks in the description logic $\mathcal{EL}$. While these problems are intractable, we propose methods for improving the reasoning times by precompiling information about the known errors and using proof-theoretic techniques for computing justifications. A prototypical implementation shows that our approach is feasible for large ontologies used in practice.

1 Introduction

Description Logics (DLs) [3] are a family of knowledge representation formalisms that have been successfully used to model many application domains, specifically in the bio-medical areas. They are also the logical formalism underlying the standard ontology language for the semantic web OWL 2 [32]. As a consequence, more and larger ontologies are being built using these formalisms. Ontology engineering is expensive and error-prone; the combination of knowledge from multiple experts, and misunderstandings between them and the knowledge engineers may lead to subtle errors that are hard to detect. For example, several iterations of Snomed CT [14, 31] classified *amputation of finger* as a subclass of *amputation of hand* [7, 8].

*Partially supported by DFG within the Cluster of Excellence 'cfAED'.

Since domain knowledge is needed for correcting an unwanted consequence, and its causes might not be obvious, it can take long before a corrected version of an ontology is released. For example, new versions of SNOMED are released every six months; one should then expect to wait at least that amount of time before an error is resolved. During that time, users should still be able to derive meaningful consequences from the ontology, while avoiding the known errors.

A related problem is *inconsistency-tolerant reasoning*, based on consistent query answering from databases [1, 9], where the goal is to obtain meaningful consequences from an inconsistent ontology $\mathcal{O}$. Inconsistency is clearly an unwanted consequence from an ontology, but it is not the only one; for instance, while SNOMED is consistent, we would still like to avoid the erroneous subclass relationship between amputation of finger and amputation of hand. We generalize the idea of inconsistency-tolerant reasoning to *error-tolerant reasoning* in which other unwanted consequences, beyond inconsistency, are considered.

We focus mainly on two kinds of error-tolerant semantics; namely brave and cautious semantics. Intuitively, *cautious semantics* refer to consequences that follow from *all* the possible repairs of $\mathcal{O}$; this guarantees that, however the ontology is repaired, the consequence will still follow. For some consequences, one might only be interested in guaranteeing that it follows from *at least one* repair; this defines the *brave semantics*. As usual in inconsistency-tolerant reasoning, the *repairs* are maximal subontologies of $\mathcal{O}$ that do not entail the unwanted consequence. Notice that brave semantics are not closed under entailment; e.g., the conjunction of two brave consequences is not necessarily a brave consequence itself. However, these consequences are still useful to guarantee that a *wanted* consequence can still be derived from at least one repair, among other cases. We also consider the IAR semantics, proposed in [22] as a means to efficiently approximate cautious reasoning; see also [11, 30].

In this paper, we focus on subsumption between concepts w.r.t. a TBox in $\mathcal{EL}$, which is known to be polynomial [13]. As every $\mathcal{EL}$ TBox is consistent, considering inconsistency-tolerant semantics makes no sense in this setting. On the other hand, SNOMED CT and other large-scale ontologies are written in tractable extensions of this logic, and being able to handle errors written in them is a relevant problem for knowledge representation and ontology development.

We show that error-tolerant reasoning in $\mathcal{EL}$ is hard. More precisely, brave semantics is NP-complete, and cautious and IAR semantics are coNP-complete. These results are similar to the complexity of inconsistency-tolerant semantics in inexpressive logics [10, 30]. We also show that hardness does not depend only on the number of repairs: there exist errors with polynomially many repairs, for which error-tolerant reasoning requires super-polynomial time (unless P = NP).

To improve the time needed for error-tolerant reasoning, we propose to precompute the information on the causes of the error. We first annotate every axiom

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

Syntax	Semantics
$\top$	$\Delta^\mathcal{I}$
$C \sqcap D$	$C^\mathcal{I} \cap D^\mathcal{I}$
$\exists r.C$	$\{x \in \Delta^\mathcal{I} \mid \exists y \in \Delta^\mathcal{I} : (x, y) \in r^\mathcal{I} \wedge y \in C^\mathcal{I}\}$

with the repairs to which it belongs. We then use a proof-theoretic approach, coupled with this annotated ontology, to derive error-tolerant consequences. We demonstrate the practical applicability of our approach for brave and cautious reasoning by applying a prototype-implementation on large ontologies used in practice. This is an extended version of [25], containing all proofs and details.

2 Preliminaries

We first briefly recall the DL $\mathcal{EL}$. Given two disjoint sets $\mathsf{N_C}$ and $\mathsf{N_R}$ of *concept-*, and *role-names*, respectively, concepts are constructed by $C ::= A \mid C \sqcap C \mid \exists r.C$, where $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$. A *TBox* is a finite set of *general concept inclusions (GCIs)* of the form $C \sqsubseteq D$, where C, D are concepts. The TBox is in *normal form* if all its GCIs are of the form $A \sqsubseteq \exists r.B$, $\exists r.A \sqsubseteq B$, or $A_1 \sqcap \ldots \sqcap A_n \sqsubseteq B$ with $n \geq 1$ and $A, A_1, \ldots, A_n, B \in \mathsf{N_C} \cup \{\top\}$.

The semantics of $\mathcal{EL}$ is defined through *interpretations* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where $\Delta^\mathcal{I}$ is a non-empty *domain* and $\cdot^\mathcal{I}$ maps each $A \in \mathsf{N_C}$ to a set $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$ and every $r \in \mathsf{N_R}$ to a binary relation $r^\mathcal{I}$ over $\Delta^\mathcal{I}$. This mapping is extended to arbitrary concepts as shown in Table 1. The interpretation $\mathcal{I}$ is a *model* of the TBox $\mathcal{T}$ $C^\mathcal{I} \subseteq D^\mathcal{I}$ for every $C \sqsubseteq D \in \mathcal{T}$. The main reasoning problem is to decide *subsumption* [2,13]: C is *subsumed* by D w.r.t. $\mathcal{T}$ (denoted $C \sqsubseteq_\mathcal{T} D$) if $C^\mathcal{I} \subseteq D^\mathcal{I}$ holds for every model $\mathcal{I}$ of $\mathcal{T}$. $\mathcal{HL}$ is the sublogic of $\mathcal{EL}$ that does not allow existential restrictions; it is a syntactic variant of Horn logic: every Horn clause can be seen as an $\mathcal{HL}$ GCI. An $\mathcal{HL}$ TBox is a *core* TBox if all its axioms are of the form $A \sqsubseteq B$ with $A, B \in \mathsf{N_C}$.

Error-tolerant reasoning refers to the task of deriving meaningful consequences from a TBox that is known to contain errors. In the scope of this paper, an erroneous consequence refers to an error in a subsumption relation. If the TBox $\mathcal{T}$ entails an unwanted subsumption $C \sqsubseteq_\mathcal{T} D$, then we are interested in finding the ways in which this consequence can be avoided. To define error-tolerant reasoning formally, we need the notion of a repair.

Definition 1 (repair)**.** Let $\mathcal{T}$ be an $\mathcal{EL}$ TBox and $C \sqsubseteq_\mathcal{T} D$. A *repair of* $\mathcal{T}$ *w.r.t.* $C \sqsubseteq D$ is a maximal (w.r.t. set inclusion) subset $\mathcal{R} \subseteq \mathcal{T}$ such that $C \not\sqsubseteq_\mathcal{R} D$. The set of all repairs of $\mathcal{T}$ w.r.t. $C \sqsubseteq D$ is denoted by $\mathsf{Rep}_\mathcal{T}(C \sqsubseteq D)$.

We will usually consider a fixed TBox $\mathcal{T}$, and hence say that $\mathcal{R}$ is a repair w.r.t. $C \sqsubseteq D$, or even simply a repair, if the consequence is clear from the context.

Example 2. The repairs of $\mathcal{T} = \{A \sqsubseteq \exists r.X, \exists r.X \sqsubseteq B, A \sqsubseteq Y, Y \sqsubseteq B, A \sqsubseteq B'\}$ w.r.t. the consequence $A \sqsubseteq B$ are the sets $\mathcal{R}_i := \mathcal{T} \setminus \mathcal{S}_i, 1 \leq i \leq 4$, where $\mathcal{S}_1 = \{A \sqsubseteq \exists r.X, A \sqsubseteq Y\}$, $\mathcal{S}_2 = \{A \sqsubseteq \exists r.X, Y \sqsubseteq B\}$, $\mathcal{S}_3 = \{\exists r.X \sqsubseteq B, A \sqsubseteq Y\}$, and $\mathcal{S}_4 = \{\exists r.X \sqsubseteq B, Y \sqsubseteq B\}$.

The number of repairs w.r.t. a consequence may be exponential, even for core TBoxes [28]. Each of these repairs is a potential way of avoiding the unwanted consequence; however, it is impossible to know *a priori* which is the best one to use for further reasoning tasks. One common approach is to be *cautious* and consider only those consequences that follow from *all* repairs. Alternatively, one can consider *brave* consequences: those that follow from at least one repair.

Definition 3 (cautious, brave)**.** Let $\mathcal{T}$ be a TBox, $C \sqsubseteq_{\mathcal{T}} D$, and C', D' be two concepts. C' is *bravely subsumed* by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$ if there is a repair $\mathcal{R} \in \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$ such that $C' \sqsubseteq_{\mathcal{R}} D'$; C' is *cautiously subsumed* by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$ if for every repair $\mathcal{R} \in \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$ it holds that $C' \sqsubseteq_{\mathcal{R}} D'$. If $\mathcal{T}$ or $C \sqsubseteq D$ are clear from the context, we usually omit them.

Example 4. Let $\mathcal{T}$, $\mathcal{R}_1, \ldots \mathcal{R}_4$ be as in Example 2. A is bravely but not cautiously subsumed by $Y \sqcap B'$ w.r.t. $\mathcal{T}$ and $A \sqsubseteq B$ since $A \sqsubseteq_{\mathcal{R}_2} Y \sqcap B'$ but $A \not\sqsubseteq_{\mathcal{R}_1} Y \sqcap B'$.

In the context of inconsistency-tolerant reasoning, other kinds of semantics which have better computational properties have been proposed [11, 22, 30]. Among these are the so-called IAR semantics, which consider the consequences that follow from the intersection of all repairs. Formally, C' is *IAR subsumed* by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$ if $C' \sqsubseteq_{\mathcal{Q}} D'$, where $\mathcal{Q} := \bigcap_{\mathcal{R} \in \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)} \mathcal{R}$.

Example 5. Let $\mathcal{T}$ and $\mathcal{R}_1, \ldots, \mathcal{R}_4$ be as in Example 2. Then A is IAR subsumed by B' w.r.t. $\mathcal{T}$ and $A \sqsubseteq B$ as $A \sqsubseteq B' \in \bigcap_{i=1}^{4} \mathcal{R}_i$.

A notion dual to repairs is that of MinAs, or justifications [7, 18]. A *MinA* for $C \sqsubseteq_{\mathcal{T}} D$ is a minimal (w.r.t. set inclusion) subset $\mathcal{M}$ of $\mathcal{T}$ such that $C \sqsubseteq_{\mathcal{M}} D$. We denote as $\mathsf{MinA}_{\mathcal{T}}(C \sqsubseteq D)$ the set of all MinAs for $C \sqsubseteq_{\mathcal{T}} D$. There is a close connection between repairs and MinAs for error-tolerant reasoning.

