A Stratification-based Approach for Inconsistency Handling in Description Logics

Guilin Qi[1], Jeff Z. Pan[2]

[1] Institute AIFB
Universität Karlsruhe
D-76128 Karlsruhe, Germany
gqi@aifb.uni-karlsruhe.de
[2] Department of Computing Science, The University of Aberdeen
Aberdeen AB24 3FX
jpan@csd.abdn.ac.uk

Abstract. Inconsistency handling is a central problem in many knowledge representation fields, such as belief revision, belief merging. Many approaches have been proposed to handle inconsistency in ontologies. In this paper, we propose a stratification-based approach for inconsistency handling in description logics (DLs), a family of ontology languages. This approach consists of two steps. In the first step, we obtain a preference relation on the axioms in the *DL knowledge base* using an algorithm. Then two existing approaches in first-order logic are adapted to resolve conflicting information in the stratified DL knowledge base.

1 Introduction

Ontologies play a crucial role for the success of the Semantic Web [12]. There are many representation languages for ontologies, such as description logics (or DLs for short) [4]. Inconsistency may occur because of several reasons, such as modelling errors, migration or merging ontologies, and ontology evolution. Current DL reasoners, such as RACER [14], can detect logical inconsistency. But they only provide lists of unsatisfiable classes and the process of *resolving* inconsistency is left to the user or ontology engineers. The need to improve DL reasoners to reasoning with inconsistency is becoming urgent to make them more applicable. Many approaches were proposed to handle inconsistency in ontologies based on existing techniques for inconsistency management in traditional logics, such as propositional logic and nonmonotonic logics [24, 21, 18].

It is well-known that priority or preference plays an important role in inconsistency handling [2, 7, 20]. In [2], the authors introduced priority to default terminological logic such that more specific defaults are preferred to more general ones. When conflicts occur in reasoning with defaults, defaults which are more specific should be applied before more general ones. In [20], an algorithm, called *refined conjunctive maxi-adjustment* (RCMA for short) was proposed to weaken conflicting information in a *stratified* DL knowledge base and some consistent DL knowledge bases were obtained. To weaken a terminological axiom, they introduced a DL expression, called *cardinality restrictions* on concepts [3]. In [26], a revision-based approach was given to resolve inconsistency in a stratified DL knowledge base. Instead of using cardinality restrictions on concepts, this approach weakens DL axioms (both terminological axioms and assertional axioms) by removing those instances which are responsible for inconsistency.

In this paper, we propose a stratification-based approach for inconsistency handling in DLs. First, we give an algorithm to obtain a preference relation on the axioms of an inconsistent DL knowledge base. The knowledge base associated with this preference relation is a *stratified DL knowledge base*. We then apply two existing approaches in first-order logic to resolve conflicting information in the stratified DL knowledge bases. The first approach is called a possibilistic logic approach and the second approach is called a lexicographic-based approach. We analyze the pros and cons of both approaches.

This paper is organized as follows. Section 2 gives a brief review of description logics. In Section 3, we provide background knowledge on stratified knowledge bases and two inconsistency handling approaches. An algorithm to stratify a DL knowledge base is proposed in Section 4. We then adapt the existing inconsistency handling approaches to DLs in Section 5. Before conclusion, we have a brief discussion on related work.

2 Description logics

In this section, we introduce some basic notions of Description Logics (DLs), a family of well-known knowledge representation formalisms [4]. We consider $\mathcal{ALC}$ [25], which is a simple yet relatively expressive DL. Let N_C and N_R be pairwise disjoint and countably infinite sets of *concept names* and *role names* respectively. We use the letters A and B for concept names, the letter R for role names, and the letters C and D for concepts. The set of $\mathcal{ALC}$ concepts is the smallest set such that: (1) every concept name is a concept; (2) if C and D are concepts, R is a role name, then the following expressions are also concepts: $\neg C$ (full negation), $C \sqcap D$ (concept conjunction), $C \sqcup D$ (concept disjunction), $\forall R.C$ (value restriction on role names) and $\exists R.C$ (existential restriction on role names). An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a set $\Delta^{\mathcal{I}}$, called the *domain* of $\mathcal{I}$, and a function $\cdot^{\mathcal{I}}$ which maps every concept C to a subset $C^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$ and every role R to a subset $R^{\mathcal{I}}$ of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ such that, for all concepts C, D, role R, the following properties are satisfied:

(1) $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$,
(2) $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$, $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$,
(3) $(\exists R.C)^{\mathcal{I}} = \{x | \exists\ y s.t. (x, y) \in R^{\mathcal{I}}$ and $y \in C^{\mathcal{I}}\}$,
(4) $(\forall R.C)^{\mathcal{I}} = \{x | \forall y (x, y) \in R^{\mathcal{I}}$ implies $y \in C^{\mathcal{I}}\}$.

A DL knowledge base consists of two components, the *terminological box (TBox)* T and the *assertional box (ABox)* A. A TBox is a finite set of terminological axioms of the form $C \sqsubseteq D$ (general concept inclusion or GCI for short) or $C \equiv D$ (equalities), where C and D are two (possibly complex) $\mathcal{ALC}$ concepts. An interpretation $\mathcal{I}$ satisfies a GCI $C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, and it satisfies an equality

