Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence

Abduction Framework for Repairing Incomplete $\mathcal{EL}$ Ontologies: Complexity Results and Algorithms

Fang Wei-Kleiner and **Zlatan Dragisic** and **Patrick Lambrix**
Department of Computer and Information Science
Swedish e-Science Research Centre
Linköping University, Sweden

Abstract

In this paper we consider the problem of repairing missing is-a relations in ontologies. We formalize the problem as a generalized TBox abduction problem (GTAP). Based on this abduction framework, we present complexity results for the existence, relevance and necessity decision problems for the GTAP with and without some specific preference relations for ontologies that can be represented using a member of the $\mathcal{EL}$ family of description logics. Further, we present algorithms for finding solutions, a system as well as experiments.

Introduction

Abduction is a reasoning method to generate explanations for observed symptoms and manifestations. When the application domain is described by a logical theory, it is called *logic-based abduction* (Eiter and Gottlob 1995). Logic-based abduction is widely applied in diagnosis, planning, and database updates (Kakas and Mancarella 1990), among others. Recently, logic-based abduction has provided the theoretical ground for the field of ontology debugging and repairing, in which inconsistent and incomplete information of ontology is discovered and repaired (Section Related Work).

In this paper, we consider ontologies that are represented by description logics (DLs), more specifically represented by TBoxes in the $\mathcal{EL}$ family. The $\mathcal{EL}$ family of description logics is highly relevant for the representation of lightweight ontologies. For instance, several of the major ontologies in the biomedical domain, e.g., SNOMED[1] and Gene Ontology (Ashburner et al. 2000), can be represented in $\mathcal{EL}$ or small extensions thereof (Baader, Brandt, and Lutz 2005).

Defects in ontologies can take different forms (e.g. (Kalyanpur et al. 2006b)), such as the *modeling defects* which require domain knowledge to detect and resolve, and *semantic* defects such as unsatisfiable concepts and inconsistent ontologies. In this paper we tackle a particular kind of modeling defects: defects in the is-a structure in ontologies. Missing is-a structure leads to valid conclusions to be missed and therefore affects the quality of the application results. Debugging defects in ontologies consists of two phases, detection and repair. In this paper we assume that the detection phase has been performed and focus on the repairing phase. There are many approaches to detect missing is-a relations (see Section Related Work). However, in general, these approaches do not detect *all* missing is-a relations and in several cases even only few. Therefore we assume that we have obtained a set of missing is-a relations for a given ontology (but not necessarily all). Under this circumstance, the easiest way to repair is to just add the missing is-a relations to the ontology. For instance, T in Figure 1 represents a small ontology inspired by Galen ontology[2], that is relevant for our discussions. Assume that we have detected set M (in Figure 1) of missing is-a relations. Obviously, adding these relations to the ontology will repair the missing is-a structure. However, there are other more interesting possibilities. For instance, adding is-a relations Carditis $\sqsubseteq$ CardioVascularDisease, GranulomaProcess $\sqsubseteq$ PathologicalProcess, Wound $\sqsubseteq$ PathologicalPhenomenon, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess and SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess also repairs the missing is-a structure. Further, these is-a relations are correct according to the domain and constitute new is-a relations (e.g. SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess) that were not derivable from the ontology and not originally detected by the detection algorithm.[3] We also note that from a logical point of view, adding Carditis $\sqsubseteq$ Fracture, GranulomaProcess $\sqsubseteq$ PathologicalProcess, Wound $\sqsubseteq$ PathologicalPhenomenon, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess and SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess also repairs the missing is-a structure. However, from the point of view of the domain, this solution is not correct. Therefore, as for all approaches for debugging modeling defects, a domain expert needs to validate the logical solutions.

The above example shows that the framework of TBox abduction defined in (Elsenbroich, Kutz, and Sattler 2006) catches the basic semantics of repairing is-a relations. Let T denote the current ontology based on a certain formalism. The set of identified missing is-a relations M (atomic concept subsumptions) represents the manifestation. To re-

[1] http://www.ihtsdo.org/snomed-ct/

[2] http://www.co-ode.org/galen/

[3] Thus, the approach in this paper can also be seen as a detection method that takes already found missing is-a relations as input.

C = { GranulomaProcess, CardioVascularDisease, PathologicalPhenomenon, Fracture, Endocarditis, Carditis, InflammationProcess, PathologicalProcess, NonNormalProcess, Wound, BurningProcess, SoftTissueTraumaProcess, TraumaticProcess}

T = { CardioVascularDisease $\sqsubseteq$ PathologicalPhenomenon, Fracture $\sqsubseteq$ PathologicalPhenomenon, $\exists$isImmediateConsequence.PathologicalProcess $\sqsubseteq$ PathologicalPhenomenon, Endocarditis $\sqsubseteq$ Carditis, Endocarditis $\sqsubseteq$ $\exists$isImmediateConsequence.InflammationProcess, PathologicalProcess $\sqsubseteq$ NonNormalProcess, hasAssociatedProcess $\sqsubseteq$ isImmediateConsequence, Wound $\sqsubseteq$ $\exists$hasAssociatedProcess.SoftTissueTraumaProcess }

M = { Endocarditis $\sqsubseteq$ PathologicalPhenomenon, GranulomaProcess $\sqsubseteq$ NonNormalProcess, Wound $\sqsubseteq$ PathologicalPhenomenon, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, BurningProcess $\sqsubseteq$ TraumaticProcess }