Theorem 6. *Let $\mathcal{T}$ be a TBox, C, C', D, D' concepts with $C \sqsubseteq_{\mathcal{T}} D$. Then*

1. *C' is cautiously subsumed by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$ iff for every repair $\mathcal{R} \in \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$ there is an $\mathcal{M}' \in \mathsf{MinA}_{\mathcal{T}}(C' \sqsubseteq D')$ with $\mathcal{M}' \subseteq \mathcal{R}$; and*

2. *C' is bravely subsumed by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$ iff there is a repair $\mathcal{R} \in \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$ and a MinA $\mathcal{M}' \in \mathsf{MinA}_{\mathcal{T}}(C' \sqsubseteq D')$ with $\mathcal{M}' \subseteq \mathcal{R}$.*

This theorem will be useful for developing a more efficient error-tolerant reasoning algorithm. Before describing this algorithm in detail, we study the complexity of this kind of reasoning.

3 Complexity

We show that deciding cautious and IAR subsumptions is intractable already for core TBoxes. Deciding brave subsumptions is intractable for $\mathcal{EL}$, but tractable for $\mathcal{HL}$. We first prove the latter claim using directed hypergraphs, which generalize graphs by connecting sets of nodes, rather than just nodes.

A *directed hypergraph* is a pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a non-empty set of *nodes*, and $\mathcal{E}$ is a set of *directed hyperedges* $e = (S, S')$, with $S, S' \subseteq \mathcal{V}$. Given $S, T \subseteq \mathcal{V}$, a *path* from S to T in $\mathcal{G}$ is a set of hyperedges $\{(S_i, T_i) \in \mathcal{E} \mid 1 \leq i \leq n\}$ such that for every $1 \leq i \leq n$, $S_i \subseteq S \cup \bigcup_{j=1}^{n-1} T_j$, and $T \subseteq \bigcup_{i=1}^{n} T_i$ hold. The *reachability problem* in hypergraphs consists in deciding the existence of a path from S to T in $\mathcal{G}$. This problem is decidable in polynomial time on $|\mathcal{V}|$ [16].

Recall that $\mathcal{HL}$ concepts are conjunctions of concept names; we can represent $C = A_1 \sqcap \cdots \sqcap A_m$ as its set of conjuncts $S_C = \{A_1, \ldots, A_m\}$. Each GCI $C \sqsubseteq D$ yields a directed hyperedge (S_C, S_D) and every $\mathcal{HL}$-TBox $\mathcal{T}$ forms a directed hypergraph $\mathcal{G}_\mathcal{T}$. Then $C \sqsubseteq_\mathcal{T} D$ iff there is a path from S_C to S_D in $\mathcal{G}_\mathcal{T}$.

Theorem 7. *Brave subsumption in $\mathcal{HL}$ can be decided in polynomial time on the size of the TBox.*

Proof. Let $\mathcal{T}$ be an $\mathcal{HL}$ TBox, and C, C', D, D' be $\mathcal{HL}$ concepts. C' is bravely subsumed by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$ iff there is a path from $S_{C'}$ to $S_{D'}$ in $\mathcal{G}_\mathcal{T}$ that does not contain any path from S_C to S_D. If no such path exists, then (i) every path from $S_{C'}$ to $S_{D'}$ passes through S_D, and (ii) every path from $S_{C'}$ to S_D passes through S_C. We need to verify whether any of these two statements is violated. The existence of a path that does not pass through a given set is decidable in polynomial time. □

However, for $\mathcal{EL}$ this problem is NP-complete. To prove this we adapt an idea from [27] for reducing the NP-hard *more minimal valuations* (MMV) problem [7, 15]: deciding, for a monotone Boolean formula φ and a set $\mathfrak{V}$ of minimal valuations satisfying φ, if there are other minimal valuations $V \notin \mathfrak{V}$ satisfying φ.

Theorem 8. *Brave subsumption in $\mathcal{EL}$ is* NP*-complete.*

We now show that the cautious and IAR semantics are intractable already for core TBoxes. This is a consequence of the intractability of the following problem.

Definition 9 (axiom relevance)**.** The *axiom relevance* problem consists in deciding, given a core TBox $\mathcal{T}$, $A \sqsubseteq B \in \mathcal{T}$, and $A_0 \sqsubseteq_{\mathcal{T}} B_0$, whether there is a repair $\mathcal{R}$ of $\mathcal{T}$ w.r.t. $A_0 \sqsubseteq B_0$ such that $A \sqsubseteq B \notin \mathcal{R}$.

Lemma 10. *Axiom relevance is* NP*-hard.*

Proof. We reduce the NP-hard *path-via-node* problem [21]: given a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and nodes $s, t, m \in \mathcal{V}$, decide if there is a simple path from s to t in $\mathcal{G}$ that goes through m. Given an instance of the path-via-node problem, we introduce a concept name A_v for every $v \in (\mathcal{V} \setminus \{m\}) \cup \{m_1, m_2\}$, and build the core TBox

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

There is a simple path from s to t in $\mathcal{G}$ through m iff there is a repair $\mathcal{R}$ of $\mathcal{T}$ w.r.t. $A_s \sqsubseteq A_t$ with $A_{m_1} \sqsubseteq A_{m_2} \notin \mathcal{R}$. □

Theorem 11. *Cautious subsumption and IAR subsumption w.r.t. core, $\mathcal{HL}$ or $\mathcal{EL}$ TBoxes are co*NP*-complete.*

Proof. If C is not cautiously subsumed by D, we can guess a set $\mathcal{R}$ and verify in polynomial time that $\mathcal{R}$ is a repair and $C \not\sqsubseteq_{\mathcal{R}} D$. If C is not IAR subsumed by D, we can guess a set $\mathcal{Q} \subseteq \mathcal{T}$, and for every GCI $C_i \sqsubseteq D_i \notin \mathcal{Q}$ a set $\mathcal{R}_i$ such that $C_i \sqsubseteq D_i \notin \mathcal{R}_i$. Verifying that each $\mathcal{R}_i$ is a repair and $C \not\sqsubseteq_{\mathcal{Q}} D$ is polynomial. Thus both problems are in coNP. To show hardness, for a GCI $C \sqsubseteq D \in \mathcal{T}$, there is a repair $\mathcal{R}$ such that $C \sqsubseteq D \notin \mathcal{R}$ iff $C \not\sqsubseteq_{\mathcal{R}} D$ iff C is neither cautiously nor IAR subsumed by D. By Lemma 10 both problems are coNP-hard. □

The hardness of error-tolerant reasoning is usually attributed to the fact that there can exist exponentially many repairs for a given consequence. However, this argument is incomplete. For instance, brave reasoning remains polynomial in $\mathcal{HL}$, although consequences may have exponentially many repairs already in this logic. We show now that cautious and brave subsumption are also hard on the *number of repairs*; i.e., they are not what we call *repair-polynomial.*

Definition 12 (repair-polynomial)**.** An error-tolerant problem w.r.t. a TBox $\mathcal{T}$ and a consequence $C \sqsubseteq D$ is *repair-polynomial* if it can be solved by an algorithm that runs in polynomial time on the size of both $\mathcal{T}$ and $\mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$.

Theorem 13. *Unless* P = NP*, cautious and brave subsumption of C' by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$ in $\mathcal{EL}$ are not repair-polynomial.*

Algorithm 1 Repairs entailing $C' \sqsubseteq D'$

Input: Unwanted consequence $C \sqsubseteq_{\mathcal{T}} D$, concepts C', D'
Output: $\mathfrak{R} \subseteq \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$: repairs entailing $C' \sqsubseteq D'$

```
  𝔑 ← Rep_T(C ⊑ D)
  for each R ∈ Rep_T(C ⊑ D) do
      if C' ⋢_R D' then
          𝔑 ← 𝔑 \ {R}
  return 𝔑
```

The proof adapts the construction from Theorem 8 to reduce the problem of enumerating maximal valuations that falsify a formula to deciding cautious subsumption. The number of repairs obtained from the reduction is polynomial on the number of maximal valuations that falsify the formula. Since this enumeration cannot be solved in time polynomial on the number of maximal falsifiers, cautious reasoning can also not be performed in time polynomial on the number of repairs. An analogous argument is used for brave reasoning. All the details can be found in the appendix. Thus, error-tolerant reasoning is hard even if only polynomially many repairs exist; i.e., there are cases where $|\mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)|$ is polynomial on $|\mathcal{T}|$, but brave and cautious reasoning require super-polynomial time. The culprit for hardness is not the number of repairs *per se*, but rather the relationships among these repairs.

We now propose a method for improving the reasoning times, by precomputing the set of all repairs, and using this information effectively.

4 Precompiling Repairs

A naïve solution for deciding brave or cautious subsumptions would be to first enumerate all repairs and then check which of them entail the relation (the set $\mathfrak{R}$ in Algorithm 1). C' is then bravely or cautiously subsumed by D' iff $\mathfrak{R} \neq \emptyset$ or $\mathfrak{R} = \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$, respectively. Each test $C' \sqsubseteq_{\mathcal{R}} D'$ requires polynomial time on $|\mathcal{R}| \leq |\mathcal{T}|$ [13], and exactly $|\mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)|$ such tests are performed. The **for** loop in the algorithm thus needs polynomial time on the sizes of $\mathcal{T}$ and $\mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$. From Theorem 13 it follows that the first step, namely the computation of all the repairs, must be expensive. In particular, these repairs cannot be enumerated in output-polynomial time; i.e., in time polynomial on the input *and the output* [17].

Corollary 14. *The set of repairs for an $\mathcal{EL}$ TBox $\mathcal{T}$ w.r.t. $C \sqsubseteq D$ cannot be enumerated in output polynomial time, unless* P = NP.

For any given error, one would usually try to decide whether several brave or cautious consequences hold. It thus makes sense to improve the execution time of each of these individual reasoning tasks by avoiding a repetition of the first, expensive, step.

The set of repairs can be computed in exponential time on the size of $\mathcal{T}$; this bound cannot be improved in general since (i) there might exist exponentially many such repairs, and (ii) they cannot be enumerated in output polynomial time. However, this set only needs to be computed once, when the error is found, and can then be used to improve the reasoning time for all subsequent subsumption relations. Once $\mathsf{Rep}_\mathcal{T}(C \sqsubseteq D)$ is known, Algorithm 1 computes $\mathfrak{R}$, and hence decides brave and cautious reasoning, in time polynomial on $|\mathcal{T}| \cdot |\mathsf{Rep}_\mathcal{T}(C \sqsubseteq D)|$. It is important to notice that this does not violate the result that cautious and brave reasoning are not repair-polynomial. The main difference is that this variant of Algorithm 1 does not need to compute the repairs; they are already given.

Clearly, Algorithm 1 does more than merely deciding cautious and brave consequences. Indeed, it computes the set of all repairs that entail $C' \sqsubseteq D'$. This information can be used to decide more complex reasoning tasks. For instance, one may be interested in knowing whether the consequence follows from *most*, or *at least k* repairs, to mention just two possible inferences. IAR semantics can also be decided in polynomial time on $\mathcal{T}$ and $\mathsf{Rep}_\mathcal{T}(C \sqsubseteq D)$: simply compute $\mathcal{Q} = \bigcap_{\mathcal{R} \in \mathsf{Rep}_\mathcal{T}(C \sqsubseteq D)} \mathcal{R}$, and test whether $C' \sqsubseteq_\mathcal{Q} D'$ holds. The first step needs polynomial time on $\mathsf{Rep}_\mathcal{T}(C \sqsubseteq D)$ while the second is polynomial on $\mathcal{Q} \subseteq \mathcal{T}$.

As we have seen, precompiling the set of repairs already yields an improvement on the time required for deciding error-tolerant subsumption relations. However, there are some obvious drawbacks for this idea. In particular, storing and maintaining a possibly exponential set of TBoxes can be a challenge for the knowledge engineer. Moreover, this method does not scale well for handling multiple errors that are found at different time points. When a new error is detected, the repairs of all the TBoxes need to be computed, potentially causing the introduction of redundant TBoxes that must later be removed. We improve on this solution by structuring all the repairs into a single labelled TBox.

