On the evolution of the instance level of *DL-Lite* knowledge bases

Maurizio Lenzerini, Domenico Fabio Savo

Dipartimento di Informatica e Sistemistica Antonio Ruberti
SAPIENZA Università di Roma
lastname@dis.uniroma1.it

Abstract. Recent papers address the issue of updating the instance level of knowledge bases expressed in Description Logic following a model-based approach. One of the outcomes of these papers is that the result of updating a knowledge base $\mathcal{K}$ is generally not expressible in the Description Logic used to express $\mathcal{K}$. In this paper we introduce a formula-based approach to this problem, by revisiting some research work on formula-based updates developed in the '80s, in particular the WIDTIO (When In Doubt, Throw It Out) approach. We show that our operator enjoys desirable properties, including that both insertions and deletions according to such operator can be expressed in the DL used for the original KB. Also, we present polynomial time algorithms for the evolution of the instance level knowledge bases expressed in *DL-Lite*$_{A,id}$, which the most expressive Description Logics of the *DL-Lite* family.

1 Introduction

Description Logics (DLs) are logics for expressing knowledge bases (KBs) constituted by two components, namely, the TBox, asserting general properties of concepts and roles (binary relations), and the ABox, which is a set of assertions about individuals that are instances of concepts and roles. It is widely accepted that such logics are well-suited for expressing ontologies, with the TBox capturing the intensional knowledge about the domain of interest, and the ABox expressing the knowledge about the instance level of the predicates defined in the TBox. Following this idea, several Knowledge Representation Systems, called DL systems, have been recently built, providing methods and tools for managing ontologies expressed in DLs [1]. Notice that numerous DLs have been studied in the last decades, with the goal of analyzing the impact of the expressive power of the DL language to the complexity of reasoning. Consequently, each DL system is tailored towards managing KB expressed in a specific DL.

By referring to the so-called *functional view of knowledge representation* [11], DL systems should be able to perform two kinds of operations, called ASK and TELL. ASK operations, such as subsumption checking, or query answering, are

[1] http://www.cs.man.ac.uk/ sattler/reasoners.html

used to extract information from the KB, whereas TELL operations aim at changing the KB according to new knowledge acquired over the domain. In other words, TELL operations should be able to cope with the *evolution of the KB*.

There are two types of evolution operators, corresponding to inserting, and deleting chunks of knowledge, respectively. In the case of insertion, the aim is to incorporate new knowledge into the KB, and the corresponding operator should be defined in such a way to compute a consistent KB that supports the new knowledge. In the case of deletion, the aim is to come up with a consistent KB where the retracted knowledge is not valid. In both cases, the crucial aspect to take into account is that evolving a consistent knowledge base should not introduce inconsistencies.

While ASK operations have been investigated in detail by the DL community, existing DL reasoners do not provide explicit services for KB evolution. Nevertheless, many recent papers demonstrate that the interest towards a well-defined approach to KB evolution is growing significantly [9, 12, 7, 13, 6]. Following the tradition of the work on knowledge revision and update [10], all the above papers advocate some minimality criterion in the changes of the KB that must be undertaken to realize the evolution operations. In other words, the need is commonly perceived of keeping the distance between the original KB and the KB resulting from the application of an evolution operator minimal. There are two main approaches to define such a distance, called *model-based* and *formula-based*, respectively. In the model-based approaches, the result of an evolution operation applied to the KB $\mathcal{K}$ is defined in terms of a set of models, with the idea that such a set should be as close as possible to the models of $\mathcal{K}$. One basic problem with this approach is to characterize the language needed to express the KB that exactly captures the resulting set of models. Conversely, in the formula-based approaches, the result is explicitly defined in terms of a formula, by resorting to some minimality criterion with respect to the formula expressing $\mathcal{K}$. Here, the basic problem is that the formula constituting the result of an evolution operation is not unique in general.

In this paper, we study the problem of DL KB evolution, by focusing our attention to scenarios characterized by the following elements:

(1) We consider the case where the evolution affects only the instance level of the KB, i.e., the ABox. In other words, we enforce the condition that the KB resulting from the application of the evolution operators has the same TBox as the original KB (similarly to [12, 7]).

(2) We aim at a situation where the KB resulting from the evolution can be expressed in the same DL as the original KB. This is coherent with our goal of providing the foundations for equipping DL systems with evolution operators: indeed, if a DL system S is able to manage KBs expressed in a DL $\mathcal{L}$, the result of evolving such KBs should be expressible in $\mathcal{L}$.

(3) The KBs resulting from the application of an evolution operator on two logically equivalent KBs should be mutually equivalent. In other words, we want the result to be independent of the syntactic form of the original KB.

Assumption (1), although limiting the generality of our approach, captures several interesting scenarios, including *ontology-based data management*, where the DL KB is used as a logic-based interface to existing data sources.

As for item (2), we note that virtually all model-based approaches suffer from the expressibility problem. This has been reported in many recent papers, including [12, 7, 6], for various DLs. For this reason, we adopt a formula-based approach, inspired in particular by the work developed in [8] for updating logical theories. As in [8], we consider both insertions and deletions. However, we differ from [8] for an important aspect. We already noted that the formula constituting the result of an evolution operation is not unique in general. While [8] essentially proposes to keep the whole set of such formulas, we take a radical approach, and consider their intersection as the result of the evolution. In other words, we follow the *When In Doubt Throw It Out* (WIDTIO) [14] principle.