The following is-a relations are correct according to the domain, i.e Or returns $true$ for:
GranulomaProcess $\sqsubseteq$ InflammationProcess, GranulomaProcess $\sqsubseteq$ PathologicalProcess, GranulomaProcess $\sqsubseteq$ NonNormalProcess, CardioVascularDisease $\sqsubseteq$ PathologicalPhenomenon, Fracture $\sqsubseteq$ PathologicalPhenomenon, Endocarditis $\sqsubseteq$ PathologicalPhenomenon, Endocarditis $\sqsubseteq$ Carditis, Endocarditis $\sqsubseteq$ CardioVascularDisease,
Carditis $\sqsubseteq$ PathologicalPhenomenon, Carditis $\sqsubseteq$ CardioVascularDisease, InflammationProcess $\sqsubseteq$ PathologicalProcess, InflammationProcess $\sqsubseteq$ NonNormalProcess, PathologicalProcess $\sqsubseteq$ NonNormalProcess, Wound $\sqsubseteq$ PathologicalPhenomenon, TraumaticProcess $\sqsubseteq$ NonNormalProcess, TraumaticProcess $\sqsubseteq$ PathologicalProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess, SoftTissueTraumaProcess $\sqsubseteq$ NonNormalProcess, SoftTissueTraumaProcess $\sqsubseteq$ PathologicalProcess, BurningProcess $\sqsubseteq$ NonNormalProcess, BurningProcess $\sqsubseteq$ PathologicalProcess, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, BurningProcess $\sqsubseteq$ TraumaticProcess.

Let $\mathcal{P}$ = GTAP(T, C, Or, M).

Figure 1: Small $\mathcal{EL}^{++}$ example.

pair the ontology, the ontology should be extended with a set S of atomic concept subsumptions (repair) such that the extended ontology is consistent and the missing is-a relations in M are derivable from the extended ontology. That is, $T \cup S \models M$ holds.

However, there are properties of ontology repairing of missing is-a relations which distinguish themselves from the classic abduction framework. We summarize them as P1 and P2, and give the intuition behind them.

P1: Oracle function Or instead of hypothesis H.

In the classic abduction framework there is a hypothesis H from which the solution S is chosen such that $S \subseteq H$ holds. The corresponding component is the set of atomic concept subsumptions that should be correct according to the domain. In general, this set is not known beforehand. In the repairing scenario, a domain expert decides whether an atomic concept subsumption is correct according to the domain, and can return $true$ or $false$ like an oracle. Consequently, we formulate this function as Or that when given an atomic concept subsumption, returns $true$ or $false$. It is then required that for every atomic concept subsumption $s \in S$, we have that $Or(s) = true$.

P2: Informativeness as one of the preference criteria.

Ontology repairing of missing is-a relations follows different preference criteria from the logic-based abduction framework, in the sense that a more *informative* solution is preferred to a less informative one. Note that the informativeness is a measurement for how much information the added subsumptions (i.e. solution S) can derive. (See Definition 2 for the precise formulation.) This is in contrast to the criteria of minimality (e.g. subset minimality, cardinality minimality) from the abduction framework. In principle this difference on the preference stems from the original purpose of the two formalisms. The abduction framework is often used for diagnostic scenarios, thus the essential goal is to confine the cause of the problem as small as possible. Whilst for ontology repairing, the goal is to add more subsumptions to enrich the ontology. As long as the added rules are correct, a more informative repairing means more enrichment to the ontology. However, there are technical difficulties in finding the most informative solution as such. A brute-force method to create a most informative solution is to check for each pair of atomic concepts A and B, whether $Or(A \sqsubseteq B) = true$. In practice, for large ontologies this is infeasible. Therefore, it is not clear how to *generate* such a solution in practice due to the missing hypothesis H. Further, we might obtain a solution with redundancy. For this purpose, we would like to add another minimality preference, namely subset minimality to the informativeness preference. That is, we prefer a solution which is both semantically maximal (most informative) and subset minimal. Combining these two preferences drives us to three distinct interpretations (Definitions 5 - 7), depending on what kind of priority we assign for the single preferences.

In this paper we focus on the formalization of the problems and conduct complexity analysis on the decision problems regarding the various preference criteria for $\mathcal{EL}^{++}$ and $\mathcal{EL}$ ontologies. We prove complexity results on all the decision problems (see Table 2). The complexity results provide a guideline on the choosing of suitable preference criteria for designing repairing algorithms in practice. As a result, the final part of the paper is dedicated to concrete algorithms for finding skyline optimal solutions, together with a system based on the algorithms as well as experiments. The contributions of this paper are the following.
- We formalize the repairing of the missing is-a structure in an ontology as a generalized version of the TBox abduction

Name	Syntax	Semantics
top	$\top$	$\Delta^{\mathcal{I}}$
bottom	$\bot$	$\emptyset$
nominal	$\{a\}$	$\{a^{\mathcal{I}}\}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} : (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
GCI	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
RI	$r_1 \circ \ldots \circ r_k \sqsubseteq r$	$r_1^{\mathcal{I}} \circ \ldots \circ r_k^{\mathcal{I}} \subseteq r^{\mathcal{I}}$

Table 1: $\mathcal{EL}^{++}$ Syntax and Semantics

problem (GTAP).
- We present complexity results for the decision problems for GTAP in both $\mathcal{EL}^{++}$ and $\mathcal{EL}$ with and without the preference relations subset minimality and semantic maximality as well as three ways of combining these (maxmin, minmax, skyline). This combination is novel and is important for GTAP.
- We provide algorithms for finding skyline optimal solutions to GTAP in $\mathcal{EL}$ and $\mathcal{EL}^{++}$, and show experiments using an implemented system.

