

Revision of DL-Lite Knowledge Bases

Zhe Wang, Kewen Wang, and Rodney Topor

Griffith University, Australia

Abstract. We address the revision problem for knowledge bases (KBs) in Description Logics (DLs). This problem has received much attention in the ontology management and DL communities, but the existing proposals are restricted in several ways. In this paper we develop a formal framework for revision of DL-Lite KBs, using techniques that are analogous to those for model-based revision in propositional logic. However, unlike propositional logic, a DL-Lite KB can have infinitely many models, which makes it hard to define and compute revision in terms of models. For this reason, we first develop an alternative semantic characterization for DL-Lite by introducing the concept of a feature and then define a specific revision operator for DL-Lite KBs based on features (instead of models). In contrast to previous approaches, we tackle the problem of revision between KBs, and the result of revision is always an unique DL-Lite KB. We also present an algorithm for computing KB revisions in DL-Lite.

1 Introduction

An ontology is a formal description of a common vocabulary for a domain of interest and the relationships between terms built from the vocabulary. Ontologies have been applied in a wide range of practical domains such as medical informatics, bio-informatics and, more recently, the Semantic Web. Description Logics (DLs) have been used as underlying formalisms of most ontology languages recently. A DL-based ontology is usually expressed as a DL-knowledge base (KB), which consists of a TBox (terminological box) and an ABox (set of assertions).

DL-Lite [1, 3] is specifically designed as a family of lightweight DLs with relatively restricted expressive power but efficient ontological reasoning algorithms. In particular, logics in the DL-Lite family have polynomial time computational complexity with respect to standard reasoning tasks, and LogSpace data complexity with respect to complex query answering. With reasonable expressive power and efficient reasoning support, especially on answering complex queries over large amounts of data, the DL-Lite family has proved to be an appealing formalism for ontology development.

A challenge in ontology maintenance is that ontologies are not static but may evolve over time. Because of limitations in the initial design, changes in users' requirements, need to reflect changes in the real world, and so on, there is a need to revise ontologies in accordance with the new knowledge. Recently, intensive interest has been shown on revision/update in DLs. Earlier approaches concentrated on adapting revision postulates for propositional logic to DLs but no specific revision operators were provided, e.g. [4, 5]. Recently, there have been several attempts to define specific revision/update operators for DLs including [7, 9, 11–16]. In particular, an update operator for DL-Lite

ABoxes is defined in [7]. In fact, most specific revision/update operators for DLs can deal only with ABoxes, like [7, 9, 11], or only with TBoxes, like [13, 14]. As shown in [8, 14, 16], the *kernel revision* operator for propositional logic can be adapted to most DLs in a straightforward way, but the result of revision is not a unique KB in most cases and thus arbitrary choices by the user are required.

It is worth mentioning that update is traditionally distinguished from revision in that update addresses changes of the actual state of the world (e.g., that resulting from an action) whereas revision addresses the incorporation of new knowledge about the world. In this paper, we focus on the revision problem.

In propositional logic, the area of *belief revision* [6] is well developed, with numerous proposals and techniques for knowledge base revision. Among them, Satoh's revision operator $\varphi *_s \mu$ [17] is well known. In his approach, the set of models of a revised knowledge base K' are those models of K' that are 'closest' to those of the initial knowledge base K , where the distance between two models is based on their symmetric difference.

However, such model-based approaches cannot be applied directly to KB revision in DLs. The difficulties in defining and computing model-based revision of KBs in DLs are that: (1) unlike a propositional theory, a KB in a DL may have infinitely many models; (2) a model of a KB can be infinite; and (3) given a collection M of models, there may not exist a unique KB \mathcal{K} such that M is exactly the set of all models of \mathcal{K} .

In this paper, we aim to define a rational revision operator for DL-Lite_{bool} ontologies in a way analogous to the model-based approaches in propositional logic. In order to achieve this goal, we first define the concept of *features* for DL-Lite_{bool}, which precisely capture most important semantic properties of DL-Lite_{bool} KBs, and are always finite. We adapt the techniques of model-based revision in propositional logic to the revision of DL-Lite_{bool} KBs, and define a specific revision operator based on the distance between features. We also show that the revision operator possesses desirable properties, and can be computed through syntactic methods.

Although our approach is based on DL-Lite_{bool} [1], we remark that the belief revision method we propose also applies to other logics in DL-Lite. In contrast to previous approaches, we address the problem of defining and computing revision of a DL-Lite KB by another KB. Also, the result of our revision is always a unique KB in DL-Lite.

Due to space limitation, proofs are omitted in this paper but a longer version of this paper with complete proofs can be found at http://www.cit.gu.edu.au/~kewen/revision_long.pdf.

2 DL-Lite Family

A *signature* is a finite set $\mathcal{S} = \mathcal{S}_C \cup \mathcal{S}_R \cup \mathcal{S}_O \cup \mathcal{S}_N$ where \mathcal{S}_C is the set of atomic concepts, \mathcal{S}_R is the set of atomic roles, \mathcal{S}_O is the set of individual names (or, objects) and \mathcal{S}_N is the set of natural numbers in \mathcal{S} . We assume 1 is always in \mathcal{S}_N . \top and \perp will not be considered as atomic concepts or atomic roles. Formally, given a signature \mathcal{S} , a

DL-Lite_{bool} language \mathcal{L} has the following syntax:

$$\begin{aligned} R &\longleftarrow P \mid P^- \\ B &\longleftarrow \top \mid \perp \mid A \mid \geq n R \\ C &\longleftarrow B \mid \neg C \mid C_1 \sqcap C_2 \end{aligned}$$

where $n \geq 1$, $A \in \mathcal{S}_C$ and $P \in \mathcal{S}_R$. B is called a *basic concept* and C is called a *general concept*. We write $\exists R$ as a shorthand for $\geq 1 R$, and let $R^+ = P$, where $P \in \mathcal{S}_R$, whenever $R = P$ or $R = P^-$.

A *literal* is a basic concept or its negation. The set of all literals on \mathcal{S} is denoted as $Lit_{\mathcal{S}}$, and the set of all basic concepts is denoted as $Lit_{\mathcal{S}}^+$.

A DL-Lite_{bool} TBox \mathcal{T} is a finite set of concept inclusions of the form $C_1 \sqsubseteq C_2$, where C_1 and C_2 are general concepts. A DL-Lite_{bool} ABox \mathcal{A} is a finite set of membership assertions of the form $C(a), R(a, b)$, where a, b are individual names. We call $C(a)$ a concept assertion and $R(a, b)$ a role assertion. A DL-Lite_{bool} knowledge base (KB) is a pair $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$.