$C{\equiv}D$ iff $C^{\mathcal{I}} = D^{\mathcal{I}}$. It is clear that $C{\equiv}D$ can be seen as an abbreviation for the two GCIs $C{\sqsubseteq}D$ and $D{\sqsubseteq}C$. Therefore, we take a TBox to contain only GCIs. We can also formulate statements about individuals. We denote individual names as $a,\ b,\ c$. A *concept* (*role*) assertion axiom has the form $C(a)$ ($R(a,b)$), where C is a concept description, R is a role name, and $a,\ b$ are *individual names*. To give a semantics to ABoxs, we need to extend interpretations to individual names. For each individual name a, $\cdot^{\mathcal{I}}$ maps it to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. An interpretation $\mathcal{I}$ satisfies a concept axiom $C(a)$ iff $a^{\mathcal{I}}{\in}C^{\mathcal{I}}$, it satisfies a role axiom $R(a,b)$ iff $(a^{\mathcal{I}}, b^{\mathcal{I}}){\in}R^{\mathcal{I}}$. An *ABox* contains a finite set of concept and role axioms. A DL knowledge base K consists of a TBox and an ABox, i.e. it is a set of GCIs and assertion axioms. An interpretation $\mathcal{I}$ is a *model* of a DL (TBox or ABox) axiom iff it satisfies this axiom, and it is a model of a DL knowledge base K if it satisfies every axiom in K. In the following, we use $M(\phi)$ (or $M(K)$) to denote the set of models of an axiom ϕ (or DL knowledge base K). K is consistent iff $M(K){\neq}\emptyset$. Let K be an inconsistent DL knowledge base. A set $K'{\subseteq}K$ is a *conflict* [1] of K if K' is inconsistent, and any sub-knowledge base $K''{\subset}K'$ is consistent. Given a DL knowledge base K and a DL axiom ϕ, we say K *entails* ϕ, denoted as $K \models \phi$, iff $M(K){\subseteq}M(\phi)$.

3 Stratified Knowledge Bases

In this section, we first provide some background knowledge on stratified knowledge bases. Then some inconsistency handling approaches in classical logic are introduced.

3.1 Stratified knowledge base

We consider a first order language $\mathcal{L}$ determined by a set of variable symbols and a set of predicate and function symbols. 0-ary functions are constants. We use uppercase letters like P and R for predicate symbols, lowercase letters like a, b, c for constant symbols, and x, y for variable symbols. The classical consequence relation is denoted as $\vdash$. We denote formulae in $\mathcal{L}$ by ϕ, ψ, γ,... A *classical knowledge base* K is a finite set of first-order formulae. K is inconsistent iff $K \vdash \phi$ and $K \vdash \neg\phi$ for some formula ϕ.

A *stratified* knowledge base, sometimes also called prioritized knowledge base [5], is a set K of (finite) propositional formulas together with a total preorder $\leq$ on K (a preorder is a transitive and reflexive relation, and $\leq$ is a total preorder if either $\phi{\leq}\psi$ or $\psi \leq \phi$ holds for any $\phi, \psi{\in}K$)[2]. Intuitively, if $\phi \leq \psi$, then ϕ is considered to be at least less or equally important than ψ. K can be equivalently defined as a sequence $K = (S_1, ..., S_n)$, such that formulas in S_i have the same

[1] The notion of conflict is different from the notion of minimal unsatisfiability-preserving sub-TBox of a DL knowledge base defined in [24] in that it concerns inconsistency instead of incoherence.

[2] For simplicity, we use K to denote a stratified knowledge base and ignore the total preorder $\leq$.

level of priority and have higher priority than the ones in S_j where $j < i$. Each subset S_i is called a stratum of K and i the priority level of each formula of S_i. Therefore, the higher the stratum, the higher the priority level of a formula in it. A subbase A of K is also stratified, that is, $A = (A_1, ..., A_n)$ such that $A_i \subseteq S_i$, $i = 1, ..., n$.

3.2 Reasoning with inconsistent stratified knowledge bases

Many approaches have been introduced to reasoning with inconsistent stratified knowledge bases [5, 7–9]. We consider two approaches, one is the possibilistic logic approach [9] and the other is the adapted lexicographic-based approach [7].

Possibilistic logic approach Possibilistic logic inference [9] is based on a suitable consistent stratified subbase of K. Suppose $K = \{S_1, ..., S_n\}$. Let $Inc(K) = max\{i : S_i \cup ... \cup S_n \textit{ is inconsistent}\}$ be the inconsistency degree of K. There are two possibilistic consequence relations.

Definition 1. *Let $K = \{S_1, ..., S_n\}$ be a stratified knowledge base. A formula ϕ is said to be a possibilistic consequence of K, denoted by $K \vdash_\pi \phi$ if and only if $S_{Inc(K)+1} \cup ... \cup S_n \vdash \phi$.*

A formula ϕ is a possibilistic consequence of K if and only if it can inferred by the set of formulas whose priority levels are greater than $Inc(K)$, the inconsistency degree of K.

Definition 2. *Let $K = \{S_1, ..., S_n\}$ be a stratified knowledge base. A formula ϕ is said to be a i-consequence of K, denoted by $K \vdash_i \phi$ if and only if the following conditions are satisfied:*

(1) $i > Inc(K)$
(2) $S_i \cup ... \cup S_n \vdash \phi$
(3) for any $j > i$, $S_j \cup ... \cup S_n \nvdash \phi$.

In Definition 2, Condition (1) ensures that the i-consequence is not trivial. Condition (2) says that ϕ can be inferred from the set of formulas whose priority levels are greater than i and Condition (3) means that i is the highest priority level which can be attached to ϕ.

To check whether a formula ϕ is a possibilistic consequence or an i-consequence of K, we first need to compute the inconsistency degree of K, which is a hard task.

Proposition 1. *[19] Computing $Inc(K)$ requires $\lceil log_2 n \rceil$ satisfiability checks, where n is the number of different strata of K.*

According to Proposition 1, it requires $\lceil log_2 n \rceil + 1$ satisfiability checks to decide whether a formula ϕ is a possibilistic consequence of K or not.

Adapted lexicographic-based approach In [7], a stratified first-order logic approach for handling inconsistency was proposed to adapt the lexicographic-based approach in propositional logic [5]. When a formula of the form $\forall x\phi(x)$ is involved in a conflict, then it is simply deleted by the lexicographic-based approach to restore consistency. In contrast, the adapted lexicographic-based approach weakens the conflicting formula by dropping only instances of this formula which are responsible of a conflict. For example, if a formula of the form $\forall x\phi(x)$ is conflicting for $x = a$, then this formula is weakened as $\forall x \neg(x = a) \rightarrow \phi(x)$. Let us explain the approach in more detail.