The description logics $\mathcal{EL}^{++}$ and $\mathcal{EL}$

Concept descriptions are constructed inductively from a set N_C of atomic concepts and a set N_R of atomic roles and (possibly) a set N_I of individual names. The concept constructors are the top concept $\top$, bottom concept $\bot$, nominals, conjunction, and existential restriction, and a restricted form of concrete domains. In this paper, we consider the version of $\mathcal{EL}^{++}$ without concrete domains. Note that this simplification does not affect the complexity results presented later on. For the syntax of the different constructors see Table 1. An interpretation $\mathcal{I}$ consists of a non-empty set $\Delta^{\mathcal{I}}$ and an interpretation function $\cdot^{\mathcal{I}}$ which assigns to each atomic concept $A \in N_C$ a subset $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, to each atomic role $r \in N_R$ a relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and to each individual name $a \in N_I$ an individual $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. The interpretation function is straightforwardly extended to complex concepts. An $\mathcal{EL}^{++}$ TBox (named CBox in (Baader, Brandt, and Lutz 2005)) is a finite set of *general concept inclusions* (GCIs) and *role inclusions* (RIs) whose syntax can be found in the lower part of Table 1. Note that a finite set of GCIs is called a *general TBox*. An interpretation $\mathcal{I}$ is a *model* of a TBox T if for each GCI and RI in T, the conditions given in the third column of Table 1 are satisfied. $\mathcal{EL}$ has the restricted form of $\mathcal{EL}^{++}$ which allows for concept constructors of top concept $\top$, conjunction and existential restriction. An $\mathcal{EL}$ TBox contains only GCIs. The main reasoning task for description logics is subsumption in which the problem is to decide for a TBox T and concepts C and D whether $T \models C \sqsubseteq D$. Subsumption in $\mathcal{EL}^{++}$ is polynomial even w.r.t. general TBoxes (Baader, Brandt, and Lutz 2005).

Abduction Framework

In the following we explain how the problem of finding possible ways to repair the missing is-a structure in an ontology is formalized as a generalized version of the TBox abduction problem as defined in (Elsenbroich, Kutz, and Sattler 2006).

Definition 1 *(GENERALIZED TBOX ABDUCTION) Let T be a TBox in $\mathcal{EL}^{++}$ and C be the set of all atomic concepts in T. Let $M = \{A_i \sqsubseteq B_i \mid A_i, B_i \in C\}$ be a finite set of TBox assertions. Let $Or : \{C_i \sqsubseteq D_i \mid C_i, D_i \in C\} \rightarrow \{true, false\}$. A solution to the generalized TBox abduction problem (GTAP) (T, C, Or, M) is any finite set $S = \{E_i \sqsubseteq F_i \mid E_i, F_i \in C \;\wedge\; Or(E_i \sqsubseteq F_i) = true\}$ of TBox assertions, such that $T \cup S$ is consistent and $T \cup S \models M$. The set of all such solutions is denoted as $\mathcal{S}(T, C, Or, M)$.*

As noted as property P1 in the Introduction, in the classic abduction problem there is usually no oracle Or, but a set of abduciles H that restricts the solution space. A major difference is that H is usually given, and finding solutions can therefore start from H. In GTAP on the other hand this is not possible, but (partial) solutions are validated using Or.

As an example, consider GTAP $\mathcal{P}$ as defined in Figure 1. Then {GranulomaProcess $\sqsubseteq$ InflammationProcess, Carditis $\sqsubseteq$ CardioVascularDisease, InflammationProcess $\sqsubseteq$ PathologicalProcess, TraumaticProcess $\sqsubseteq$ NonNormalProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess, SoftTissueTraumaProcess $\sqsubseteq$ PathologicalProcess, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess} is a solution for $\mathcal{P}$. Another solution is {Carditis $\sqsubseteq$ CardioVascularDisease, GranulomaProcess $\sqsubseteq$ PathologicalProcess, Wound $\sqsubseteq$ PathologicalPhenomenon, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess} as shown in Section Introduction.

There can be many solutions for a GTAP and not all solutions are equally interesting. Therefore, we propose two preference criteria on the solutions as well as different ways to combine them. The first criterion is a criterion that is not used in other abduction problems, but that is particularly important for GTAP. In GTAP it is important to find solutions that add to the ontology as much information as possible that is correct according to the domain. Therefore, the first criterion prefers solutions that imply more information.

Definition 2 *(MORE INFORMATIVE) Let S and S' be two solutions to the GTAP (T, C, Or, M). S is said to be* more informative *than S' iff $T \cup S \models S'$ and $T \cup S' \not\models S$.*

Further, we say that S is equally informative *as S' iff $T \cup S \models S'$ and $T \cup S' \models S$.*

Consider two solutions to $\mathcal{P}$, S_1 = {Carditis $\sqsubseteq$ CardioVascularDisease, GranulomaProcess $\sqsubseteq$ PathologicalProcess, SoftTissueTraumaProcess $\sqsubseteq$ PathologicalProcess, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess} and S_2 = {Carditis $\sqsubseteq$ CardioVascularDisease, GranulomaProcess $\sqsubseteq$ PathologicalProcess, Wound $\sqsubseteq$ PathologicalPhenomenon, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess}. In this case solution S_1 is more informative than S_2.