Let $\mathsf{Rep}_\mathcal{T}(C \sqsubseteq D) = \{\mathcal{R}_1, \ldots, \mathcal{R}_n\}$. We label every GCI $E \sqsubseteq F \in \mathcal{T}$ with $\mathsf{lab}(E \sqsubseteq F) = \{i \mid E \sqsubseteq F \in \mathcal{R}_i\}$. Conversely, for every subset $I \subseteq \{1, \ldots, n\}$ we define the TBox $\mathcal{T}_I = \{E \sqsubseteq F \in \mathcal{T} \mid \mathsf{lab}(E \sqsubseteq F) = I\}$. A set I is a *component* if $\mathcal{T}_I \neq \emptyset$. Every axiom belongs to exactly one component and hence the number of components is bounded by $|\mathcal{T}|$. One can represent these components using only polynomial space and all repairs can be read from them via a directed acyclic graph expressing dependencies between components. For simplicity we keep the representation as subsets of $\{1, \ldots, n\}$.

The labelled TBox has full information on the repairs, and on their relationship with each other. For $\mathcal{S} \subseteq \mathcal{T}$, $\mathsf{lab}(\mathcal{S}) := \bigcap_{E \sqsubseteq F \in \mathcal{S}} \mathsf{lab}(E \sqsubseteq F)$ yields all repairs containing $\mathcal{S}$. If $\mathcal{M}$ is a MinA for $C' \sqsubseteq D'$, $\mathsf{lab}(\mathcal{M})$ is a set of repairs entailing this subsumption. Moreover, $\nu(C' \sqsubseteq D') := \bigcup_{\mathcal{M} \in \mathsf{MinA}_\mathcal{T}(C' \sqsubseteq D')} \mathsf{lab}(\mathcal{M})$ is the set of all

Algorithm 2 Decide cautious and brave subsumption

Input: Labelled TBox $\mathcal{T}$, concepts C', D'

```
procedure IS-BRAVE(T, C', D')
    for each M ∈ MinA_T(C' ⊑ D') do
        if lab(M) ≠ ∅ then
            return true
    return false
procedure IS-CAUTIOUS(T, C', D')
    ν ← ∅
    for each M ∈ MinA_T(C' ⊑ D') do
        ν ← ν ∪ lab(M)
        if ν = {1, ..., n} then
            return true
    return false
```

repairs entailing $C' \sqsubseteq D'$. Thus, C' is bravely subsumed by D' iff $\nu(C' \sqsubseteq D') \neq \emptyset$ and is cautiously subsumed iff $\nu(C' \sqsubseteq D') = \{1, \ldots, n\}$ (recall Theorem 6).

The set $\nu(C' \sqsubseteq D')$ corresponds to the so-called *boundary* for the subsumption $C' \sqsubseteq D'$ w.r.t. the labelled TBox $\mathcal{T}$ [4]. Several methods for computing the boundary exist. Since we are only interested in deciding whether this boundary is empty or equal to $\{1, \ldots, n\}$, we can optimize the algorithm to stop once this decision is made. This optimized method is described in Algorithm 2. The algorithm first computes all MinAs for $C' \sqsubseteq_{\mathcal{T}} D'$, and their labels iteratively. If one of this labels is not empty, then the subsumption is a brave consequence; the procedure IS-BRAVE then returns true. Alternatively, IS-CAUTIOUS accumulates the union of all these labels in a set ν until this set contains all repairs, at which point it returns true.

The main difference between Algorithm 1 and Algorithm 2 is that the former iterates over the set of repairs of the unwanted consequences, while the latter iterates over $\mathsf{MinA}_{\mathcal{T}}(C' \sqsubseteq D')$. Typically, consequences have a small number of MinAs, which only contain a few axioms, while repairs are usually large and numerous. Thus, although Algorithm 2 has the overhead of computing the MinAs for the wanted consequence, it then requires less and cheaper iterations. As confirmed by our experimental results, this approach does show an advantage in practice.

Using the labelled TBox, it is also possible to decide IAR semantics through one subsumption test, and hence in polynomial time on the size of $\mathcal{T}$, regardless of the number of repairs.

Theorem 15. *Let $n = |\mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)|$. Then C' is IAR-subsumed by D' iff $C' \sqsubseteq_{\mathcal{T}_J} D'$, where $J = \{1, \ldots, n\}$.*

This shows that precompiling all repairs into a labelled ontology can help reducing

Table 2: Metrics of the ontologies used in the experiments

Ontology	#axioms	#conc. names	#role names
GALEN-OWL	45 499	23 136	404
NCI	159 805	104 087	92
SNOMED	369 194	310 013	58

the overall complexity and execution time of reasoning. Next, we exploit the fact that the number of MinAs for consequences in ontologies used in practice is relatively small and compute them using a saturation-based approach.

5 Implementation and Experiments

We ran two separate series of experiments. The goal of the first series was to investigate the feasibility of error-tolerant reasoning in practice. We implemented a prototype tool in Java that checks whether a concept subsumption $C \sqsubseteq D$ is brave or cautious w.r.t. a given TBox $\mathcal{T}$ and a consequence $C' \sqsubseteq D'$. The tool uses Theorem 6 and the duality between MinAs and repairs, i.e. the repairs for $C' \sqsubseteq D'$ w.r.t $\mathcal{T}$ can be obtained from the MinAs for $C' \sqsubseteq D'$ w.r.t $\mathcal{T}$ by consecutively removing the minimal hitting sets [29] of the MinAs from $\mathcal{T}$. The tool first computes all the MinAs for both inclusions $C \sqsubseteq D$ and $C' \sqsubseteq D'$ w.r.t. $\mathcal{T}$, and then verifies whether some inclusions between the MinAs for $C \sqsubseteq D$ and $C' \sqsubseteq D'$ hold to check for brave or cautious subsumptions. Note that the inclusion conditions only depend on the MinAs for the wanted consequence $C \sqsubseteq D$ and the erroneous subsumption $C' \sqsubseteq D'$ and *not* on the repairs of $C' \sqsubseteq D'$. Consequently, the repairs for $C' \sqsubseteq D'$ do not have to be explicitly computed in our tool. For the computation of the MinAs we used a saturation-based approach based on a consequence-based calculus [19]. More details regarding the computation of MinAs can be found in [24].

We selected three ontologies that are expressed mainly in $\mathcal{EL}$ and are typically considered to pose different challenges to DL reasoners. These are the January 2009 international release of SNOMED CT, version 13.11d of the NCI thesaurus,[1] and the GALEN-OWL ontology.[2] All non-$\mathcal{EL}$ axiom (including axioms involving roles only, e.g. role inclusion axioms) were first removed from the ontologies. The number of axioms, concept names, and role names in the resulting ontologies is shown in Table 2.

For every ontology $\mathcal{T}$ we selected a number of inclusion chains of the form $A_1 \sqsubseteq_\mathcal{T} A_2 \sqsubseteq_\mathcal{T} A_3 \sqsubseteq_\mathcal{T} A_4$, which were then grouped into

- *Type I* inclusions, where $A_2 \sqsubseteq_\mathcal{T} A_4$ was set as the unwanted consequence,

[1] http://evs.nci.nih.gov/ftp1/NCI_Thesaurus

[2] http://owl.cs.manchester.ac.uk/research/co-ode/

Table 3: Experimental results obtained for checking brave and cautious subsumption

ontology	type	#succ. comp.	#brave	#cautious	avg. #MinAs	max #MinAs	avg. time (s)
GALEN	I	498 / 500	495	39	1.707 | 1.663	4 | 4	335.680
	II	500 / 500	268	48	2.068 | 1.388	6 | 2	331.823
NCI	I	26 / 26	26	2	1.269 | 1.154	2 | 3	13.465
	II	36 / 36	16	8	3.111 | 1.111	7 | 3	15.338
SNOMED	I	302 / 500	296	17	1.652 | 1.656	42 | 12	161.471
	II	314 / 500	154	34	3.908 | 1.879	54 | 54	150.566

and

- *Type II* inclusions, where $A_2 \sqsubseteq_{\mathcal{T}} A_3$ was the unwanted consequence.

For the NCI and SNOMED CT ontologies we chose inclusions $A_2 \sqsubseteq A_4$ (for Type I) and $A_2 \sqsubseteq A_3$ (for Type II) that were *not* entailed by the consecutive version of the considered ontology, i.e. those that can be considered to be "mistakes" fixed in the consecutive release (the July 2009 international release of SNOMED CT and version 13.12e of the NCI Thesaurus). 500 inclusions of each type were found for SNOMED CT, but only 26 Type-I inclusions and 36 Type-II inclusions were detected in the case of NCI. For the GALEN-OWL ontology 500 inclusions chains of each type were chosen at random. For every Type-I chain, we then used our tool to check whether the inclusion $A_1 \sqsubseteq A_3$ is a brave or cautious consequence w.r.t. $A_2 \sqsubseteq A_4$. Similarly, for every Type-II inclusion we checked whether $A_1 \sqsubseteq A_4$ is a brave or cautious consequence w.r.t. $A_2 \sqsubseteq A_3$.

All experiments were conducted on a PC with an Intel Xeon E5-2640 CPU running at 2.50GHz. An execution timeout of 30 CPU minutes was imposed on each problem in this experiment series. The results obtained are shown in Table 3. The first two columns indicate the ontology that was used and the inclusion type. The next three columns show the number of successful computations within the time limit, and the number of brave and cautious subsumptions, respectively. The average and the maximal number of MinAs over the considered set of inclusions are shown in the next two columns. The left-hand side of each of these columns refers to the MinAs obtained for the consequence for which its brave or cautious entailment status should be checked, and the right-hand side refers to the unwanted consequence. The last column shows the average CPU time needed for the computations over each considered set of inclusions. All times shown correspond to the total computation time used.

The number of successful computations was the lowest for the experiments involving SNOMED, whereas no timeouts were incurred for NCI. Moreover, the highest average number of MinAs was found for Type-II inclusions for SNOMED with a maximal number of 54. GALEN-OWL required the longest computation times, which could be a consequence of the fact that the (full) GALEN ontology is generally seen as being difficult to classify by DL reasoners. The shortest computation times were reported for experiments involving NCI. It is important to notice, however, that the standard deviations of the computation times for

GALEN and Snomed were quite high. This indicates a large variation between problem instances; for example, some instances relating to GALEN required less than 9 seconds, and over one third of the experiments finished in sixty seconds or less. All the successful computations required at most 11 GiB of main memory.

In a second series of experiments we evaluated the advantages of performing precompilation when deciding several brave and cautious entailments w.r.t. an unwanted consequence. We therefore implemented a slightly improved version of Algorithm 1 which iterates over all the repairs for the unwanted consequence and determines whether a consequence that should be checked is brave or cautious by using the conditions from Definition 3. The implemented algorithm stops as quickly as possible, e.g. when a non-entailing repair has been found, we conclude immediately that the consequence is not cautious. The computation of the repairs is implemented by making use of the duality between MinAs and repairs (via the minimal hitting sets of the MinAs) as described above. The minimal hitting sets were computed using the Boolean algebraic algorithm from [23]. In the following we refer to this improved algorithm as the *naïve* approach. We used the reasoner ELK [20] to check whether a given inclusion follows from a repair. In particular, the incremental classification feature offered by ELK allowed us to further reduce reasoning times. When switching from a repair $\mathcal{R}$ to the next $\mathcal{R}'$, the knowledge about removed ($\mathcal{R} \setminus \mathcal{R}'$) and added axioms ($\mathcal{R}' \setminus \mathcal{R}$) was utilised by ELK to (potentially) avoid a complete reclassification.

Algorithm 2 was implemented in a straightforward way. The computation of the repairs for the unwanted consequence was implemented analogously to the naïve algorithm. Note that unlike with the naïve algorithm, all the MinAs for the wanted consequences had to be computed.

