2016 12th International Conference on Signal-Image Technology & Internet-Based Systems

Characterizing Non-Defeated Repairs in Inconsistent Lightweight Ontologies

Salem Benferhat
Artois University - Nord de France,
CRIL CNRS UMR 8188, France.
Email: benferhat@cril.univ-artois.fr

Zied Bouraoui
Computer Science and Informatics
Cardiff University, UK
Email: BouraouiZ@cardiff.ac.uk

Huma Chadhry
Technology Malaysiay
81310 UTM Johor Bahru, Johor, Malaysia
Email: huma.bicse@gmail.com

Mohd Shafry Bin Mohd Rahim Fc
Technology Malaysiay
81310 UTM Johor Bahru, Johor, Malaysia
Email: shafry@utm.my

Karim Tabia
Artois University - Nord de France,
CRIL CNRS UMR 8188, France.
Email: tabia@cril.univ-artois.fr

Abdelmoutia Telli
Artois University - Nord de France,
CRIL CNRS UMR 8188, France.
Email: telli@cril.univ-artois.fr

***Abstract*—In many applications, such as video processing, data are often provided by several and potentially conflicting sources having different reliability levels. This paper deals with the problem of handling inconsistency in lightweight ontologies when the set of assertions (facts) is prioritized. We propose a safe and efficient way to restore consistency using the concept of free assertions; assertions that are not involved in conflicts. Our approach allows the selection of one consistent assertional base, called a preferred repair. The last part of the paper contains an illustrative example using video data.**

***Keywords*-Inconsistency handling; Prioritized knowledge bases; Lightweight ontologies.**

I. Introduction

Ontologies are efficient frameworks to structure knowledge of a given domain. Typically, ontologies are expressed using description logics as knowledge bases containing two distinct components : a terminological base (called *TBox*) representing generic knowledge, and an assertional base (called *ABox*) containing facts or assertions. Recently, a particular interest was given to *Ontology-Mediated Query Answering* (e.g, [11], [15], [17]), which aims to exploit knowledge available in the TBox in order to get back as much complete answers as possible to queries from the ABox. In such a setting, the TBox is often assumed to be verified and validated while the ABox, typically provided in large quantities by various and unreliable sources, may be inconsistent with respect to the TBox.

In the presence of inconsistency, the standard query answering process becomes no longer appropriate, since all queries would be positively answered. This problem has led to a number of inconsistency-tolerant inference relations (e.g, [2], [9], [14]), called semantics, that aim to cope with inconsistency by providing meaningful answers to queries. These semantics are based on the notion of ABox repair [13], defined as an inclusion-maximal subset of the facts consistent with the TBox. This definition of a repair is closely related to the notion of database repair (e.g, [1]) or a maximally consistent subbase used in the propositional logic setting (*e.g.* [16]).

The seminal paper [13] introduced four semantics, namely AR, IAR, CAR and ICAR, for inconsistency-tolerant query answering in *DL-Lite* setting. Since then, several semantics have been proposed, e.g. ICR [10], k-sup and k-def [9], non-objection inference [7]. Recently, a general framework for inconsistency-tolerant semantics was proposed in [2]. This framework considers two key notions: *modifiers* and *inference strategies* where an inconsistency-tolerant query answering method is seen as made out of a modifier and an inference strategy. Finally, a comparaison of the main semantics (following three criteria : productivity, rational properties and complexity) was carried out in [3].

The problem dealt with in this paper is the one of handling inconsistency in lightweight ontologies when the ABox is prioritized. A prioritized ABox simply means an ABox involving assertions with different levels of reliability or priority. For instance, this can be the results of gathering data provided by different sources having different reliability levels. Dealing with inconsistency when the ABox is prioritized comes down first to compute *preferred* repairs (e.g, [4], [8]) and then using them for query answering. Using the whole set of preferred repairs to perform inference is a hard task (coNP-hard in data complexity) for DL-Lite, a tractable fragment of DLs.