Let ϕ be a formula which is universally quantified with a set of variable $X = \{x_1, ..., x_n\}$. Let $I = \{i_1, ..., i_n\}$ be such that i_k $(k = 1, ..., n)$ are instances of x_k respectively. Let us denote the formula $\neg(\bigwedge_{k=1,...,n}(x_k = i_k))$ as $Different(I, X)$. The following definitions can be found in [7].

Definition 3. *Let ϕ be a first-order formula which is universally quantified with a set of variable $X = \{x_1, ..., x_n\}$ where n is finite. ϕ_{weak} is called a weakened formula of ϕ if it has the form: $A \rightarrow \phi$, where $A = \{Different(I_j, X) : j = 1, ..., m\}$ (or A can be seen as the conjunction of formulas in it). The degree of a weakened formula ϕ_{weak} is defined as $degree(\phi_{weak}) = |A|$, i.e., it is the cardinality of A.*

The degree of a weakened formula is used to count the number of instances that cannot be applied, i.e. instances that are ignored.

The weakened base of a first-order knowledge base is defined as follows.

Definition 4. *Let $K = \{S_1, ..., S_n\}$ be a stratified knowledge base, where S_n contains formulas that are completely certain (that is, they cannot be deleted or weakened if they are involved in a conflict). A first-order knowledge base $K' = \{S'_1, ..., S'_n\}$ is said to be a weakened base of K if 1) K' is consistent, and 2) K' is only obtained by replacing some formula ϕ of $\{S_1, ..., S_{n-1}\}$ by their weakened counterpart ϕ_{weak}.*

The degree of a stratum S'_i of a weakened base K' is defined as: $degree(S'_i) = \Sigma_{\phi_{weak} \in S'_i} degree(\phi_{weak})$. We then can define the ranking between weakened bases as follows.

Definition 5. *Let K' and K'' be two weakened bases of K. K' is said to be lexicographically preferred to K'', denoted by $K' >_{lex} K''$, if $\exists i$, $1 \leq i \leq n$ such that i) $degree(S'_i) < degree(S''_i)$, and ii) $\forall j > i$, $degree(S'_j) = degree(S''_j)$. K' is said to be lexicographically preferred weakened base of K if there is no consistent weakened base K'' such that $K'' >_{lex} K'$. A formula ψ is said to be a lex conclusion of K, denoted $K \vdash_{lex} \psi$, if ψ is a consequence of all lexicographically preferred weakened bases of K.*

4 Stratification of DL Knowledge Bases

In this section, we define an algorithm transform an inconsistent DL knowledge base into a stratified DL knowledge base, i.e. each element of the base is assigned a rank, based on the weakening-based revision operator. More precisely, a stratified DL knowledge base is of the form $K = S_1 \cup ... \cup S_n$, where for each $i \in \{1, ..., n\}$, S_i is a finite multi-set of DL sentences. Sentences in each stratum S_i have the same rank or reliability, while sentences contained in S_j such that $j > i$ are seen as more reliable.

There are many ways to obtain a stratified DL knowledge base. For example, the stratification can be given by an expert or by ontology learning [15]. The stratification can also be computed automatically. In this section, we propose an algorithm to stratify a DL knowledge base. We assume that a TBox T consists of two adjoint subsets: a set of completely sure terminology axioms T_c, i.e., axioms which will not be involved in any conflict, and a set of default terminology axioms T_d. We also assume that the information in an ABox is completely sure. The knowledge base is called a default DL knowledge base. That is, a default DL knowledge base K is already stratified as $K = \{T_d, A \cup T_c\}$, where T_c contains completely sure terminology axioms, A contains assertion axioms, and T_d contains default terminology axioms. This assumption is often adopted in default theories [1, 7, 23]. In default theories, *specificity* is a commonly used criterion for ranking a set of default rules [22, 23, 2]. Many methods have been proposed to compute specificity in default theories. In [22], Pearl gives a method to rank a set of default rules such that a more specific default is preferred to a more general one. This method is then revised and applied to stratify a knowledge base consisting of a set of default and hard rules in [6]. In this section, we propose a stratification algorithm based on the stratification method in [6]. Given a set of terminology axioms $T = T_c \cup T_d$, where T_c contains completely sure terminology axioms, and T_d contains default terminology axioms, we say that a default terminology axiom $C_1 \sqsubseteq D_1$ is more specific than another one $C_2 \sqsubseteq D_2$ iff $T \models C_1 \sqsubseteq C_2$ but $T \not\models C_2 \sqsubseteq C_1$. Note that the ordering relation defined by specificity is not necessarily a total preorder.

Stratification Algorithm

Input: Default terminology axioms base T_d, completely sure terminology axioms base T_c

Output: Stratified default terminology axiom base T_s

begin

  m=1;

  while $T_d \neq \emptyset$ **do**

  begin

    $S_m = \{C_i \sqsubseteq D_i | C_i \sqsubseteq D_i \in T_d,\ and\ T_c \cup\ T_d \cup \{C_i(a)\}\ is\ consistent,\ a\ is\ a\ new\ instance\}$;

    If $S_m = \emptyset$ **then** stop (inconsistent terminology axioms).

    $T_d = T_d \setminus S_m$; $m = m + 1$;

  end begin

  end while

  Return $T_s = \{S_1, S_2, ..., S_m\}$.

end

In the stratification algorithm, when there exists m such that S_m is empty, then we say that T_d is inconsistent with T_c and we end the algorithm (because all the other element in T_d are blocked to be stratified). In the following, we assume that T_c is always consistent with T_d. Given a default DL knowledge base $K = \{T_d, A \cup T_c\}$, suppose T_d is stratified as $T_s = \{S_1, ..., S_m\}$ using the stratification algorithm, we get a stratified DL knowledge base $K' = \{S_1, ..., S_{m+1}\}$, where $S_{m+1} = A \cup T_c$.