Definition 3 *(SEMANTIC MAXIMALITY) A solution S to the GTAP (T, C, Or, M) is said to be semantically maximal iff there is no solution S' which is more informative than S. The set of all semantically maximal solutions is denoted as $\mathcal{S}^{max}(T, C, Or, M)$.*

An example of a semantically maximal solution to $\mathcal{P}$ is {BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, GranulomaProcess $\sqsubseteq$ InflammationProcess, Carditis $\sqsubseteq$ CardioVascularDisease, InflammationProcess $\sqsubseteq$ PathologicalProcess, TraumaticProcess $\sqsubseteq$ NonNormalProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess, TraumaticProcess $\sqsubseteq$ PathologicalProcess}.

The second criterion is a classical criterion in abduction problems. It requires that no element in a solution is redundant.

Definition 4 *(SUBSET MINIMALITY) A solution S to the GTAP (T, C, Or, M) is said to be subset minimal iff there is no proper subset $S' \subsetneq S$ such that S' is a solution. The set of all subset minimal solutions is denoted as $\mathcal{S}_{min}(T, C, Or, M)$.*

An example of a subset minimal solution for $\mathcal{P}$ is {GranulomaProcess $\sqsubseteq$ InflammationProcess, InflammationProcess $\sqsubseteq$ PathologicalProcess, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess, SoftTissueTraumaProcess $\sqsubseteq$ PathologicalProcess}. On the other hand, solution {TraumaticProcess $\sqsubseteq$ NonNormalProcess, GranulomaProcess $\sqsubseteq$ InflammationProcess, InflammationProcess $\sqsubseteq$ PathologicalProcess, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess, SoftTissueTraumaProcess $\sqsubseteq$ PathologicalProcess} is not subset minimal as it contains TraumaticProcess $\sqsubseteq$ NonNormalProcess which is redundant for repairing the missing is-a relations.

In practice, both of the above two criteria are desirable. We therefore define ways to combine them depending on what kind of priority we assign for the single preferences.

Definition 5 *(COMBINING WITH PRIORITY FOR SEMANTIC MAXIMALITY) A solution S to the GTAP (T, C, Or, M) is said to be maxmin optimal iff S is semantically maximal and there does not exist another semantically maximal solution S' such that S' is a proper subset of S. The set of all maxmin optimal solutions is denoted as $\mathcal{S}_{min}^{\mathbf{max}}(T, C, Or, M)$.*

As an example, {GranulomaProcess $\sqsubseteq$ InflammationProcess, InflammationProcess $\sqsubseteq$ PathologicalProcess, TraumaticProcess $\sqsubseteq$ PathologicalProcess, Carditis $\sqsubseteq$CardioVascularDisease, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess} is a maxmin optimal solution for $\mathcal{P}$. The advantage of maxmin optimal solutions is that a maximal body of correct information is added to the ontology and without redundancy. For GTAP these are the most attractive solutions, but as mentioned before it is not clear how to generate such a solution for large ontologies in practice.

Definition 6 *(COMBINING WITH PRIORITY FOR SUBSET MINIMALITY) A solution S to the GTAP (T, C, Or, M) is said to be minmax optimal iff S is subset minimal and there does not exist another subset minimal solution S' such that S' is more informative than S. The set of all minmax optimal solutions is denoted as $\mathcal{S}_{\mathbf{min}}^{max}(T, C, Or, M)$.*

As an example, {BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, GranulomaProcess $\sqsubseteq$ InflammationProcess, InflammationProcess $\sqsubseteq$ PathologicalProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess, SoftTissueTraumaProcess $\sqsubseteq$ PathologicalProcess} is a minmax optimal solution for $\mathcal{P}$. In practice, minmax optimal solutions ensure fewer is-a relations to be added, thus avoiding redundancy. This is desirable if the domain expert would prefer to look at as small solutions as possible. The disadvantage is that there may be correct relations that are not derivable when they are not included in the solution.

Definition 7 *(SKYLINE OPTIMAL) A solution S to the GTAP (T, C, Or, M) is said to be skyline optimal iff there does not exist another solution S' such that S' is a proper subset of S and S' is equally informative as S. The set of all skyline optimal solutions is denoted as $\mathcal{S}_{min}^{max}(T, C, Or, M)$.*

All subset minimal, minmax optimal and maxmin optimal solutions are also skyline optimal solutions. However, there are semantically maximal solutions that are not skyline optimal. For example, S = {BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, GranulomaProcess $\sqsubseteq$ InflammationProcess, Carditis $\sqsubseteq$ CardioVascularDisease, InflammationProcess $\sqsubseteq$ PathologicalProcess, TraumaticProcess $\sqsubseteq$ NonNormalProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess, TraumaticProcess $\sqsubseteq$ PathologicalProcess} is a semantically maximal solution for $\mathcal{P}$, but it is not skyline optimal as its subset S $\setminus$ {TraumaticProcess $\sqsubseteq$ NonNormalProcess} is equally informative. There also exist skyline optimal solutions that are not subset minimal solutions. For instance, {TraumaticProcess $\sqsubseteq$ NonNormalProcess, GranulomaProcess $\sqsubseteq$ InflammationProcess, InflammationProcess $\sqsubseteq$ PathologicalProcess, BurningProcess $\sqsubseteq$ SoftTissueTraumaProcess, SoftTissueTraumaProcess $\sqsubseteq$ TraumaticProcess, SoftTissueTraumaProcess $\sqsubseteq$ PathologicalProcess} is a skyline optimal solution that is not subset minimal as removing TraumaticProcess $\sqsubseteq$ NonNormalProcess would still yield a solution (although not as informative). Skyline optimal is a relaxed criterion. It requires subset minimality for some level of informativeness. Although maxmin solutions are preferred, in practice, it is not clear how to generate a maxmin solution, except for a brute-force method that would query the oracle with, for larger ontologies, unfeasibly many questions. Therefore, a skyline solution is the next best thing and, in the case solutions exist, it is easy to generate *a* skyline optimal solution. However, the difficulty lies in reaching an as high level of informativeness as possible.