Finally, to deal with item (3), we sanction that the notion of distance between KBs refers to the closure of the ABox of a KB, rather than to the ABox itself. The closure of an ABox $\mathcal{A}$ with respect to an TBox $\mathcal{T}$ is defined as the set of all ABox assertions that logically follows from $\mathcal{T}$ and $\mathcal{A}$. By basing the definition of distance on the closure of ABoxes, we achieve the goal of making the result of our operators independent of the form of the original KB.

After a brief introduction to DLs (Section 2), we provide the definition of our evolution operators in Section 3. The remaining sections are devoted to illustrating algorithms for deletion (Section 4), and insertion (Section 5) for KBs expressed in the DL *DL-Lite*$_{A,id}$, which is the most expressive logic in the *DL-Lite* family [4]. The *DL-Lite* family[2] has been specifically designed to keep all reasoning tasks polynomially tractable, and we show that this property still holds for the evolution operators proposed in this paper.

2 Preliminaries

Let $\mathcal{S}$ be a signature of symbols for individual (object and value) constants, and atomic elements, i.e., concepts, value-domains, attributes, and roles. If $\mathcal{L}$ is a DL, then an $\mathcal{L}$-KB $\mathcal{K}$ over $\mathcal{S}$ is a pair $\langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T}$, called *TBox*, is a finite set of intensional assertions over $\mathcal{S}$ expressed in $\mathcal{L}$, and $\mathcal{A}$, called *ABox*, is a finite set of instance assertions, i.e, assertions on individuals, over $\mathcal{S}$ expressed in $\mathcal{L}$. Different DLs allow for different kinds of concept, attribute, and role expressions, and different kinds of TBox and ABox assertions over such expressions. In this paper we assume that ABox assertions are always *atomic*, i.e., they correspond to ground atoms, and therefore we omit to refer to $\mathcal{L}$ when we talk about ABox assertions.

The semantics of a DL KB is given in terms of interpretations. An interpretation is a *model* of a KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ if it satisfies all assertions in $\mathcal{T} \cup \mathcal{A}$, where the notion of satisfaction depends on the constructs allowed by the specific DL in which $\mathcal{K}$ is expressed. We denote the set of models of $\mathcal{K}$ with $Mod(\mathcal{K})$.

[2] Not to be confused with the set of DLs studied in [2], which form the *DL-Lite*$_{bool}$ family.

Let $\mathcal{T}$ be a TBox in $\mathcal{L}$, and let $\mathcal{A}$ be an ABox. We say that $\mathcal{A}$ is $\mathcal{T}$*-consistent* if $\langle \mathcal{T}, \mathcal{A} \rangle$ is satisfiable, i.e. if $Mod(\langle \mathcal{T}, \mathcal{A} \rangle) \neq \emptyset$, $\mathcal{T}$-inconsistent otherwise. The $\mathcal{T}$*-closure* of $\mathcal{A}$ with respect to $\mathcal{T}$, denoted $\mathsf{cl}_{\mathcal{T}}(\mathcal{A})$, is the set of all atomic ABox assertion that are formed with individuals in $\mathcal{A}$, and are logically implied by $\langle \mathcal{T}, \mathcal{A} \rangle$. Note that if $\langle \mathcal{T}, \mathcal{A} \rangle$ is an $\mathcal{L}$-KB, then $\langle \mathcal{T}, \mathsf{cl}_{\mathcal{T}}(\mathcal{A}) \rangle$ is an $\mathcal{L}$-KB as well, and is logically equivalent to $\langle \mathcal{T}, \mathcal{A} \rangle$, i.e., $Mod(\langle \mathcal{T}, \mathcal{A} \rangle) = Mod(\langle \mathcal{T}, \mathsf{cl}_{\mathcal{T}}(\mathcal{A}) \rangle)$. $\mathcal{A}$ is said to be $\mathcal{T}$*-closed* if $\mathsf{cl}_{\mathcal{T}}(\mathcal{A}) = \mathcal{A}$. Finally, for an ABox assertion γ_1, we denote by $\mathsf{Subsumee}_{\langle \mathcal{T}, \mathcal{A} \rangle}(\gamma_1)$ the set of atoms $\gamma_2 \in \mathsf{cl}_{\mathcal{T}}(\mathcal{A})$ such that $\langle \mathcal{T}, \mathcal{A} \rangle \models \gamma_2 \supset \gamma_1$.

The *DL-Lite* family [4] is a family of low complexity DLs particularly suited for dealing with KBs with very large ABoxes, and forms the basis of OWL 2 QL, one of the profile of OWL 2, the official ontology specification language of the World-Wide-Web Consortium (W3C)[3].

We now present the DL *DL-Lite*$_{A,id}$, which is the most expressive logic in the family. Expressions in *DL-Lite*$_{A,id}$ are formed according to the following syntax:

$$
\begin{array}{lll}
B \longrightarrow A \mid \exists Q \mid \delta(U) & E \longrightarrow \rho(U) & C \longrightarrow B \mid \neg B \\
Q \longrightarrow P \mid P^- & V \longrightarrow U \mid \neg U & R \longrightarrow Q \mid \neg Q \\
T \longrightarrow \top_D \mid T_1 \mid \cdots \mid T_n & &
\end{array}
$$

where A, P, and U are symbols in $\mathcal{S}$ denoting respectively an atomic concept name, an atomic role name and an attribute name, $T_1, \ldots, T_n$ are all the value-domains allowed in the logic (those corresponding to the data types adopted by Resource Description Framework (RDF)[4]), $\top_D$ denotes the union of all domain values, P^- denotes the inverse of P, $\exists Q$ denotes the objects related to by the role Q, $\neg$ denotes negation, $\delta(U)$ denotes the *domain* of U, i.e., the set of objects that U relates to values, and $\rho(U)$ denotes the *range* of U, i.e., the set of values related to objects by U.

