Approximating Model-Based ABox Revision in DL-Lite: Theory and Practice

Guilin Qi[1,2] **Zhe Wang**[3] **Kewen Wang**[3] **Xuefeng Fu** **Zhiqiang Zhuang**[3]
[1] School of Computer Science and Engineering, Southeast University, China
[2] State Key Lab for Novel Software Technology, Nanjing University, Nanjing, China
[3] School of Information and Communication Technology, Griffith University, Australia

Abstract

Model-based approaches provide a semantically well justified way to revise ontologies. However, in general, model-based revision operators are limited due to lack of efficient algorithms and inexpressibility of the revision results. In this paper, we make both theoretical and practical contribution to efficient computation of model-based revisions in DL-Lite. Specifically, we show that maximal approximations of two well-known model-based revisions for DL-Lite$_{\mathcal{R}}$ can be computed using a syntactic algorithm. However, such a coincidence of model-based and syntactic approaches does not hold when role functionality axioms are allowed. As a result, we identify conditions that guarantee such a coincidence for DL-Lite$_{\mathcal{FR}}$. Our result shows that both model-based and syntactic revisions can co-exist seamlessly and the advantages of both approaches can be taken in one revision operator. Based on our theoretical results, we develop a graph-based algorithm for the revision operators and thus graph database techniques can be used to compute ontology revisions. Preliminary evaluation results show that the graph-based algorithm can efficiently handle revision of practical ontologies with large data.

Introduction

The latest version of OWL (Ontology Web Language) recommended by W3C is OWL 2[1], which has three profiles with fine-tuned expressive power to support tractable reasoning. OWL 2 QL, one of the three profiles, is designed for ontology-based data access. The logic that underpins OWL 2 QL is DL-Lite, which is a family of tractable description logics[2]. With the development of the Semantic Web, more and more data are published as linked data, and the data is often accompanied with lightweight ontologies[3] which provide extended vocabularies and logical constraints for the data. With new data published and incorporated into the existing data, one typical problem is how to deal with logical inconsistencies caused by the violation of the constraints posed by the ontologies. This problem can be formalized as the problem of ABox revision in description logics, which deals with the removal of assertions in the old ABox to accommodate the new ABox and to resolve inconsistencies, under the assumption that the new ABox is more reliable.

Recently, there has been an increasing interest in ontology revision in DL-Lite. Some model-based revision operators in DL-Lite are proposed (Qi and Du 2009; Kharlamov and Zheleznyakov 2011), which select models of newly received ontology that are 'closest' to the existing ontology. However, model-based operators for ontology revision often suffer from two drawbacks: the inexpressibility problem, i.e., in general, the result of revision cannot be expressed in the same DL, and the computation problem, i.e., the computation of revision is inefficient except for some special cases. In (Calvanese et al. 2010), the authors propose a syntactic algorithm for ABox revision in DL-Lite$_{\mathcal{FR}}$ which runs in polynomial time. It has been shown in (Kharlamov, Zheleznyakov, and Calvanese 2013) that for a fragment of DL-Lite$_{core}$, the core language of DL-Lite, this algorithm outputs the result of two model-based revision operators. However, this result is only shown to hold for the restricted language. An open problem is whether the equivalence of model-based and syntactic revisions still holds in more expressive languages, such as DL-Lite$_{\mathcal{FR}}$, and whether similar result holds for other model-based revision operators.

In this paper, we present the first theoretical work on approximation of two well-known model-based revision operators in DL-Lite$_{\mathcal{FR}}$, as well as a practical graph-based algorithm and some experimental results. We first show that the result of the syntactic revision algorithm given in (Kharlamov, Zheleznyakov, and Calvanese 2013) can be used to approximate two model-based revision operators in DL-Lite$_{\mathcal{R}}$. While both revision operators suffer from the inexpressibility problem, their approximations can be computed efficiently. We then give a counterexample to illustrate that our results on approximation breaks down when role functionality axioms are included. In order to accommodate role functionality axioms, we propose to either disallow the appearance of some role names or modify the syntactic algorithm by removing some axioms. On the practical aspect, we propose a way of implementing the algorithm by transforming a DL-Lite$_{\mathcal{FR}}$ ontology to a graph, and making use of graph database techniques to compute ontology revisions.

[1] http://www.w3.org/TR/owl2-profiles/

[2] http://dl.kr.org/

[3] Here we take ontologies as schema of the data.

Preliminary evaluation results show that the graph-based revision algorithm is able to deal with practical ontologies with large data efficiently.

Preliminaries

In this section, we first briefly recall some basics of DL-Lite and then introduce two model-based revision operators.

DL-Lite

We start with the introduction of DL-Lite$_{core}$, the core language for the DL-Lite family. The complex concepts and roles of DL-Lite$_{core}$ are defined as follows: (1) $B ::= A \mid \exists R$, (2) $R ::= P \mid P^{-}$, (3) $C ::= B \mid \neg B$, where A denotes an atomic concept, P an atomic role, B a basic concept, and C a general concept.

In DL-Lite$_{core}$, an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$, where $\mathcal{T}$ is a finite set of *concept inclusion assertions* of the form: $B \sqsubseteq C$; and $\mathcal{A}$ is a finite set of *membership assertions* of the form: $A(a)$, $P(a, b)$. DL-Lite$_{\mathcal{FR}}$ extends DL-Lite$_{core}$ with inclusion assertions between roles of the form $R \sqsubseteq E$ where E is a role or its inverse and functionality on roles or on their inverses of the form ($\texttt{Func}\ R$). To keep the logic tractable, whenever a role inclusion $R_1 \sqsubseteq R_2$ logically follows from $\mathcal{T}$, neither ($\texttt{Func}\ R_2$) nor ($\texttt{Func}\ R_2^{-}$) can appear in it. We call assertions of the form $B_1 \sqsubseteq \neg B_2$ as negative inclusions (NIs). We use $\mathrm{adom}(\mathcal{O})$ to denote the set of all constants occurring in $\mathcal{O}$.

The semantics of DL-Lite is defined in a standard way. Following the work given in (Calvanese et al. 2010), we assume that all interpretations are defined over the same infinite countable domain Δ. Given an interpretation $\mathcal{I}$ and an assertion α, $\mathcal{I} \vDash \alpha$ denotes that $\mathcal{I}$ is a *model* of α. An interpretation is called a *model* of an ontology $\mathcal{O}$, iff it is a model for each assertion in $\mathcal{O}$. We use $\mathrm{Mod}(\mathcal{O})$ to denote the set of all models of $\mathcal{O}$. An ontology is satisfiable if it has at least one model. An ontology $\mathcal{O}$ logically implies an assertion α, written $\mathcal{O} \models \alpha$, if all models of $\mathcal{O}$ are also models of α. The deductive closure of an ABox $\mathcal{A}$, denoted $cl_{\mathcal{T}}(\mathcal{A})$, is the set of all ABox assertions α such that $\mathcal{T} \cup \mathcal{A} \models \alpha$.