Let us look at an example.

Example 1. Let $K = \{T_d, A \cup T_c\}$, where $T_d = \{bird \sqsubseteq flies, penguin \sqsubseteq \neg flies\}$, $T_c = \{penguin \sqsubseteq bird\}$ and $A = \{penguin(Cheeky)\}$. We now apply the stratification algorithm to stratify T_d. First, since $T_d \cup T_c \cup \{bird(a)\}$ is consistent and $T_d \cup T_c \cup \{penguin(a)\}$ is inconsistent, where a is an arbitrary bird name, we have $S_1 = \{bird \sqsubseteq flies\}$. There is only one element left in T_d, so $S_2 = \{penguin \sqsubseteq \neg flies\}$. That is, T_d is stratified as $T_s = \{S_1, S_2\}$. Note that *penguin* $\sqsubseteq \neg flies$ is more specific than $bird \sqsubseteq flies$ because we have $penguin \sqsubseteq bird$ in T_c. K is then further stratified as $K' = \{S_1, S_2, A \cup T_c\}$.

In Example 1, the ranking obtained by the stratification algorithm agrees with the notion of specificity. More generally, suppose $C_i \sqsubseteq D_i$ is a terminology axiom in T_d such that $T_d \cup T_c \cup \{C_i(a)\}$ is inconsistent. Then the assertion $C_i(a)$ triggers a more general default terminology axiom in T_d which is responsible for the inconsistency. Therefore, the higher the rank is, the more specific the default terminology axiom is.

5 Inconsistency Handling in Stratified DL Knowledge Bases

5.1 Possibilistic logic approach

We apply the possibilistic logic approach to deal with inconsistency in a stratified DL knowledge base K. We have the following two definitions.

Definition 6. *Let $K = \{S_1, ..., S_n\}$ be a stratified DL knowledge base. Let $Inc(K) = max\{i : S_i \cup ... \cup S_n$ is inconsistent$\}$ be the inconsistency degree of K. For any DL statement ϕ, ϕ is a possibilistic consequence of K, denoted $K \models_\pi \phi$, if and only if, $S_{Inc(K)+1} \cup ... \cup S_n \models_\pi \phi$.*

Definition 7. *Let $K = \{S_1, ..., S_n\}$ be a stratified DL knowledge base. Let $Inc(K)$ be the inconsistency degree of K. For any DL statement ϕ, ϕ is a i-consequence of K, denoted $K \models_i \phi$, if and only if the following conditions are satisfied:*

(1) $i > Inc(K)$
(2) $S_i \cup ... \cup S_n \models \phi$
(3) for any $j > i$, $S_j \cup ... \cup S_n \not\models \phi$.

By Definition 6 and Definition 7, both the possibilistic consequence and the i-consequence relation are independent of DL reasoners, i.e., we can treat the DL reasoner as a black box and use it to check knowledge base consistency. Another advantage of the possibilistic approaches is that they are independent of DL languages, although we restrict our discussion to DL $\mathcal{ALC}$ in this paper.

The main task of possibilistic inferences defined above is to compute the inconsistency degree of K, which requires $\lceil log_2 n \rceil$ DL consistency checks, where n is the number of different strata of K.

Let us go back to Example 1.

Example 2. (Continuing Example 1) Suppose that K is stratified as $K' = \{S_1, S_2, S_3\}$, where $S_1 = \{bird \sqsubseteq flies\}$, $S_2 = \{penguin \sqsubseteq \neg flies\}$ and $S_3 = \{penguin(Cheeky), penguin \sqsubseteq bird\}$. Let us check if $Cheeky$ can fly. First, we compute the inconsistency degree of K'. Since $\cup_{i=1}^{3} S_i$ is inconsistent and $S_2 \cup S_3$ is consistent, $Inc(K') = 1$. It is clear that $S_2 \cup S_3 \models_\pi \neg flies(Cheeky)$. So we can conclude that $Cheeky$ cannot fly. Furthermore, we can conclude that $K' \models_2 \neg flies(Cheeky)$, that is, the priority level of the proposition that $Cheeky$ cannot fly is two.

Possibilistic logic approach simply blocks a default terminology axiom if it is responsible for the conflict and its priority level is not larger than the inconsistency degree. This may result in unnecessary loss of information. Let us continue to consider Example 3. Suppose we are told that $Kelly$ is a bird. We add $bird(Kelly)$ to S_3, that is, $S_3 = \{penguin(Cheeky), bird(Kelly), penguin \sqsubseteq bird\}$. Since $S_2 \cup S_3 \not\models flies(Kelly)$. So we cannot conclude that $Kelly$ can fly. This is because $bird \sqsubseteq flies$ is blocked and cannot be used to infer that $flies(Kelly)$.

5.2 Lexicographic-based approach

In this section, we apply the adapted lexicographic-based approach to the description logic setting. To do this, we need to extend the logic $\mathcal{ALC}$ with *cardinality restrictions* on concepts, which was proposed in [3]. Cardinality restrictions on a concept C are of the form $\geq m\ C$ and $\leq n\ C$, which express that the concept C has at least m elements and at most n elements respectively. We only consider cardinality restriction of the form $\leq n\ C$. An interpretation $\mathcal{I}$ is said to satisfy $\leq\ n\ C$ iff $|C^{\mathcal{I}}| \leq n$. Each GCI $C \sqsubseteq D$ can be equivalently transformed into a cardinality restriction of the form $\leq\ 0\ C \sqcap \neg D$, which says that the concept $C \sqcap \neg D$ is empty.