The semantics of a DL-Lite KB is given by interpretations. An interpretation \mathcal{I} is a pair $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty set called the *domain* and $\cdot^{\mathcal{I}}$ is an interpretation function which associates each atomic concept A with a subset $A^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$, each atomic role P with a binary relation $P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and each individual name a with an element $a^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$ such that $a^{\mathcal{I}} \neq b^{\mathcal{I}}$ for each pair of individual names a, b (unique name assumption).

The interpretation function $\cdot^{\mathcal{I}}$ can be extended to general concept descriptions:

$$\begin{aligned} (P^-)^{\mathcal{I}} &= \{(a^{\mathcal{I}}, b^{\mathcal{I}}) \mid (b^{\mathcal{I}}, a^{\mathcal{I}}) \in P^{\mathcal{I}}\} \\ (\geq n R)^{\mathcal{I}} &= \{a^{\mathcal{I}} \mid |\{b^{\mathcal{I}} \mid (a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}\}| \geq n\} \\ (\neg C)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\ (C_1 \sqcap C_2)^{\mathcal{I}} &= C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}} \end{aligned}$$

An interpretation \mathcal{I} is a model of inclusion $C_1 \sqsubseteq C_2$ if $C_1^{\mathcal{I}} \subseteq C_2^{\mathcal{I}}$; \mathcal{I} is a model of an assertion $C(a)$ if $a^{\mathcal{I}} \in C^{\mathcal{I}}$; \mathcal{I} is a model of an assertion $R(a, b)$ if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$. \mathcal{I} is a model of a TBox \mathcal{T} (or ABox \mathcal{A}) if \mathcal{I} is a model of each inclusion in \mathcal{T} (resp., each assertion in \mathcal{A}). Two inclusions (or resp., assertions, TBoxes, ABoxes) are said to be *equivalent* if they have exactly the same models.

\mathcal{I} is a model of a KB $\langle \mathcal{T}, \mathcal{A} \rangle$, if \mathcal{I} is a model of both \mathcal{T} and \mathcal{A} . A KB \mathcal{K} is consistent if it has at least one model. Two KBs $\mathcal{K}_1, \mathcal{K}_2$ that have the same models are said to be equivalent, denoted $\mathcal{K}_1 \equiv \mathcal{K}_2$. A KB \mathcal{K} logically implies an inclusion or assertion α , denoted $\mathcal{K} \models \alpha$, if all models of \mathcal{K} are also models of α . We will use $\text{Mod}(\mathcal{K})$ to denote the set of models of a KB \mathcal{K} . $\text{Sig}(\mathcal{K})$ is the signature of \mathcal{K} .

As in propositional logic, a DL-Lite_{bool} concept C can always be transformed into an equivalent disjunction of conjunctions of literal concepts (DNF).

3 Features in DL-Lite_{bool}

In this section, we introduce the concept of features in DL-Lite_{bool}, which provides an alternative semantic characterization for DL-Lite_{bool}. An advantage of semantic features over models is that the number of all features for a DL-Lite_{bool} knowledge base

is finite and each feature is finite as we will see later. These finiteness properties make it possible to recast key approaches to revision for classical propositional logic into DL-Lite_{bool}.

Features for DL-Lite_{bool} are based on the notion of *types* defined in [10].

In the following sections, we assume \mathcal{S} is a fixed signature. Recall that a \mathcal{S} -literal (or, literal) is a basic concept on \mathcal{S} or its negation. $Lit_{\mathcal{S}}$ is the set of all \mathcal{S} -literals.

A \mathcal{S} -type is a subset τ of $Lit_{\mathcal{S}}$ containing \top and satisfying the following conditions:

- for every \mathcal{S} -literal L , $L \in \tau$ iff $\neg L \notin \tau$;
- for any $m, n \in \mathcal{S}_N$ with $m < n$, $\geq n R \in \tau$ implies $\geq m R \in \tau$;
- for any $m, n \in \mathcal{S}_N$ with $m < n$, $\neg(\geq m R) \in \tau$ implies $\neg(\geq n R) \in \tau$.

A type τ can be equivalently represented by the conjunction of all literals in τ , which we also call a *type* when the context is clear and no confusion is caused. An important observation is that every general concept C over \mathcal{S} corresponds to a set $\text{Ts}(C)$ of \mathcal{S} -types, in light of the fact that C can be transformed into an equivalent disjunction of \mathcal{S} -types. For example, concept $A \sqcap (B \sqcup \exists P)$ can be transformed into $(A \sqcap B \sqcap \exists P) \sqcup (A \sqcap \neg B \sqcap \exists P) \sqcup (A \sqcap B \sqcap \neg \exists P)$.

Informally speaking, types correspond to propositional models whereas $\text{Ts}(C)$ correspond to the set of models of formula C . For simplicity, we assume that a type τ consists of only its basic concepts, following similar common practices for propositional models. For example, let $\mathcal{S}_C = \{A, B\}$, $\mathcal{S}_R = \{P\}$, and $\mathcal{S}_N = \{1, 3\}$. Then $\tau = \{A, \exists P, \geq 3P, \exists P^-\}$ is a type instead of $\tau = \{\top, A, \neg B, \exists P, \geq 3P, \exists P^-, \neg(\geq 3P^-)\}$.

We say a type τ satisfies a concept C if $\tau \in \text{Ts}(C)$, and τ satisfies concept inclusion $C_1 \sqsubseteq C_2$ if $\tau \in \text{Ts}(\neg C_1 \sqcup C_2)$. In this way, we define that τ satisfies a TBox \mathcal{T} if it satisfies each inclusion in \mathcal{T} .

However, in order to capture the membership relations in a KB, we need to extend the notion of types and thus define *Herbrand sets* in DL-Lite.

Definition 1. A \mathcal{S} -Herbrand set \mathcal{H} is a finite set of assertions of the form $B(a)$ or $P(a, b)$, where $a, b \in \mathcal{S}_O$, $B \in Lit_{\mathcal{S}}^+$, $P \in \mathcal{S}_R$, satisfying the following conditions

1. for each $a \in \mathcal{S}_O$, if $B_1(a), \dots, B_k(a)$ are all the concept assertions about a in \mathcal{H} , then the set $\{B_1, \dots, B_k\}$ forms an \mathcal{S} -type.
2. for each $P \in \mathcal{S}_R$, if $P(a, b_i)$ ($1 \leq i \leq n$) are all the assertions in \mathcal{H} of such form, then for any $m \in \mathcal{S}_N$ with $m \leq n$, $(\geq m P)(a)$ is in \mathcal{H} ;
3. for each $P \in \mathcal{S}_R$, if $P(b_i, a)$ ($1 \leq i \leq n$) are all the assertions in \mathcal{H} of such form, then for any $m \in \mathcal{S}_N$ with $m \leq n$, $(\geq m P^-)(a)$ is in \mathcal{H} .