Model-based revision operators in DL-Lite

Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ and a new ABox $\mathcal{N}$, suppose $\mathcal{O}$ is consistent and $\mathcal{T} \cup \mathcal{N}$ is consistent. The problem of ABox revision is, how to modify $\mathcal{A}$ (by deletion or insertion of assertions) such that $\mathcal{T} \cup \mathcal{N}$ is consistent with the modified ABox? We reformulate the definition of model-based revision operators given in (Calvanese et al. 2010; Kharlamov, Zheleznyakov, and Calvanese 2013). Since we have assumed that all the interpretations are defined over the same domain Δ, we can consider an interpretation as a set of atoms. Given two interpretations $\mathcal{I}$ and $\mathcal{J}$, the set of atoms that are assigned different truth values is denoted as $\mathrm{diff}(\mathcal{I}, \mathcal{J}) = \mathcal{I} \ominus \mathcal{J}$, where $\mathcal{I} \ominus \mathcal{J} = (\mathcal{I} \setminus \mathcal{J}) \cup (\mathcal{J} \setminus \mathcal{I})$ is the symmetric difference between $\mathcal{I}$ and $\mathcal{J}$. $\mathrm{diff}(\mathcal{I}, \mathcal{J})$ can be extended to two ontologies $\mathcal{O}$ and $\mathcal{O}'$ as $\mathrm{diff}(\mathcal{O}, \mathcal{O}') = \{\mathrm{diff}(\mathcal{I}, \mathcal{J}) | \mathcal{I} \models \mathcal{O},\ \mathcal{J} \models \mathcal{O}'\}$.

We introduce two well-known distance functions given in the literature (see (Katsuno and Mendelzon 1992) and (Kharlamov, Zheleznyakov, and Calvanese 2013)).

$\mathrm{dist}_{\subseteq}(\mathcal{I}, \mathcal{J}) = \mathrm{diff}(\mathcal{I}, \mathcal{J})$ and $\mathrm{dist}_{\#}(\mathcal{I}, \mathcal{J}) = |\mathrm{diff}(\mathcal{I}, \mathcal{J})|$.

Furthermore, let $\mathrm{dist}_{\subseteq}(\mathcal{O}, \mathcal{O}')$ be the set of minimal elements in $\mathrm{diff}(\mathcal{O}, \mathcal{O}')$ w.r.t. the set inclusion and $\mathrm{dist}_{\#}(\mathcal{O}, \mathcal{O}') = min_{\mathcal{I} \models \mathcal{O}, \mathcal{J} \models \mathcal{O}'} \mathrm{dist}_{\#}(\mathcal{I}, \mathcal{J})$.

Based on these two distance functions, we can define two revision operators in a model-theoretic way as follows.

Definition 1. *Given an ontology $\mathcal{O}$ and an ABox $\mathcal{N}$, define*
$\mathrm{Mod}(\mathcal{O} \circ_{\subseteq} \mathcal{N}) = \{\mathcal{J} \in \mathrm{Mod}(\langle \mathcal{T}, \mathcal{N} \rangle) \mid$ *there exists* $\mathcal{I} \in \mathrm{Mod}(\mathcal{O})$ *s.t.* $\mathrm{dist}_{\subseteq}(\mathcal{I}, \mathcal{J}) \in \mathrm{dist}_{\subseteq}(\mathcal{O}, \langle \mathcal{T}, \mathcal{N} \rangle)\}$.
$\mathrm{Mod}(\mathcal{O} \circ_{\#} \mathcal{N}) = \{\mathcal{J} \in \mathrm{Mod}(\langle \mathcal{T}, \mathcal{N} \rangle) \mid$ *there exists* $\mathcal{I} \in \mathrm{Mod}(\mathcal{O})$ *s.t.* $\mathrm{dist}_{\#}(\mathcal{I}, \mathcal{J}) = \mathrm{dist}_{\#}(\mathcal{O}, \langle \mathcal{T}, \mathcal{N} \rangle)\}$.

In DL-Lite, for any set $\mathcal{M}$ of interpretations, we may not be able to find an ontology whose models are exactly those interpretations in $\mathcal{M}$. Therefore, the notion of sound approximation is defined in (Kharlamov, Zheleznyakov, and Calvanese 2013).

Definition 2. *(Kharlamov, Zheleznyakov, and Calvanese 2013) Let $\mathcal{M}$ be a set of models and $\mathcal{D}$ a DL. We say that a $\mathcal{D}$ ontology $\mathcal{O}$ is a* sound $\mathcal{D}$-approximation *of $\mathcal{M}$ if* $\mathcal{M} \subseteq \mathrm{Mod}(\mathcal{O})$. *Furthermore, such an ontology $\mathcal{O}$ is a* maximal sound $\mathcal{D}$-approximation *if for every* sound $\mathcal{D}$-approximation $\mathcal{O}'$ *of $\mathcal{M}$ it holds that* $\mathrm{Mod}(\mathcal{O}') \not\subset \mathrm{Mod}(\mathcal{O})$.

Approximating Model-based ABox Revision in DL-Lite$_{\mathcal{R}}$

In this section we show that the approximations of two model-based revision operators $\circ_{\subseteq}$ and $\circ_{\#}$ coincide with the syntactic revision operator introduced in (Kharlamov, Zheleznyakov, and Calvanese 2013) for DL-Lite$_{\mathcal{R}}$. Given a DL-Lite$_{\mathcal{FR}}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ and ABox $\mathcal{N}$, an algorithm, denoted as AtAlg, is given in (Kharlamov, Zheleznyakov, and Calvanese 2013) (see Algorithm 1). The algorithm AtAlg determines a *syntactic ABox revision* operator defined as $\mathcal{O} \circ_{MCS} \mathcal{N} = \mathcal{T} \cup \mathcal{N} \cup \mathrm{AtAlg}(\mathcal{O}, \mathcal{N})$.

```
Algorithm 1: $\mathrm{AtAlg}(\mathcal{O},\mathcal{N})$)
Data: DL-Lite$_{FR}$ ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ and ABox $\mathcal{N}$
Result: A set of ABox assertions
begin
    $\mathcal{A}' = \emptyset$; $X := cl_{\mathcal{T}}(\mathcal{A})$;
    repeat
        choose some $g \in X$; $X$:= $X \setminus \{g\}$;
        if $\{g\} \cup \mathcal{N} \not\models_{\mathcal{T}} \bot$ then
            $\mathcal{A}' := \mathcal{A}' \cup \{g\}$;
    until $X$=$\emptyset$ ;
    return $\mathcal{A}'$;
end
```

The following theorem shows that $\mathcal{O} \circ_{MCS} \mathcal{N}$ is a maximal sound approximation of $\mathcal{O} \circ \mathcal{N}$ in DL-Lite$_{\mathcal{R}}$, for $\circ = \circ_{\#}$ and $\circ = \circ_{\subseteq}$.