When a GCI is debugged to be erroneous, it is generally deleted to restore consistency in current methods [18, 21, 24]. However, as we can see from Example 1, this can result in unnecessary loss of information. In [20], a method is proposed to weaken a conflicting GCI rather than delete it. The idea is that we first transform every GCI $C \sqsubseteq D$ into an equivalent cardinality restriction $\leq\ 0\ C \sqcap \neg D$. For a cardinality restriction which is involved in conflict, we simply weaken it as $\leq\ n\ C \sqcap \neg D$, where $n \geq 1$. We adopt this method to weaken a GCI.

Definition 8. *Let $C \sqsubseteq D$ be a GCI. A weakening $(C \sqsubseteq D)_{weak}$ of $C \sqsubseteq D$ has the form $\leq\ n\ C \sqcap \neg D$, where $n \geq 0$. We use $d((C \sqsubseteq D)_{weak}) = n$ to denote the degree of $(C \sqsubseteq D)_{weak}$.*

It is clear that $d((C \sqsubseteq D)_{weak}) = 0$ if $(C \sqsubseteq D)_{weak} = \leq\ 0\ C \sqcap \neg D$.

We now consider the weakening of a stratified DL knowledge base.

Definition 9. *Let $K = \{S_1, ..., S_n\}$ be a stratified DL knowledge base, where S_n contains completely sure terminology axioms and assertion axioms. A stratified DL knowledge base $K' = \{S'_1, ..., S'_n\}$ is said to be a weakened base of K if it satisfies the following conditions:*

- *K' is consistent,*
- *There is a bijection from $S_1 \cup ... \cup S_{n-1}$ to $S'_1 \cup ... \cup S'_{n-1}$ such that for each $\phi \in K$, $f(\phi)$ is a weakening of ϕ,*
- *$S'_n = S_n$.*

The degree of a stratum S'_i of a weakened base K' is defined as: $degree(S'_i) = \Sigma_{\phi_{weak} \in S'_i} degree(\phi_{weak})$.

The ranking between weakened bases is defined as follows.

Definition 10. *Let K be a stratified DL knowledge base. Let $K' = \{S'_1, ..., S'_n\}$ and $K'' = \{S''_1, ..., S''_n\}$ be two weakened bases of K. K' is said to be lex-preferred to K'', denoted by $K' >_{lex} K''$, if $\exists i$, $1 \leq i \leq n$ such that* 1) *$degree(S'_i) < degree(S''_i)$, and* 2) *$\forall j > i$, $degree(S'_j) = degree(S''_j)$.*

Similar to the adapted lexicographic-based approach, we can define the following inference.

Definition 11. *Let K' be a weakened base of K. K' is a lex-preferred weakened base of K if there is no consistent weakened base K'' of K such that $K'' >_{lex} K'$. A formula ψ is said to be a lexicographic conclusion of K, denoted $K \models_{lex} \psi$, if ψ is a consequence of all lex-preferred weakened bases of K.*

We illustrate the lexicographic-based approach by the following example.

Example 3. (Continuing Example 2) K' has three weakened bases: $K_1 = \{S_{11}, S_{12}, S_{13}\}$, where $S_{11} = \{\leq 1\ bird \sqcap \neg flies\}$, $S_{12} = S_2$ and $S_{13} = S_3$; $K_2 = \{S_{21}, S_{22}, S_{23}\}$, where $S_{21} = S_1$, $S_{22} = \{\leq 1\ penguin \sqcap \neg flies\}$ and $S_{23} = S_3$; $K_3 = \{S_{31}, S_{32}, S_{33}\}$, where $S_{31} = S_1$, $S_{32} = S_2$ and $S_{33} = \{penguin(Cheeky), \leq 1\ penguin \sqcap \neg bird\}$. It is easy to check that K_1 is the only lex-preferred weakened base of K. Since $K_1 \models bird(Kelly)$, we have $K \models_{lex} bird(Kelly)$.

Next, we consider the semantic computation of the lexicographic-based approach.

Definition 12. *Let $\mathcal{W}$ be a non-empty set of interpretations and $\mathcal{I} \in \mathcal{W}$, ϕ a terminology axiom of the form $C \sqsubseteq D$, and K be a DL knowledge base (K is not stratified here). The number of ϕ-exceptions for $\mathcal{I}$ is:*

$$e^{\phi}(\mathcal{I}) = \begin{cases} |C^{\mathcal{I}} \cap (\neg D^{\mathcal{I}})| & \text{if } C^{\mathcal{I}} \cap (\neg D^{\mathcal{I}}) \text{ is finite} \\ \infty & \text{otherwise.} \end{cases} \quad (1)$$

The number of K-exceptions for $\mathcal{I}$ is $e^K(\mathcal{I}) = \Sigma_{\phi\in K} e^{\phi}(\mathcal{I})$. The ordering $\preceq_K$ on $\mathcal{W}$ is: $\mathcal{I} \preceq_K \mathcal{I}'$ iff $e^K(\mathcal{I}){\leq}e^K(\mathcal{I}')$, for $\mathcal{I}' \in \mathcal{W}$. $\mathcal{I} \equiv_K \mathcal{I}'$ denotes $\mathcal{I} \preceq_K \mathcal{I}'$ and $\mathcal{I}' \preceq_K \mathcal{I}$

The definition of ϕ-exception originates from Definition 6 in [20]. However, in [20], it is used to define an ordering $\preceq^{\pi}_K$ on a set of interpretations with the same *pre-interpretation* $\pi = (\Delta^{\pi}, d^{\pi})$, where Δ^{π} is a domain and d^{π} is a *denotation function* which maps every individual name a to a different element in Δ^{π}.

We define the lexicographical preference ordering as follows.