In the presence of conflicting information, there is always a tradeoffs that one needs to reach between the expressiveness and computational issues. Having multiple repairs often allows to derive more conclusions than if only one repair is used. However, query answering from multiple repairs is inevitably more expensive than query answering from a single repair. In fact, reasoning from a single repair can be viewed as an approximation of reasoning from multiple repairs. Recently, a so-called non-defeated repair has been proposed in [4].

This paper proposes a characterization of the non-defeated repair using the concept of accepted assertions. This leads to select a unique preferred repair from an

978-1-5090-5698-9/16 $31.00 © 2016 IEEE
DOI 10.1109/SITIS.2016.53

inconsistent knowledge base. Selecting only one preferred repair is important since, once computed, it allows an efficient query answering.

The rest of this paper is organized as follows: Section II provides the needed background on *DL-Lite*. Section III presents some elementary concepts on inconsistency handling such as the concepts of conflicts, repairs and free assertions. Section IV presents the so-called non-defeated repair and its characterization using the concept of accepted assertions. Section V provides a potential application of our approach to video data. Section VI concludes the paper.

II. *DL-Lite* WITH PRIORITIZED ASSERTIONAL BASES

This section briefly recalls *DL-Lite* framework and introduces the concept of prioritised assertional base.

A. DL-Lite: A Brief Refresher

We only consider $DL\text{-}Lite_R$ language [12] and we will simply use *DL-Lite* instead of $DL\text{-}Lite_R$. Note that the results of this paper can be extended in a straightforward way to any tractable *DL-Lite* as far as computing ABox conflicts is done in polynomial time. This is true for $DL\text{-}Lite_{core}$ (a particular case of $DL\text{-}Lite_R$) and $DL\text{-}Lite_F$. The *DL-Lite* language is defined as follows:

$$\begin{array}{llllllll} R \longrightarrow & P & \mid & P^- & E \longrightarrow & R & \mid & \neg R \\ B \longrightarrow & A & \mid & \exists R & C \longrightarrow & B & \mid & \neg B \end{array}$$

where A is an atomic concept, P is an atomic role and P^- is the inverse of P. B (*resp.* C) is called basic (*resp.* complex) concept and role R (*resp.* E) is called basic (*resp.* complex) role. A knowledge base (KB) is a couple $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ where $\mathcal{T}$ is a TBox and $\mathcal{A}$ is an ABox.

A TBox includes a finite set of inclusion axioms on concepts and on roles respectively of the form: $B \sqsubseteq C$ and $R \sqsubseteq E$. The $ABox$ contains a finite set of atomic concepts and role assertions respectively of the form $A(a)$ and $P(a, b)$ where a and b are two individuals.

The semantics of a *DL-Lite* knowledge base is given in terms of interpretations. An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, .^{\mathcal{I}})$ consists of a non-empty domain $\Delta^{\mathcal{I}}$ and an interpretation function $.^{\mathcal{I}}$ that maps each individual a to $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, each A to $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and each role P to $P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. Furthermore, the interpretation function $.^I$ is extended in a straightforward way for concepts and roles as follows:

$$\begin{array}{rcl} A^I & \subseteq & \Delta^{\mathcal{I}} \\ P^I & \subseteq & \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \\ (P^-)^I & = & \{(y,x) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} | (x,y) \in P^{\mathcal{I}}\} \\ (\exists R)^I & = & \{x \in \Delta^{\mathcal{I}} | \exists y \in \Delta^{\mathcal{I}}\, such\, that\, (x,y) \in R^{\mathcal{I}}\} \\ (\neg B)^I & = & \Delta^{\mathcal{I}} \setminus B^{\mathcal{I}} \\ (\neg R)^I & = & \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \setminus R^{\mathcal{I}} \end{array}$$