Informally, $B(a) \notin \mathcal{H}$ with $B \in Lit_{\mathcal{S}}^+$ means that $\neg B(a)$ holds in \mathcal{H} , and $P(a, b) \notin \mathcal{H}$ with $P \in \mathcal{S}_R$ means that a, b are not related by P in \mathcal{H} . Thus, Herbrand sets are different from ABoxes by adopting a close world assumption. It is not hard to see that, conditions 2 and 3 in Definition 1 preserve consistency of a Herbrand set. And condition 2 in Definition 1 requires that $\geq n P \in \mathcal{H}$ implies $\geq m P \in \mathcal{H}$ for all $m \in \mathcal{S}_N$ with $m < n$. These conditions are introduced mainly for simplifying the definition of our revision operator in the following section.

For simplicity, we denote the group of all the concept assertions $B_1(a), \dots, B_k(a)$ about a in \mathcal{H} as $\tau(a)$, where $\tau = \{B_1, \dots, B_k\}$ is a type. We say $\tau(a)$ is in \mathcal{H} if $B_i(a)$ is in \mathcal{H} for all $1 \leq i \leq k$.

Herbrand sets can be used to capture membership relations in a KB. We say a Herbrand set \mathcal{H} satisfies a membership assertion $C(a)$ if $\tau(a)$ is in \mathcal{H} and $\tau \in \text{Ts}(C)$; and \mathcal{H} satisfies assertion $P(a, b)$ (resp., $P^-(b, a)$) if $P(a, b)$ (resp., $P^-(b, a)$) is in \mathcal{H} . \mathcal{H} satisfies an ABox \mathcal{A} if \mathcal{H} satisfies all the assertions in \mathcal{A} .

To provide an alternative characterization for reasoning in a KB, we could use pairs $\langle \tau, \mathcal{H} \rangle$, where τ is a type and \mathcal{H} is a Herbrand set, to replace standard interpretations, such that $\langle \tau, \mathcal{H} \rangle$ satisfies KB $\langle \mathcal{T}, \mathcal{A} \rangle$ if τ satisfies \mathcal{T} and \mathcal{H} satisfies \mathcal{A} . To this end, we want it to guarantee that \mathcal{K} is consistent iff there exists some pair $\langle \tau, \mathcal{H} \rangle$ satisfying \mathcal{K} . However, the following example shows that this approach does not work.

Example 1. Let $\mathcal{K} = \langle \{ \exists P^- \sqsubseteq \perp \}, \{ \exists P(a) \} \rangle$ be a DL-Lite_{bool} KB. Obviously, \mathcal{K} is inconsistent and thus has no model. However, let $\mathcal{S} = \{P, a, 1\}$ and $\tau = \{\exists P\}$, then $\langle \tau, \{\tau(a)\} \rangle$ satisfies \mathcal{K} .

The problem with using pairs $\langle \tau, \mathcal{H} \rangle$ as alternative semantic characterization for DL-Lite is that a single type τ is not sufficient to capture the semantic connection between the TBox and the ABox of a KB. Our investigation shows that it is plausible to use a pair of a set of types and a Herbrand set, as an alternative semantic characterization for a DL-Lite_{bool} KB. Using sets of types as semantic characterizations of DL-Lite TBoxes is also suggested in [10].

Thus, we introduce the definition of *features* as follows.

Definition 2. Given a signature \mathcal{S} , a feature on \mathcal{S} , or simply \mathcal{S} -feature, is defined as a pair $\mathcal{F} = \langle \Xi, \mathcal{H} \rangle$, where Ξ is a non-empty set of \mathcal{S} -types and \mathcal{H} a \mathcal{S} -Herbrand set, such that the following conditions are satisfied:

1. for each $P \in \mathcal{S}_R$, $\exists P \in \bigcup \Xi$ iff $\exists P^- \in \bigcup \Xi$;
2. for each $a \in \mathcal{S}_O$ and $\tau(a)$ in \mathcal{H} , s. t. τ is an \mathcal{S} -type, $\tau \in \Xi$.

Informally, the first condition in Definition 2 says that $\exists P$ is a satisfiable concept in \mathcal{F} if and only if $\exists P^-$ is also a satisfiable concept. And the second condition requires that the interpretation of the ABox must be in accordance to the TBox.

We use a feature $\mathcal{F} = \langle \Xi, \mathcal{H} \rangle$ as an alternative for an interpretation in DL-Lite_{bool}:

- \mathcal{F} satisfies an inclusion $C_1 \sqsubseteq C_2$ over \mathcal{S} , if $\Xi \subseteq \text{Ts}(\neg C_1 \sqcup C_2)$.
- \mathcal{F} satisfies an assertion $C(a)$ over \mathcal{S} , if $\tau(a)$ is in \mathcal{H} and $\tau \in \text{Ts}(C)$.
- \mathcal{F} satisfies assertion $P(a, b)$ (resp., $P^-(b, a)$) over \mathcal{S} , if $P(a, b)$ is in \mathcal{H} .

We say \mathcal{F} is a *model feature* of DL-Lite_{bool} KB \mathcal{K} if \mathcal{F} satisfies each inclusion and each assertion in \mathcal{K} . $\Omega_{\mathcal{K}}$ denotes the set of all model features of \mathcal{K} .

From the definition of features, a model feature is finite and the number of model features of a KB is also finite.

Example 2. Consider the KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T} = \{ A \sqsubseteq \exists P, B \sqsubseteq \exists P, \exists P^- \sqsubseteq B, A \sqcap B \sqsubseteq \perp, \geq 2 P^- \sqsubseteq \perp \}$ and $\mathcal{A} = \{ A(a), P(a, b) \}$. It is shown in [2] that \mathcal{K} is a KB having no finite model.

An infinite model \mathcal{I} of \mathcal{K} is defined as follows: $\Delta^{\mathcal{I}} = \{d_a, d_b, d_1, d_2, d_3 \dots\}$, $a^{\mathcal{I}} = d_a$ and $b^{\mathcal{I}} = d_b$; the concept A is interpreted as a singleton $\{d_a\}$ and B as $\{d_b, d_1, d_2, d_3 \dots\}$; and role P is interpreted as $\{(d_a, d_b), (d_b, d_1), (d_1, d_2), \dots, (d_i, d_{i+1}), \dots\}$.