Theorem 1. *Let $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an ontology in DL-Lite$_{\mathcal{R}}$ and $\mathcal{N}$ a new ABox. $\mathcal{O} \circ_{MCS} \mathcal{N}$ is a maximal sound DL-Lite$_{\mathcal{R}}$-approximation of both $\mathcal{O} \circ_{\subseteq} \mathcal{N}$ and $\mathcal{O} \circ_{\#} \mathcal{N}$.*

Proof. (Sketch) Let $M_{\subseteq} = \mathrm{Mod}(\mathcal{O} \circ_{\subseteq} \mathcal{N})$ and $M_{MCS} = \mathrm{Mod}(\mathcal{O} \circ_{MCS} \mathcal{N})$. For any $\mathcal{J} \in M_{\subseteq}$, clearly $\mathcal{J} \models \mathcal{T} \cup \mathcal{N}$. We only need to show that $\mathcal{J} \models \mathrm{AtAlg}(\mathcal{O}, \mathcal{N})$. We show this by absurdity. Suppose there exists $f^* \in \mathrm{AtAlg}(\mathcal{O}, \mathcal{N})$ such that $\mathcal{J} \not\models f^*$. Since $\mathcal{J} \in M_{\subseteq}$, there exists $\mathcal{I} \models \mathcal{O}$ (thus $\mathcal{I} \models f^*$) such that $\mathrm{dist}_{\subseteq}(\mathcal{I}, \mathcal{J})$ is minimal among all pairs of models of $\mathcal{T} \cup \mathcal{N}$ and $\mathcal{O}$. We construct a model $\mathcal{I}'$ of $\mathcal{O}$ and a model $\mathcal{J}'$ of $\mathcal{T} \cup \mathcal{N}$ such that $\mathrm{diff}_{\subseteq}(\mathcal{I}', \mathcal{J}') \subset \mathrm{diff}_{\subseteq}(\mathcal{I}, \mathcal{J})$. In DL-Lite, a standard method to construct a model is to use the notion of a chase. If we take $\mathcal{I}' = \mathit{chase}_{\mathcal{T}}(\mathcal{A})$ and $\mathcal{J}' = \mathit{chase}_{\mathcal{T}}(\mathcal{N})$, where $\mathit{chase}_{\mathcal{T}}(\mathcal{A})$ is the chase of $\mathcal{A}$ w.r.t. $\mathcal{T}$, then we may not have $\mathrm{diff}_{\subseteq}(\mathcal{I}', \mathcal{J}') \subset \mathrm{diff}_{\subseteq}(\mathcal{I}, \mathcal{J})$. Let us look at an example: let $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, $\mathcal{T} = \{A \sqsubseteq \exists R, B \sqsubseteq \neg \exists R^-\}$ and $\mathcal{A} = \{A(c), R(c,d)\}$, and an ABox $\mathcal{N} = \{B(d)\}$. Clearly, $\mathit{chase}_{\mathcal{T}}(\mathcal{A}) = \mathcal{A}$ and $\mathit{chase}_{\mathcal{T}}(\mathcal{N}) = \mathcal{N}$. However, we can find $\mathcal{I} = \{A(c), R(c,d), R(c,e)\}$ and $\mathcal{J} = \{B(d), A(c), R(c,e)\}$ and we can check that $\mathrm{diff}_{\subseteq}(\mathcal{I}, \mathcal{J})$ is minimal w.r.t. the set inclusion. That is, we should add $R(c,e)$ to the model of $\mathcal{A} \cup \mathcal{T}$ and we should add $A(c)$ and $R(c,e)$ to the model of $\mathcal{N} \cup \mathcal{T}$. Based on this intuition, we propose the following method to construct $\mathcal{I}'$ and $\mathcal{J}'$. We first take $\mathcal{I}_1 = \mathit{chase}_{\mathcal{T}}(\mathcal{A})$ and $\mathcal{J}_1 = \mathit{chase}_{\mathcal{T}}(\mathcal{N})$, we then update $\mathcal{J}_1$ using $\mathcal{I}_1$ by adding those $A(c)$ and $P(c,d)$ in $\mathcal{I}_1$ that are not in conflict with $\mathcal{N}$ w.r.t. $\mathcal{T}$ to $\mathcal{J}_1$. Suppose we get $\mathcal{J}_2$. We then update $\mathcal{I}_1$ by $\mathcal{J}_2$, and so on, until we get a fixed point. We can check that $\mathrm{diff}_{\subseteq}(\mathcal{I}', \mathcal{J}') \subset \mathrm{diff}_{\subseteq}(\mathcal{I}, \mathcal{J})$, which is a contradiction. □

One may wonder if $\mathcal{O} \circ_{MCS} \mathcal{N}$ is the syntactic counterpart of $\mathrm{Mod}(\mathcal{O} \circ \mathcal{N})$, for $\circ = \circ_{\subseteq}$ or $\circ = \circ_{\#}$. The following example shows that this is not the case.

Example 1. *Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T} = \{A \sqsubseteq \neg \exists P\}$ and $\mathcal{A} = \{A(c)\}$, and an ABox $\mathcal{N} = \{\exists P(c)\}$. Then $cl_{\mathcal{T}}(\mathcal{A}) = \{A(c)\}$ and $\mathrm{AtAlg}(\mathcal{O}, \mathcal{N}) = \emptyset$. So $\mathcal{O} \circ_{MCS} \mathcal{N} = \mathcal{T} \cup \mathcal{N}$. It is easy to check that $\mathcal{J} = \{P(c,d), P(c,e)\}$ and $\mathcal{J}' = \{P(c,d)\}$ are two models of $\mathcal{T} \cup \mathcal{N}$. Thus they are both models of $\mathcal{O} \circ_{MCS} \mathcal{N}$. The chase of $\mathcal{A}$ w.r.t. $\mathcal{T}$ is $\mathcal{I} = \{A(c)\}$. We have $\mathrm{diff}(\mathcal{I}, \mathcal{J}') \subset \mathrm{diff}(\mathcal{I}, \mathcal{J})$ (resp. $|\mathrm{diff}(\mathcal{I}, \mathcal{J}')| < |\mathrm{diff}(\mathcal{I}, \mathcal{J})|$). Since $\mathcal{J}$ or $\mathcal{J}'$ cannot be expanded with $A(c)$ and any model of $\mathcal{T} \cup \mathcal{A}$ must contain $A(c)$ and can be expanded with neither $P(c,d)$ nor $P(c,e)$, $\mathrm{diff}(\mathcal{I}_i, \mathcal{J}') \subset \mathrm{diff}(\mathcal{I}_i, \mathcal{J})$ (resp. $|\mathrm{diff}(\mathcal{I}_i, \mathcal{J}')| < |\mathrm{diff}(\mathcal{I}_i, \mathcal{J})|$) for any model $\mathcal{I}_i$ of $\mathcal{T} \cup \mathcal{A}$. Thus, $\mathcal{J}$ cannot be a model of $\mathcal{O} \circ \mathcal{N}$, for $\circ = \circ_{\subseteq}$ or $\circ = \circ_{\#}$.*

According to Theorem 1 and Example 1, we can show that operators $\circ_{\subseteq}$ and $\circ_{\#}$ suffer from the problem of inexpressibility, i.e., $\mathrm{Mod}(\mathcal{O} \circ \mathcal{N})$ is not axiomatizable in DL-Lite$_{\mathcal{R}}$.

Approximating model-based revision in DL-Lite$_{\mathcal{FR}}$

We consider the question if Theorem 1 still holds when functionality axioms are included. Unfortunately, the following example gives a negative answer to this question.