Definition 13. *Let $K = (S_1, ..., S_n)$ be a stratified DL knowledge base, where S_n contains completely sure terminology axioms and assertion axioms, and Ω be the set of models of S_n. The lexicographical preference ordering $\preceq_{lex,K}$ is defined as $\mathcal{I}{\preceq_{lex,K}}\mathcal{I}'$ iff $\forall i{\in}\{1, ..., n-1\}$, $\mathcal{I}{\equiv_{S_i}}\mathcal{I}'$ or $\exists i$ such that $\mathcal{I} \prec_{S_i} \mathcal{I}'$, and $\mathcal{I}{\equiv_{S_j}}\mathcal{I}'$ for all $n > j > i$. The set of minimal models of K w.r.t $\preceq_{lex,K}$ is denoted as $min(\Omega, \preceq_{lex,K})$.*

The following results give semantic interpretation of the lexicographic-based inference. We first prove a lemma.

Definition 14. *Let K and K' be two consistent DL knowledge bases (K and K' are not stratified), where K consists of terminology axioms. A DL knowledge base $K_{weak,K'}$ is a weakened knowledge base of K w.r.t K' if it satisfies:*

- *$K_{weak,K'} \cup K'$ is consistent, and*
- *There is a bijection f from K to $K_{weak,K'}$ such that for each $\phi{\in}K$, $f(\phi)$ is a weakening of ϕ.*

The set of all weakened bases of K w.r.t K' is denoted by $Weak_{K'}(K)$.

Lemma 1. *Let K and K' be two consistent DL knowledge bases, where K consists of terminology axioms, and $\mathcal{I}$ be an interpretation such that $\mathcal{I} \models K'$. Let $l = min(d(K_{weak,K'}) : K_{weak,K'}{\in}Weak_{K'}(K), \mathcal{I} \models K_{weak,K'})$. Then $e^K(\mathcal{I}) = l$.*

Proof. Suppose $K_{weak,K'} \in Weak_{K'}(K)$ such that $d(K_{weak,K'}) = l$ and $\mathcal{I} \models K_{weak,K'}$. Let $\phi = C{\sqsubseteq}D \in K$ and $\phi_{weak}{\in}K_{weak,K'}$. Suppose $d(\phi_{weak}) = n$, that is, $\phi_{weak} ={\leq_n} C{\sqcap}\neg D$. Since $\mathcal{I} \models K_{weak,K'}$, $\mathcal{I} \models \phi_{weak}$. Moreover, for any other weakening ϕ'_{weak} of ϕ, if $d(\phi'_{weak}) < n$, then $\mathcal{I} \not\models \phi'_{weak}$ (because otherwise, we find another weakening $K'_{weak,K'} = (K_{weak,K'} \setminus \{\phi_{weak}\}){\cup}\{\phi'_{weak}\}$ such that $d(K'_{weak,K'}) < d(K_{weak,K'})$ and $\mathcal{I} \models K'_{weak,K'}$). So $|C^{\mathcal{I}}{\sqcap}\neg D^{\mathcal{I}}|{\leq}n$. We further have $|C^{\mathcal{I}}{\sqcap}\neg D^{\mathcal{I}}|{\geq}n$. Otherwise, suppose $|C^{\mathcal{I}}{\sqcap}\neg D^{\mathcal{I}}|{<}n$. Then there exists ϕ_{weak} of ϕ such that $d(\phi'_{weak}) < n$, this is a contradiction. Therefore, $e^{\phi}(\mathcal{I}) = |C^{\mathcal{I}}{\sqcap}\neg D^{\mathcal{I}}| = n = d(\phi_{weak})$. That is, $e^K(\mathcal{I}) = l$.

Proposition 2. *Let $K = (S_1, ..., S_n)$ be a stratified DL knowledge base, where S_n contains completely sure terminology axioms and assertion axioms. ϕ is a DL statement and Ω is the set of models of S_n. Then $K \models_{lex} \phi$ iff $\mathcal{I} \models \phi$, for all $\mathcal{I} \in min(\Omega, \preceq_{lex,K})$.*

Proof. *Suppose $\mathcal{K}$ contains all the lex-preferred weakened bases of K. We need to prove that for every interpretation $\mathcal{I}$, $\mathcal{I} \models \mathcal{K}$ iff $\mathcal{I} \in min(\Omega, \preceq_{lex,K})$, where $\mathcal{I} \models \mathcal{K}$ iff $\mathcal{I} \models K_i$ for all $K_i \in \mathcal{K}$.*

"Only if part"

Suppose $\mathcal{I} \models \mathcal{K}$ and $\mathcal{I} \notin min(\Omega, \preceq_{lex,K})$. Then $\exists \mathcal{I}'$ such that $\mathcal{I}' \prec_{lex,K} \mathcal{I}$. That is, there exists some i such that $\mathcal{I}' \prec_{S_i} \mathcal{I}$ and $\mathcal{I}' \equiv_{S_j} \mathcal{I}$ for all $n > j > i$. Suppose $K' = \{S'_1, ..., S'_n\} \in \mathcal{K}$, then $\mathcal{I} \models K'$. Since $\mathcal{I}' \equiv_{S_j} \mathcal{I}$ for all $n > j > i$, by Lemma 1, there exists a weakened base S''_j of S_j such that $\mathcal{I}' \models S''_j$ and $degree(S'_j) = degree(S''_j)$. This can be proved by induction over priority level k of K.

For $k = n-1$. Since $\mathcal{I}' \equiv_{S_{n-1}} \mathcal{I}$, we have $e^{S_{n-1}}(\mathcal{I}) = e^{S_{n-1}}(\mathcal{I}')$. By Lemma 1, we have $degree(S'_{n-1}) = e^{S_{n-1}}(\mathcal{I})$. Moreover, there exists a weakened base S''_{n-1} of S_{n-1} such that $degree(S''_{n-1}) = e^{S_{n-1}}(\mathcal{I}')$. So $degree(S'_{n-1}) = degree(S''_{n-1})$.