For comparing the performance of the naïve approach (Algorithm 1) against Algorithm 2 in practice, we selected 226 inclusions between concept names from Snomed having more than 10 MinAs, with a maximum number of 223. For each inclusion $A \sqsubseteq B$ we randomly chose five inclusions $A'_i \sqsubseteq B'_i$ entailed by Snomed, and tested whether $A'_i \sqsubseteq B'_i$ is a brave or cautious subsumption w.r.t. $A \sqsubseteq B$ for every $i \in \{1, \ldots, 5\}$ using the naïve approach and Algorithm 2. In this series of experiments we allowed each problem instance to run for at most 3600 CPU seconds, and 3 GiB of heap memory (with 16 GiB of main memory in total) were allocated to the Java VM. Each problem instance was run three times, and the best result was recorded.

The results obtained are depicted in Figure 1. The problem instances $A \sqsubseteq B$ are sorted ascendingly along the x-axis according to the number of repairs for $A \sqsubseteq B$. The required computation times for each problem instance (computing all repairs for the unwanted consequence and checking whether the five subsumptions are brave or cautious entailments w.r.t. the unwanted consequence) are shown along the y-axis on the left-hand side of the graph. If no corresponding y-value is shown for a given problem instance, the computation either timed out or ran out

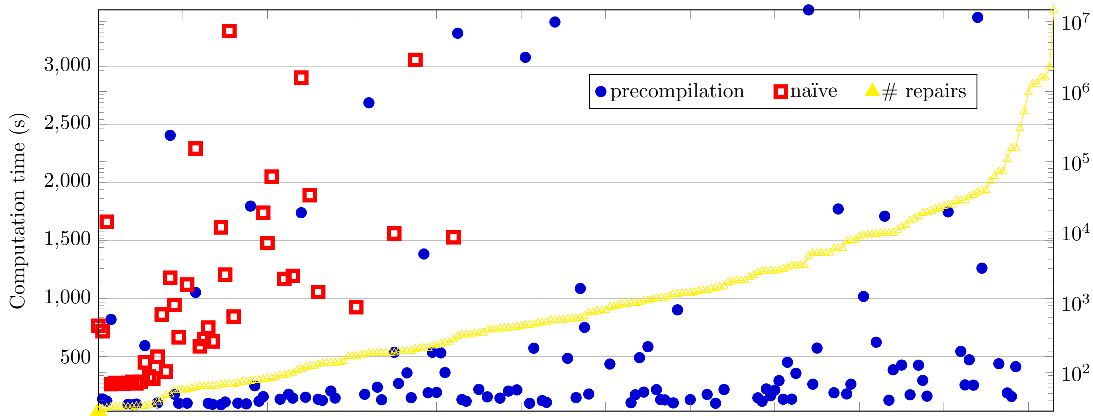

Figure 1: Comparison of approaches for error-tolerant reasoning.

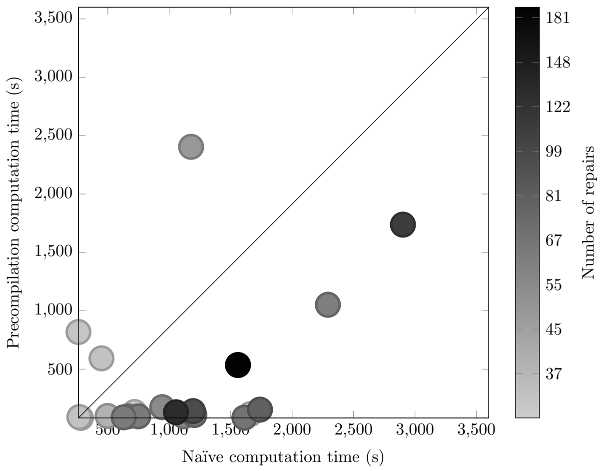

Figure 2: Comparative performance according to the number of repairs.

of memory in all three calls. The number of repairs for the unwanted consequences appears on the right-hand side.

One can see that a relatively small number of repairs can lead to several thousands (up to over 14 millions) of repairs. Also, if the number of repairs remains small, i.e. below 400, the naïve approach performs fairly well, even outperforming the precompilation approach on a few problem instances. For larger number of repairs, however, none of the computations for the naïve approach succeeded. The time required to perform reasoning with ELK outweighs the computation times of all the MinAs for the precompilation approach. In total 118 instances could be solved by at least one run of the precompilation approach, whereas only 42 computations finished when the naïve approach was used. Figure 2 shows the comparative behaviour of the two approaches over the 22 instances that succeeded in both methods. The tone of each point depicts the number of repairs of the unwanted consequence, as shown on the scale on the right. In the figure, points below the diagonal line correspond to instances where the precompilation approach performed better than the naïve approach. As it can be seen, the precompilation approach typically outperforms the naïve one, even in these simple cases, although there exist instances where the opposite behaviour is observed. However, there are also 20 instances where only the naïve approach succeeded. In our experiments the computation of the MinAs was typically the most time consuming part; the computation of the repairs once all the MinAs were available could be done fairly quickly.

6 Conclusions

We introduced error-tolerant reasoning inspired by inconsistency-tolerant semantics from DLs and consistent query answering over inconsistent databases. The main difference is that we allow for a general notion of *error* beyond inconsistency. We studied brave, cautious, and IAR reasoning, which depend on the class of repairs from which a consequence can be derived. Although we focused on subsumption w.r.t. $\mathcal{EL}$ TBoxes, these notions can be easily extended to any kind of monotonic consequences from a logical language.

Our results show that error-tolerant reasoning is hard in general for $\mathcal{EL}$, although brave reasoning remains polynomial for some of its sublogics. Interestingly, IAR semantics, introduced to regain tractability of inconsistency-tolerant query answering in light-weight DLs, is coNP-hard, even for the basic logic $\mathcal{HL}$ with core axioms. Moreover, the number of repairs is not the only culprit for hardness of these tasks: for both brave and cautious reasoning there is no polynomial-time algorithm on the size of $\mathcal{T}$ *and the number of repairs* that can solve these problems unless P = NP.

To overcome the complexity issues, we propose to compile the repairs into a labeled ontology. While the compilation step may require exponential time, after its execution IAR semantics can be decided in polynomial time, and brave and cautious semantics become repair-polynomial. Surprisingly, the idea of precomputing the set of all repairs to improve the efficiency of reasoning seems to have been overlooked by the inconsistency-tolerant reasoning community.

To investigate the feasibility of error-tolerant reasoning in practice, we developed prototype tools based on computing all MinAs, and annotating axioms with the repairs they belong to. Our experiments show that despite their theoretical complexity, brave and cautious reasoning can be performed successfully in many practical cases, even for large ontologies. Our saturation-based procedure can detect a large number of MinAs for some consequences in a fairly short amount of time. We plan to study optimizations that can help us reduce the reasoning times further. A deeper analysis to our experimental results will be a first step in this direction. There is a close connection between error-tolerant reasoning and axiom-pinpointing [6, 7]; our labelled ontology method also relates to context-based reasoning [4]. Techniques developed for those areas, like e.g. automata-based pinpointing methods [5], could be useful in this setting.

It is known that for some inexpressive DLs, all MinAs can be enumerated in output-polynomial time [26, 27]; the complexity of enumerating their repairs has not, to the best of our knowledge, been studied. We will investigate if enumerating repairs is also output-polynomial in those logics, and hence error-tolerant reasoning is repair-polynomial.

We will study the benefits of using labelled axioms for ontology contraction [12] and ontology evolution. Contraction operations can be simulated by modifying axiom labels, and minimal insertion operations add a labelled axiom. We will also extend our algorithms for more expressive logics. A full implementation and testing of these approaches is under development.

References

[1] Marcelo Arenas, Leopoldo Bertossi, and Jan Chomicki. Consistent query answers in inconsistent databases. In *Proceedings of the 18th ACM SIGMOD-SIGACT-SIGART symposium on Principles of Database Systems (PODS 1999)*, pages 68–79. ACM, 1999.

[2] Franz Baader. Terminological cycles in a description logic with existential restrictions. In Georg Gottlob and Toby Walsh, editors, *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 325–330. Morgan Kaufmann, 2003.

[3] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2nd edition, 2007.

[4] Franz Baader, Martin Knechtel, and Rafael Peñaloza. Context-dependent views to axioms and consequences of semantic web ontologies. *Journal of Web Semantics*, 12–13:22–40, 2012. Available at http://dx.doi.org/10.1016/j.websem.2011.11.006.

[5] Franz Baader and Rafael Peñaloza. Automata-based axiom pinpointing. *Journal of Automated Reasoning*, 45(2):91–129, August 2010.

[6] Franz Baader and Rafael Peñaloza. Axiom pinpointing in general tableaux. *Journal of Logic and Computation*, 20(1):5–34, 2010.

[7] Franz Baader, Rafael Peñaloza, and Boontawee Suntisrivaraporn. Pinpointing in the description logic $\mathcal{EL}^+$. In *Proceedings of the 30th German Conference on Artificial Intelligence (KI2007)*, volume 4667 of *LNAI*, pages 52–67, Osnabrück, Germany, 2007. Springer.

[8] Franz Baader and Boontawee Suntisrivaraporn. Debugging SNOMED CT using axiom pinpointing in the description logic $\mathcal{EL}^+$. In *Proceedings of the 3rd Knowledge Representation in Medicine (KR-MED'08): Representing and Sharing Knowledge Using SNOMED*, volume 410 of *CEUR-WS*, 2008.

[9] Leopoldo Bertossi. Database repairing and consistent query answering. *Synthesis Lectures on Data Management*, 3(5):1–121, 2011.

[10] Meghyn Bienvenu. On the complexity of consistent query answering in the presence of simple ontologies. In *Proceedings of the 26st Natonal Conference on Artificial Intelligence (AAAI 2012)*, 2012.

[11] Meghyn Bienvenu and Riccardo Rosati. Tractable approximations of consistent query answering for robust ontology-based data access. In Francesca Rossi, editor, *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*. AAAI Press, 2013.

[12] Richard Booth, Thomas Meyer, and Ivan José Varzinczak. First steps in $\mathcal{EL}$ contraction. In *Proceedings of the 2009 Workshop on Automated Reasoning About Context and Ontology Evolution (ARCOE 2009)*, 2009.

[13] Sebastian Brandt. Polynomial time reasoning in a description logic with existential restrictions, GCI axioms, and - what else? In Ramon López de Mántaras and Lorenza Saitta, editors, *Proceedings of the 16th European Conference on Artificial Intelligence, (ECAI 2004)*, pages 298–302. IOS Press, 2004.

[14] R Cote, D Rothwell, J Palotay, R Beckett, and L Brochu. The systematized nomenclature of human and veterinary medicine. Technical report, SNOMED International, Northfield, IL: College of American Pathologists, 1993.

[15] Thomas Eiter and Georg Gottlob. Identifying the minimal transversals of a hypergraph and related problems. Technical Report CD-TR 91/16, Christian Doppler Laboratory for Expert Systems, TU Vienna, 1991.

[16] Giorgio Gallo, Giustino Longo, and Stefano Pallottino. Directed hypergraphs and applications. *Discrete Applied Mathematics*, 42(2):177–201, 1993.

[17] David S. Johnson, Mihalis Yannakakis, and Christos H. Papadimitriou. On generating all maximal independent sets. *Information Processing Letters*, 27(3):119–123, 1988.

[18] Aditya Kalyanpur, Bijan Parsia, Matthew Horridge, and Evren Sirin. Finding all justifications of OWL DL entailments. In *Proceedings of the 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference, ISWC 2007, ASWC 2007*, volume 4825 of *LNCS*, pages 267–280. Springer, 2007.