A *DL-Lite*$_{A,id}$ TBox $\mathcal{T}$ contains intensional assertions of three types, namely inclusion assertions, functionality assertions, and identification assertions [5] (IDs). More precisely, *DL-Lite*$_{A,id}$ assertions are of the form:

$$
\begin{array}{llll}
B \sqsubseteq C & (\textit{concept inclusion}) & E \sqsubseteq T & (\textit{value-domain inclusion}) \\
Q \sqsubseteq R & (\textit{role inclusion}) & (\mathsf{funct}\ U) & (\textit{attribute functionality}) \\
(id\ B\ \pi_1, ..., \pi_n) & (\textit{identification}) & &
\end{array}
$$

In the identification assertions, π denotes a *path*, which is an expression built according to the following syntax rule:

$$\pi \longrightarrow S \mid B? \mid \pi_1 \circ \pi_2$$

where S denotes an atomic role, the inverse of an atomic role, or an atomic attribute, $\pi_1 \circ \pi_2$ denotes the composition of the paths π_1 and π_2, and $B?$, called *test relation*, represents the identity relation on instances of the concept B. In our logic, identification assertions are *local*, i.e., at least one $\pi_i \in \{\pi_1, ..., \pi_n\}$ has length 1, i.e., it is an atomic role, the inverse of an atomic role, or an atomic attribute. In what follows, we only refer to IDs which are local.

[3] `http://www.w3.org/TR/2008/WD-owl2-profiles-20081008/`

[4] `http://www.w3.org/RDF/`

The set of positive (resp., negative) inclusions in $\mathcal{T}$ will be denoted by $\mathcal{T}^+$ (resp., $\mathcal{T}^-$), and the set of identification assertions in $\mathcal{T}$ will be denoted by $\mathcal{T}_{id}$.

A concept inclusion assertion expresses that a (basic) concept B is subsumed by a (general) concept C. Analogously for the other types of inclusion assertions. Inclusion assertions that do not contain (resp. contain) the symbols '$\neg$' in the right-hand side are called *positive inclusions* (resp. *negative inclusions*). Attribute functionality assertions are used to impose that attributes are actually functions from objects to domain values. An ID $(id\ B\ \pi_1, ..., \pi_n)$ asserts that for any two different instances a,b of B, there is at least one π_i such that a and b differ in the set of their π_i-fillers. Note that IDs can be used to assert functionality of roles. Specifically, the assertion $(id\ \exists Q^-\ Q^-)$ imposes that Q is functional.

Finally, a TBox *DL-Lite*$_{A,id}$ $\mathcal{T}$ satisfies the following condition: every role or attribute that occurs (in either direct or inverse direction) in a path of an ID $\alpha \in \mathcal{T}_{id}$ or in a functional assertion, is not specialized in $\mathcal{T}'$, i.e., it does not appear in the right-hand side of assertions of the form $Q \sqsubseteq Q'$ or $U \sqsubseteq U'$.

A *DL-Lite*$_{A,id}$ ABox $\mathcal{A}$ is a finite set of assertions of the form $A(a)$, $P(a,b)$, and $U(a,v)$, where A, P, and U are as above, a and b are object constants in $\mathcal{S}$, and v is a value constant in $\mathcal{S}$.

Example 1. We consider a portion of the Formula One domain. We know that official drivers (OD) and test drivers (TD) are both team members (TM), and official drivers are not test drivers. Every team member is a member of (mf) a exactly one team (FT), and every team has at most one official driver. Finally, no race director (RD) is a member of a team. We also know that s is the official driver of team t_1, that b is a test driver, and that p is a team member. The corresponding *DL-Lite*$_{A,id}$-KB $\mathcal{K}$ is:

$\mathcal{T}$: $OD \sqsubseteq TM\ \ TD \sqsubseteq TM\ \ OD \sqsubseteq \neg TD\ \ RD \sqsubseteq \neg TM\ \ TM \sqsubseteq \exists mf$
$TM \sqsubseteq \neg FT\ \ \exists mf \sqsubseteq TM\ \ \exists mf^- \sqsubseteq FT\ \ (id\ OD\ mf)\ \ (id\ FT\ mf^-)$
$\mathcal{A}$: $OD(s)\quad mf(s,t_1)\quad TD(b)\quad TM(p)$ □

We conclude this section with a brief discussione on the complexity of reasoning about a *DL-Lite*$_{A,id}$-KB $\langle \mathcal{T}, \mathcal{A} \rangle$. Satisfiability can be checked in polynomial time with respect to $|\mathcal{T} \setminus \mathcal{T}_{id}|$ and $|\mathcal{A}|$, and in NP with respect to $|\mathcal{T}_{id}|$. Moreover, if $\langle \mathcal{T}, \mathcal{A} \rangle$ is satisfiable, then answering a query q posed to $\langle \mathcal{T}, \mathcal{A} \rangle$ can be done in polynomial time with respect to $|\mathcal{T}|$ and $|\mathcal{A}|$, and in NP with respect to $|q|$. Finally, $\mathsf{cl}_{\mathcal{T}}(\mathcal{A})$ can be computed in quadratic time with respect to $|\mathcal{T}|$ and $|\mathcal{A}|$.

3 WIDTIO approach to KB evolution in DLs

In this section we first present our semantics for the evolution of DL knowledge bases at the instance level, and then we provide a comparison between our operator and other work in the literature.

In the rest of this section, $\mathcal{L}$ is a DL, and $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ is a satisfiable $\mathcal{L}$-KB. In other words, we do not consider the evolution of unsatisfiable KBs. In addition, F is a finite set of atomic ABox assertions in $\mathcal{L}$.