Further, in addition to finding solutions, traditionally, there are three main decision problems for logic-based abduction: existence, relevance and necessity.

Definition 8 *Given a GTAP (T, C, Or, M) we define the following decision problems:*

Existence *$\mathcal{S}(T, C, Or, M) \neq \emptyset$?*

	$\mathcal{EL}$			$\mathcal{EL}^{++}$		
Decision problems	Existence	Relevance	Necessity	Existence	Relevance	Necessity
General	in P	in P	in P	NP-complete	NP-complete	co-NP-complete
Subset Minimality	in P	NP-complete	in P	NP-complete	NP-complete	co-NP-complete
Semantic Maximality	in P	in P	in P	NP-complete	NP-complete	co-NP-complete
Minmax	in P	NP-complete	in P	NP-complete	Σ_2^P-complete	Π_2^P-complete
Maxmin	in P	in P	in P	NP-complete	NP-complete	co-NP-complete
Skyline	in P	NP-complete	in P	NP-complete	NP-complete	co-NP-complete

Table 2: Complexity Results of GTAP

Relevance *Given ψ, does a solution $S \in \mathcal{S}(T, C, Or, M)$ exist such that $\psi \in S$?*

Necessity *Given ψ, do all the solutions in $\mathcal{S}(T, C, Or, M)$ contain ψ?*

If we replace $\mathcal{S}$ in Definition 8 with $\mathcal{S}_{min}$, $\mathcal{S}^{max}$, $\mathcal{S}^{max}_{\mathbf{min}}$, $\mathcal{S}^{\mathbf{max}}_{min}$ and $\mathcal{S}^{max}_{min}$, respectively, we obtain the GTAP decision problems under the criteria of subset minimality, semantic maximality and the combinations.

Complexity Results

The summary of the complexity results of the GTAP decision problems for both $\mathcal{EL}$ and $\mathcal{EL}^{++}$ is shown in Table 2. All the proofs can be found in (Wei-Kleiner, Dragisic, and Lambrix 2014). It is confirmed that abduction is harder than deduction over the same formalism. The two properties P1 and P2 in Introduction provide certain guidelines for choosing the suitable preferences. According to P1, we can not generate all the correct subsumptions. Thus the preferences of semantic maximality and maxmin are not applicable, although the complexity is low. As a consequence, minmax and skyline are the suitable candidates. For $\mathcal{EL}^{++}$, the complexity for the relevance problems of minmax is unfortunately high, thus skyline turns out to be the best choice.

Algorithm

In this section we present algorithms for solving GTAP (T, C, Or, M) for $\mathcal{EL}$ and $\mathcal{EL}^{++}$. The algorithms guarantee a skyline-optimal solution when $\forall m \in M : Or(m) = true$ and $T \cup M$ is consistent. We note that the latter is always true in $\mathcal{EL}$. The main intuition for the algorithms is to first solve a GTAP with one missing is-a relation for each $m \in M$. This means, for each missing is-a relation m in the original GTAP, we find is-a relations that are correct according to the oracle Or and that when added to the ontology would make m derivable. Then we combine the results for each $m \in M$ to obtain a solution for the original GTAP. Finally, the algorithms try to find more informative solutions by treating the newly added is-a relations as missing is-a relations and solving a new GTAP. It can be shown that a skyline-optimal solution for the new GTAP is also a skyline-optimal solution for the original GTAP.