[19] Yevgeny Kazakov. Consequence-driven reasoning for Horn SHIQ ontologies. In Craig Boutilier, editor, *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI'09)*, pages 2040–2045, 2009.

[20] Yevgeny Kazakov, Markus Krötzsch, and František Simančík. The incredible ELK: From polynomial procedures to efficient reasoning with $\mathcal{EL}$ ontologies. *Journal of Automated Reasoning*, 2013. To appear.

[21] Andrea S. Lapaugh and Christos H. Papadimitriou. The even-path problem for graphs and digraphs. *Networks*, 14(4):507–513, 1984.

[22] Domenico Lembo, Maurizio Lenzerini, Riccardo Rosati, Marco Ruzzi, and Domenico Fabio Savo. Inconsistency-tolerant semantics for description logics. In Pascal Hitzler and Thomas Lukasiewicz, editors, *Proceedings of the 4th International Conference on Web Reasoning and Rule Systems (RR'10)*, volume 6333 of *LNCS*, pages 103–117. Springer, 2010.

[23] Li Lin and Yunfei Jiang. The computation of hitting sets: Review and new algorithms. *Information Processing Letters*, 86(4):177–184, 2003.

[24] Michel Ludwig. Just: a tool for computing justifications w.r.t. $\mathcal{EL}$ ontologies. In *Proceedings of the 3rd International Workshop on OWL Reasoner Evaluation (ORE 2014)*, 2014.

[25] Michel Ludwig and Rafael Peñaloza. Error-tolerant reasoning in the description logic el. In *Proceedings of the 14th European Conference on Logics in Artificial Intelligence (JELIA'14)*, 2014. To appear.

[26] Rafael Peñaloza and Barış Sertkaya. Complexity of axiom pinpointing in the DL-Lite family of description logics. In Helder Coelho, Rudi Studer, and Michael Wooldridge, editors, *Proceedings of the 19th European Conference on Artificial Intelligence, (ECAI 2010)*, volume 215 of *Frontiers in Artificial Intelligence and Applications*, pages 29–34. IOS Press, 2010.

[27] Rafael Peñaloza and Barış Sertkaya. On the complexity of axiom pinpointing in the $\mathcal{EL}$ family of description logics. In Fangzhen Lin, Ulrike Sattler, and Miroslaw Truszczynski, editors, *Proceedings of the Twelfth International Conference on Principles of Knowledge Representation and Reasoning (KR 2010)*. AAAI Press, 2010.

[28] Rafael Peñaloza. *Axiom-Pinpointing in Description Logics and Beyond*. PhD thesis, Dresden University of Technology, Germany, 2009.

[29] Raymond Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57–95, 1987.

[30] Riccardo Rosati. On the complexity of dealing with inconsistency in description logic ontologies. In Toby Walsh, editor, *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11)*, pages 1057–1062. AAAI Press, 2011.

[31] K. Spackman. Managing clinical terminology hierarchies using algorithmic calculation of subsumption: Experience with SNOMED-RT. *Journal of the American Medical Informatics Association*, 2000. Fall Symposium Special Issue.

[32] W3C OWL Working Group. OWL 2 web ontology language document overview. W3C Recommendation, 2009. `http://www.w3.org/TR/owl2-overview/`.

7 Appendix

7.1 Proofs regarding the Theoretical Results

Lemma 16. *Let $\mathcal{T}$ be a TBox, let C, D be concepts, and let $S \subseteq \mathcal{T}$. Then $C \sqsubseteq_S D$ holds iff there exists $\mathcal{M} \in \mathsf{MinA}_{\mathcal{T}}(C \sqsubseteq D)$ such that $\mathcal{M} \subseteq S$.*

Proof. Follows immediately from the definition of $\mathsf{MinA}_{\mathcal{T}}(C \sqsubseteq D)$. □

Lemma 17. *Let $\mathcal{T}$ be a TBox and let C, D be concepts such that $C \sqsubseteq_{\mathcal{T}} D$. Moreover, let $\mathsf{MinA}_{\mathcal{T}}(C \sqsubseteq D) = \{\mathcal{M}_1, \ldots, \mathcal{M}_n\}$ $(n \geq 1)$. Then*

$$\mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D) = \{\, \mathcal{T} \setminus M \mid M \in S \,\}$$

where S is the set of all the minimal (w.r.t. set inclusion) sets $M \subseteq \bigcup_{i=1}^{n} \mathcal{M}_i$ such that for every $1 \leq i \leq n$, $M \cap \mathcal{M}_i \neq \emptyset$.

Proof. "$\subseteq$"

Let $\mathcal{R} \in \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$. Then $C \not\sqsubseteq_{\mathcal{R}} D$, i.e. for every $1 \leq i \leq n$ there exists $a_i \in \mathcal{M}_i$ such that $a_i \notin \mathcal{R}$ by Lemma 16. Let $M = \{a_1, \ldots, a_n\}$. We can infer that $\mathcal{R} \subseteq \mathcal{T} \setminus M$ holds. If we assume that there exists $M' \subsetneq M$ such that $M' \cap \mathcal{M}_i \neq \emptyset$ for every $1 \leq j \leq n$, it would follow that $C \not\sqsubseteq_{\mathcal{T} \setminus M'} D$ by Lemma 16, contradicting the maximality of $\mathcal{R}$ as $\mathcal{R} \subsetneq \mathcal{T} \setminus M'$. Hence, $M \in S$. Moreover, as $C \not\sqsubseteq_{\mathcal{T} \setminus M} D$, it holds that $\mathcal{T} \setminus M \subseteq \mathcal{R}$ by the maximality of $\mathcal{R}$, which implies that $\mathcal{R} = \mathcal{T} \setminus M$.

"$\supseteq$"

Let $M \in S$. By Lemma 16 we have $C \not\sqsubseteq_{\mathcal{T} \setminus M} D$. Additially, as for every $\mathcal{R} \subseteq \mathcal{T}$ with $\mathcal{T} \setminus M \subsetneq \mathcal{R}$, it holds that there exists $\mathcal{M} \in \mathsf{MinA}_{\mathcal{T}}(C \sqsubseteq D)$ with $\mathcal{M} \subseteq \mathcal{R}$ by the definition of S and therefore $C \sqsubseteq_{\mathcal{R}} D$, we can conclude that $\mathcal{T} \setminus M \in \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$. □

Theorem 6. *Let $\mathcal{T}$ be a TBox, C, C', D, D' concepts with $C \sqsubseteq_{\mathcal{T}} D$. Then*

1. *C' is cautiously subsumed by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$ iff for every repair $\mathcal{R} \in \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$ there is an $\mathcal{M}' \in \mathsf{MinA}_{\mathcal{T}}(C' \sqsubseteq D')$ with $\mathcal{M}' \subseteq \mathcal{R}$; and*
2. *C' is bravely subsumed by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$ iff there is a repair $\mathcal{R} \in \mathsf{Rep}_{\mathcal{T}}(C \sqsubseteq D)$ and a MinA $\mathcal{M}' \in \mathsf{MinA}_{\mathcal{T}}(C' \sqsubseteq D')$ with $\mathcal{M}' \subseteq \mathcal{R}$.*

Proof. Follows immediately from Definition 3 and Lemma 16. □

Theorem 8. *Brave subsumption in $\mathcal{EL}$ is* NP*-complete.*

Table 4: The TBoxes $\mathcal{T}_\psi$ for complex formulas

$\psi = \psi_1 \land \psi_2$	$\psi = \psi_1 \lor \psi_2$
$A \sqsubseteq \exists r_\psi.C_\psi$	$A \sqsubseteq \exists r_\psi.B_{\psi_1}$
$C_\psi \sqsubseteq B_{\psi_1}$	$A \sqsubseteq \exists s_\psi.B_{\psi_2}$
$C_\psi \sqsubseteq B_{\psi_2}$	$B_{\psi_1} \sqsubseteq B_\psi$
$B_{\psi_1} \sqcap B_{\psi_2} \sqsubseteq B_\psi$	$B_{\psi_2} \sqsubseteq B_\psi$
$\exists r_\psi.B_\psi \sqsubseteq D_\psi$	$\exists r_\psi.B_\psi \sqsubseteq B'_{\psi_1}$
	$\exists s_\psi.B_\psi \sqsubseteq B'_{\psi_2}$
	$B'_{\psi_1} \sqcap B'_{\psi_2} \sqsubseteq D_\psi$

Proof. To decide brave subsumption of C' by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$, we guess a set $\mathcal{R} \subseteq \mathcal{T}$ and verify in polynomial time that $\mathcal{R}$ is a repair of $\mathcal{T}$ w.r.t. $C \sqsubseteq D$ and that $C \sqsubseteq_\mathcal{R} D$. Thus, the problem is in NP.

To prove hardness, we adapt an idea from [27], to reduce the *more minimal valuations* (MMV) problem, which is NP-hard [7, 15]. This problem consists in deciding, for a monotone Boolean formula φ and a set $\mathfrak{V}$ of minimal valuations satisfying φ, whether there exist other minimal valuations $V \notin \mathfrak{V}$ that satisfy φ.

Let $\varphi, \mathfrak{V}$ be an instance of MMV with $|\mathfrak{V}| = n$, and define the formula $\varphi' := \bigvee_{V\in\mathfrak{V}} \bigwedge_{p\in V} p$. To decide MMV, it suffices to find out whether there exists a minimal valuation satisfying φ that does not satisfy φ'. Let sub denote the set of all subformulas of φ and φ' and csub the subset of sub without the propositional variables. We introduce the concept names $B_\psi, B'_\psi, C_\psi, D_\psi$, and two role names r_ψ, s_ψ for every $\psi \in \mathsf{sub}$, and the additional concept names A and E For each $\psi \in \mathsf{sub}$ we define a TBox $\mathcal{T}_\psi$ as follows: if ψ is the propositional variable p, then $\mathcal{T}_\psi := \{A \sqsubseteq B_p\}$; if ψ is a complex formula, then $\mathcal{T}_\psi$ contains the axioms described in Table 4. Finally, we set

$$\mathcal{T} := \bigcup_{\psi\in\mathsf{sub}} \mathcal{T}_\psi \cup \{\bigsqcap_{\psi\in\mathsf{csub}} D_\psi \sqcap B_\varphi \sqsubseteq E\}.$$

Notice that for every $\mathcal{T}' \subseteq \mathcal{T}$, if $\mathcal{T}' \models A \sqsubseteq E$, then also $A \sqsubseteq D_\psi$ for every $\psi \in \mathsf{csub}$. In order to have $A \sqsubseteq D_\psi$, all the axioms in $\mathcal{T}_\psi$ are necessary, and thus $\mathcal{T}_\psi \subseteq \mathcal{T}'$. In particular, if $\psi = \psi_1 \land \psi_2$, then $B_{\psi_1} \sqcap B_{\psi_2} \sqsubseteq B_\psi \in \mathcal{T}'$, and if $\psi = \psi_1 \lor \psi_2$, then $\{B_{\psi_1} \sqsubseteq B_\psi, B_{\psi_2} \sqsubseteq B_\psi\} \subseteq \mathcal{T}'$. Thus, a valuation V satisfies φ iff

$$\mathcal{T}_V := \{A \sqsubseteq B_p \mid p \in V\} \cup \bigcup_{\psi\in\mathsf{csub}} \mathcal{T}_\psi \cup \{\bigsqcap_{\psi\in\mathsf{csub}} D_\psi \sqcap B_\varphi \sqsubseteq E\}$$

entails $A \sqsubseteq E$. This in particular shows that there is a minimal valuation V satisfying φ but violating φ' iff there is a repair $\mathcal{R}$ of $\mathcal{T}$ w.r.t. $A \sqsubseteq B_{\varphi'}$ such that $\mathcal{T}_V \subseteq \mathcal{R}$; that is, iff A is bravely subsumed by E w.r.t. $\mathcal{T}$ and $A \sqsubseteq B_{\varphi'}$. □

Theorem 13. *Unless* P = NP, *cautious and brave subsumption of C' by D' w.r.t. $\mathcal{T}$ and $C \sqsubseteq D$ in $\mathcal{EL}$ are not repair-polynomial.*

Proof. We consider the NP-hard *more maximal falsifiers* (MMF) problem [28]: given a monotone Boolean formula φ and a set $\mathfrak{F}$ of maximal valuations falsifying φ, decide if there exists a maximal valuation V falsifying φ with $V \notin \mathfrak{F}$. Given an instance $\varphi, \mathfrak{F}$ of MMF, let sub denote the set of all subformulas of φ. For each $\psi \in \mathsf{sub}$, construct the TBox $\mathcal{T}_\psi$ as in the proof of Theorem 8. Finally, we create the TBoxes:

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

It is easy to see that if n is the number of maximal valuations falsifying φ, then $|\mathsf{Rep}_{\mathcal{T}}(A \sqsubseteq E)| \leq 8(n+1) \cdot |\mathsf{sub}|$.