The following definition specifies when a set of ABox assertions "realizes" the insertion or deletion of a set of ABox assertions with respect to $\mathcal{K}$.

Definition 1. *Let $\mathcal{A}'$ be an ABox. Then, $\mathcal{A}'$ accomplishes the insertion of F into $\langle \mathcal{T}, \mathcal{A} \rangle$ if $\mathcal{A}'$ is $\mathcal{T}$-consistent, and $\langle \mathcal{T}, \mathcal{A}' \rangle \models F$ (i.e., $F \subseteq \mathsf{cl}_{\mathcal{T}}(\mathcal{A}')$). Similarly, $\mathcal{A}'$ accomplishes the deletion of F from $\langle \mathcal{T}, \mathcal{A} \rangle$ if $\mathcal{A}'$ is $\mathcal{T}$-consistent, and $\langle \mathcal{T}, \mathcal{A}' \rangle \not\models F$ (i.e., $F \not\subseteq \mathsf{cl}_{\mathcal{T}}(\mathcal{A}')$).*

Obviously, we are interested in KBs which accomplish the evolution of a KB with a *minimal change*. In order to formalize the notion of *minimal change*, we first need to provide some definitions.

Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two ABoxes. Then, we say that $\mathcal{A}_1$ has fewer deletions than $\mathcal{A}_2$ with respect to $\langle \mathcal{T}, \mathcal{A} \rangle$ if $\mathsf{cl}_{\mathcal{T}}(\mathcal{A}) \setminus \mathsf{cl}_{\mathcal{T}}(\mathcal{A}_1) \subset \mathsf{cl}_{\mathcal{T}}(\mathcal{A}) \setminus \mathsf{cl}_{\mathcal{T}}(\mathcal{A}_2)$. Similarly, we say that $\mathcal{A}_1$ and $\mathcal{A}_2$ have the same deletions with respect to $\langle \mathcal{T}, \mathcal{A} \rangle$ if $\mathsf{cl}_{\mathcal{T}}(\mathcal{A}) \setminus \mathsf{cl}_{\mathcal{T}}(\mathcal{A}_1) = \mathsf{cl}_{\mathcal{T}}(\mathcal{A}) \setminus \mathsf{cl}_{\mathcal{T}}(\mathcal{A}_2)$. Finally, we say that $\mathcal{A}_1$ has fewer insertions than $\mathcal{A}_2$ with respect to $\langle \mathcal{T}, \mathcal{A} \rangle$ if $\mathsf{cl}_{\mathcal{T}}(\mathcal{A}_1) \setminus \mathsf{cl}_{\mathcal{T}}(\mathcal{A}) \subset \mathsf{cl}_{\mathcal{T}}(\mathcal{A}_2) \setminus \mathsf{cl}_{\mathcal{T}}(\mathcal{A})$.

Definition 2. *Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two ABoxes. Then, $\mathcal{A}_1$ has* fewer changes *than $\mathcal{A}_2$ with respect to $\langle \mathcal{T}, \mathcal{A} \rangle$ if $\mathcal{A}_1$ has fewer deletions than $\mathcal{A}_2$ with respect to $\langle \mathcal{T}, \mathcal{A} \rangle$, or $\mathcal{A}_1$ and $\mathcal{A}_2$ have the same deletions with respect to $\langle \mathcal{T}, \mathcal{A} \rangle$, and $\mathcal{A}_!$ has fewer insertions than $\mathcal{A}_2$ with respect to $\langle \mathcal{T}, \mathcal{A} \rangle$.*

Now that we have defined the relation of *fewer changes* between two KBs w.r.t. another one, we can define the notion of a KB which accomplishes the insertion (resp. deletion) of a set of facts into (resp. from) another KB minimally.

Definition 3. *Let $\mathcal{A}'$ be an ABox. Then $\mathcal{A}'$ accomplishes the insertion (deletion) of F into (from) $\langle \mathcal{T}, \mathcal{A} \rangle$ minimally if $\mathcal{A}'$ accomplishes the insertion (deletion) of F into (from) $\langle \mathcal{T}, \mathcal{A} \rangle$, and there is no $\mathcal{A}''$ that accomplishes the insertion (deletion) of F into (from) $\langle \mathcal{T}, \mathcal{A} \rangle$, and has fewer changes than $\mathcal{A}'$ with respect to $\langle \mathcal{T}, \mathcal{A} \rangle$.*

With these notions in place, we can now define our evolution operator.

Definition 4. *Let $\mathcal{U} = \{\mathcal{A}_1, \ldots, \mathcal{A}_n\}$ be the set of all ABoxes accomplishing the insertion (deletion) of F into (from) $\langle \mathcal{T}, \mathcal{A} \rangle$ minimally, and let $\mathcal{A}'$ be an ABox. Then, $\langle \mathcal{T}, \mathcal{A}' \rangle$ is the result of changing $\langle \mathcal{T}, \mathcal{A} \rangle$ with the insertion (deletion) of F if (1) $\mathcal{U}$ is empty, and $\langle \mathcal{T}, \mathsf{cl}_{\mathcal{T}}(\mathcal{A}') \rangle = \langle \mathcal{T}, \mathsf{cl}_{\mathcal{T}}(\mathcal{A}) \rangle$, or (2) $\mathcal{U}$ is nonempty, and $\langle \mathcal{T}, \mathsf{cl}_{\mathcal{T}}(\mathcal{A}') \rangle = \langle \mathcal{T}, \bigcap_{1 \leq i \leq n} \mathsf{cl}_{\mathcal{T}}(\mathcal{A}_i) \rangle$.*

It is immediate to verify that, up to logical equivalence, the result of changing $\langle \mathcal{T}, \mathcal{A} \rangle$ with the insertion or the deletion of F is unique. In the rest of this paper, the result of changing $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ with the insertion (resp. deletion) of F according to our semantics will be denoted by $\mathcal{K} \oplus_{\cap}^{\mathcal{T}} F$ (resp. $\mathcal{K} \ominus_{\cap}^{\mathcal{T}} F$). Notice that, by definition of our operator, in the case where F is $\mathcal{T}$-inconsistent, the result of changing $\langle \mathcal{T}, \mathcal{A} \rangle$ with both the insertion and the deletion of F is logically equivalent to $\langle \mathcal{T}, \mathcal{A} \rangle$ itself.