Our algorithms use normalized TBoxes (Baader, Brandt, and Lutz 2005). These contain only axioms of the forms $A_1 \sqcap \ldots \sqcap A_n \sqsubseteq B$, $A \sqsubseteq \exists r.B$, and $\exists r.A \sqsubseteq B$ for $\mathcal{EL}$, and additionally for $\mathcal{EL}^{++}$, role inclusions of the forms $r \sqsubseteq s$ and $r_1 \circ r_2 \sqsubseteq s$ where $A, A_1, \ldots, A_n$ and B are atomic concepts and r, r_1 and r_2 are roles. To solve a GTAP for a single missing is-a relation $E \sqsubseteq F$, superconcepts of E are collected in a *Source* set and subconcepts of F are collected in a *Target* set. *Source* contains expressions of the forms A and $\exists r.A$ while *Target* contains expressions of the forms A, $A_1 \sqcap \ldots \sqcap A_n$ and $\exists r.A$ where A, A_1, …, A_n are atomic concepts and r is a role. Adding an is-a relation between an element in Source and an element in Target to the ontology would make $E \sqsubseteq F$ derivable. As we are interested in solutions containing is-a relations between atomic concepts, we check for every pair (A,B) ∈ Source × Target whether A and B are atomic concepts and $Or(A \sqsubseteq B) = true$. If so, then this is a possible solution. Further, if A is of the form $\exists r.N$ and B is of the form $\exists r.O$, then making $N \sqsubseteq O$ derivable would also make $A \sqsubseteq B$ derivable. In $\mathcal{EL}^{++}$ there are two more possibilities when A is of the form $\exists r.N$ and B is of the form $\exists s.O$. If T contains $r \sqsubseteq s$, then making $N \sqsubseteq O$ derivable would also make $A \sqsubseteq B$ derivable. Further, if T contains $r \circ r_1 \sqsubseteq s$ and $N \sqsubseteq \exists r_1.P$, then making $P \sqsubseteq O$ derivable would also make $A \sqsubseteq B$ derivable. $\mathcal{EL}$ TBoxes are always consistent, but this is not the case for $\mathcal{EL}^{++}$ TBoxes. This means that for $\mathcal{EL}^{++}$ we need to check consistency of possible solutions in each step of the algorithm. Further, we remove redundancy in the solutions while maintaining the same level of informativeness. It can be shown that the algorithms are sound. In Algorithm 1 we show the algorithm for $\mathcal{EL}^{++}$.[4]

Experiments

We have run experiments on an Intel Core i7-2620M Processor at 3.07 GHz with 4 GB RAM under Windows 7 Professional and Java 1.7 compiler. In all experiments the validation phase took the most time while the computations between iterations took less than 10 seconds. The results are summarized in Table 3. It shows the number of missing is-a relations in each iteration in the algorithm ('It' columns). These are repaired by adding itself, or by adding new knowledge to the ontology. When new relations are added we also indicate in parentheses how many were found using ∃.

In the first experiment we used the Biotop ontology (2013 OWL Reasoner Evaluation Workshop) with 280 concepts and 42 object properties. We randomly chose 47 is-a rela-

[4] In (Dragisic, Lambrix, and Wei-Kleiner 2014) we show the algorithm for $\mathcal{EL}$ and describe an implemented system.

```
Procedure RepairSingleIsa begin
    Input: $E \sqsubseteq F$, T, Or, C
    Output: Solution for GTAP (T, C, Or, $\{E \sqsubseteq F\}$)
    Sol := $\emptyset$;
    Source := find superconcepts of E;
    Target := find subconcepts of F;
    foreach $A \in Source$ do
        foreach $B \in Target$ do
            if $T \cup Sol \cup \{A \sqsubseteq B\}$ is consistent then
                if A and B are atomic concepts & $A \sqsubseteq B \in Or$ then
                    if there exists $K \sqsubseteq L \in Sol$ such that $T \models A \sqsubseteq K$
                    and $T \models L \sqsubseteq B$ then
                        do nothing;
                    else
                        remove every $K \sqsubseteq L \in$ Sol s.t. $T \models K \sqsubseteq A$
                        and $T \models B \sqsubseteq L$;
                        Sol := Sol $\cup$ $\{A \sqsubseteq B\}$;
                else if A is of the form $\exists r.N$ & B is of the form $\exists s.O$ then
                    Extra_Sols := FindExistsSolutions(T, r, N, s, O);
                    foreach $Rel \in Extra\_Sols$ do
                        Sol := Sol $\cup$ RepairSingleIsa(Rel, T, Or, C);
    return Sol;
Procedure RepairMultipleIsa begin
    Input: M, T, Or, C
    Output: Solution for GTAP (T, C, Or, M)
    foreach $E_i \sqsubseteq F_i \in M$ do
        SingleSol$_i$ := RepairSingleIsa($E_i \sqsubseteq F_i$, T, Or, C);
    Solution := $\bigcup_i$ SingleSol$_i$;
    if $T \cup$ Solution is inconsistent then
        return M;
    remove redundancy in Solution within same level of informativeness;
    return Solution;
Procedure Repair begin
    Input: M, T, Or, C
    Output: Solution for GTAP (T, C, Or, M)
    Missing := M;
    Solution := RepairMultipleIsa(Missing, T, Or, C);
    Final-Solution := Solution;
    while Solution $\neq$ Missing do
        Missing := Solution;
        Solution := RepairMultipleIsa(Missing, T $\cup$ Missing, Or, C);
        Final-Solution := Final-Solution $\cup$ Solution;
        remove redundancy in Final-Solution within same level of
        informativeness;
    return Final-Solution;
Procedure FindExistsSolutions begin
    Input: T, r, N, s, O
    Output: Set of is-a relations
    CandidateSols := $\emptyset$;
    Compositions := find all role inclusions of form $r \sqsubseteq s$ or $r \circ r_1 \sqsubseteq s$ in
    TBox T;
    foreach $Comp \in Compositions$ do
        if Comp is of form $r \sqsubseteq s$ then
            CandidateSols := CandidateSols $\cup$ $\{N \sqsubseteq O\}$;
        else
            Cs := $\{ P \mid T \models N \sqsubseteq \exists r1.P \}$;
            CandidateSols := CandidateSols $\cup$ $\{P \sqsubseteq O \mid P \in Cs\}$;
    return CandidateSols;
```

Algorithm 1: Algorithm for solving GTAP in $\mathcal{EL}^{++}$.