Example 2. *Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T} = \{A \sqsubseteq \exists R, B \sqsubseteq \neg \exists R^-, (\texttt{Func } R)\}$ and $\mathcal{A} = \{A(c), B(e), R(c,d)\}$, and an ABox $\mathcal{N} = \{B(d)\}$. Then $cl_{\mathcal{T}}(\mathcal{A}) = \{A(c), B(e), R(c,d), \exists R(c)\}$ and $\mathrm{AtAlg}(\mathcal{O}, \mathcal{N}) = \{A(c), B(e), \exists R(c)\}$. So $\mathcal{O} \circ_{MCS} \mathcal{N} = \mathcal{T} \cup \{A(c), B(d), B(e), \exists R(c)\}$. Consider the following four interpretations:*

$\mathcal{I} = \{A(c), R(c,d), B(e)\}$;
$\mathcal{J}_1 = \{(B(d), B(e), A(c), R(c,f)\}$;
$\mathcal{J}_2 = \{B(d), B(e)\}$;
$\mathcal{J}_3 = \{B(d), A(c), R(c,e)\}$.

It is easy to check that $\mathcal{I}$ is a model of $\mathcal{T} \cup \mathcal{A}$ and $\mathcal{J}_i$ ($i = 1, 2, 3$) are models of $\mathcal{T} \cup \mathcal{N}$. We have $\mathrm{diff}(\mathcal{I}, \mathcal{J}_1) = \{B(d), R(c,d), R(c,f)\}$, $\mathrm{diff}(\mathcal{I}, \mathcal{J}_2) = \{A(c), B(d), R(c,d)\}$ and $\mathrm{diff}(\mathcal{I}, \mathcal{J}_3) = \{B(d), B(e), R(c,d), R(c,e)\}$. Thus, $\mathrm{dist}_{\#}(\mathcal{I}, \mathcal{J}_1) = \mathrm{dist}_{\#}(\mathcal{I}, \mathcal{J}_2) = 3$. Any model of $\mathcal{T} \cup \mathcal{A}$ must contain $A(c)$, $R(c,d)$ and $B(e)$ and any model of $\mathcal{T} \cup \mathcal{N}$ must contain $B(d)$. Since $R(c,d)$ and $B(d)$ are in conflict w.r.t. $\mathcal{T}$, any model $\mathcal{I}$ of $\mathcal{T} \cup \mathcal{A}$ and any model $\mathcal{J}$ of $\mathcal{T} \cup \mathcal{N}$ will not contain both of them. For any such $\mathcal{J}$, suppose it contains $A(c)$, then it will contain $R(c, y_{new})$ where y_{new} is a fresh individual. Since $(\texttt{Func } R) \in \mathcal{T}$, $R(c, y_{new})$ cannot exist in any model $\mathcal{I}$ of $\mathcal{T} \cup \mathcal{A}$. Thus, $\mathrm{diff}(\mathcal{I}, \mathcal{J})$ must contain $R(c,d)$, $B(d)$ and $R(c, y_{new})$. If $\mathcal{J}$ does not contain $A(c)$, then $\mathrm{diff}(\mathcal{I}, \mathcal{J})$ must contain $R(c,d)$, $B(d)$ and $A(c)$. Therefore, $\mathcal{J}_2$ and $\mathcal{J}_3$ are models of $\mathcal{O} \circ_{\#} \mathcal{N}$. Clearly, $\mathcal{J}_3$ is not a model of $\mathrm{AtAlg}(\mathcal{O}, \mathcal{N})$. So it is not a model of $\mathcal{O} \circ_{MCS} \mathcal{N}$. Since $\mathcal{J}_2$ is not a model of $A(c)$, $A(c)$ cannot be inferred from $\mathcal{O} \circ_{\#} \mathcal{N}$. Similarly, we can show that $\mathcal{J}_1$, $\mathcal{J}_2$ and $\mathcal{J}_3$ are models of $\mathcal{O} \circ_{\subseteq} \mathcal{N}$. Since $\mathcal{J}_1$ is not a model of $B(e)$ and $\mathcal{J}_3$ is not a model of $A(c)$, neither $A(c)$ nor $B(e)$ can be inferred from $\mathcal{O} \circ_{\subseteq} \mathcal{N}$. Thus, Theorem 1 does not hold for DL-Lite$_{\mathcal{FR}}$.

To see why adding functionalities on roles or their inverses causes the inexpressibility problem, we analyze Example 2 again. Since $(\texttt{Func } R)$ exists, when applying the model construction method given in the proof of Theorem 1, we cannot update models of $\mathcal{O}$ and $\mathcal{T} \cup \mathcal{N}$ successfully because of the functionality constraints. For example, when constructing the chase of $\mathcal{A}$ w.r.t. $\mathcal{T}$, we need to add $R(c, x_{new})$ to the chase of $\mathcal{N}$ w.r.t. $\mathcal{T}$. However, since $(\texttt{Func } R)$) exists, we may not be able to do that. Based on this analysis, we define a notion called *triggered roles*, which are role names that causes the inexpressibility problem.

Definition 3. *(Set of triggered roles). Let $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an ontology in DL-Lite$_{\mathcal{FR}}$ and $\mathcal{N}$ a new ABox. The set of* triggered roles *in $\mathcal{O}$, denoted as $\mathrm{TR}[\mathcal{O}, \mathcal{N}]$, is the set of all roles that satisfy one of the following conditions:*

- *Condition 1: (1) $(\texttt{Func } P) \in \mathcal{T}$, (2) $P(c,d) \in cl_{\mathcal{T}}(\mathcal{A})$, and $\mathcal{A} \setminus \{P(c,d)\} \models_{\mathcal{T}} \exists P(c)$, and (3) $\mathcal{N} \cup \{\exists P(c)\} \not\models_{\mathcal{T}} \bot$ and $\mathcal{N} \cup \{\exists P^-(d)\} \models_{\mathcal{T}} \bot$, where $c, d \in \mathrm{adom}(\mathcal{O})$.*
- *Condition 2: (1) $(\texttt{Func } P) \in \mathcal{T}$, (2) $P(c,d) \in cl_{\mathcal{T}}(\mathcal{A})$, and $\mathcal{A} \setminus \{P(c,d)\} \models_{\mathcal{T}} \exists P^-(d)$, and (3) $\mathcal{N} \cup \{\exists P^-(d)\} \not\models_{\mathcal{T}} \bot$ and $\mathcal{N} \cup \{\exists P(c)\} \models_{\mathcal{T}} \bot$, where $c, d \in \mathrm{adom}(\mathcal{O})$.*

We show that if the set of triggered roles is empty, then $\mathcal{O} \circ_{MCS} \mathcal{N}$ is an approximation of $\mathcal{O} \circ \mathcal{N}$, for $\circ = \circ_{\subseteq}$ or $\circ = \circ_{\#}$.