Example 2. Consider the *DL-Lite$_{A,id}$* KB $\mathcal{K}$ of the Example 1, and suppose that p becomes now a race director, and b becomes the new official driver of the team t_1. To reflect this new information, we change $\mathcal{K}$ with the insertion of $F_1 = \{RD(p), OD(b), mf(b, t_1)\}$. Since the TBox implies that a race director cannot be a team member, $RD(p)$ contradicts $TM(p)$. Also, since every team has at most one official driver, $OD(b)$ and $mf(b, t_1)$ contradict $mf(s, t)$. According to Definition 3, the KBs accomplishing the insertion of F_1 into $\mathcal{K}$ minimally are:

$$\mathcal{K}_1 = \langle \mathcal{T}, \{RD(p), OD(b), mf(b, t_1), TM(s), mf(s, t_1)\}\rangle$$
$$\mathcal{K}_2 = \langle \mathcal{T}, \{RD(p), OD(b), mf(b, t_1), TM(s), OD(s)\}\rangle$$

Thus, $\mathcal{K} \oplus_{\cap}^{\mathcal{T}} F_1$ is:

$$\mathcal{K}_3 = \langle \mathcal{T}, \{RD(p), OD(b), mf(b, t_1), TM(s)\}\rangle.$$

Now, suppose that we do not know anymore whether b is a member of t_1, and, even more, whether b is a team member at all. Then, we change $\mathcal{K}_3$ with the deletion of $F_2 = \{TM(b), mf(b, t_1)\}$, thus obtaining

$$\mathcal{K}_3 \ominus_{\cap}^{\mathcal{T}} F_2 = \langle \mathcal{T}, \{RD(p), TM(s), OD(b)\}\rangle. \qquad \square$$

The following theorem is an adaptation to our setting of two results reported in [8], and will be used in the next two sections.

Theorem 1. *Let $\mathcal{A}'$ be an ABox. Then*

1. *$\mathcal{A}'$ accomplishes the deletion of F from $\langle \mathcal{T}, \mathcal{A}\rangle$ minimally if and only if $\mathsf{cl}_{\mathcal{T}}(\mathcal{A}')$ is a maximal $\mathcal{T}$-closed subset of $\mathsf{cl}_{\mathcal{T}}(\mathcal{A})$ such that $F \not\subseteq \mathsf{cl}_{\mathcal{T}}(\mathcal{A}')$.*
2. *$\mathcal{A}'$ accomplishes the insertion of F from $\langle \mathcal{T}, \mathcal{A}\rangle$ minimally if and only if $\mathsf{cl}_{\mathcal{T}}(\mathcal{A}') = \mathcal{A}'' \cup \mathsf{cl}_{\mathcal{T}}(F)$, where $\mathcal{A}''$ is a maximal $\mathcal{T}$-closed subset of $\mathsf{cl}_{\mathcal{T}}(\mathcal{A})$ such that $\mathcal{A}'' \cup F$ is $\mathcal{T}$-consistent.*

We end this section with a brief discussion on related work. We mentioned in the introduction several model-based approaches to DL KB evolution, and noticed that they all suffer from the expressibility problem. This problem is also shared by [13], that uses *features* instead of models, and proposes the notion of approximation to cope with the expressibility problem, similarly to [7].

Related to our proposal are several formula-based approaches presented in the literature. Perhaps, the closest approach to the one proposed in this paper is that reported in [6], where formula-based evolution (actually, insertion) of *DL-Lite* KBs is studied. The main difference with our work is that we base our semantics on the WIDTIO principles, and therefore we compute the intersection of all KBs accomplishing the change minimally. Conversely, in the *bold* semantics discussed in [6], the result of the change is chosen non-deterministically among the KBs accomplishing the change minimally. Another difference is that while [6] addresses the issue of evolution of both the TBox and the ABox, we only deal with the case of fixed TBox (in the terminology of [6], this corresponds to keep the TBox *protected*). It is interesting to observe that the specific DL considered in [6] is *DL-Lite*$_{\mathcal{FR}}$, and for this logic, exactly one KB accomplishes the insertion of a set of ABox assertions minimally. It follows that for instance-level insertion, their bold semantics coincides with ours. On the other hand, the presence of identification assertions in *DL-Lite*$_{A,id}$ changes the picture considerably, since with such assertions in the TBox, many KBs may exist accomplishing the insertion minimally. In this case, the two approaches are indeed different. Finally, [6] proposes a variant of the bold semantics, called *careful semantics*, for instance-level insertion in *DL-Lite*$_{\mathcal{FR}}$. Intuitively, such a semantics aims at disregarding knowledge that is entailed neither by the original KB, nor by the set of newly asserted facts. Although such principle is interesting, we believe that the careful semantics is too drastic, as it tends to eliminate too much information from the original KB.