Biotop	It1	It2	It3	It4
Missing	47	41	42	41
Repaired by itself	19	31	38	41
Repaired using new knowledge	28	10	4	0
New relations	26(3)	11	3(1)	0

AMA	It1	It2	It3	
Missing	94	101	101	
Repaired by itself	57	98	101	
Repaired using new knowledge	37	3	0	
New relations	44	3	0	

NCI-A	It1	It2	It3	
Missing	58	55	54	
Repaired by itself	49	50	54	
Repaired using new knowledge	9	5	0	
New relations	6	4	0	

Table 3: Experiments results.

tions, and modified the ontology by removing is-a relations which would make the selected is-a relations derivable. The unmodified ontology was used as domain knowledge in the experiment. Further, we debugged the two ontologies from the 2013 OAEI Anatomy track, i.e. AMA containing 2744 concepts and NCI-A containing 3304 concepts. The input was a validated set of 94 and 58 missing is-a relations, respectively, for AMA and NCI-A.

The experiments have shown that our iterative approach is beneficial as in all our experiments additional relations were added to the ontology in subsequent iterations. New knowledge was often added to the ontologies. In this case, the added is-a relations could also be considered as missing is-a relations and used for further completing the structure. The first experiment also showed a way to complete the structure even when no missing is-a relations are available. This methodology also allows a domain expert to deal with existing is-a relations which the domain expert has identified as relations which need to be revised or investigated further. For a larger description and discusssion we refer to (Dragisic, Lambrix, and Wei-Kleiner 2014).

Related Work

There are works on abductive reasoning problems in (simple) description logics including concept abduction (Colucci et al. 2004; Bienvenu 2008; Donini et al. 2009) and ABox abduction (Du et al. 2011; Klarman, Endriss, and Schlobach 2011; Calvanese et al. 2012; 2011) as defined in (Elsenbroich, Kutz, and Sattler 2006).

There is not much work on the repairing of missing is-a structure. In (Lambrix and Liu 2013) this was addressed in the setting of taxonomies where the problem as well as some preference criteria were defined.

There is work that addresses *related topics* but not directly the problem that is addressed in this paper. There is much work on the *detection of missing (is-a) relations* in e.g. ontology learning (Cimiano, Buitelaar, and Magnini 2005), using linguistic (Hearst 1992) and logical (Corcho et al. 2009) patterns, or by using knowledge inherent in an ontology network (Lambrix, Liu, and Tan 2009; Ivanova et al. 2012). As mentioned before, these approaches, in general, do not detect all missing is-a relations. There is also much work on a dual problem to the one addressed in this paper, i.e. the *debugging of semantic defects*. Most of the work on debugging semantic defects aims at identifying and removing logical contradictions from an ontology (Haase and Stojanovic 2005; Schlobach 2005; Kalyanpur et al. 2006b; 2006a; Flouris et al. 2008), from mappings between ontologies (Meilicke, Stuckenschmidt, and Tamilin 2007; Wang and Xu 2008; Ji et al. 2009; Qi, Ji, and Haase 2009) or ontologies in a network (Jimenez-Ruiz et al. 2009; Ivanova et al. 2012). The work in (Lambrix and Ivanova 2013; Ivanova and Lambrix 2013) deals with debugging both missing and wrong is-a structure and mappings for the case of taxonomies in a network.

Conclusions and Future Work

We have studied the GTAP in the context of ontology repairing. We first defined a model of GTAP and extended it with various preferences. Then we presented complexity results on the existence, relevance and necessity decision problems for ontologies that can be represented as TBoxes using a member of the $\mathcal{EL}$ family. Unless the polynomial hierarchy collapses, GTAP is much harder than the classical deduction problem, which is tractable for $\mathcal{EL}^{++}$. Further, we provided an algorithm and system for finding skyline optimal solutions to the GTAP and showed its usefulness through experiments.

In the future, we are interested in studying the GTAP for other knowledge representation languages. Further, we will investigate variants of the GTAP with different preference relations and restrictions of the signature. Another interesting topic is to study the GTAP in the context of modular ontologies where it may not be possible to introduce changes in the imported ontologies. Further, we will look into the integration of different abduction frameworks to deal with both modeling and semantic defects.

Acknowledgements

We thank the Swedish Research Council (Vetenskapsrådet), the Swedish e-Science Research Centre (SeRC) and the Swedish National Graduate School in Computer Science (CUGS) for financial support.

References

Ashburner, M.; Ball, C.; Blake, J.; Botstein, D.; Butler, H.; Cherry, J.; Davis, A.; Dolinski, K.; Dwight, S.; Eppig, J.; Harris, M.; Hill, D.; Issel-Tarver, L.; Kasarskis, A.; Lewis, S.; Matese, J.; Richardson, J.; Ringwald, M.; Rubin, G.; and Sherlock, G. 2000. Gene Ontology: Tool for the Unification of Biology. *Nature Genetics* 25(1):25–29.

Baader, F.; Brandt, S.; and Lutz, C. 2005. Pushing the $\mathcal{EL}$ envelope. In *19th International Joint Conference on Artificial Intelligence*, 364–369.

Bienvenu, M. 2008. Complexity of abduction in the $\mathcal{EL}$ family of lightweight description logics. In *11th International Conference on Principles of Knowledge Representation and Reasoning*, 220–230.

Calvanese, D.; Ortiz, M.; Simkus, M.; and Stefanoni, G. 2011. The complexity of conjunctive query abduction in DL-lite. In *International Workshop on Description Logics*, 81–91.