An interpretation I is said to be a model of a concept (*resp.* role) inclusion axiom, denoted by $\mathcal{I} \models B \sqsubseteq C$ (*resp.* $\mathcal{I} \models R \sqsubseteq E$), if and only if $B^{\mathcal{I}} \subseteq C^{\mathcal{I}}$ (*resp.* $R^{\mathcal{I}} \subseteq E^{\mathcal{I}}$). Similarly, we say that an interpretation $\mathcal{I}$ is a model of a membership assertion $A(a)$ (*resp.* $P(a, b)$), denoted by $\mathcal{I} \models A(a)$ (*resp.* $\mathcal{I} \models P(a, b)$), if and only if $a^{\mathcal{I}} \in A^{\mathcal{I}}$ (*resp.* $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in P^{\mathcal{I}}$). A knowledge base $\mathcal{K}$ is called consistent if it admits at least one model, otherwise $\mathcal{K}$ is said to be inconsistent. A TBox $\mathcal{T}$ is said to be incoherent if there exists at least a concept C such that for each interpretation $\mathcal{I}$ which is a model of $\mathcal{T}$, we have $C^{\mathcal{I}}$=$\emptyset$.

B. Prioritized Assertional Bases

A prioritized assertional base (or a prioritized ABox), simply denoted by $\mathcal{A}$=$(\mathcal{S}_1, \cdots, \mathcal{S}_n)$, is a tuple of sets of assertions. The sets S_i's are called layers or strata. Each layer $\mathcal{S}_i$ contains the set of assertions having the same level of priority i and they are considered as more reliable than the ones present in a layer $\mathcal{S}_j$ when $j > i$. Hence, $\mathcal{S}_1$ contains the most important assertions while $\mathcal{S}_n$ contains the least important assertions. Throughout this paper and when there is no ambiguity we simply use 'prioritized *DL-Lite* KB $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$' to refer to a *DL-Lite* KB with a prioritized ABox of the form $\mathcal{A} = (\mathcal{S}_1, \cdots, \mathcal{S}_n)$.

This paper proposes methods to deal with prioritized and inconsistent *DL-Lite* KB. The input of our method is a prioritized *DL-Lite* knowledge base $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ with $\mathcal{A} = (\mathcal{S}_1, \cdots, \mathcal{S}_n)$. The output of our approach is a standard *DL-Lite* knowledge base $\mathcal{K}' = \langle \mathcal{T}, \mathcal{R} \rangle$, where $\mathcal{R}$ is not a prioritized assertional base (namely, just a set of assertions). $\mathcal{K}$ and $\mathcal{K}'$ have the same terminological base and $\mathcal{R}$ will be called a preferred repair. Then a query q is said to follow from $\mathcal{K}$ if it can be derived, using the standard *DL-Lite* inference, from $\mathcal{K}'$. Let us first recall important concepts for handling inconsistency when no priority is available between assertions.

III. INCONSISTENCY-TOLERANT INFERENCE FOR FLAT *DL-Lite* ASSERTIONAL BASES.

A. The Concept of Inclusion-Maximal Repair

We assume that the TBox $\mathcal{T}$ is coherent. Namely, elements of $\mathcal{T}$ are not questionable in the presence of conflicts. One natural way to deal with inconsistency is to first compute the set of maximal consistent subsets of assertions, called repairs, then use these repairs to perform inference (i.e. query answering). More formally, a repair is defined as follows:

Definition 1: Let $\mathcal{K}$=$\langle \mathcal{T}, \mathcal{A} \rangle$ be a flat (not prioritized) *DL-Lite* KB. A subbase $\mathcal{R} \subseteq \mathcal{A}$ is said to be a repair if $\langle \mathcal{T}, \mathcal{R} \rangle$ is consistent. And $\mathcal{R}$ is said to be a maximally inclusion-based repair of $\mathcal{K}$, denoted by *MAR*, if $\langle \mathcal{T}, \mathcal{R} \rangle$ is consistent and $\forall \mathcal{R}'$: $\mathcal{R} \subsetneq \mathcal{R}'$,$\langle \mathcal{T}, \mathcal{R}' \rangle$ is inconsistent.