Suppose now that cautious subsumption is repair-polynomial. Then there must exist an algorithm A with runtime bounded by some polynomial $p(t, r)$, where t is the size of the TBox and r the number of repairs. Then we can decide MMF as follows: we run the algorithm A on $\mathcal{T}$ for the subsumption $A \sqsubseteq F$ for at most $p(|\mathcal{T}|, 8|\mathfrak{F}||\mathsf{sub}|)$ steps. If it answers *yes*, then there is no new falsifying valuation. If it answers *no* or has not terminated after this number of steps, then there is at least one more maximal falsifying valuation. Thus, this would decide MMF in polynomial time.

The result for brave subsumption can be shown analogously, using the construction from Theorem 8. □

7.2 Proofs regarding the Implementation

In the following we write $\bar{A}$ for upper-case letters $A, B, X, \ldots$, etc., to denote that $A, B, X, \ldots \in \mathsf{N_C} \cup \{\top\}$. We also assume that normalised TBoxes do not contain axioms of the form $C \sqsubseteq \top$, or $\bar{A}_1 \sqcap \ldots \sqcap \bar{A}_n \sqsubseteq \bar{B}$ where $\bar{A}_i = \top$ for some $1 \leq i \leq n$.

The calculus $\mathfrak{T}$ that we used for the computation of the MinAs is now depicted in Figure 3. Note that it operates on normalised terminologies and it derives consequences of the form $\bar{A} \sqsubseteq \bar{B}$ only.

For a normalised $\mathcal{EL}$ TBox $\mathcal{T}$, a $\mathfrak{T}$-derivation (tree) Γ w.r.t. $\mathcal{T}$ is a finite tree where each leaf is labelled with a consequence $\bar{X} \sqsubseteq \bar{X}$ or $\bar{X} \sqsubseteq \top$, and each non-leaf node n is labelled with a consequence $\bar{A} \sqsubseteq \bar{B}$ such that $\bar{A} \sqsubseteq \bar{B}$ results from applying an inference rule of $\mathfrak{T}$ (w.r.t. $\mathcal{T}$) on the parent(s) of n in Γ. We say that Γ is a derivation of (a consequence) $\bar{A} \sqsubseteq \bar{B}$ if the root of Γ is labelled with $\bar{A} \sqsubseteq \bar{B}$. Moreover, we write $\bar{A} \vdash_{\mathcal{T}} \bar{B}$ iff there exists a derivation of $\bar{A} \sqsubseteq \bar{B}$ w.r.t. $\mathcal{T}$.

We prove the completeness of $\mathfrak{T}$ next. Note that the inference rule (Merge) is

- $$\frac{}{\bar{A} \sqsubseteq \bar{A}} \text{ (Ax)} \qquad \frac{}{\bar{A} \sqsubseteq \top} \text{ (AxTop)}$$

- $$\frac{\bar{A} \sqsubseteq \bar{B}_1}{\bar{A} \sqsubseteq \bar{B}_2} \text{ (Succ)}$$

  if $\bar{B}_1 \sqsubseteq \bar{B}_2 \in \mathcal{T}$

- $$\frac{\bar{A} \sqsubseteq \bar{B}_1 \quad \bar{X} \sqsubseteq \bar{Y}}{\bar{A} \sqsubseteq \bar{B}_2} \text{ (Ex)}$$

  if $\bar{B}_1 \sqsubseteq \exists r.\bar{X} \in \mathcal{T}$ and $\exists r.\bar{Y} \sqsubseteq \bar{B}_2 \in \mathcal{T}$

- $$\frac{\bar{A} \sqsubseteq X_1 \quad \ldots \quad \bar{A} \sqsubseteq X_n}{\bar{A} \sqsubseteq B} \text{ (And)}$$

  if $X_1 \sqcap \ldots \sqcap X_n \sqsubseteq B \in \mathcal{T}$

- $$\frac{\bar{A} \sqsubseteq \bar{X} \quad \bar{X} \sqsubseteq \bar{B}}{\bar{A} \sqsubseteq \bar{B}} \text{ (Merge)}$$

Figure 3: Calculus $\mathfrak{T}$

not necessary to prove the completeness of $\mathfrak{T}$, but it is used in the implementation to speed up the computation of derivations.

Lemma 18. *Let $\mathcal{T}$ be a normalised TBox and let $\bar{A}, \bar{B} \in \mathsf{N_C} \cup \{\top\}$. Then $\bar{A} \sqsubseteq_{\mathcal{T}} \bar{B}$ iff there exists a derivation of $\bar{A} \sqsubseteq \bar{B}$ using the inference rules of the calculus depicted in Fig. 3.*

Proof. It is easy to see that the direction "$\Leftarrow$" follows the soundness of the inference rules.

For the direction "$\Rightarrow$", we define an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where

$$\begin{aligned} \Delta^{\mathcal{I}} &= \mathsf{sig}(\mathcal{T}) \cup \{\bar{A}, \bar{B}, \top\} \\ A^{\mathcal{I}} &= \{\, \bar{B} \mid \bar{B} \vdash_{\mathcal{T}} A \,\} \\ r^{\mathcal{I}} &= \{\, (\bar{A}, \bar{X}) \mid \bar{A} \vdash_{\mathcal{T}} \bar{B} \text{ and } \bar{B} \sqsubseteq \exists r.\bar{X} \in \mathcal{T} \,\} \end{aligned}$$

for every $A \in \mathsf{N_C}$ and for every $r \in \mathsf{N_R}$.

First we observe that for every $\bar{A}, \bar{B} \in \Delta^{\mathcal{I}}$, $\bar{B} \in (\bar{A})^{\mathcal{I}}$ iff $\bar{B} \vdash_{\mathcal{T}} \bar{A}$ holds (either by definition of $\cdot^{\mathcal{I}}$ or by applying the (Ax)- or (AxTop)-rules). We now show that $\mathcal{I}$ is a model of $\mathcal{T}$.

For $\bar{A} \sqsubseteq B \in \mathcal{T}$, let $\bar{X} \in A^{\mathcal{I}}$, i.e. $\bar{X} \vdash_{\mathcal{T}} \bar{A}$ holds. By applying the (Succ)-rule, we obtain $\bar{X} \vdash_{\mathcal{T}} B$, which allows us to conclude that $\bar{X} \in B^{\mathcal{I}}$ holds.

For $\bar{A} \sqsubseteq \exists r.\bar{B} \in \mathcal{T}$, let $\bar{X} \in \bar{A}^{\mathcal{I}}$. Then as $\bar{X} \vdash_{\mathcal{T}} \bar{A}$, we have $(\bar{X}, \bar{B}) \in r^{\mathcal{I}}$ by the definition of $\mathcal{I}$. It now suffices to observe that $\bar{B} \in (\bar{B})^{\mathcal{I}}$ holds, and we can conclude that $\bar{X} \in (\exists r.\bar{B})^{\mathcal{I}}$.

For $\exists r.\bar{A} \sqsubseteq B \in \mathcal{T}$, let $\bar{X} \in (\exists r.\bar{A})^{\mathcal{I}}$. Then there exists $\bar{Y} \in \Delta^{\mathcal{I}}$ such that $(\bar{X}, \bar{Y}) \in r^{\mathcal{I}}$ and $\bar{Y} \in (\bar{A})^{\mathcal{I}}$. Hence, by definition of $r^{\mathcal{I}}$, there exists $\bar{X}' \sqsubseteq \exists r.\bar{Y} \in \mathcal{T}$ with $\bar{X} \vdash_{\mathcal{T}} \bar{X}'$. Moreover, it follows that $\bar{Y} \vdash_{\mathcal{T}} \bar{A}$. Hence, by applying the (Ex)-rule we get $\bar{X} \vdash_{\mathcal{T}} B$, i.e. $\bar{X} \in B^{\mathcal{I}}$ holds.

For $A_1 \sqcap \ldots \sqcap A_n \sqsubseteq B \in \mathcal{T}$, let $\bar{X} \in (A_i)^{\mathcal{I}}$ for every $1 \leq i \leq n$. We then have $\bar{X} \vdash_{\mathcal{T}} A_i$ for every $1 \leq i \leq n$. We can infer that $\bar{X} \vdash_{\mathcal{T}} B$ holds by applying the (And)-rule, i.e. $\bar{X} \in B^{\mathcal{I}}$ holds. This concludes the proof that $\mathcal{I}$ is a model of $\mathcal{T}$.

Now, as $\bar{A} \vdash_{\mathcal{T}} \bar{A}$, i.e. $\bar{A} \in (\bar{A})^{\mathcal{I}}$, as $\mathcal{I}$ is a model of $\mathcal{T}$, and as $\bar{A} \sqsubseteq_{\mathcal{T}} \bar{B}$, we obtain $\bar{A} \in (\bar{A})^{\mathcal{I}} \subseteq (\bar{B})^{\mathcal{I}}$, i.e. $\bar{A} \vdash_{\mathcal{T}} \bar{B}$. □

We now show that it is sufficient to consider so-called *admissible* derivations only to decide whether a normalised TBox $\mathcal{T}$ entails a consequence $\bar{A} \sqsubseteq \bar{B}$.

Definition 19 (Admissible derivations)**.** Let $\mathcal{T}$ be a normalised TBox. A derivation Γ is said to be *admissible* iff for every sub-derivation Γ' of a consequence $\bar{A} \sqsubseteq \bar{B}$ in Γ, the consequence $\bar{A} \sqsubseteq \bar{B}$ does not occur as a premise in Γ'.

We write $A \vdash^{a}_{\mathcal{T}} B$ iff there exists an admissible derivation of $A \sqsubseteq B$ w.r.t. $\mathcal{T}$.

It is easy to see that there can only exist finitely many derivations w.r.t. a normalised TBox $\mathcal{T}$ for a consequence $\bar{A} \sqsubseteq_{\mathcal{T}} \bar{B}$.