Finally, we point out that, to our knowledge, the evolution operator presented in this work is the first tractable evolution operator based on the WIDTIO principle.

4 Deletion in *DL-Lite*$_{A,id}$

We study deletion under the assumption that the DL language $\mathcal{L}$ is *DL-Lite*$_{A,id}$. Thus, in this section, we implicitly refer to a *DL-Lite*$_{A,id}$-KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, and we address the problem of changing $\mathcal{K}$ with the deletion of a finite set F of ABox assertions. We assume that both $\langle \mathcal{T}, \mathcal{A} \rangle$ and $\langle \mathcal{T}, F \rangle$ are satisfiable.

We first consider the case where the set F is constituted by just one assertion f. By exploiting Theorem 1, it is easy to conclude that there is exactly one KB accomplishing the deletion of $\{f\}$ from a given KB.

Theorem 2. *Let f be an ABox assertion. Up to logical equivalence, there is exactly one KB of the form $\langle \mathcal{T}, \mathcal{A}' \rangle$ that accomplishes the deletion of $\{f\}$ from $\langle \mathcal{T}, \mathcal{A} \rangle$ minimally, and such KB can be computed in polynomial time with respect to $|\mathcal{T}|$ and $|\mathcal{A}|$.*

Let us now consider the case of arbitrary F, i.e., the case where $F = \{f_1, \ldots, f_m\}$, for $m \geq 0$. Suppose that, for every $1 \leq i \leq m$, $\mathcal{A}_i$ accomplishes the deletion of $\{f_i\}$ from $\langle \mathcal{T}, \mathcal{A} \rangle$ minimally. One might wonder whether the set $\Gamma_1 = \{\langle \mathcal{T}, \mathcal{A}_j \rangle \mid A_j$ accomplishes the deletion of F minimally from $\langle \mathcal{T}, \mathcal{A} \rangle\}$ coincides (modulo logical equivalence) with $\Gamma_2 = \{\langle \mathcal{T}, \mathcal{A}_1 \rangle, \ldots \langle \mathcal{T}, \mathcal{A}_m \rangle\}$. The next theorem tells us that one direction is indeed valid: for each KB $\mathcal{K}_1 \in \Gamma_1$ there exists a KB $\mathcal{K}_2 \in \Gamma_2$ such that $Mod(\mathcal{K}_1) = Mod(\mathcal{K}_2)$.

Theorem 3. *If $\langle \mathcal{T}, \mathcal{A}' \rangle$ accomplishes the deletion of $\{f_1, \ldots, f_m\}$ from $\langle \mathcal{T}, \mathcal{A} \rangle$ minimally, then there exists $i \in \{1..m\}$ such that $\langle \mathcal{T}, \mathcal{A}' \rangle$ accomplishes the deletion of $\{f_i\}$ from $\langle \mathcal{T}, \mathcal{A} \rangle$ minimally.*

However, the following example shows that the other direction does not hold: there may exist a $\mathcal{K}_2 \in \Gamma_2$ that is not logically equivalent to any $\mathcal{K}_1 \in \Gamma_1$.

Example 3. Let $\mathcal{T} = \{B \sqsubseteq C, C \sqsubseteq D, E \sqsubseteq D\}$, $\mathcal{A} = \{B(a), E(a)\}$, and $F = \{C(a), D(a)\}$. It is easy to see that the deletion of $D(a)$ from $\langle \mathcal{T}, \mathcal{A} \rangle$ is accomplished minimally by $\langle \mathcal{T}, \emptyset \rangle$, while the deletion of $C(a)$ from $\langle \mathcal{T}, \mathcal{A} \rangle$ is accomplished minimally by $\langle \mathcal{T}, \{E(a)\} \rangle$. Therefore, in this case, we have $\Gamma_2 = \{\langle \mathcal{T}, \emptyset \rangle, \langle \mathcal{T}, \{E(a)\} \rangle\}$. Also, one can verify that $\langle \mathcal{T}, \{E(a)\} \rangle$ is the only (up to logical equivalence) KB accomplishing the deletion of F minimally, i.e., $\Gamma_1 = \{\langle \mathcal{T}, \{E(a)\}\rangle\}$. Thus, there is a KB in Γ_2, namely $\langle \mathcal{T}, \emptyset\rangle$, that is not logically equivalent to any KB in Γ_1. $\Box$

The next theorem characterizes when a given $\langle \mathcal{T}, \mathcal{A}_i\rangle \in \Gamma_2$ accomplishes the deletion of F minimally.

Theorem 4. *Let $F = \{f_1, \ldots, f_m\}$, and, for every $1 \leq i \leq m$, let $\langle \mathcal{T}, \mathcal{A}_i\rangle$ accomplish the deletion of $\{f_i\}$ from $\langle \mathcal{T}, \mathcal{A}\rangle$ minimally. Then, $\langle \mathcal{T}, \mathcal{A}_j\rangle$, where $j \in \{1..m\}$, accomplishes the deletion of F from $\langle \mathcal{T}, \mathcal{A}\rangle$ minimally if and only if there is no $h \in \{1..m\}$ such that $h \neq j$, and $\langle \mathcal{T}, \{f_h\}\rangle \models f_j$.*

By exploiting Theorems 2, 3, and 4, we can directly prove that $\mathcal{K} \ominus^{\mathcal{T}}_{\cap} F$ can be computed by the algorithm ComputeDeletion below. It is easy to see that the time complexity of the algorithm is $O(|\mathcal{T}|^2 \times |F|^2 + |\mathcal{A}|^2)$.