According to the definition of *MAR*, adding any assertion f from $\mathcal{A} \setminus \mathcal{R}$ to $\mathcal{R}$ entails the inconsistency of $\langle \mathcal{T}, \mathcal{R} \cup \{f\} \rangle$. Moreover, the maximality in *MAR* is used in the sense of set inclusion. We denote by *MAR*($\mathcal{A}$) the set of *MAR* of $\mathcal{A}$ with respect to $\mathcal{T}$. The definition of *MAR* meets the definition of $ABox$ repair proposed in [13].

Using the notion of repair, coping with inconsistency in flat *DL-Lite* knowledge bases can be done by applying standard query answering either using the whole set of repairs (universal entailment or AR-entailment [13]) or only using one repair.

B. Free Assertions and Conflict Sets

Let us introduce the notion of a conflict. It is a minimal subset $\mathcal{C}$ of assertions of $\mathcal{A}$ such that $\mathcal{K} = \langle \mathcal{T}, \mathcal{C} \rangle$ is inconsistent. Definition 2 introduces the notion of defeated assertions.

Definition 2: Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a *DL-Lite* KB. A subbase $\mathcal{C} \subseteq \mathcal{A}$ is said to be an assertional conflict of $\mathcal{K}$ iff $\langle \mathcal{T}, \mathcal{C} \rangle$ is inconsistent and $\forall\ f \in \mathcal{C}$, $\langle \mathcal{T}, \mathcal{C} \setminus \{f\} \rangle$ is consistent. An assertion f is said to be defeated if there exists a conflict $\mathcal{C}$ such that $f \in \mathcal{C}$.

From Definition 2, removing any fact f from $\mathcal{C}$ restores the consistency of $\langle \mathcal{T}, \mathcal{C} \rangle$. In *DL-Lite*, when the TBox is coherent, a conflict involves exactly two assertions. We denote by $\mathcal{C}(\mathcal{A})$ the set of conflicts in $\mathcal{A}$. A nice feature of *DL-Lite* is that computing the set of conflicts is done in polynomial time. We now introduce the notion of non-conflicting or free elements.

Definition 3: Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be *DL-Lite* KB. An assertion $f \in \mathcal{A}$ is said to be *free* or non-defeated if and only if $\forall c \in \mathcal{C}(\mathcal{A}) : f \notin c$.

Intuitively, *free* (or non-defeated) assertions correspond to elements that are not involved in any conflict. We denote by *free*($\mathcal{A}$) the set of *free* assertions in $\mathcal{A}$. The notion of *free* elements is originally proposed in [5] in a propositional logic setting. Within a *DL-Lite* setting, $free(\mathcal{A})$ is computed in polynomial time thanks to the fact that computing conflicts is done in polynomial time.

The set of free elements can be obtained using the set of maximally inclusion-based repairs as follows:

$$\text{free}(\mathcal{A}) = \bigcap_{X \in MAR(\mathcal{A})} X.$$

For flat *DL-Lite* knowledge bases, the $free$-entailment (entailment based on free assertions) is equivalent to the IAR-entailment proposed in [13]. Besides, the concept of $free$ entailment have been introduced in [5].

IV. The Concept of Non-Defeated Prioritized Assertions

The aim of this section is to propose a characterization of the so-called non-defeated repair. This non-defeated repair has been recently proposed in [4] in the context of handling inconsistency in a prioritized *DL-Lite* knowledge bases.

The $free$ entailment repair can be viewed as a safe way to deal with inconsistency. The term *safe* is used by opposition to the term *risky* or *adventurous* with respect to the derived conclusions. This section provides an extension of the notion of $free$ or non-defeated elements, given in Definition 2 and 3 when assertional bases are prioritized.