Lemma 20. *Let $\mathcal{T}$ be a normalised TBox and let $\bar{A}, \bar{B} \in \mathsf{N_C} \cup \{\top\}$. Then every derivation of $\bar{A} \sqsubseteq \bar{B}$ w.r.t. $\mathcal{T}$ can be transformed into an admissible derivation of $\bar{A} \sqsubseteq \bar{B}$ w.r.t. $\mathcal{T}$.*

Proof. It is easy to see that if a derivation Γ contains a sub-derivation Γ' of a consequence $\bar{A} \sqsubseteq \bar{B}$ in which a consequence $\bar{A} \sqsubseteq \bar{B}$ occurs as a premise, the complete sub-derivation Γ' can simply be replaced with the derivation for $\bar{A} \sqsubseteq \bar{B}$ occurring as premise. As this process can only be repeated finitely many times, one can see that every derivation of $\bar{A} \sqsubseteq \bar{B}$ w.r.t. $\mathcal{T}$ can be transformed into an admissible derivation of $\bar{A} \sqsubseteq \bar{B}$ w.r.t. $\mathcal{T}$. □

Corollary 21. *Let $\mathcal{T}$ be a normalised TBox. and let $\bar{A}, \bar{B} \in \mathsf{N_C} \cup \{\top\}$. Then $\bar{A} \sqsubseteq_{\mathcal{T}} \bar{B}$ iff $\bar{A} \vdash^a_{\mathcal{T}} \bar{B}$.*

Definition 22 (Axioms(Γ))**.** Let $\mathcal{T}$ be a normalised TBox. For every derivation Γ w.r.t. $\mathcal{T}$, we define Axioms(Γ) $\subseteq \mathcal{T}$ as the set consisting of exactly the axioms required for applying the inference rules in order to obtain Γ.

Theorem 23. *Let $\mathcal{T}$ be a normalised TBox and let $\bar{A}, \bar{B} \in \mathsf{N_C} \cup \{\top\}$. Moreover, let S be the set of all the admissible derivations of $\bar{A} \sqsubseteq \bar{B}$ w.r.t. $\mathcal{T}$. Then* $\mathsf{MinA}_{\mathcal{T}}(\bar{A} \sqsubseteq \bar{B}) = \min_{\subseteq}\{\, \text{Axioms}(\Gamma) \mid \Gamma \in S \,\}$.

Proof. "$\subseteq$" Let $\mathcal{M} \in \mathsf{MinA}_{\mathcal{T}}(\bar{A} \sqsubseteq_{\mathcal{T}} \bar{B})$. Then we have $\bar{A} \vdash^a_{\mathcal{M}} \bar{B}$ by Corollary 21, i.e. there exists a derivation $\Gamma \in S$ such that Axioms(Γ) $\subseteq \mathcal{M}$. We can hence infer that Axioms(Γ) $= \mathcal{M}$ as otherwise we would obtain a contradiction with $\mathcal{M} \in \mathsf{MinA}_{\mathcal{T}}(\bar{A} \sqsubseteq_{\mathcal{T}} \bar{B})$. Similarly, if we assume that there exists $\Gamma' \in S$ such that Axioms(Γ') $\subsetneq$ Axioms(Γ) $= \mathcal{M}$, we would again obtain a contradiction with $\mathcal{M} \in \mathsf{MinA}_{\mathcal{T}}(\bar{A} \sqsubseteq_{\mathcal{T}} \bar{B})$. We have thus concluded this direction of the proof.

"$\supseteq$" Let $\Gamma \in S$ such that there does not exist $\Gamma' \in S$ with Axioms(Γ') $\subsetneq$ Axioms(Γ). As $\bar{A} \sqsubseteq_{\text{Axioms}(\Gamma)} \bar{B}$, there exists $\mathcal{M} \in \mathsf{MinA}_{\mathcal{T}}(\bar{A} \sqsubseteq \bar{B})$ with $\mathcal{M} \subseteq$ Axioms(Γ). If we assume that $\mathcal{M} \subsetneq$ Axioms(Γ) holds, it would follow from Corollary 21 that there exists a derivation $\Gamma' \in S$ with Axioms(Γ') $\subsetneq$ Axioms(Γ), contradicting our assumption. Hence, we have Axioms(Γ) $= \mathcal{M} \in \mathsf{MinA}_{\mathcal{T}}(\bar{A} \sqsubseteq \bar{B})$. □

In the following we show how MinAs for inclusions $C \sqsubseteq D$ w.r.t. general TBoxes $\mathcal{T}$ can be obtained from MinAs for inclusions $A \sqsubseteq B$ (where A and B are concept names) w.r.t. normalised TBoxes.

To this end, we first have to introduce some additional notions. We denote by $\mathsf{sig}(C), \mathsf{sig}(\mathcal{T}), \mathsf{sig}(C \sqsubseteq D), \ldots$ the set consisting of exactly the concept and role names that are used in the construction of C, $\mathcal{T}$, $C \sqsubseteq D$, ..., respectively. We assume that $\mathsf{N_N} \subseteq \mathsf{N_C}$ is a countably-infinite set of concept names that are used for normalisation only. Furthermore, we assume that Norm is a function that maps every GCI $C \sqsubseteq D$ with $\mathsf{sig}(C \sqsubseteq D) \cap \mathsf{N_N} = \emptyset$ into a normalised $\mathcal{EL}$ TBox $\mathrm{Norm}(C \sqsubseteq D)$ such that

(i) for every $\mathcal{EL}$ TBox $\mathcal{T}$ with $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$ and for every two $\mathcal{EL}$ concepts C and D with $(\mathsf{sig}(C) \cup \mathsf{sig}(D)) \cap \mathsf{N_N} = \emptyset$ it holds that

$$\mathcal{T} \models C \sqsubseteq D \text{ iff } \mathcal{T}_{\mathrm{Norm}} \models C \sqsubseteq D,$$

where $\mathcal{T}_{\mathrm{Norm}} = \bigcup_{C' \sqsubseteq D' \in \mathcal{T}} \mathrm{Norm}(C' \sqsubseteq D')$; and

(ii) for every $\mathcal{EL}$ TBox $\mathcal{T}$ with $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$ there exists an injective function $m \colon \mathcal{T} \to \mathcal{T}_{\mathrm{Norm}}$ such that $m(C \sqsubseteq D) \in \mathrm{Norm}(C \sqsubseteq D)$ for every $C \sqsubseteq D \in \mathcal{T}$.

Such a normalisation function can be obtained by defining a function that introduces new defining GCIs (using concept names from $\mathsf{N_N}$) for complex concepts until the normal form is reached. We refer the reader to [3] for more details. The normalisation in our prototype implementation does not “reuse” previously introduced definitions for complex concepts, i.e. in our implementation fresh definitions are also introduced for complex concepts that were normalised previously already. For instance, one would obtain $\mathrm{Norm}(A_1 \sqsubseteq \exists r.\exists s.B) = \{A_1 \sqsubseteq \exists r.Z_1, Z_1 \sqsubseteq \exists s.B\}$ and $\mathrm{Norm}(A_2 \sqsubseteq \exists r.\exists s.B) = \{A_2 \sqsubseteq \exists r.Z_2, Z_2 \sqsubseteq \exists s.B\}$ instead of $\mathrm{Norm}(A_2 \sqsubseteq \exists r.\exists s.B) = \{A_2 \sqsubseteq \exists r.Z_1, Z_1 \sqsubseteq \exists s.B\}$ with $Z_1, Z_2 \in \mathsf{N_N}$. Hence, Condition (ii) can be ensured w.r.t. a normalisation function that does not reuse introduced definitions by defining $m(C \sqsubseteq D)$ to be an arbitrary element of $\mathrm{Norm}(C \sqsubseteq D)$ for every $\mathcal{EL}$ TBox $\mathcal{T}$ with $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$ and for every $C \sqsubseteq D \in \mathcal{T}$.

For an $\mathcal{EL}$ TBox $\mathcal{T}$ with $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$ we define a mapping $n_{b,\mathcal{T}}$ which assigns every axiom $C \sqsubseteq D \in \mathcal{T}_{\mathrm{Norm}}$ with

$$n_{b,\mathcal{T}}(C \sqsubseteq D) = \{\, C' \sqsubseteq D' \in \mathcal{T} \mid C \sqsubseteq D \in \mathrm{Norm}(C' \sqsubseteq D') \,\}.$$

Intuitively, $n_{b,\mathcal{T}}(C \sqsubseteq D)$ consists of all the axioms $C' \sqsubseteq D' \in \mathcal{T}$ whose normalisation contains the axiom $C \sqsubseteq D$. Finally, for $\{a_1, \ldots, a_n\} \subseteq \mathcal{T}_{\mathrm{Norm}}$ we set

$$[\{a_1, \ldots, a_n\}]_{n_{b,\mathcal{T}}} = \{\, \{\tilde{a}_1, \ldots, \tilde{a}_n\} \mid \forall\, 1 \le i \le n \colon \tilde{a}_i \in n_{b,\mathcal{T}}(a_i) \,\}$$

Intuitively, for a set of axioms $\{a_1, \ldots, a_n\} \subseteq \mathcal{T}_{\mathrm{Norm}}$, $[\{a_1, \ldots, a_n\}]_{n_{b,\mathcal{T}}}$ contains all the subsets $S \subseteq \mathcal{T}$ such that $\{a_1, \ldots, a_n\} \subseteq S_{\mathrm{Norm}}$.

Lemma 24. *Let $\mathcal{T}$ be a TBox such that $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$, and let $\mathcal{M}, \mathcal{M}' \subseteq \mathcal{T}_{Norm}$ such that $\mathcal{M} \subseteq \mathcal{M}'$. Then for every $X' \in [\mathcal{M}']_{n_b,\mathcal{T}}$ there exists $X \in [\mathcal{M}]_{n_b,\mathcal{T}}$ such that $X \subseteq X'$.*

Proof. Follows immediately from the definition of $[\mathcal{M}']_{n_b,\mathcal{T}}$. □

Lemma 25. *Let $\mathcal{T}$ be a TBox such that $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$, and let $\mathcal{M} \subseteq \mathcal{T}$. Then $\mathcal{M} \in [\mathcal{M}_{Norm}]_{n_b,\mathcal{T}}$.*

Proof. Follows immediately from the definition of $[\mathcal{M}_{\text{Norm}}]_{n_b,\mathcal{T}}$ using Condition (ii). □

The next major result will be established in Lemma 28, where we show how MinAs for inclusions $C \sqsubseteq D$ w.r.t. general TBoxes $\mathcal{T}$ can be obtained from MinAs for inclusions $C \sqsubseteq D$ w.r.t. normalised TBoxes.

Lemma 26. *Let $\mathcal{T}$ be a TBox such that $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$, and let C, D be concepts such that $(\mathsf{sig}(C) \cup \mathsf{sig}(D)) \cap \mathsf{N_N} = \emptyset$. Then*

$$\{\, \mathcal{M} \mid \mathcal{M} \subseteq \mathcal{T} \text{ such that } C \sqsubseteq_{\mathcal{M}} D \,\} = \bigcup_{\substack{\mathcal{M} \subseteq \mathcal{T}_{Norm} \\ \text{such that } C \sqsubseteq_{\mathcal{M}} D}} [\mathcal{M}]_{n_b,\mathcal{T}}$$

Proof. "$\subseteq$" Let $\mathcal{M} \subseteq \mathcal{T}$ such that $\mathcal{C} \sqsubseteq_{\mathcal{M}} \mathcal{D}$. Then $\mathcal{C} \sqsubseteq_{\mathcal{M}_{\text{Norm}}} \mathcal{D}$. It remains to observe that $\mathcal{M} \in [\mathcal{M}_{\text{Norm}}]_{n_b,\mathcal{T}}$ holds by Lemma 25.