Theorem 2. *Let $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an ontology in DL-Lite$_{\mathcal{FR}}$ and $\mathcal{N}$ a new ABox. If $\mathrm{TR}[\mathcal{O},\mathcal{N}] = \emptyset$, then we have $\mathrm{Mod}(\mathcal{O} \circ \mathcal{N}) \subseteq \mathrm{Mod}(\mathcal{O} \circ_{MCS} \mathcal{N})$, for $\circ = \circ_{\subseteq}$ or $\circ = \circ_{\#}$. Otherwise, we cannot have $\mathrm{Mod}(\mathcal{O} \circ \mathcal{N}) \subseteq \mathrm{Mod}(\mathcal{O} \circ_{MCS} \mathcal{N})$, for $\circ = \circ_{\subseteq}$ or $\circ = \circ_{\#}$.*

If the set of triggered roles is not empty, then we can modify $\mathcal{O} \circ_{MCS} \mathcal{N}$ such that it is an approximation of $\mathcal{O} \circ \mathcal{N}$, for $\circ = \circ_{\subseteq}$ or $\circ = \circ_{\#}$. Consider Example 2, $\mathcal{J}_2$ is a model of $\mathcal{O} \circ \mathcal{N}$, but it may not be a model of $A(c)$ and $R(c, d)$, where R is a triggered role. However, it is easy to check that $\mathcal{J}_2$ is a model of $\mathrm{AtAlg}(\mathcal{O},\mathcal{N}) \setminus \{A(c), R(c, d)\}$. This inspired us to give a method to compute the maximal approximation of $\mathcal{O} \circ \mathcal{N}$, for $\circ = \circ_{\subseteq}$ or $\circ = \circ_{\#}$. The idea is to remove $\exists P(c)$ and those assertions that can infer it from $\mathrm{AtAlg}(\mathcal{O},\mathcal{N})$ for each triggered role P. We need a lemma that generalizes Proposition A.1 given in (Kharlamov, Zheleznyakov, and Calvanese 2013).

Lemma 1. *Let $\mathcal{T} \cup \mathcal{A}$ be a consistent DL-Lite$_{\mathcal{FR}}$ ontology and let $g = \exists R(c)$. If $\mathcal{A} \models_{\mathcal{T}} \exists R(c)$, then there exists a membership assertion $f \in \mathcal{A}$ such that $f \models_{\mathcal{T}} \exists R(c)$.*

Lemma 1 infers that each justification for $\exists R(c)$ contains only one membership assertion, where a justification for $\exists R(c)$ is the minimal subset of $\mathcal{A}$ that can entail $R(c)$. We define the notion of removing set of an assertion of the form $\exists R(c)$, which consists of all membership assertions that infer $\exists R(c)$.

Definition 4. *(Removing set) Let $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an ontology and $\mathcal{N}$ be an ABox in DL-Lite$_{\mathcal{FR}}$. Suppose $\mathcal{A} \models_{\mathcal{T}} \exists R(c)$. The* removing set *of $\exists R(c)$, denoted as $\mathrm{remove}_{\mathcal{T}}(\exists R(c))$, is defined as*

$$\mathrm{remove}_{\mathcal{T}}(\exists R(c)) = \{f \in \mathrm{AtAlg}(\mathcal{O},\mathcal{N}) | \{f\} \models_{\mathcal{T}} \exists R(c)\}.$$

In Example 2, we have $\mathrm{remove}_{\mathcal{T}}(\exists R(c)) = \{A(c), R(c, d), \exists R(c)\}$.

Theorem 3. *Let $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ be an ontology in DL-Lite$_{\mathcal{FR}}$ and $\mathcal{N}$ a new ABox. Suppose $\mathrm{TR}[\mathcal{O},\mathcal{N}] \neq \emptyset$. Let $\mathrm{AtAlg}(\mathcal{O},\mathcal{N})_{new} = \mathrm{AtAlg}(\mathcal{O},\mathcal{N}) \setminus \cup_{R \in \mathrm{TR}[\mathcal{O},\mathcal{N}]} \mathrm{remove}_{\mathcal{T}}(\exists R(c))$ and $\mathcal{O}' = \mathcal{T} \cup \mathcal{N} \cup \mathrm{AtAlg}(\mathcal{O},\mathcal{N})_{new}$. Then $\mathcal{O}'$ is a maximal approximation of $\mathcal{O} \circ_{\#} \mathcal{N}$ but it is not an approximation of $\mathcal{O} \circ_{\subseteq} \mathcal{N}$.*

Proof. (Sketch) We cannot apply the same method to construct models of $\mathcal{T} \cup \mathcal{A}$ and $\mathcal{T} \cup \mathcal{N}$ as in the proof of Theorem 1. We give a new method to construct models as follows. The construction of $\mathcal{I}_1$ and $\mathcal{J}_1$ is the same as the proof of Theorem 1. However, when updating $\mathcal{J}_1$ using $\mathcal{I}_1$, we need to exclude those role assertions where the role is in $\mathrm{TR}[\mathcal{O},\mathcal{N}]$. □

A Graph-based Algorithm for ABox Revision in DL-Lite

In the previous section, we have shown that the result of the revision operator given in (Kharlamov, Zheleznyakov, and Calvanese 2013) can be used to approximate the result of two model-based revision operators in DL-Lite$_{\mathcal{FR}}$. A variant of this operator is given in (Calvanese et al. 2010), which is defined by a more practical algorithm called *FastEvo*. This algorithm runs in polynomial time. However, as Algorithm AtAlg, it needs to compute the ABox closure w.r.t. the TBox, which will hinder their applicability for ontologies with large ABoxes. In this section, we propose a revision algorithm that does not compute the ABox closure beforehand and it can utilize state of the art graph databases to compute the result of revision. Before we give the algorithm, we present a revision operator that removes one assertion from each *minimal inconsistent subset* of $\mathcal{A}$ w.r.t. $\mathcal{N}$ and $\mathcal{T}$.

Definition 5. *Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ and an ABox $\mathcal{N}$. A minimal inconsistent subset (MIS) $\mathcal{D}$ of $\mathcal{A}$ w.r.t. $\mathcal{N}$ and $\mathcal{T}$ is a sub-ABox of $\mathcal{A}$ which satisfies (1) $\mathcal{D} \cup \mathcal{T} \cup \mathcal{N}$is inconsistent; (2) $\forall \mathcal{D}' \subset \mathcal{D}$, $\mathcal{D}' \cup \mathcal{T} \cup \mathcal{N}$ is consistent. We denote the set of all the MISs of $\mathcal{A}$ w.r.t. $\mathcal{N}$ by MIS$_{\mathcal{N}}(\mathcal{A})$ (we omit $\mathcal{T}$ to simplify the notation).*

Example 3. *(originally from (Giacomo et al. 2009)) Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ and a new ABox $\mathcal{N}$, where $\mathcal{T} = \{\exists WillPlay \sqsubseteq AvailablePlayer, AvailablePlayer \sqsubseteq Player, Injured \sqsubseteq \neg AvailablePlayer\}$, $\mathcal{A} = \{WillPlay(Peter, game06)\}$, $\mathcal{N} = \{Injured(Peter)\}$. It is easy to check that there exists one MIS of $\mathcal{A}$ w.r.t. $\mathcal{N}$: $\{WillPlay(Peter, game06)\}$.*

According to (Calvanese et al. 2010), every MIS of MIS$_{\mathcal{N}}(\mathcal{A})$ contains only one assertion. Thus, to restore consistency, we can simply remove $\cup_{\mathcal{D}_i \in MIS_{\mathcal{N}}(\mathcal{A})} \mathcal{D}_i$. However, this may delete much more information than necessary. Consider Example 1 again, we can find that Peter is injured implies that he is not an available player anymore, but he remains a player, and this would not be captured by simply removing $\cup_{\mathcal{D}_i \in MIS_{\mathcal{N}}(\mathcal{A})} \mathcal{D}_i$. Consequently, we will add $Player(Peter)$ to the result of revision as it does not contradict $\mathcal{N} \cup \mathcal{T}$ and it can be inferred from $\cup_{\mathcal{D}_i \in MIS_{\mathcal{N}}(\mathcal{A})} \mathcal{D}_i \cup \mathcal{T}$.