Suppose for all $k \geq l$, where $l > i+1$, there exists a weakened base S''_k of S_k such that $degree(S''_k) = degree(S'_k)$ and $\mathcal{I}' \models S_k$. Since $k-1 > i$, we have $\mathcal{I} \equiv_{S_{k-1}} \mathcal{I}'$. That is, $e^{S_{k-1}}(\mathcal{I}) = e^{S_{k-1}}(\mathcal{I}')$. Similarly, by Lemma 1, there exists a weakened base S''_{k-1} of S_{k-1} such that $\mathcal{I}' \models S''_{k-1}$ and $degree(S''_{k-1}) = degree(S'_{k-1})$.

Since $\mathcal{I}' \prec_{S_i} \mathcal{I}$, we have $e^{S_i}(\mathcal{I}') < e^{S_i}(\mathcal{I})$. By Lemma 1, there exists a weakened base S''_i of S_i such that $\mathcal{I}' \models S''_i$ and $degree(S''_i) < degree(S'_i)$. This is a contradiction because we then can find a weakened base $K'' = \{S''_1, ..., S''_n\}$ such that $K'' >_{lex} K'$. Therefore, if $\mathcal{I} \models \mathcal{K}$, then $\mathcal{I} \in min(\Omega, \preceq_{lex,K})$.

"If part"

Suppose $\mathcal{I} \in min(\Omega, \preceq_{lex,K})$. Let us assume that $\mathcal{I} \not\models \mathcal{K}$. Suppose K' is a weakened base of K such that $\mathcal{I} \models K'$, and for there does not exist a weakened base K'' of K such that $\mathcal{I} \models K''$ and $K'' >_{lex} K'$. Since $\mathcal{I} \not\models \mathcal{K}$, we have $degree(K'') < degree(K')$ for all $K'' \in \mathcal{K}$. Let $K'' \in \mathcal{K}$ and there exists an interpretation $\mathcal{I}'$ such that $\mathcal{I}' \models K''$. By Definition 10, there exists i such that $degree(S''_i) < degree(S'_i)$ and for all $n > j > i$, $degree(S''_j) = degree(S'_j)$. By Lemma 1, it is easy to show that $e^{S_j}(\mathcal{I}') = e^{S_j}(\mathcal{I})$ for all $n > j > i$ and $e^{S_i}(\mathcal{I}') < e^{S_i}(\mathcal{I})$. So $\mathcal{I}' \prec_{lex,K} \mathcal{I}$, which is a contradiction.

This completes the proof.

According to Proposition 2, we can define the lexicographic-based inference in a semantic way.

Definition 15. *Let $K = (S_1, ..., S_n)$ be a stratified DL knowledge base. ϕ is a DL statement. Then K lexicographically entails ϕ, denoted $K \models_{lex} \phi$, iff $\omega \models \phi$, for all $\omega \in min(\Omega, \preceq_{lex,K})$.*

Compared with possibilistic approaches, the lexicographic-based approach is more fine-grained and can keep more original information. However, it is based on cardinality restrictions on concepts, so it cannot be used to deal with inconsistency in DLs which disallow cardinality restrictions on concepts. Furthermore, to implement the lexicographic-based approach, we need to pinpoint the instances which are responsible for the inconsistency, which is usually a hard task.

6 Related Work

A lot of work has been done on handling inconsistency in DLs [1, 2, 21, 16, 24, 18, 20]. In [1], Reiter's default logic is embedded into terminological representation formalisms. In their paper, conflicting information is treated as *exceptions*. To deal with conflicting default rules, they instantiated each rule using individuals appearing in the ABox and applied two existing default reasoning methods to compute all extensions. Then, in [2], priorities were introduced to default terminological logic such that more specific defaults are preferred to more general ones. In our stratification algorithm, we also give priority to a more specific default terminology. However, when handling inconsistency, we do not need the instantiation step. Furthermore, in [1, 2], the resolution of conflicting ABox assertions was not considered. Recently, some methods for repairing inconsistencies [24, 21] or reasoning with inconsistent ontologies [16, 18] have been proposed. A common problem with these methods is that they do not take advantage of DL expressions. If a terminological axiom is detected to be erroneous (that is, it is involved in a conflict), then it is simply deleted. In contrast, we introduce an important DL expression, i.e. cardinality restrictions, to deal with an erroneous terminological axiom. Our lexicographical-based approach is closely related to the adaptive lexicographic-based approach in [8]. However, our approach is more general than the adaptive lexicographic-based approach. The later can only deal with inconsistencies arising due to instances (or individual names in DLs) explicitly introduced in the facts (or ABox assertions), while our approach is also applicable when inconsistencies result from TBox axioms. In [20], the authors proposed an algorithm, called refined conjunctive maxi-adjustment (RCMA), for inconsistency handling in a stratified knowledge base based on cardinality restrictions. Our second inconsistency handling method is also based on cardinality restrictions. However, our method differs from RCMA method in that we only weaken those GCIs which are involved in conflict and RCMA method weakens not only conflicting GCIs but also GCIs not involved in conflict. This work is also related to some other approaches to extend DLs with nonmonotonic theories, such as defeasible description logics [13, 17, 27] and belief change in DLs [10, 11]. Defeasible description logics combines defeasible logic and description logics by adding a layer of rules from defeasible logic on top of ontologies in description logics. As in defeasible logic, an acyclic relation on the set of rules is assumed to deal with conflicting rules. This preference relation may not be a total preorder as we have assumed in the paper. The default terminology axioms are similar to the defeasible rules in defeasible description logics. However, rules are not terminology axioms. In [10, 11], AGM's theory of belief change has been applied to description logics. However, they only studied the feasibility of applying the generalized AGM postulates for belief change to DLs. No explicit belief change operators were proposed in their papers.

7 Conclusions

In this paper, we first proposed an approach to stratifying a DL knowledge base such that a more specific conflicting terminology axiom is preferred to a more general one. Then two inconsistency handling approaches first-order logic are adapted to deal with inconsistency in a stratified DL knowledge base. The first approach is the possibilistic logic approach, which drops formulas whose priority level is not larger than the inconsistency degree. The deficiency of this approach is that it suffers form the drowning problem and it will result in undesirable conclusions. In contrast, the second approach weakens the conflicting terminology axioms instead of deleting them. The semantics of the approach is also discussed.