"$\supseteq$" Let $\mathcal{M} \in [\mathcal{M}']_{n_b,\mathcal{T}}$ such that there exists $\mathcal{M}' \subseteq \mathcal{T}_{\text{Norm}}$ with $C \sqsubseteq_{\mathcal{M}'} D$. As $\mathcal{M} \in [\mathcal{M}']_{n_b,\mathcal{T}}$, it holds that $\mathcal{M}' \subseteq \mathcal{M}_{\text{Norm}}$. Hence, we obtain $C \sqsubseteq_{\mathcal{M}_{\text{Norm}}} D$, which allows us to conclude that $C \sqsubseteq_{\mathcal{M}} D$ holds. □

Lemma 27. *Let $\mathcal{T}$ be a TBox such that $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$, and let C, D be concepts such that $(\mathsf{sig}(C) \cup \mathsf{sig}(D)) \cap \mathsf{N_N} = \emptyset$. Then*

$$\min_{\subseteq} \left(\bigcup_{\mathcal{M} \in \mathsf{MinA}_{\mathcal{T}_{Norm}}(C \sqsubseteq D)} [\mathcal{M}]_{n_b,\mathcal{T}} \right) = \min_{\subseteq} \left(\bigcup_{\substack{\mathcal{M} \subseteq \mathcal{T}_{Norm} \\ \text{such that } C \sqsubseteq_{\mathcal{M}} D}} [\mathcal{M}]_{n_b,\mathcal{T}} \right).$$

Proof. Let S_l be the set on the left-hand side of the equality, and let S_r be the set on the right-hand side.

"$\subseteq$" Let $\mathcal{M} \in S_1$, i.e. $\mathcal{M}$ is a minimal set (w.r.t. set inclusion) such that there exists $\mathcal{M}' \in \mathsf{MinA}_{\mathcal{T}_{\text{Norm}}}(C \sqsubseteq D)$ with $\mathcal{M} \in [\mathcal{M}']_{n_b,\mathcal{T}}$. If we now assume towards a contradiction that $\mathcal{M} \notin S_r$, i.e. there exists $X' \subseteq \mathcal{T}_{\text{Norm}}$ such that $C \sqsubseteq_{X'} D$ and there exists $\mathcal{M}'' \in [X']_{n_b,\mathcal{T}}$ with $\mathcal{M}'' \subsetneq \mathcal{M}$, then by Lemma 24 there would exist $\mathcal{M}''' \in S_l$ with $\mathcal{M}''' \subseteq \mathcal{M}'' \subsetneq \mathcal{M}$, contradicting $\mathcal{M} \in S_l$.

"$\supseteq$" Let $\mathcal{M} \in S_r$, i.e. $\mathcal{M}$ is a minimal set (w.r.t. set inclusion) such that there exists $X' \subseteq \mathcal{T}_{\text{Norm}}$ with $C \sqsubseteq_{X'} D$ and $\mathcal{M} \in [X']_{n_b,\mathcal{T}}$. Hence there exists $X \in \mathsf{MinA}_{\mathcal{T}_{\text{Norm}}}(C \sqsubseteq D)$ with $X \subseteq X'$ and there exists $\mathcal{M}_l \in [X]_{n_b,\mathcal{T}}$ with $\mathcal{M}_l \subseteq \mathcal{M}$ by Lemma 24. Furthermore, there exists $\mathcal{M}_l' \in S_l$ with $\mathcal{M}_l' \subseteq \mathcal{M}_l \subseteq \mathcal{M}$. If we now assume towards a contradiction that $\mathcal{M}_l' \subsetneq \mathcal{M}$, there would exist $\mathcal{M}'' \in S_r$ with $\mathcal{M}'' \subseteq \mathcal{M}_l' \subsetneq \mathcal{M}$, contradicting $\mathcal{M} \in S_r$. □

Lemma 28. *Let $\mathcal{T}$ be a TBox such that $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$, and let C, D be concepts such that $(\mathsf{sig}(C) \cup \mathsf{sig}(D)) \cap \mathsf{N_N} = \emptyset$. Then*

$$\mathsf{MinA}_{\mathcal{T}}(C \sqsubseteq D) = \min_{\subseteq} \left(\bigcup_{\mathcal{M} \in \mathsf{MinA}_{\mathcal{T}_{Norm}}(C \sqsubseteq D)} [\mathcal{M}]_{n_b, \mathcal{T}} \right)$$

Proof. Follows from Lemmata 26 and 27. □

The next important result will now be established in Lemma 31. We show how MinAs for inclusions $C \sqsubseteq D$ w.r.t. a TBox $\mathcal{T}$ can be obtained from MinAs for inclusions $A \sqsubseteq B$ (where A and B are special concept names) w.r.t. an extended TBox $\mathcal{T}' \supseteq \mathcal{T}$.

Lemma 29. *Let $\mathcal{T}$ be a TBox such that $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$, and let C, D be concepts such that $(\mathsf{sig}(C) \cup \mathsf{sig}(D)) \cap \mathsf{N_N} = \emptyset$. Additionally, let $A_C, A_D \in \mathsf{N_N}$. Then*

$$\{ \mathcal{M} \subseteq \mathcal{T} \mid C \sqsubseteq_{\mathcal{M}} D \} = \{ \mathcal{M} \setminus X \mid \mathcal{M} \subseteq \mathcal{T} \cup X \text{ and } A_C \sqsubseteq_{\mathcal{M}} A_D \}$$

where $X = \{A_C \sqsubseteq C, D \sqsubseteq A_D\}$.

Proof. "$\subseteq$" Let $\mathcal{M} \subseteq \mathcal{T}$ such that $C \sqsubseteq_{\mathcal{M}} D$. Then it is easy to see that $A_C \sqsubseteq_{\mathcal{M} \cup X} A_D$ holds.

"$\supseteq$" Let $\mathcal{M} \subseteq \mathcal{T} \cup X$ such that $A_C \sqsubseteq_{\mathcal{M}} A_D$. Then it is easy to see that $C \sqsubseteq_{\mathcal{M} \setminus X} D$ holds. □

Lemma 30. *Let $\mathcal{T}$ be a TBox such that $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$, and let C, D be concepts such that $(\mathsf{sig}(C) \cup \mathsf{sig}(D)) \cap \mathsf{N_N} = \emptyset$. Additionally, let $A_C, A_D \in \mathsf{N_N}$. Then*

$$\min_{\subseteq} \{ \mathcal{M} \setminus X \mid \mathcal{M} \in \mathsf{MinA}_{\mathcal{T}'}(A_C \sqsubseteq A_D) \} = \min_{\subseteq} \{ \mathcal{M} \setminus X \mid \mathcal{M} \subseteq \mathcal{T}' \text{ and } A_C \sqsubseteq_{\mathcal{M}} A_D \}$$

where $\mathcal{T}' = \mathcal{T} \cup X$ with $X = \{A_C \sqsubseteq C, D \sqsubseteq A_D\}$.

Proof. "$\subseteq$" Let $\mathcal{M} \in \mathsf{MinA}_{\mathcal{T}'}(A_C \sqsubseteq A_D)$ such that there does not exist $\mathcal{M}' \in \mathsf{MinA}_{\mathcal{T}'}(A_C \sqsubseteq A_D)$ with $\mathcal{M}' \setminus X \subsetneq \mathcal{M} \setminus X$. Thus $A_C \sqsubseteq_{\mathcal{M}} A_D$. If we now assume towards a contradiction that there exists $\mathcal{M}'' \subseteq \mathcal{T}'$ with $A_C \sqsubseteq_{\mathcal{M}''} A_D$ and $\mathcal{M}'' \setminus X \subsetneq \mathcal{M} \setminus X$, it would follow that there exists $\mathcal{M}''' \in \mathsf{MinA}_{\mathcal{T}'}(A_C \sqsubseteq A_D)$ with $\mathcal{M}''' \subseteq \mathcal{M}''$, i.e. $\mathcal{M}''' \setminus X \subsetneq \mathcal{M} \setminus X$, contradicting our assumptions.

"$\supseteq$" Let $\mathcal{M} \subseteq \mathcal{T}'$ such that $A_C \sqsubseteq_{\mathcal{M}} A_D$ and such that there does not exist $\mathcal{M}' \subseteq \mathcal{T}'$ with $A_C \sqsubseteq_{\mathcal{M}'} A_D$ and $\mathcal{M}' \setminus X \subsetneq \mathcal{M} \setminus X$. We can infer that there exists $\mathcal{M}'' \in \mathsf{MinA}_{\mathcal{T}'}(A_C \sqsubseteq A_D)$ with $\mathcal{M}'' \subseteq \mathcal{M}$, which implies that $\mathcal{M}'' \setminus X \subseteq \mathcal{M} \setminus X$. We can immediately infer that $\mathcal{M}'' \setminus X = \mathcal{M} \setminus X$ as we would obtain a contradiction with our assumptions otherwise. Similarly, if we assume towards a contradiction that there exists $\mathcal{M}''' \in \mathsf{MinA}_{\mathcal{T}'}(A_C \sqsubseteq A_D)$ with $\mathcal{M}''' \setminus X \subsetneq \mathcal{M} \setminus X$, we would again obtain a contradiction with our assumptions. □

Algorithm 3

procedure FINDMINAS($\mathcal{T}, C, D$)
  $X := \{A_C \sqsubseteq C, D \sqsubseteq A_D\}$ with $A_C, A_D \in \mathsf{N_C}$
  $\tilde{\mathcal{T}} := \mathrm{Norm}(\mathcal{T})$
  $\tilde{X} := \mathrm{Norm}(X)$
  $S_{\tilde{\mathcal{T}}\cup\tilde{X}} := \min_{\subseteq}\{\mathrm{Axioms}(\Delta) \mid \Delta$ admissible derivation of $A_C \sqsubseteq A_D$ w.r.t. $\tilde{\mathcal{T}} \cup \tilde{X}\}$
  $S_{\mathcal{T}\cup X} := \min_{\subseteq} \bigcup_{\mathcal{M}\in S_{\tilde{\mathcal{T}}\cup\tilde{X}}} [\mathcal{M}]_{n_b,\mathcal{T}\cup X}$
  return $\min_{\subseteq}\{\, \mathcal{M} \setminus X \mid \mathcal{M} \in S_{\mathcal{T}\cup X} \,\}$

Lemma 31. *Let $\mathcal{T}$ be a TBox such that $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$, and let C, D be concepts such that $(\mathsf{sig}(C) \cup \mathsf{sig}(D)) \cap \mathsf{N_N} = \emptyset$. Additionally, let $A_C, A_D \in \mathsf{N_N}$. Then*

$$\mathsf{MinA}_{\mathcal{T}}(C \sqsubseteq D) = \min_{\subseteq}\{\, \mathcal{M} \setminus X \mid \mathcal{M} \in \mathsf{MinA}_{\mathcal{T}'}(A_C \sqsubseteq A_D) \,\}$$

where $\mathcal{T}' = \mathcal{T} \cup X$ with $X = \{A_C \sqsubseteq C, D \sqsubseteq A_D\}$.

Proof. Follows from Lemmata 29 and 30. □

We can now put all our results together. The following algorithm FINDMINAS($\mathcal{T}, C, D$) can be used to compute all the MinAs for the inclusion $C \sqsubseteq D$ w.r.t. $\mathcal{T}$ by first computing all the MinAs for the inclusion $A_C \sqsubseteq A_D$ w.r.t. $\mathcal{T}_{\mathrm{Norm}} \cup \mathrm{Norm}(A_C \sqsubseteq C) \cup \mathrm{Norm}(D \sqsubseteq A_D)$ (where $A_C, A_D \in \mathsf{N_N}$).

Theorem 32. *Let $\mathcal{T}$ be a TBox such that $\mathsf{sig}(\mathcal{T}) \cap \mathsf{N_N} = \emptyset$, and let C, D be concepts such that $(\mathsf{sig}(C) \cup \mathsf{sig}(D)) \cap \mathsf{N_N} = \emptyset$. Additionally, let $A_C, A_D \in \mathsf{N_N}$. Then*

$$\mathsf{MinA}_{\mathcal{T}}(C \sqsubseteq D) = \textsc{FindMinAs}(\mathcal{T}, C, D).$$

Proof. Follows from Theorem 23 and Lemmata 28 and 31. □