Definition 6. *Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ and an ABox $\mathcal{N}$, a maximal consistent set S of $cl_{\mathcal{T}}(\cup_{\mathcal{D}_i \in MIS_{\mathcal{N}}(\mathcal{A})} \mathcal{D}_i)$ w.r.t. $\mathcal{N}$ is a sub-ABox of $cl_{\mathcal{T}}(\cup_{\mathcal{D}_i \in MIS_{\mathcal{N}}(\mathcal{A})} \mathcal{D}_i)$ which satisfies (1) $S \cup \mathcal{T} \cup \mathcal{N}$ is consistent; (2) $\forall \alpha \in cl_{\mathcal{T}}(\cup_{\mathcal{D}_i \in MIS_{\mathcal{N}}(\mathcal{A})} \mathcal{D}_i)$ and $\alpha \notin S$, $S \cup \{\alpha\} \cup \mathcal{T} \cup \mathcal{N}$ is inconsistent.*

Definition 7. *Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ and an ABox $\mathcal{N}$. The revision operator $\circ_{MIS}$ for $\mathcal{O}$ is defined as follows:*

$$\mathcal{O} \circ_{MIS} \mathcal{N} = \mathcal{T} \cup (\mathcal{A} \setminus \cup_{\mathcal{D}_i \in MIS_{\mathcal{N}}(\mathcal{A})} \mathcal{D}_i) \cup S \cup \mathcal{N}$$

We can show that the deductive closure of the resulting ABox of our operator is the same as the ABox obtained by operator $\circ_{MCS}$ and the revision operator defined by algorithm *FastEvo* given in (Calvanese et al. 2010).

Theorem 4. *Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ and an ABox $\mathcal{N}$ in DL-Lite$_{\mathcal{FR}}$, suppose $\mathcal{O} \circ_{MIS} \mathcal{N} = (\mathcal{T}, \mathcal{A}', \mathcal{N})$, where $A' = (\mathcal{A} \setminus \cup_{\mathcal{D}_i \in MIS_{\mathcal{N}}(\mathcal{A})} \mathcal{D}_i) \cup S$, then $\mathcal{T} \cup cl_{\mathcal{T}}(\mathcal{A}') \cup \mathcal{N} = \mathcal{O} \circ_{MCS} \mathcal{N}$.*

A Graph-based Algorithm

Given a DL-Lite ontology $\mathcal{O}$ over a signature Σ, which can be partitioned into two disjoint signatures, Σ_P, containing symbols for atomic elements, i.e., atomic concept and atomic roles, and Σ_C, containing symbols for individuals, the digraph $G_{\mathcal{O}} = \langle N, E\rangle$ constructed from ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A}\rangle$ over the signature Σ as follows:

(1) for each atomic concept B in Σ_P, N contains the node B;

(2) for each atomic role P in Σ_P, N contains the node $P, P^-, \exists P, \exists P^-$;

(3) for each concept inclusion $B_1 \sqsubseteq B_2 \in T$, E contains the *arc*(B_1, B_2);

(4) for each role inclusion $P_1 \sqsubseteq P_2 \in T$, E contains the *arc*(P_1, P_2), *arc*(P_1^-, P_2^-), *arc*$(\exists P_1, \exists P_2)$, *arc* $(\exists P_1^-, \exists P_2^-)$;

(5) for each individual c in Σ_C, N contains leaf node c;

(6) for each concept membership assertion $B(c) \in \mathcal{A}$, E contains *arc*(c, B);

(7) for each role membership assertion $P(a, b)$, N contains node $(a, b), (b, a)$, and E contains the *arc*$((a, b), P)$, *arc*$((b, a), P^-)$, *arc*$(a, \exists P)$, *arc*$(b, \exists P^-)$;

In our graph, each node represents a basic concept or a basic role, while each arc represents an inclusion assertion or a membership assertion, i.e. the start node of the arc corresponds to the left-hand side of the inclusion assertion (resp. individual or individual-pair of the membership assertion) and the end node of the arc corresponds to the right-hand side of the inclusion assertion (resp. concept or role of the membership assertion). Items (4) and (7) are used to ensure that the information represented in the ontology is preserved by the graph.

Algorithm 2: GraphRevi($\mathcal{T}$, $\mathcal{A}$, $\mathcal{N}$)

```
   Input: TBox $\mathcal{T}$ and ABoxes $\mathcal{A}$, $\mathcal{N}$, each consistent
          with $\mathcal{T}$
   Output: $\mathcal{T} \cup (\mathcal{A} \backslash D) \cup M \cup \mathcal{N}$
1  begin
2      $D = \emptyset$;
3      $M = \emptyset$;
4      $A_{all} = \mathcal{A} \cup \mathcal{N}$;
5      for each $(functR) \in \mathcal{T}$ do
6          if $\{R(a,b), R(a,c)\} \subseteq A_{all}$ then
7              if $R(a,b) \notin \mathcal{N}$ then
8                  $D = D \cup \{R(a,b)\}$;
9              else
10                 $D = D \cup \{R(a,c)\}$;
11     construct $G_{\langle \mathcal{T}, A_{all} \backslash D\rangle} = \langle V, E\rangle$;
12     $D = D \cup Search(G_{\langle \mathcal{T}, A_{all} \backslash D\rangle}, \mathcal{A})$;
13     $M = cl_{\mathcal{T}}(D) \backslash D$;
14     construct HG $G_{\langle \mathcal{T}, M \cup \mathcal{A}_1\rangle} = \langle V, E\rangle$;
15     $M = M \backslash Search(G_{\langle \mathcal{T}, M \cup \mathcal{A}_1\rangle}, M)$;
16     return $\mathcal{T} \cup (\mathcal{A} \backslash D) \cup M \cup \mathcal{N}$;
17 end
```

Algorithm 3: Search(G, $\mathcal{A}$)

```
Data: Graph:$G$; ABox:$\mathcal{A}$
Result: finite set of membership assertions $N$
begin
    $N = \emptyset$;
    for each $\neg X \in V$ do
        $U = leafChild(X) \cap leafChild(\neg X)$;
        for each $a \in U$ do
            for each $v \in$
            $(Children(X) \cup Children(\neg X) \cup \{X\})$
            do
                if $v(a) \in \mathcal{A}$ then
                    if $v(a) = \exists R(a)$ then
                        for each $R(a,b) \in \mathcal{A}$ do
                            $N = N \cup \{R(a,b)\}$;
                    else
                        $N = N \cup \{v(a)\}$;
    return $N$;
end
```