Take $\mathcal{S} = \text{Sig}(\mathcal{K}) = \{A, B, P, 1, 2, a, b\}$. The (finite) model feature of \mathcal{K} that corresponds to \mathcal{I} is $\mathcal{F} = \langle \Xi, \mathcal{H} \rangle$ where the Herbrand set $\mathcal{H} = \{\tau_1(a), \tau_2(b), P(a, b)\}$ and $\Xi = \{\tau_1, \tau_2\}$ with $\tau_1 = \{A, \exists P\}$ and $\tau_2 = \{B, \exists P, \exists P^-\}$.

Given an inclusion or assertion α over $\mathcal{S} = \text{Sig}(\mathcal{K})$, we define $\mathcal{K} \models_f \alpha$ if for each $\mathcal{F} \in \Omega_{\mathcal{K}}$, \mathcal{F} satisfies α . Given two KBs \mathcal{K}_1 and \mathcal{K}_2 , let $\mathcal{S} = \text{Sig}(\mathcal{K}_1 \cup \mathcal{K}_2)$, define $\mathcal{K}_1 \models_f \mathcal{K}_2$ if $\Omega_{\mathcal{K}_1} \subseteq \Omega_{\mathcal{K}_2}$; and $\mathcal{K}_1 \equiv_f \mathcal{K}_2$ if $\Omega_{\mathcal{K}_1} = \Omega_{\mathcal{K}_2}$.

The following two results show that model features do capture the semantic properties of DL-Lite_{bool} KBs.

Proposition 1. *Let \mathcal{K} be a DL-Lite_{bool} KB and $\mathcal{S} = \text{Sig}(\mathcal{K})$. Then*

1. \mathcal{K} is consistent iff \mathcal{K} has a model feature;
2. for any concept inclusion $C_1 \sqsubseteq C_2$ over \mathcal{S} , $\mathcal{K} \models (C_1 \sqsubseteq C_2)$ iff $\mathcal{K} \models_f (C_1 \sqsubseteq C_2)$;
3. for any membership assertion $C(a)$ (resp., $R(a, b)$) over \mathcal{S} , $\mathcal{K} \models C(a)$ (resp., $\mathcal{K} \models R(a, b)$) iff $\mathcal{K} \models_f C(a)$ (resp., $\mathcal{K} \models_f R(a, b)$);

Proposition 2. *Let $\mathcal{K}_1, \mathcal{K}_2$ be two DL-Lite_{bool} KBs and $\mathcal{S} = \text{Sig}(\mathcal{K}_1 \cup \mathcal{K}_2)$. Then*

1. $\mathcal{K}_1 \models \mathcal{K}_2$ iff $\mathcal{K}_1 \models_f \mathcal{K}_2$;
2. $\mathcal{K}_1 \equiv \mathcal{K}_2$ iff $\mathcal{K}_1 \equiv_f \mathcal{K}_2$.

In summary, the advantages of model features over standard models can be described as follows. First, each model feature of a KB is finite in structure. Second, there are a finite number of model features for each KB. Third, each non-empty set of model features determines a unique KB up to KB equivalence. In particular, given a non-empty set Ω of \mathcal{S} -features, we can always construct a unique DL-Lite_{bool} KB \mathcal{K} over \mathcal{S} such that $\Omega \subseteq \Omega_{\mathcal{K}}$ and $\Omega_{\mathcal{K}}$ is minimal among all supersets $\Omega_{\mathcal{K}'}$ of Ω . In Section 5, we will introduce a method for constructing such a DL-Lite_{bool} KB from a set of features.

4 Feature-based Revision

In this section, we introduce a specific revision operator for DL-Lite KBs based on model features rather than models.

Before introducing the definition of our feature-based revision operator, we first extend the definition of symmetric difference Δ so that it can be used for model features.

Recall that $S_1 \Delta S_2 = (S_1 - S_2) \cup (S_2 - S_1)$ for any two sets S_1 and S_2 . Given two \mathcal{S} -features $\mathcal{F}_1 = \langle \Xi_1, \mathcal{H}_1 \rangle$ and $\mathcal{F}_2 = \langle \Xi_2, \mathcal{H}_2 \rangle$, the *distance* between \mathcal{F}_1 and \mathcal{F}_2 , denoted $\mathcal{F}_1 \Delta \mathcal{F}_2$, is a pair $\langle \Xi_1 \Delta \Xi_2, \mathcal{H}_1 \Delta \mathcal{H}_2 \rangle$. Note that $\mathcal{H}_1 \Delta \mathcal{H}_2$ may not be a Herbrand set anymore. For example, let $\mathcal{S} = \{P, a, b_1, b_2, b_3, 1, 2, 3\}$, $\mathcal{H}_1 = \{\exists P(a), P(a, b_1)\}$ and $\mathcal{H}_2 = \{\exists P(a), (\geq 2 P)(a), P(a, b_2), P(a, b_3)\}$. Then we get $\mathcal{H}_1 \Delta \mathcal{H}_2 = \{(\geq 2 P)(a), P(a, b_1), P(a, b_2), P(a, b_3)\}$, which is not a Herbrand set because it does not contain $\exists P(a)$ and $(\geq 3 P)(a)$.

To compare two distances, we could define $\mathcal{F}_1 \triangle \mathcal{F}_2 \subseteq_d \mathcal{F}_3 \triangle \mathcal{F}_4$ if $\Xi_1 \triangle \Xi_2 \subseteq_d \Xi_3 \triangle \Xi_4$ and $\mathcal{H}_1 \triangle \mathcal{H}_2 \subseteq_d \mathcal{H}_3 \triangle \mathcal{H}_4$, where $\mathcal{F}_i = \langle \Xi_i, \mathcal{H}_i \rangle$ for $i = 1, 2, 3, 4$; and $\mathcal{F}_1 \triangle \mathcal{F}_2 \subset_d \mathcal{F}_3 \triangle \mathcal{F}_4$ if $\mathcal{F}_1 \triangle \mathcal{F}_2 \subseteq_d \mathcal{F}_3 \triangle \mathcal{F}_4$ and $\mathcal{F}_3 \triangle \mathcal{F}_4 \not\subseteq_d \mathcal{F}_1 \triangle \mathcal{F}_2$. However, we will show in an example later that such a measure is too weak for KB revision and cannot preserve enough information. Instead, we set a preference on Herbrand sets over type sets: $\mathcal{F}_1 \triangle \mathcal{F}_2 \subset_d \mathcal{F}_3 \triangle \mathcal{F}_4$ iff

1. $\mathcal{H}_1 \triangle \mathcal{H}_2 \subset \mathcal{H}_3 \triangle \mathcal{H}_4$, or
2. $\mathcal{H}_1 \triangle \mathcal{H}_2 = \mathcal{H}_3 \triangle \mathcal{H}_4$ and $\Xi_1 \triangle \Xi_2 \subset \Xi_3 \triangle \Xi_4$.