When there is no priority between assertions, an assertion f is said to be defeated as soon as it is conflicting with some rule in $\mathcal{T}$ or with some assertion g in $\mathcal{A}$. When there is a priority relation between elements of $\mathcal{A}$, an assertion f may be considered as non-defeated even if it is conflicting with some assertion g. This happens when g is considered as having less priority (or being less important) than f. The following definition formally introduces the notion of non-defeated assertions when the ABox is prioritized:

Definition 4: Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* KB. Let f be an assertion of $\mathcal{A}$. Then the assertion $f \in S_i$ is said to be defeated if:

- $\langle \mathcal{T}, \{f\} \rangle$ is inconsistent, or
- There exists an assertion $g \in S_j$ such that $\langle \mathcal{T}, \{f, g\} \rangle$ is inconsistent and $j < i$.

Basically, an assertion f is said to be defeated if it solely contradicts the terminological base (first condition in Definition 4). It is also considered as defeated if it is contradicted by some higher priority assertion. In the case where an assertion is not defeated, it is called accepted. More precisely:

Definition 5: Let $\mathcal{K}$=$\langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* KB. Let f be an assertion of $\mathcal{A}$. Then the assertion $f \in S_i$ is said to be non-defeated if and only if it is not defeated (in the sense of Definition 4).

It turns out that the set of accepted assertions is exactly equal to the so-called non-defeated repair introduced in [4]. The idea in the construction of non-defeated repair is to iteratively retrieve, layer per layer, the set of $free$ elements. More precisely:

Definition 6: Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* KB. We define the non-defeated repair, denoted by $nd(\mathcal{A}) = \mathcal{S}'_1 \cup \ldots \cup \mathcal{S}'_n$, as follows:

$$\forall i = 1, .., n : \mathcal{S}'_i = free(\mathcal{S}_1 \cup \ldots \cup \mathcal{S}_i).$$

Namely, $nd(\mathcal{A}) =$

$$free(\mathcal{S}_1) \cup free(\mathcal{S}_1 \cup \mathcal{S}_2) \cup \ldots \cup free(\mathcal{S}_1 \cup \ldots \cup \mathcal{S}_n).$$

Hence an important result of this paper is to provide a characterization of the so-called non-defeated repair using the concept of non-defeated beliefs. The following proposition summarizes the first main contribution of this paper.

Proposition 1: Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a prioritized *DL-Lite* KB. Then:

$$nd(\mathcal{A}) = \{f : f \text{ is a non-defeated assertion}\}$$

where $nd(\mathcal{A})$ is given in Definition 6 and the concept of non-defeated assertion is given in Definition 5.

Proof: Let $f \in nd(\mathcal{A})$. Assume that $f \in S_i$ is defeated. By Definition 4, we have two cases:

- Either $\langle \mathcal{T}, \{f\} \rangle$ is inconsistent, but this contradicts the fact $f \in free(S_1 \cup \cdots \cup S_n)$.
- Or there exists an assertion $g \in S_j$ such that $\langle \mathcal{T}, \{f, g\} \rangle$ is consistent and $j < i$.

First this means that $f \notin free(S_1) \cup \cdots \cup free(S_1 \cup \cdots \cup S_{i-1})$ (since $f \in S_i$). This also means that $f \notin free(S_1 \cup \cdots \cup S_i) \cup \cdots free(S_1 \cup \cdots \cup S_n)$. Since $\langle \mathcal{T}, \{f, g\} \rangle$ is inconsistent. Hence, this contradicts the fact that $f \in nd(\mathcal{A})$.

For the converse, assume that $f \in S_i$ is non-defeated but $f \notin nd(\mathcal{A})$. This means that $f \notin free(S_1 \cup \cdots \cup S_i)$. This also means that there exists a conflict $\mathcal{C}$ of $\langle \mathcal{T}, S_1 \cup \cdots \cup S_i \rangle$ such that $f \in \mathcal{C}$. But this contradicts the fact that f is non-defeated.

■

The definition of non-defeated subbase is an adaptation of the definition proposed in [6] within a propositional logic setting. However, contrarily to the propositional setting, as we will see later, non-defeated repair can be applied on $\mathcal{A}$ or its deductive closure $cl(\mathcal{A})$ which leads to two different ways to select a single preferred repair. Besides the non-defeated repair is computed in polynomial time in a *DL-Lite* setting while its computation is hard in a propositional logic setting.