Input: a satisfiable $\mathit{DL\text{-}Lite}_{A,id}$ KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A}\rangle$, a finite set of ABox assertions F such that $\langle \mathcal{T}, F\rangle$ is satisfiable
Output: a $\mathit{DL\text{-}Lite}_{A,id}$ KB
begin
  $F' \leftarrow F$;
  foreach $f_i \in F'$ and $f_j \in F$ such that $i \neq j$ **do**
    if $\langle \mathcal{T}, \{f_j\}\rangle \models f_i$ **then** $F' \leftarrow F' \setminus \{f_i\}$;
  return $\langle \mathcal{T}, \mathsf{cl}_{\mathcal{T}}(\mathcal{A}) \setminus \{\alpha \in \mathsf{Subsumee}_{\mathcal{K}}(f) \mid f \in F'\}\rangle$;
end

Algorithm 1: *ComputeDeletion*$(\langle \mathcal{T}, \mathcal{A}\rangle, F)$

Theorem 5. *ComputeDeletion$(\langle \mathcal{T}, \mathcal{A}\rangle, F)$ terminates, and computes $\langle \mathcal{T}, \mathcal{A}\rangle \ominus^{\mathcal{T}}_{\cap} F$ in polynomial time with respect to $|\mathcal{T}|$, $|\mathcal{A}|$ and $|F|$.*

5 Insertion in *DL-Lite*$_{A,id}$

In this section, we refer to a $\mathit{DL\text{-}Lite}_{A,id}$-KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A}\rangle$, and address the problem of changing $\mathcal{K}$ with the insertion of a finite set F of ABox assertions. As in the previous section, we assume that both $\langle \mathcal{T}, \mathcal{A}\rangle$ and $\langle \mathcal{T}, F\rangle$ are satisfiable.

Theorem 1 tells us that, in principle, we can compute the KB resulting from the insertion of F into $\langle \mathcal{T}, \mathcal{A}\rangle$ by building all maximal subsets of $\mathcal{A}$ which are $\mathcal{T}$-consistent with F, and then computing their intersection. The main problem to be faced with this method is that, depending on the DL used, there can be an exponential number of maximal subsets $\mathcal{A}'$ of $\mathsf{cl}_{\mathcal{T}}(\mathcal{A})$ such that $\mathcal{A}' \cup \{f\}$ is $\mathcal{T}$-consistent[5].In particular, in $\mathit{DL\text{-}Lite}_{A,id}$, building all maximal subsets of $\mathcal{A}$ which are $\mathcal{T}$-consistent with F, and then computing their intersection is computationally costly. Fortunately, we show in the following that $\mathcal{K} \oplus^{\mathcal{T}}_{\cap} F$ can be computed without computing all maximal consistent subsets of $\mathcal{A}$ with F.

[5] Note that this cannot happen in those DLs of the *DL-Lite* family which do not admit the use of identification assertions (such as the DL studied in [6]).

To describe our method, we need some preliminary notions. A set V of ABox assertions is called a *$\mathcal{T}$-violation set for* $t \in \mathcal{T} \setminus \mathcal{T}^+$ if $\langle \mathcal{T}^+ \cup \{t\}, V \rangle$ is unsatisfiable, while for every proper subset V' of V, $\langle \mathcal{T}^+ \cup \{t\}, V' \rangle$ is satisfiable. Any set V of ABox assertions that is a $\mathcal{T}$-violation set for a $t \in \mathcal{T} \setminus \mathcal{T}^+$ is simply called a *$\mathcal{T}$-violation set*.

We know from Theorem 1 that the ABox $\mathcal{A}'$ accomplishes the insertion of F from $\langle \mathcal{T}, \mathcal{A} \rangle$ minimally if and only if $\mathsf{cl}_\mathcal{T}(\mathcal{A}') = \mathcal{A}'' \cup \mathsf{cl}_\mathcal{T}(F)$, where $\mathcal{A}''$ is a maximal $\mathcal{T}$-closed subset of $\mathsf{cl}_\mathcal{T}(\mathcal{A})$ such that $\mathcal{A}'' \cup F$ is $\mathcal{T}$-consistent. Since we must compute the intersection of all such ABoxes $\mathcal{A}'$, it is sufficient to compute those assertions in $\mathsf{cl}_\mathcal{T}(\mathcal{A})$ that are not in the intersection, and remove them from $\mathsf{cl}_\mathcal{T}(\mathcal{A}) \cup \mathsf{cl}_\mathcal{T}(\mathcal{F})$. All the assertions in $\mathsf{cl}_\mathcal{T}(F)$ are obviously in the intersection of the ABoxes $\mathcal{A}'$. As for the ABox assertions in $\mathsf{cl}_\mathcal{T}(\mathcal{A}) \setminus \mathsf{cl}_\mathcal{T}(F)$, it is easy to see that one such assertion α is not in the intersection of the ABoxes $\mathcal{A}'$ if and only if there exists a maximal subset Σ of $\mathsf{cl}_\mathcal{T}(\mathcal{A})$ such that $\Sigma \cup F$ is $\mathcal{T}$-consistent, and Σ does not contain α.

Taking into account the above observation, the next theorem is the key to our solution.

Theorem 6. *Let α be an assertion in $\mathsf{cl}_\mathcal{T}(\mathcal{A}) \setminus \mathsf{cl}_\mathcal{T}(F)$. There exists a maximal subset Σ of $\mathsf{cl}_\mathcal{T}(\mathcal{A})$ such that $\Sigma \cup F$ is $\mathcal{T}$-consistent, and Σ does not contain α if and only if there is a $\mathcal{T}$-violation set V in $\mathsf{cl}_\mathcal{T}(\mathcal{A}) \cup \mathsf{cl}_\mathcal{T}(F)$ such that $\alpha \in V$, and $F \cup (V \setminus \{\alpha\})$ is $\mathcal{T}$-consistent.*

Theorem 6 suggests immediately the algorithm $\mathsf{ComputeInsertion}$ below for computing $\mathcal{K} \oplus_\cap^\mathcal{T} F$.