Our weakening method is based on cardinality restrictions. However, from the implementation point of view, the cardinality restriction is not very promising as no main-stream DL reasoners supports it yet. In a future work, we will explore other DL constructors such as nominals to weaken terminology axioms. Finally, to implement our approaches, an important problem is to detect GCIs and and assertions which are responsible for the conflict. Some existing techniques on debugging of unsatisfiable classes, such as debugging methods in [24, 21], may be adapted to pinpoint the conflicting axioms in a stratified DL knowledge base.

Acknowledgements Research presented in this paper was partially supported by the European Commission NeOn (IST-2006-027595, http://www.neon-project.org/) and the Knowledge Web (IST-2004-507842, http://knowledgeweb.semanticweb.org/) projects.

References

1. F. Baader and B. Hollunder. Embedding defaults into terminological knowledge representation formalisms, *Journal of Automated Reasoning*, 14(1):149-180, 1995.
2. F. Baader and B. Hollunder. Priorities on defaults with prerequisites, and their application in treating specificity in terminological default logic, *Journal of Automated Reasoning*, 15(1): 41-68, 1995.
3. F. Baader, M. Buchheit, and B. Hollander. Cardinality restrictions on concepts. *Artificial Intelligence*, 88: 195-213, 1996.
4. F. Baader, D.L. McGuiness, D. Nardi, and Peter Patel-Schneider. *The Description Logic Handbook: Theory, implementation and application*, Cambridge University Press, 2003.
5. S. Benferhat, C. Cayrol, D. Dubois, L. Lang, and H. Prade. Inconsistency management and prioritized syntax-based entailment. In *Proc. of IJCAI'93*, 640-645, 1993.
6. S. Benferhat, R.E. Baida, and F. Cuppens. A stratification-based approach for handling conflicts in access control. *Proceedings of 8th ACM Symposium on Access Control Models and Technologies*, 189-195, 2003.
7. S. Benferhat, and R.E. Baida. A stratified first order logic approach for access control. *International Journal of Intelligent Systems*, 19:817-836, 2004.
8. S. Benferhat, S. Kaci, D.L. Berre, and M.A. Williams. Weakening conflicting information for iterated revision and knowledge integration. *Artificail Intelligence*, vol. 153(1-2):339-371, 2004.

9. Dubois, D.; Lang, J.; and Prade, H. 1994. Possibilistic logic. In *Handbook of logic in Aritificial Intelligence and Logic Programming*, Volume 3. Oxford University Press, 439-513.
10. G. Flouris, D. Plexousakis and G. Antoniou. On applying the AGM theory to DLs and OWL, In *Proc. of ISWC'05*, 216-231, 2005.
11. G. Flouris, Z. Huang, J.Z. Pan, D. Plexousakis, and H. Wache. Inconsistencies, negation and changes in ontologies. In *Proc. of AAAI'06*, to appear, 2006.
12. T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web, *Scientific American*, 284(5):3443, 2001.
13. G. Governotori. Defeasible description logic, In G. Antoniou and H. Boley, editors, *Rules and Rule Markup Languages for the Semantic Web*, LNCS 3323, pages 98-112, 2004. Springer Verlag.
14. V. Haarslev and R. Möller, RACER System Description, In *Proc. of IJCAR'01*, 701-706, 2001.
15. P. Haase and J. Völker, Ontology Learning and Reasoning - Dealing with Uncertainty and Inconsistency In *Proc. of URSW'05*, 45-55. November 2005.
16. P. Haase, F. van Harmelen, Z. Huang, H. Stuckenschmidt, and Y. Sure. A framework for handling inconsistency in changing ontologies, In *Proc. of ISWC'05, LNCA3729*, 353-367, 2005.
17. S. Heymans and D. Vermeir. A defeasible ontology language, In R. Meersman and Z. Tari, editors, *Confederated International Conferences: CoopIS, DOA and ODBASE 2002*, number 2519 in LNCS, page 1033-1046, Berlin, 2002. Springer Verlag.
18. Z. Huang, F. van Harmelen, and A. ten Teije. Reasoning with inconsistent ontologies, In *Proc. of IJCAI'05*, 254-259, 2005.
19. J. Lang, "Possibilistic logic: complexity and algorithm" D. Gabbay, Ph. Smets, ed., *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, vol. 5, pp: 179-220, 2000.
20. T. Meyer, K. Lee, and R. Booth. Knowledge integration for description logics, In *Proc. of AAAI'05*, 645-650, 2005.
21. B. Parsia, E. Sirin and A. Kalyanpur. Debugging OWL ontologies, In *Proc. of WWW'05*, 633-640, 2005.
22. J. Pearl. System Z: A Natural Ordering of Defaults with Tractable Applications to Nonmonotonic Reasoning. *Proc. of TARK'90*, 121-135, 1990.
23. J. Quantz and V. Royer. A Preference Semantics for Defaults in Terminological Logics, In *Proc. of KR'92*, 294-305, 1992.
24. S. Schlobach, and R. Cornet. Non-standard reasoning services for the debugging of description logic terminologies, In *Proc. of IJCAI'03*, 355-360, 2003.
25. M. Schmidt-Schauß, and G. Smolka. Attributive Concept descriptions with complements, *Artificial Intelligence*, 48:1-26, 1991.
26. G. Qi, W. Liu and D.A. Bell. A revision-based approach to handling inconsistency in description logics. In *Proc. of NMR06*, 2006.
27. K. Wang, D. Billington, J. Blee, and G. Antoniou. Combining description logic and defeasible logic for the semantic web, In Grigoris Antoniou and Harlod Boley, editors, *RuleML 2004*, LNCS, Berlin, 2004. Springer Verlag.