We are now ready to provide a simple proof where the set:

$$X = \{f : f \in \mathcal{A} \text{ and } f \text{ is non-defeated}\} \text{ is consistent.}$$

The proof is immediate.
Assume that X is inconsistent. We have two cases:

- Either there exists $f \in X$ such that $\langle \mathcal{T}, \{f\} \rangle$ is inconsistent. This contradicts the fact f is non-defeated.
- Or there exists $f \in X$ and $g \in X$ such that $\langle \mathcal{T}, \{f, g\} \rangle$ is inconsistent. This means that either f is defeated (if g is as prioritary as f) or g is defeated (if f is as prioritary as g). This is impossible by the contradiction of X.

The above proposition clearly provides a nice characterisation of the non-defeated repair proposed in [4] using the concept of non-defeated assertion.

V. Application of Non-Defeated Repairs to Video Processing

This section contains a brief description of a potential application of our approach to classifying and solving conflicts in a collection of dance videos issued from different sources. Motion in video carries important information which is of a multi-fold nature. Motion sometimes needs to be interpreted for understanding or anticipating any immediate reaction to the motion. Hence, the significance of motion perception in a video is seen in several current systems and it remains to be an active research area.

Movements in dances, especially traditional ones, encompass different types of information; sacred rituals, social dialogue or cultural expressions and more. This calls for the need to completely and precisely describe and process motion information and make it digitally available for further processing. Several applications can be reached by dance video automatic annotation, such as dance video retrieval, classification and indexing of video databases or animation of dances using models of stored information.

A video is a sequence of frames which extend over time. A set of frames can be referred to as a dance segment once a human completes a dance step in a composed dance. Sequence of actions that display a motion which is isolated and makes up a dancing vocabulary can be referred to as a dance step. Improvisation in a dance refers to simultaneous dance moves, which are not previously composed.

Dance moves need to categorized as expressions such as Time Steps in *Cha-Cha-Cha* or *Passo Basico* in Samba dance [1]. Our aim is to explore how different dance steps in a traditional Malaysian dance video can be categorized and described to extract knowledge from the video. For instance, in the dance Inang, we can describe the step *Side Bend* as a gesture in which the upper body and the raised arms arch in left or right direction and the feet tap toe in the opposite direction, creating the form of an arch on right and then left side. The dance step with the name *Side Bend* are segmented from the entire video and this segment of dance movement is labeled as *Side Bend*.

In the presence of a collection of dance segments, issued from different sources (cameras for instance), the problem of conflicting data assertions may arise due for instance to camera positions or persepctives, occlusions, etc. To illustrate this situation, let us consider an example where we assume that we only have the following concepts:

- Forwardmove,
- Backgroundmove,
- FastForwardmove,
- SlowForwardmove,

[1] http://www.dancadesalao.com/agenda/ingles.php

- FastBackgroundmove,
- SlowBackgroundmove,
- StartForwardmove, and
- StartBackgroundmove.

These concepts concern dance segments that represent direct moves. The concept Forwardmove (resp. SlowForwardmove and FastForwardmove) lists the set of dance segments that represent a forward move (resp. a slow and a fast forward move). Similarly, The concept Backgroundmove (resp. Slowbackgroundmove and Fastbackgroundmove) lists the set of dance segments that represent a background move (resp. a slow and a fast background move). The concept StartForwardmove (resp. StartBackgroundmove) lists the set of dance segments that represent a starting forward (resp. background) move in a video.