Note that here we set a preference on Herbrand set over type set, such that assertions in the original ABox are prior to be preserved than inclusions in the TBox. The rationale behind this preference is that assertions represent more specific information than inclusions and thus from the view point of knowledge representation, should have higher priority.

According to our new semantic characterization for DL-Lite_{bool} KBs, the semantics of each KB \mathcal{K} is determined by the set $\Omega_{\mathcal{K}}$ of all model features of \mathcal{K} .

Given two KBs \mathcal{K} and \mathcal{K}' , to define our revision operator $\mathcal{K} \circ \mathcal{K}'$, we need to specify the subset of $\Omega_{\mathcal{K}'}$ that is closest to $\Omega_{\mathcal{K}}$.

Let Ω_1 and Ω_2 be two sets of features. Then we define

$$\sigma(\Omega_1, \Omega_2) = \{ \mathcal{F}_1 \in \Omega_1 \mid \text{there exists } \mathcal{F}_2 \in \Omega_2 \text{ s.t.} \\ (\mathcal{F}_1 \triangle \mathcal{F}_2) \not\subseteq_d (\mathcal{F}_1 \triangle \mathcal{F}_2) \text{ for all } \mathcal{F}_1' \in \Omega_1 \text{ and } \mathcal{F}_2' \in \Omega_2 \}$$

Now we are ready to define our revision operator in terms of model features.

Definition 3. Let $\mathcal{K}, \mathcal{K}'$ be two DL-Lite_{bool} KBs and $\mathcal{S} = \text{Sig}(\mathcal{K} \cup \mathcal{K}')$. The (feature-based) revision of \mathcal{K} by \mathcal{K}' is defined as the DL-Lite_{bool} KB $\mathcal{K} \circ \mathcal{K}'$ such that if $\Omega_{\mathcal{K}} = \emptyset$, then $\Omega_{\mathcal{K} \circ \mathcal{K}'} = \Omega_{\mathcal{K}'}$; otherwise

1. $\sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}}) \subseteq \Omega_{\mathcal{K} \circ \mathcal{K}'}$, and
2. $\Omega_{\mathcal{K} \circ \mathcal{K}'}$ is minimal in the sense that, for any DL-Lite_{bool} KB \mathcal{K}'' satisfying the above condition 1, $\Omega_{\mathcal{K} \circ \mathcal{K}'} \subseteq \Omega_{\mathcal{K}''}$.

Note that the definition is slightly different from Satoh's as $\sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}})$ may not correspond exactly to a DL-Lite_{bool} KB. By requiring $\Omega_{\mathcal{K} \circ \mathcal{K}'}$ to be the minimal superset of $\sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}})$, we define $\mathcal{K} \circ \mathcal{K}'$ in a conservative manner. Another option is to define $\Omega_{\mathcal{K} \circ \mathcal{K}'}$ to be a maximal subset of $\sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}})$. However, as $\Omega_{\mathcal{K} \circ \mathcal{K}'}$ can be very small if defined this way, there is a risk that unexpected logical consequences are introduced. Thus, it is more plausible to use supersets instead of subsets.

For any two DL-Lite_{bool} KBs $\mathcal{K}_1, \mathcal{K}_2$ that are both revisions of \mathcal{K} by \mathcal{K}' , we have $\Omega_{\mathcal{K}_1} = \Omega_{\mathcal{K}_2}$. By Proposition 2, this definition uniquely defines the result of $\mathcal{K} \circ \mathcal{K}'$ up to equivalence. We will show in Section 5 that $\mathcal{K} \circ \mathcal{K}'$ always exists.

Example 3. Consider a DL-Lite_{bool} KB

$$\mathcal{K} = \langle \{ \text{PhD} \sqsubseteq \text{Student}, \text{Student} \sqsubseteq \neg \exists \text{teaches} \}, \{ \text{PhD}(\text{Tom}) \} \rangle.$$

The TBox of \mathcal{K} specifies that PhD students are students, and students are not allowed to teach any courses, while the ABox states that *Tom* is a PhD student. Suppose PhD students are actually allowed to teach, and we want to revise \mathcal{K} with $\mathcal{K}' = \langle \{ PhD \sqsubseteq \exists teaches \}, \emptyset \rangle$.

Then $\mathcal{F}' = \langle \{\tau_1\}, \{\tau_1(Tom)\} \rangle$, where $\tau_1 = \{Student\}$ is a model feature of \mathcal{K}' . Take $\mathcal{F} = \langle \{\tau_1, \tau_2\}, \{\tau_2(Tom)\} \rangle$ where $\tau_2 = \{PhD, Student\}$. It can be verified that \mathcal{F} is a model feature of \mathcal{K} and it is closest to \mathcal{F}' .

The distance between \mathcal{F} and \mathcal{F}' is a minimal one among those between model features of \mathcal{K} and \mathcal{K}' . Thus, \mathcal{F}' is a model feature of $\mathcal{K} \circ \mathcal{K}'$.

In fact, we can show that $\mathcal{K} \circ \mathcal{K}'$ is $\langle \{ PhD \sqsubseteq Student, PhD \sqsubseteq \exists teaches, Student \sqcap \exists teaches \sqsubseteq PhD \}, \{ Student(Tom) \} \rangle$.

In Example 3, the new inclusion added causes concept *PhD* to be unsatisfiable, i. e., *PhD* must be interpreted as an empty set, and thus contradicts *PhD(Tom)*. The contradiction is resolved through revision by replacing $Student \sqsubseteq \neg \exists teaches$ with $Student \sqcap \exists teaches \sqsubseteq PhD$. Also, *PhD(Tom)* is weakened to be *Student(Tom)*, because $\neg \exists teaches(Tom)$ is also (implicitly) expressed in \mathcal{K} . However, if we treat type sets and Herbrand sets equally in measuring distances, as we can show, *Student(Tom)* will be lost from the result of revision.