Calvanese, D.; Ortiz, M.; Simkus, M.; and Stefanoni, G. 2012. The complexity of explaining negative query answers in DL-Lite. In *13th International Conference on Principles of Knowledge Representation and Reasoning*, 583–587.

Cimiano, P.; Buitelaar, P.; and Magnini, B. 2005. *Ontology Learning from Text: Methods, Evaluation and Applications*. IOS Press.

Colucci, S.; Di Noia, T.; Di Sciascio, E.; Donini, F.; and Mongiello, M. 2004. A uniform tableaux-based approach to concept abduction and contraction in $\mathcal{ALN}$. In *International Workshop on Description Logics*, 158–167.

Corcho, O.; Roussey, C.; Vilches, L. M.; and Pérez, I. 2009. Pattern-based OWL ontology debugging guidelines. In *Workshop on Ontology Patterns*, 68–82.

Donini, F.; Colucci, S.; Di Noia, T.; and Di Sciasco, E. 2009. A tableaux-based method for computing least common subsumers for expressive description logics. In *21st International Joint Conference on Artificial Intelligence*, 739–745.

Dragisic, Z.; Lambrix, P.; and Wei-Kleiner, F. 2014. Completing the is-a structure of biomedical ontologies. In *10th International Conference on Data Integration in the Life Sciences*.

Du, J.; Qi, G.; Shen, Y.-D.; and Pan, J. 2011. Towards practical Abox abduction in large OWL DL ontologies. In *25th AAAI Conference on Artificial Intelligence*, 1160–1165.

Eiter, T., and Gottlob, G. 1995. The complexity of logic-based abduction. *Journal of the ACM* 42(1):3–42.

Elsenbroich, C.; Kutz, O.; and Sattler, U. 2006. A case for abductive reasoning over ontologies. In *OWL: Experiences and Directions*.

Flouris, G.; Manakanatas, D.; Kondylakis, H.; Plexousakis, D.; and Antoniou, G. 2008. Ontology Change: Classification and Survey. *Knowledge Engineering Review* 23(2):117–152.

Haase, P., and Stojanovic, L. 2005. Consistent Evolution of OWL Ontologies. In *2nd European Semantic Web Conference*, 182–197.

Hearst, M. 1992. Automatic acquisition of hyponyms from large text corpora. In *14th International Conference on Computational Linguistics*, 539–545.

Ivanova, V., and Lambrix, P. 2013. A unified approach for aligning taxonomies and debugging taxonomies and their alignments. In *10th Extended Semantic Web Conference*, 1–15.

Ivanova, V.; Laurila Bergman, J.; Hammerling, U.; and Lambrix, P. 2012. Debugging taxonomies and their alignments: the ToxOntology - MeSH use case. In *1st International Workshop on Debugging Ontologies and Ontology Mappings*, 25–36.

Ji, Q.; Haase, P.; Qi, G.; Hitzler, P.; and Stadtmuller, S. 2009. RaDON - repair and diagnosis in ontology networks. In *6th European Semantic Web Conference*, 863–867.

Jimenez-Ruiz, E.; Grau, B. C.; Horrocks, I.; and Berlanga, R. 2009. Ontology Integration Using Mappings: Towards Getting the Right Logical Consequences. In *6th European Semantic Web Conference*, 173–187.

Kakas, A. C., and Mancarella, P. 1990. Database updates through abduction. In *16th International Conference on Very Large Data Bases*, 650–661.

Kalyanpur, A.; Parsia, B.; Sirin, E.; and Cuenca-Grau, B. 2006a. Repairing Unsatisfiable Concepts in OWL Ontologies. In *3rd European Semantic Web Conference*, 170–184.

Kalyanpur, A.; Parsia, B.; Sirin, E.; and Hendler, J. 2006b. Debugging Unsatisfiable Classes in OWL Ontologies. *Journal of Web Semantics* 3(4):268–293.

Klarman, S.; Endriss, U.; and Schlobach, S. 2011. Abox abduction in the description logic $\mathcal{ALC}$. *Journal of Automated Reasoning* 46:43–80.

Lambrix, P., and Ivanova, V. 2013. A unified approach for debugging is-a structure and mappings in networked taxonomies. *Journal of Biomedical Semantics* 4:10.

Lambrix, P., and Liu, Q. 2013. Debugging the missing is-a structure within taxonomies networked by partial reference alignments. *Data & Knowledge Engineering* 86:179–205.

Lambrix, P.; Liu, Q.; and Tan, H. 2009. Repairing the Missing is-a Structure of Ontologies. In *4th Asian Semantic Web Conference*, 76–90.

Meilicke, C.; Stuckenschmidt, H.; and Tamilin, A. 2007. Repairing Ontology Mappings. In *22th National Conference on Artificial Intelligence*, 1408–1413.

Qi, G.; Ji, Q.; and Haase, P. 2009. A Conflict-Based Operator for Mapping Revision. In *8th International Semantic Web Conference*, 521–536.

Schlobach, S. 2005. Debugging and Semantic Clarification by Pinpointing. In *2nd European Semantic Web Conference*, 226–240.

Wang, P., and Xu, B. 2008. Debugging ontology mappings: a static approach. *Computing and Informatics* 27:21–36.

Wei-Kleiner, F.; Dragisic, Z.; and Lambrix, P. 2014. Extended version of this paper. http://www.ida.liu.se/~patla/publications/AAAI14-extended.pdf.