We also assume that we only have one relation:

- DanceSegment: gives for each dance video the list of dance segments

The terminological base is expressed by the following TBox:

Paxiom 1:	FastForwardmove $\sqsubseteq$ Forwardmove
Paxiom 2:	SlowForwardmove $\sqsubseteq$ Forwardmove
Paxiom 3:	StartForwardmove $\sqsubseteq$ Forwardmove
Paxiom 4:	FastBackgroundmove $\sqsubseteq$ Backgroundmove
Paxiom 5:	SlowBackgroundmove $\sqsubseteq$ Backgroundmove
Paxiom 6:	StartBackgroundmove $\sqsubseteq$ Backgroundmove
Paxiom 7:	StartForwardmove $\sqsubseteq$ SlowForwardmove
Paxiom 8:	StartBackgroundmove $\sqsubseteq$ SlowBackgroundmove
Paxiom 9:	Forwardmove $\sqsubseteq \exists$ DanceSegment$^-$
Paxiom 10:	backgrounddmove $\sqsubseteq \exists$ DanceSegment$^-$
Naxiom 1:	Forwardmove $\sqsubseteq \neg$ Backgroundmove

The ten first axioms (Paxiom 1 - Paxiom 10) are positive axioms that do not involve a negative symbol while the last axiom (Naxiom 1) is a negative axiom that expresses an integrity constraint that the knowledge base should satisfy.

The meaning of six first axioms is immediate. For instance, Paxiom 1 expresses that a fast forward move is a forward move.
Paxiom 7 (resp. Paxiom 8) expresses that a start forward move (resp. start background move) should be a show move.
Paxiom 9 (resp. Paxiom 10) expresses that each forward move (resp. background move) is contained in at least a dance segment.
The least axiom expresses that a forward move is not a background move.

Assume now that we have four individuals:

i) Two dances segments s_1 and s_2, and
ii) Two videos v_1 and v_2.

Assume that we only have four assertions facts given by the ABox $\mathcal{A} = (S_1, S_2)$ with:
S_1={ Backgroundmove(s_1), DanceSegment(v_1, s_1)},
S_2={ StartForwardmove(s_1), DanceSegment(v_1, s_2)}, and
S_3={ SlowBackgroundmove(s_1), FastBackgroundmove(s_2)}.

This assertional base is assumed to be provided by different sources (different cameras for example) having different reliability levels. This knowledge base is clearly inconsistent. Indeed, using the assertion StartForwardmove(s_1) and the positive inclusion axiom Paxiom 3 one can derive Forwardmove(s_1). This derived fact together with the assertion Backgroundmove(s_1) contradict the negative axiom Naxiom 1. Without the use of priorities, the free entailment will simply lead to:

$$Free(\mathcal{A}) = \{ \text{DanceSegment}(v_1, s_1), \text{DanceSegment}(v_1, s_2), \text{FastBackgroundmove}(s_2)\}.$$

Namely, only assertions that are not conserned by the conflict are preserved. When there is a priority relation between assertional facts, one can go one step further in solving conflicts. In our example, Backgroundmove(s_1) is assumed to be issued from a source more reliable than the one that delivers the assertion StartForwardmove(s_1). Hence, Backgroundmove(s_1) is non-defeated assertion. Hence, the set of non-defeated beliefs is:

$$nd(\mathcal{A}) = \{ \text{Backgroundmove}(s_1), \text{DanceSegment}(v_1, s_1), \text{DanceSegment}(v_1, s_2), \text{FastBackgroundmove}(s_2)\}.$$

Clearly, $nd(\mathcal{A})$ is larger than $Free(\mathcal{A})$ since it allows to recover Backgroundmove(s_1), due to the presence of priorities in the assertional base. The nice feature of our approach is once $nd(\mathcal{A})$ is computed, query the initial inconsistent knowledge base can be done efficiently.

VI. Conclusion

This paper proposed a characterisation of the so-called non-defeated repair proposed in [4] using the notion of non-defeated assertions. We provided a simple proof of the consistency of the set of non-defeated assertions. The paper also contained an example showing a potential application of our approach to classifying dance videos in a presence of multi-source conflicting information.

A future work is to apply our approach to query traditional Malaysian dance videos in presence of possible conflicts.

VII. Acknowledgments

This work is supported by the european project H2020 Marie Sklodowska-Curie Actions (MSCA) research and Innovation Staff Exchange (RISE): AniAge (High Dimensional Heterogeneous Data based Animation Techniques for Southeast Asian Intangible Cultural Heritage.