The standard AGM postulates (R1) – (R6) for propositional belief revision have been adapted to DLs in [15]. However, the authors present the postulates in terms of models of KBs. In the following, we reformulate them using KB combinations and entailments, in a manner analogous to classical AGM postulates.

- (R1) $\mathcal{K} \circ \mathcal{K}' \models \mathcal{K}'$;
- (R2) if $\mathcal{K} \cup \mathcal{K}'$ is consistent, then $\mathcal{K} \circ \mathcal{K}' \equiv \mathcal{K} \cup \mathcal{K}'$;
- (R3) if \mathcal{K}' is consistent, then $\mathcal{K} \circ \mathcal{K}'$ is consistent;
- (R4) if $\mathcal{K}_1 \equiv \mathcal{K}_2$ and $\mathcal{K}'_1 \equiv \mathcal{K}'_2$, then $\mathcal{K}_1 \circ \mathcal{K}'_1 \equiv \mathcal{K}_2 \circ \mathcal{K}'_2$;
- (R5) if $\Omega_{\mathcal{K} \circ \mathcal{K}'} = \sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}})$, then $(\mathcal{K} \circ \mathcal{K}') \cup \mathcal{K}'' \models \mathcal{K} \circ (\mathcal{K}' \cup \mathcal{K}'')$;
- (R6) if $(\mathcal{K} \circ \mathcal{K}') \cup \mathcal{K}''$ is consistent, then $\mathcal{K} \circ (\mathcal{K}' \cup \mathcal{K}'') \models (\mathcal{K} \circ \mathcal{K}') \cup \mathcal{K}''$.

The following theorem states the first five postulates are satisfied by our revision operator.

Theorem 1. *The revision operator defined in Definition 3 satisfies postulates (R1) – (R5).*

Note that (R5) is conditional, and in this sense varies from its corresponding classical AGM postulate. We remark that the condition in (R5) cannot be removed. That is, $(\mathcal{K} \circ \mathcal{K}') \cup \mathcal{K}'' \models \mathcal{K} \circ (\mathcal{K}' \cup \mathcal{K}'')$ is not always satisfied. Also, (R6) is not always satisfied by our revision operator, as with Satoh's revision operator in propositional logic. This can be seen from the following example.

Example 4. Let $\mathcal{K} = \langle \{ A \sqsubseteq B, B \sqsubseteq C \}, \{ A(a) \} \rangle$, $\mathcal{K}' = \langle \{ B \sqsubseteq A, A \sqsubseteq \neg C \}, \{ (A \sqcup C)(a) \} \rangle$, and $\mathcal{K}'' = \langle \emptyset, \{ \neg B(a) \} \rangle$. Then $\mathcal{K} \circ \mathcal{K}' = \langle \{ A \sqsubseteq B, B \sqsubseteq A, A \sqsubseteq \neg C \}, \{ (A \sqcup C)(a) \} \rangle$, and after combined with \mathcal{K}'' , $(A \sqcup C)(a)$ in the ABox is replaced with $C(a)$. However, $\mathcal{K} \circ (\mathcal{K}' \cup \mathcal{K}'') = \langle \{ B \sqsubseteq C, B \sqsubseteq A, A \sqsubseteq \neg C \}, \{ (A \sqcup C)(a) \} \rangle$. That is, $(\mathcal{K} \circ \mathcal{K}') \cup \mathcal{K}'' \not\models \mathcal{K} \circ (\mathcal{K}' \cup \mathcal{K}'')$ and $\mathcal{K} \circ (\mathcal{K}' \cup \mathcal{K}'') \not\models (\mathcal{K} \circ \mathcal{K}') \cup \mathcal{K}''$.

5 Algorithm for Computing Revision

In this section, we investigate the problem of computing revision and as a result, present an algorithm that can compute the result of revision syntactically.

The revision operator is based on model features of KBs. In what follows, we introduce a method for computing all the model features of a DL-Lite_{bool} KB \mathcal{K} , by first computing the assembling of all the model features, which can be obtained directly from \mathcal{K} through syntactic transformations.

Given two features $\mathcal{F}_1 = \langle \Xi_1, \mathcal{H}_1 \rangle$ and $\mathcal{F}_2 = \langle \Xi_2, \mathcal{H}_2 \rangle$, we define the *assembling* of \mathcal{F}_1 and \mathcal{F}_2 as $\mathcal{F}_1 \oplus \mathcal{F}_2 = \langle \Xi_1 \cup \Xi_2, \mathcal{H}_1 \oplus \mathcal{H}_2 \rangle$, where $\mathcal{H}_1 \oplus \mathcal{H}_2$ is called a *multiple Herbrand set*, consisting of:

- $\{\tau_1, \tau_2\}(a)$, for each $a \in \mathcal{S}_O$, $\tau_1(a) \in \mathcal{H}_1$ and $\tau_2(a) \in \mathcal{H}_2$; and
- $P(a, b)$, for $a, b \in \mathcal{S}_O$, $P(a, b) \in \mathcal{H}_1$ and $P(a, b) \in \mathcal{H}_2$.

In a multiple Herbrand set, $\{\tau_1, \tau_2\}$ can be viewed as a disjunction of types τ_1 and τ_2 . Since role disjunction is not allowed in DL-Lite_{bool}, we only need to consider those role assertions appearing in both \mathcal{H}_1 and \mathcal{H}_2 .

Given a DL-Lite_{bool} KB \mathcal{K} , we use $\bigoplus \Omega_{\mathcal{K}} = \langle \Xi_{\mathcal{K}}, \mathcal{M}_{\mathcal{K}} \rangle$ to denote the assembling of all the model features of \mathcal{K} , where $\Xi_{\mathcal{K}}$ is a type set and $\mathcal{M}_{\mathcal{K}}$ a multiple Herbrand set. For simplicity, given $\{\tau_1, \dots, \tau_n\}(a)$ in a multiple Herbrand set \mathcal{M} , we also represent it as $\Xi(a)$ with $\Xi = \{\tau_1, \dots, \tau_n\}$.

Example 5. Recall the KB \mathcal{K} in Example 2. $\Xi_{\mathcal{K}}$ consists of nine types: $\{\emptyset, \tau_1, \tau_2, \{A, \exists P, \geq 2 P\}, \{B, \exists P\}, \{B, \exists P, \geq 2 P, (\exists P^-)\}, \{\exists P, (\geq 2 P)\}\}$, where a type containing a basic concept in brackets, e.g., $(\exists P^-)$, represents two types, one containing $\exists P^-$ and the other not. The multiple Herbrand set $\mathcal{M}_{\mathcal{K}}$ consists of $\{\tau_1, \{A, \exists P, \geq 2 P\}\}(a)$, $\{\tau_2, \{B, \exists P, \geq 2 P, \exists P^-\}\}(b)$, and $P(a, b)$.