We now introduce algorithm *GraphRevi* (see Algorithm 2), which takes $\mathcal{O} = \langle \mathcal{T}, \mathcal{A}\rangle$ and $\mathcal{N}$ as its input. The algorithm can be explained as follows. Let $\mathcal{A}_{all} = \mathcal{A} \cup \mathcal{N}$ (line 4). It first computes the set D of all the membership assertions in $\mathcal{A}$ that are in conflict with functionality axioms and $\mathcal{N}$ (lines 5-10). It then constructs a digraph from $\langle \mathcal{T}, \mathcal{A}_{all} \backslash D\rangle$ and uses function *Search* to compute the set of all the membership assertions in $\mathcal{A}$ that are in conflict with some negative inclusion assertions of the form $B \sqsubseteq \neg B'$ and some assertions in $\mathcal{N}$, and use this set to update D (see Algorithm 3). Thus, D is actually $\cup_{A_i \in MIS_{\mathcal{N}}(\mathcal{A})} A_i$. Let $M = cl_{\mathcal{T}}(D) \backslash D$ (lines 11-12). The algorithm deletes all the membership assertions in M that are in conflict with NI assertions and $\mathcal{N}$ (lines 13-15). Finally, $\mathcal{T} \cup (\mathcal{A} \backslash D) \cup M \cup \mathcal{N}$ is the result of revision (line 16).

Example 4. *Given an ontology $\mathcal{O} = \langle \mathcal{T}, \mathcal{A}\rangle$ and a new ABox $\mathcal{N}$, where $\mathcal{T} = \{(\texttt{Func } R), R \sqsubseteq R_1, \exists R^- \sqsubseteq B, A \sqsubseteq C, C \sqsubseteq \neg D, B \sqsubseteq \neg C\}$, $\mathcal{A} = \{R(a, b), C(d), A(e)\}$, $\mathcal{N} = \{R(a, f), A(b), D(d)\}$.*

1. *$A_{all} = \mathcal{A} \cup \mathcal{N} = \{R(a, b), C(d), A(e), R(a, f), A(b), D(d)\}$ (see Algorithm 2 line 4);*
2. *$D = \{R(a, b)\}$ (see Algorithm 1 lines 5-10);*
3. *By constructing digraph $G_{\langle \mathcal{T}, A_{all} \backslash D\rangle}$ (see Figure 1), we can obtain that $D = \{R(a, b), C(d)\}$(see Algorithm 2 lines 11-12);*
4. *$M = cl_{\mathcal{T}}(D) \backslash D = \{R_1(a, b), B(b), A(e), C(e)\}$ (see Algorithm 2 line 13);*
5. *By constructing digraph $G_{\langle \mathcal{T}, M \cup \mathcal{N}\rangle}$ (similar to Figure 1), then we can know that $M = \{R_1(a, b), A(e), C(e)\}$(see Algorithm 2 lines 14-15);*
6. *The result is: $\mathcal{T} \cup \{R(a, f), A(b), D(d), R_1(a, b), A(e), c(e)\}$ (see Algorithm 2 line 16).*

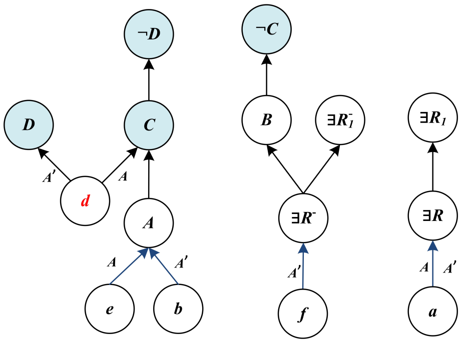

Figure 1: Digraph $G_{\langle \mathcal{T}, A_{all} \setminus D \rangle}$ of Example 4

Table 1: Generated ABoxes

Data	**univ4-1**	**univ4-2**	**univ6-1**	**univ6-2**
#axiom	636086	635581	876342	879631
Data	**univ8-1**	**univ8-2**	**univ10-1**	**univ10-2**
#axiom	1139949	1138337	1415102	1416326

Theorem 5. *Algorithm 2 runs in polynomial time. Given an ontology* $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ *and an ABox* $\mathcal{N}$*, we have* $\mathcal{O} \circ_{MIS} \mathcal{N} = GraphRevi(\mathcal{T}, \mathcal{A}, \mathcal{N})$.

Experimental Results

We have implemented our graph-based algorithm for ABox revision in Java. We first transform a DL-Lite ontology into a graph and store it in a Neo4j graph database, which is an open-source and high-performance graph database supported by Neo Technology. We also implemented the revision algorithm presented in (Calvanese et al. 2010) using Neo4j, which we denote as *FastEvo*.

We conducted some experiments on a data set constructed from UOBM benchmark ontology[4] (see Table 1 for details of the data set). We generated ABoxes by using the UOBM generator. We divided each generated ABox into two parts. We used the Random class of Java to control the dividing procedure. Since the original UOBM ontologies are consistent, we modified them by inserting some ”inconsistency-generating” axioms, such as disjointness axioms. We generated different percentage of the disjoint classes for each university ontology. After that, for each pair of disjoint concepts or roles, we generated a common instance or pair of instances and added the two conflicting assertions to two ABoxes partitioned from an ABox.

All experiments have been performed on a PC with Intel Corei5-2400 3.1 GHz CPU and 6GB of RAM, running Microsoft window 7 operating system, and Java 1.7 with 6GB of heap space.

Table 2: Execution Time (s) of ABox Revision

#univ	**Disjointness(%)**	**GraphRevi**	**FastEvo**
4	20	17.693	51.66
	30	26.591	68.702
	40	41.423	110.251
6	20	19.802	69.001
	30	27.874	86.685
	40	44.833	135.518
8	20	23.353	89.606
	30	31.070	105.483
	40	49.313	167.695
10	20	26.309	119.781
	30	33.945	133.852
	40	59.798	201.149

We did four experiments to compare the execution time of GraphRevi and *FastEvo*. The results of our experiments are shown in Table 2. According to Table 2, GraphRevi outperforms *FastEvo* when the number of universities and the percentage of disjointness axioms are increasing. In many cases, GraphRevi is 4 to 5 times faster than *FastEvo*. When the percentage of the disjointness axioms is low, GraphRevi runs very efficiently, i.e., in less than 30 seconds. Another observation is that with the increasing percentage of disjointness axioms, GraphRevi becomes less efficient, this is because it takes more time to find the membership assertions in $\mathcal{A}$ that are in conflict with $\mathcal{N}$ w.r.t. $\mathcal{T}$. We also observe that in almost all the test cases, GraphRevi can finish the computation in less than 60 seconds (even when 10 universities are considered). Furthermore, GraphRevis performs well for ontologies with 12 universities, but FastEvol cannot handle them.