References

[1] M. Arenas, E. Leopoldo. Bertossi, and J. Chomicki. Consistent query answers in inconsistent databases. In *Proceedings of the Eighteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, Philadelphia, Pennsylvania, USA*, pages 68–79, 1999.

[2] Jean-François Baget, Salem Benferhat, Zied Bouraoui, Madalina Croitoru, Marie-Laure Mugnier, Odile Papini, Swan Rocher, and Karim Tabia. A general modifier-based framework for inconsistency-tolerant query answering. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Fifteenth International Conference, KR 2016, Cape Town, South Africa, April 25-29, 2016.*, pages 513–516, 2016.

[3] Jean-François Baget, Salem Benferhat, Zied Bouraoui, Madalina Croitoru, Marie-Laure Mugnier, Odile Papini, Swan Rocher, and Karim Tabia. Inconsistency-tolerant query answering : Rationality properties and computational complexity analysis. In *Logics in Artificial Intelligence - 15th European Conference, JELIA 2016*, 2016.

[4] S. Benferhat, Z. Bouraoui, and K. Tabia. How to select one preferred assertional-based repair from inconsistent and prioritized DL-Lite knowledge bases? In Qiang Yang and Michael Wooldridge, editors, *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 1450–1456. AAAI Press, 2015.

[5] S. Benferhat, D. Dubois, and H. Prade. Representing default rules in possibilistic logic. In *Knowledge Representation and Reasoning*, pages 673–684. Morgan Kaufmann, 1992.

[6] S. Benferhat, D. Dubois, and H. Prade. *Some syntactic approaches to the handling of inconsistent knowledge bases : A comparative study. Part 2 : the prioritized case*, volume 24, pages 473–511. Physica-Verlag, Heidelberg, 1998.

[7] Salem Benferhat, Zied Bouraoui, Madalina Croitoru, Odile Papini, and Karim Tabia. Non-objection inference for inconsistency-tolerant query answering. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 3684–3690, 2016.

[8] M. Bienvenu, C. Bourgaux, and F. Goasdoué. Querying inconsistent description logic knowledge bases under preferred repair semantics. In *AAAI*, pages 996–1002, 2014.

[9] M. Bienvenu and R. Rosati. Tractable approximations of consistent query answering for robust ontology-based data access. In *International Joint Conference on Artificial Intelligence*. IJCAI/AAAI, 2013.

[10] Meghyn Bienvenu. On the complexity of consistent query answering in the presence of simple ontologies. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[11] Meghyn Bienvenu and Magdalena Ortiz. Ontology-mediated query answering with data-tractable description logics. In *Reasoning Web. Web Logic Rules - 11th International Summer School 2015, Berlin, Germany, July 31 - August 4, 2015, Tutorial Lectures*, pages 218–307, 2015.

[12] 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.

[13] D. Lembo, M. Lenzerini, R. Rosati, M. Ruzzi, and D. Fabio Savo. Inconsistency-tolerant semantics for description logics. In Pascal Hitzler and Thomas Lukasiewicz, editors, *RR*, volume 6333 of *LNCS*, pages 103–117. Springer, 2010.

[14] D. Lembo, M. Lenzerini, R. Rosati, M. Ruzzi, and D. Fabio Savo. Inconsistency-tolerant query answering in ontology-based data access. *Journal of Web Semantics*, 33:3–29, 2015.

[15] M. Lenzerini. Ontology-based data management. In *Proceedings of the 6th Alberto Mendelzon International Workshop on Foundations of Data Management 2012*, pages 12–15, 2012.

[16] R. Nicholas and M. Ruth. On inference from inconsistent premisses. *Theory and Decision*, 1(2):179–217, 1970.

[17] A. Poggi, D. Lembo, D. Calvanese, G. De Giacomo, M. Lenzerini, and R. Rosati. Linking data to ontologies. *J. Data Semantics*, 10:133–173, 2008.