To compute the model features of a KB \mathcal{K} , a naive method is to check for each \mathcal{S} -feature \mathcal{F} whether \mathcal{F} satisfies \mathcal{K} . However, we can show that $\bigoplus \Omega_{\mathcal{K}}$ can be computed directly from \mathcal{K} through syntactic transformations, and $\Omega_{\mathcal{K}}$ can be easily obtained from $\bigoplus \Omega_{\mathcal{K}}$.

Now we show how to compute $\bigoplus \Omega_{\mathcal{K}}$ for a DL-Lite_{bool} KB \mathcal{K} through syntactic method (ref. Figure 1).

Note that 2 and 3 of Step 3 ensure that \mathcal{M} is a multiple Herbrand set, and thus, together with Step 2, ensure $\langle \Xi, \mathcal{M} \rangle$ is an assembling of features.

Theorem 2. *Given a consistent DL-Lite_{bool} KB \mathcal{K} , Algorithm 1 always returns $\bigoplus \Omega_{\mathcal{K}}$.*

The following result shows that all the model features of \mathcal{K} can be easily obtained from $\bigoplus \Omega_{\mathcal{K}}$.

Proposition 3. *Given a DL-Lite_{bool} KB \mathcal{K} , if $\bigoplus \Omega_{\mathcal{K}} = \langle \Xi_{\mathcal{K}}, \mathcal{M}_{\mathcal{K}} \rangle$, then for any feature $\mathcal{F} = \langle \Xi, \mathcal{H} \rangle$, \mathcal{F} is a model feature of \mathcal{K} iff the following conditions are satisfied:*

1. $\Xi \subseteq \Xi_{\mathcal{K}}$;
2. $\tau_a \in \Xi_a$, for each $a \in \mathcal{S}_O$, $\tau_a(a) \in \mathcal{H}$ and $\Xi_a(a) \in \mathcal{M}_{\mathcal{K}}$; and

Algorithm 1 (Compute the assembling of all the model features of a DL-Lite_{bool} KB)

Input: A DL-Lite_{bool} KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ and a signature \mathcal{S} .

Output: $\bigoplus \Omega_{\mathcal{K}}$.

Method:

Step 1. Compute \mathcal{S} -type set $\Xi = \bigcap_{(C_1 \sqsubseteq C_2) \in \mathcal{T}} \text{Ts}(\neg C_1 \sqcup C_2)$.

Step 2. Eliminate from Ξ all the types τ such that $\exists R \in \tau$ for some R but $\exists R^- \notin \bigcup \Xi$.

Step 3. Initially, take $\mathcal{M} = \{P(a, b) \mid P(a, b) \text{ or } P^-(b, a) \in \mathcal{A}\}$. Then, for each $a \in \mathcal{S}_O$, perform the following procedure:

1. add $\top(a)$ into \mathcal{A} , and compute $\Xi_a = \bigcap_{C(a) \in \mathcal{A}} \text{Ts}(C) \cap \Xi$;
2. for each $P \in \mathcal{S}_R$, if n is the number of P -successors of a in \mathcal{M} and m the largest number in \mathcal{S}_N with $m \leq n$, then eliminate from Ξ_a all the types not containing $\geq m P$;
3. for each $P \in \mathcal{S}_R$, if n is the number of P -predecessors of a in \mathcal{M} and m the largest number in \mathcal{S}_N with $m \leq n$, then eliminate from Ξ_a all the types not containing $\geq m P^-$;
4. add $\Xi_a(a)$ into \mathcal{M} .

Step 4. Return $\langle \Xi, \mathcal{M} \rangle$ as $\bigoplus \Omega_{\mathcal{K}}$.

Fig. 1. Compute assembling of model features.

3. $P(a, b) \in \mathcal{H}$, for each $P(a, b) \in \mathcal{M}_{\mathcal{K}}$.

From Proposition 3, $\Omega_{\mathcal{K}}$ is the set consisting of all the features satisfying 1 – 3 of Proposition 3.

Given $\Omega_{\mathcal{K}}$ and $\Omega_{\mathcal{K}'}$, we can select those features in $\Omega_{\mathcal{K}'}$ that are closet to $\Omega_{\mathcal{K}}$, i. e., $\sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}})$, through the definition. To provide a complete algorithm for our revision operator, we only need a method to construct $\mathcal{K} \circ \mathcal{K}'$ from $\sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}})$.

Each KB uniquely determines an assembling of features, that is, the assembling of all its model features. The converse is also true, as shown in the following proposition.

Proposition 4. *Given a set Ω of features, there always exists a DL-Lite_{bool} KB \mathcal{K} such that*

1. $\bigoplus \Omega_{\mathcal{K}} = \bigoplus \Omega$;
2. $\Omega \subseteq \Omega_{\mathcal{K}}$; and for any DL-Lite_{bool} KB \mathcal{K}' with $\Omega \subseteq \Omega_{\mathcal{K}'}$, $\Omega_{\mathcal{K}} \subseteq \Omega_{\mathcal{K}'}$.

The above proposition suggests that, $\mathcal{K} \circ \mathcal{K}'$ can be constructed from $\bigoplus \Omega_{\mathcal{K} \circ \mathcal{K}'}$. Also, to compute $\bigoplus \Omega_{\mathcal{K} \circ \mathcal{K}'}$, we can avoid computing all the features in $\Omega_{\mathcal{K} \circ \mathcal{K}'}$, but obtain it directly from $\sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}})$. Indeed, we have $\bigoplus \Omega_{\mathcal{K} \circ \mathcal{K}'} = \bigoplus \sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}})$.

Finally, we present an algorithm for DL-Lite_{bool} KB revision in Figure 2, which follows from the definition.

Theorem 3. *Given two consistent DL-Lite_{bool} KBs \mathcal{K} and \mathcal{K}' , Algorithm 2 always returns $\mathcal{K} \circ \mathcal{K}'$.*

The problem of computing the revision is decidable although it may be exponential in the worst case.