Related Work

This work is closely related to the work presented in (Calvanese et al. 2010) and (Kharlamov, Zheleznyakov, and Calvanese 2013). In (Calvanese et al. 2010), the authors propose an ABox revision algorithm *FastEvol* for DL-Lite$_{\mathcal{FR}}$ ontologies, but no implementation is provided. Our experimental results show that our graph-based algorithm significantly outperforms *FastEvol*. It is proven in (Kharlamov, Zheleznyakov, and Calvanese 2013) that $Mod(\mathcal{O} \circ_{MCS} \mathcal{N})$ and $Mod(\mathcal{O} \circ_{\#} \mathcal{N})$ coincide on a fragment of DL-Lite$_{core}$. In this paper, we show that while this result does not hold in the full DL-Lite$_{core}$, $\mathcal{O} \circ_{MCS} \mathcal{N}$ is a maximal approximation of the model-based revision $\mathcal{O} \circ_{\#} \mathcal{N}$ in DL-Lite$_{\mathcal{R}}$.

Revision of DL-based ontologies has been widely discussed in the literature. Most of the work on model-based revision in DLs is devoted to proving the inexpressibility of model-based revision operators (see (Qi and Du 2009), (Calvanese et al. 2010) and (Grau et al. 2012) for example). However, very few of them discuss the approximation of model-based revision. One exception is the work in (Wang, Wang, and Topor 2010), where a revision operator is defined by a new semantics called *features*. However, feature-based revision also suffers from the inexpressibility problem and the algorithm to approximating the result of revision is intractable and is inefficient to deal with large ABoxes. Recently, there are some works on TBox revision based on a new semantics, called *type semantics* (see (Zhuang et al.

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

2014) and (Wang et al. 2015)).

Another line of work is to adapt the well-known AGM (Alchourrón, Gärdenfors and Markinson) framework to DLs (see (Flouris, Plexousakis, and Antoniou 2005), (Flouris et al. 2006) and (Ribeiro et al. 2013)) and adapt Hansson's postulates for revision (see (Hansson 1999)) to DLs (Ribeiro and Wassermann 2007). However, model-based revision operators are not discussed.

Most of practical revision operators proposed in the literature are syntax dependant, i.e., if two logically dependant ontologies are revised by another ontology, the results of revision may not be logically equivalent. Representative work on syntax-based revision operators can be found in (Haase et al. 2005), (Halaschek-Wiener, Katz, and Parsia 2006) and (Qi et al. 2008). Syntax-based revision operators are not fine-grained because an axiom is removed even if only part of it is involved in the inconsistencies. In this work, we prove that a syntactic revision operator can be used to approximate two model-based revision operators in DL-Lite$_{\mathcal{R}}$, and fill the gap between syntax-based revision and model-based revision.

Conclusion

In this paper, we presented some theoretical work on approximation of model-based ABox revision operators in DL-Lite$_{\mathcal{FR}}$ and proposed an algorithm for computing the result of approximation efficiently. On the theoretical aspect, we discussed whether the result of the syntactic revision algorithm given in (Kharlamov, Zheleznyakov, and Calvanese 2013) can be used to approximate the result of two model-based revision operators. We showed that this property holds in DL-Lite$_{\mathcal{R}}$ but fails in DL-Lite$_{\mathcal{FR}}$ when role functionality axioms are included. In the failure case, we showed that the property still holds if we disallow "triggering roles" in the TBox. We also showed that the result of a modification of the syntactic algorithm can be used to approximate the result of one of the model-based revision operators. On the practical aspect, we proposed a graph-based algorithm that can output the same result as algorithm *FastEvol* but does not need to compute the ABox closure w.r.t. the TBox beforehand. Our algorithm applies new methodology to perform revision using graph databases. We implemented a revision system based on the graph-based algorithm, called GraphRevi, and conducted experiments over a benchmark dataset. Preliminary experimental results show that our system can handle the revision of large DL-Lite ABoxes efficiently and outperforms *FastEvol*.

As a future work, we will optimize our system by exploring distributed index. As another future work, we will work on approximation of ABox revision in other DL-Lite languages, such as DL-Lite$^{\mathcal{N}}_{bool}$.

Acknowledgement

This work was partially supported by NSFC grant 61272378 and Australian Research Council (ARC) under DP130102302 and DP1093652.

References

Calvanese, D.; Kharlamov, E.; Nutt, W.; and Zheleznyakov, D. 2010. Evolution of dl-lite knowledge bases. In *Proc. of ISWC*, 112–128.

Flouris, G.; Huang, Z.; Pan, J.; Plexousakis, D.; and Wache, H. 2006. Inconsistencies, negations and changes in ontologies. In *Proc. of AAAI*, 1295–1300.

Flouris, G.; Plexousakis, D.; and Antoniou, G. 2005. On applying the AGM theory to DLs and OWL. In *Proc. of ISWC*, 216–231.

Giacomo, G. D.; Lenzerini, M.; Poggi, A.; and Rosati, R. 2009. On instance-level update and erasure in description logic ontologies. *J. Log. Comput.* 19(5):745–770.

Grau, B. C.; Ruiz, E. J.; Kharlamov, E.; and Zhelenyakov, D. 2012. Ontology evolution under semantic constraints. In *Proc.of KR*, 137–147.

Haase, P.; van Harmelen, F.; Huang, Z.; Stuckenschmidt, H.; and Sure, Y. 2005. A framework for handling inconsistency in changing ontologies. In *Proc. of ISWC*, 353–367.

Halaschek-Wiener, C.; Katz, Y.; and Parsia, B. 2006. Belief base revision for expressive description logics. In *Proc. of OWL-ED*.

Hansson, S. 1999. *A Textbook of Belief Dynamics: Theory Change and Database Updating*. Kluwer Academic Publishers.

Katsuno, H., and Mendelzon, A. 1992. Propositional knowledge base revision and minimal change. *Artif. Intell.* 52(3):263–294.

Kharlamov, E., and Zheleznyakov, D. 2011. Capturing instance level ontology evolution for DL-Lite. In *Proc. of ISWC*, 321–337.

Kharlamov, E.; Zheleznyakov, D.; and Calvanese, D. 2013. Capturing model-based ontology evolution at the instance level: The case of dl-lite. *J. Comput. Syst. Sci.* 79(6):835–872.

Qi, G., and Du, J. 2009. Model-based revision operators for terminologies in description logics. In *Proc. of IJCAI*, 891–897.

Qi, G.; Haase, P.; Huang, Z.; Ji, Q.; Pan, J.; and Völker, J. 2008. A kernel revision operator for terminologies–algorithms and evaluation. In *Proc. of ISWC*. 419–434.

Ribeiro, M. M., and Wassermann, R. 2007. Base revision in description logics - preliminary results. In *Proc. of IWOD*, 69–82.

Ribeiro, M. M.; Wassermann, R.; Flouris, G.; and Antoniou, G. 2013. Minimal change: Relevance and recovery revisited. *Artificial Intelligence* 201:59–80.

Wang, Z.; Wang, K.; Zhuang, Z.; and Qi, G. 2015. Instance-driven ontology evolution in DL-Lite. In *Proc. of AAAI, to appear*.

Wang, Z.; Wang, K.; and Topor, R. W. 2010. A new approach to knowledge base revision in DL-Lite. In *Proc. of AAAI*, 369–374.

Zhuang, Z.; Wang, Z.; Wang, K.; and Qi, G. 2014. Contraction and revision over DL-Lite tboxes. In *Proc. of AAAI*, 1149–1156.