Input: a satisfiable *DL-Lite*$_{A,id}$ KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, a finite set of ABox assertions F such that $\langle \mathcal{T}, F \rangle$ is satisfiable
Output: a *DL-Lite*$_{A,id}$ KB.
begin
  $F' = \emptyset$;
  foreach $\alpha \in \mathsf{cl}_\mathcal{T}(\mathcal{A}) \setminus \mathsf{cl}_\mathcal{T}(F)$ **do**
    if $\exists$ a $\mathcal{T}$-violation set V in $\mathsf{cl}_\mathcal{T}(\mathcal{A}) \cup \mathsf{cl}_\mathcal{T}(F)$ s.t. $\alpha \in V$ and $\langle \mathcal{T}, F \cup (V \setminus \{\alpha\}) \rangle$ is satisfiable
    then $F' \leftarrow F' \cup \{\alpha\}$;
  return $\langle \mathcal{T}, F \cup \mathsf{cl}_\mathcal{T}(\mathcal{A}) \setminus F' \rangle$;
end

Algorithm 2: *ComputeInsertion*$(\langle \mathcal{T}, \mathcal{A} \rangle, \mathcal{F})$

Algorithm $\mathsf{ComputeInsertion}$ requires to compute all $\mathcal{T}$-violation sets in $\mathsf{cl}_\mathcal{T}(\mathcal{A}) \cup \mathsf{cl}_\mathcal{T}(F)$. It can be shown that this can be done by computing the results of suitable conjunctive queries posed to $\mathsf{cl}_\mathcal{T}(\mathcal{A}) \cup \mathsf{cl}_\mathcal{T}(F)$. Such queries are built out of the negative inclusion assertions and the identification assertions $\mathcal{T}_{id}$ in $\mathcal{T}$, and essentially look for tuples that satisfy the negation of such assertions. From this observation, one can derive the following theorem.

Theorem 7. *ComputeInsertion*$(\langle \mathcal{T}, \mathcal{A} \rangle, \mathcal{F})$ *terminates, and computes $\langle \mathcal{T}, \mathcal{A} \rangle \oplus_\cap^\mathcal{T} F$ in polynomial time with respect to $|\mathcal{T} \setminus \mathcal{T}_{id}|$, $|\mathcal{A}|$, and $|F|$, and in NP with respect to $|\mathcal{T}_{id}|$.*

6 Conclusions

We plan to continue our work along several directions. First, we aim at extending our approach to the problem of evolution of the whole KB, as opposed to the ABox only. Also, we will add the notion of protected part to our approach, to model situations where one wants to prevent changes on specific parts of the KB when applying insertions or deletions. Finally, we aim at studying the case where the KB contains other kinds of constraints, so as to capture the scenario where updates are expressed on a conceptual model used as a global schema in a data integration system [3]. In this context, one of the major challenges is to deal with the problem of pushing the updates to the data sources.

References

1. Marcelo Arenas, Leopoldo E. Bertossi, and Jan Chomicki. Consistent query answers in inconsistent databases. In *Proc. of PODS'99*, pages 68–79, 1999.
2. Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyaschev. The *DL-Lite* family and relations. *J. of Artificial Intelligence Research*, 36:1–69, 2009.
3. Andrea Calì, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini. Accessing Data Integration Systems through Conceptual Schemas. *International Conference on Conceptual Modeling*, LNCS 2224, 270–284, ER 2001.
4. Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *J. of Automated Reasoning*, 39(3):385–429, 2007.
5. Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Path-based identification constraints in description logics. In *Proc. of KR 2008*, pages 231–241, 2008.
6. Diego Calvanese, Evgeny Kharlamov, Werner Nutt, and Dmitriy Zheleznyakov. Evolution of *DL-Lite* knowledge bases. In *Proc. of ISWC 2010*, 2010.
7. Giuseppe De Giacomo, Maurizio Lenzerini, Antonella Poggi, and Riccardo Rosati. On instance-level update and erasure in description logic ontologies. *J. of Logic and Computation, Special Issue on Ontology Dynamics*, 19(5):745–770, 2009.
8. Ronald Fagin, Jeffrey D. Ullman, and Moshe Y. Vardi. On the semantics of updates in databases. In *Proc. of PODS'83*, pages 352–365, 1983.
9. Giorgos Flouris, Dimitris Manakanatas, Haridimos Kondylakis, Dimitris Plexousakis, and Grigoris Antoniou. Ontology change: Classification and survey. *Knowledge Engineering Review*, 23(2):117–152, 2008.
10. Hirofumi Katsuno and Alberto Mendelzon. On the difference between updating a knowledge base and revising it. In *Proc. of KR'91*, pages 387–394, 1991.
11. Hector J. Levesque. Foundations of a functional approach to knowledge representation. *Artificial Intelligence*, 23:155–212, 1984.
12. H. Liu, C. Lutz, M. Milicic, and F. Wolter. Updating description logic ABoxes. In *Proc. of KR 2006*, pages 46–56, 2006.
13. Zhe Wang, Kewen Wang, and Rodney W. Topor. A new approach to knowledge base revision in *DL-Lite*. In *Proc. of AAAI 2010*. AAAI Press, 2010.
14. Marianne Winslett. *Updating Logical Databases.* Cambridge University Press, 1990.