Algorithm 2 (Compute the result of revising a DL-Lite_{bool} KB)**Input:** Two DL-Lite_{bool} KBs \mathcal{K} and \mathcal{K}' , $\mathcal{S} = \text{Sig}(\mathcal{K} \cup \mathcal{K}')$.**Output:** $\mathcal{K} \circ \mathcal{K}'$.**Method:***Step 1.* Use Algorithm 1 to compute $\bigoplus \Omega_{\mathcal{K}}$ and $\bigoplus \Omega_{\mathcal{K}'}$.*Step 2.* Compute $\Omega_{\mathcal{K}}$ and $\Omega_{\mathcal{K}'}$ through Proposition 3.*Step 3.* Obtain $\sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}})$ and compute $\bigoplus \Omega_{\mathcal{K} \circ \mathcal{K}'} = \bigoplus \sigma(\Omega_{\mathcal{K}'}, \Omega_{\mathcal{K}})$. Suppose $\bigoplus \Omega_{\mathcal{K} \circ \mathcal{K}'} = \langle \Xi, \mathcal{M} \rangle$.*Step 4.* Construct from $\langle \Xi, \mathcal{M} \rangle$ the DL-Lite_{bool} KB $\langle \mathcal{T}, \mathcal{A} \rangle$ with

$$\mathcal{T} = \left\{ \prod_{B \in \tau} B \sqcap \prod_{B \notin \tau} \neg B \sqsubseteq \perp \mid B \in \text{Lit}_{\mathcal{S}}^+; \tau \text{ is an } \mathcal{S}\text{-type s.t. } \tau \notin \Xi \right\},$$

and

$$\mathcal{A} = \left\{ \left(\prod_{\tau \in \Xi_a(a)} \left(\prod_{B \in \tau} B \sqcap \prod_{B \notin \tau} \neg B \right) \right)(a) \mid a \in \mathcal{S}_O, \Xi_a(a) \in \mathcal{M} \right\} \cup \{ P(a, b) \in \mathcal{M} \}.$$

Step 5. Return $\langle \mathcal{T}, \mathcal{A} \rangle$ as $\mathcal{K} \circ \mathcal{K}'$.

Fig. 2. Compute revision.

6 Conclusion

Approaches on adapting classical model-based revision to DLs have proposed in [13, 15]. In [15], the author discuss a special case in KB revision, that is, when inconsistency is caused by ABox assertions. The distances between two models are defined through the number of individuals whose membership differ in the two models. A TBox revision operator that can handle *incoherence* is introduced in [13], in which the distance between two models are defined through the number of atomic concepts they disagree. In both of these two approaches, the results of revision are not single KBs, but *disjunctive KBs*. Also, when applying to Example 3 (with only the TBox revised in the second approach), neither of these two approaches has *Student(Tom)* as a logical consequence of the revision. Same problem also occurs in those kernel revision operators with disjunctive KBs as the results of revision.

We have developed a formal framework for revising general KBs (with no specific restriction) in DL-Lite_{bool}, based on the notion of features. We have defined a specific revision operator, which is a natural adaptation of revision approaches from propositional logic. We have also developed algorithms for computing KB revisions in DL-Lite_{bool}, and the result of our revision is always a unique DL-Lite_{bool} KB. We note that other propositional revision operators, e. g., Dalal's revision, belief contraction and update can also be easily adopted in our framework.

There are still several problems remaining for future work. One is to extend our approach to KB revision in other DLs. The major difficulty here is that a general concept in most expressive DLs cannot be transformed into an equivalent disjunction of conjunctions of literal concepts. Another problem is to look at applications of our revision operator in nonmonotonic reasoning problems in DLs.

References

1. A. Artale, D. Calvanese, R. Kontchakov, and M. Zakharyashev. DL-Lite in the light of first-order logic. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence (AAAI07)*, pages 361–366, 2007.
2. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Data complexity of query answering in description logics. In *Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR-06)*, pages 260–270, 2006.
3. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *J. Autom. Reasoning*, 39(3):385–429, 2007.
4. G. Flouris, Z. Huang, J. Z. Pan, D. Plexousakis, and H. Wache. Inconsistencies, negations and changes in ontologies. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI-06)*, 2006.
5. G. Flouris, D. Plexousakis, and G. Antoniou. On applying the AGM theory to dls and OWL. In *Proceedings of the International Semantic Web Conference*, pages 216–231, 2005.
6. P. Gärdenfors, editor. *Belief Revision*. Cambridge University Press, 1992.
7. G. De Giacomo, M. Lenzerini, A. Poggi, and R. Rosati. On the approximation of instance level update and erasure in description logics. In *Proceedings of the 22th National Conference on Artificial Intelligence (AAAI-07)*, pages 403–408, 2007.
8. C. Halaschek-Wiener, Y. Katz, and B. Parsia. Belief base revision for expressive description logics. In *Proceedings of Workshop on OWL Experiences and Directions*, 2006.
9. C. Halaschek-Wiener, B. Parsia, E. Sirin, and A. Kalyanpur. Description logic reasoning for dynamic ABoxes. In *Proceedings of the 2006 International Workshop on Description Logics (DL2006)*, 2006.
10. R. Kontchakov, F. Wolter, and M. Zakharyashev. Can you tell the difference between DL-Lite ontologies? In *Proceedings of the 11th International Conference on Principles of Knowledge Representation and Reasoning (KR08)*, pages 285–295. AAAI Press, 2008.
11. H. Liu, C. Lutz, M. Milicic, and F. Wolter. Updating description logic ABoxes. In *Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR-06)*, pages 46–56, 2006.
12. T. Meyer, K. Lee, and R. Booth. Knowledge integration for description logics. In *Proceedings of The 20th National Conference on Artificial Intelligence (AAAI05)*, pages 645–650, 2005.
13. G. Qi and J. Du. Model-based revision operators for terminologies in description logics. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI-09)*, to appear.
14. G. Qi, P. Haase, Z. Huang, Q. Ji, J. Z. Pan, and J. Völker. A kernel revision operator for terminologies - algorithms and evaluation. In *Proceedings of the 7th International Semantic Web Conference (ISWC08)*, pages 419–434, 2008.
15. G. Qi, W. Liu, and D. A. Bell. Knowledge base revision in description logics. In *Proceedings of 10th European Conference on Logics in Artificial Intelligence (JELIA06)*, 2006.
16. M. Ribeiro and R. Wassermann. Base revision in description logics - preliminary results. In *Proceedings of International Workshop on Ontology Dynamics (IWOD07)*, 2007.
17. K. Satoh. Nonmonotonic reasoning by minimal belief revision. In Institute for New Generation Computer Technology (ICOT), editor, *Proceedings of the International Conference on Fifth Generation Computer Systems*, pages 455–462. Springer-Verlag, 1988.