A Framework for Handling Revisions in Distributed Ontologies

Faezeh Ensan
Faculty of Computer Science
University of New Brunswick
faezeh.ensan@unb.ca

Ebrahim Bagheri
Institute for Information Technology
National Research Council Canada
ebrahim.bagheri@nrc-cnrc.gc.ca

ABSTRACT

One of the important issues in ontology management is handling incoming updates and dealing with possible inconsistencies that they may induce. This is even more challenging in the context of a modular and distributed representation, because of the side-effects of the propagation of changes to the other connected or related ontologies. In this paper, we analyze the notion of ontology revision in a distributed ontology representation. We introduce a revision operator for distributed ontologies and show that it satisfies important postulates for knowledge base revision. In addition, based on a tableau algorithm for $\mathcal{ALC}$ ontologies, we propose an algorithm for applying the received changes and revising the original ontology through the proposed operator.

Categories and Subject Descriptors

I.2.4 [**Knowledge Representation Formalisms and Methods**]: Representation languages, Representations; H.4.m [**Information Systems Applications**]: Miscellaneous; H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Theory

Keywords

Modular Ontologies, Revision, Description Logic

1. INTRODUCTION

Description logic ontologies provide a strong basis for the representation of and reasoning over structured knowledge. More recently, modular and distributed ontologies [11, 15, 5] have received great attention. Modularity is an important issue in the development and management of ontologies due to the need for better knowledge reuse, more efficient reasoning, and applicable ways for integrating distributed and independent ontology components, especially in large and complex domains [21]. An important aspect of ontology management in such domains concerns with handling incoming changes and dealing with possible inconsistencies that they may induce. This issue is more challenging in the context of a modular representation for ontologies, because of the side-effects of the propagation of changes to the connected ontologies. For instance, an incoming modification in the knowledge base of an ontology may lead to inconsistencies in the other ontologies, while keeping the original knowledge base consistent.

The issues related to knowledge change as well as dealing with their consequential inconsistencies have been considerably addressed in the belief revision community, where fundamental postulates [2] and principles such as 'the minimality of change' or 'irrelevance of syntax' [14] have been introduced and discussed for the purpose of revising and updating belief sets and belief bases. The principle of minimal change in belief sets has been characterized by a model-theoretic approach in [14]. Accordingly, for revising a belief set ψ by a sentence μ, the revision methods select from the models of μ those that are *closest* to models of ψ. A *faithful* pre-ordering relationship among the models of the knowledge set, specifies the closest models.

Nonetheless and despite the importance of updating and revising DL based ontologies, there are few applicable works that analyze change and revision and their consequences in ontologies and even more rarely modular and distributed representations of DL knowledge bases have been addressed in this regard. The few notable works include the proposal of [18] that recasts the weakening approach of [4] for description logics, [6, 7] that compute and approximate instance-level updates for DL-Lite and the algorithm which is proposed in [21] that *recompiles* the connected ontologies to an ontology where *harmful* changes accrue.

The objective of this paper is to investigate the revision of ontologies in a distributed representation. We analyze how a modification which is originated from an external ontology may lead to inconsistencies in local knowledge base. We illustrate our approach for 'revising' instead of 'updating' distributed ontologies. The main contributions of this paper can be enumerated as follows:

- We provide an exact definition for the revision of distributed ontologies. We consider revising an ontology by a modification, that originated from the foreign ontologies from where it imports its external symbols. The modification is itself represented as an ontology. We define the principle of *persistency for local knowledge* in order to ensure that local knowledge of an ontology is not effected by any external modification.
- We define an operator for revising distributed ontologies. We show that it supports some of the important AGM postulates for knowledge revision. In addition, it satisfies the introduced principle of persistency for local knowledge.
- We introduce an algorithm for revising $\mathcal{ALC}$ knowledge bases

by a *plain* modification. We show correctness and completeness of the the proposed algorithm in the context of the introduced semantics for revision.

2. PRELIMINARIES AND BACKGROUND

A DL knowledge base is defined as $\psi = \langle \mathcal{T}, \mathcal{A} \rangle$, where $\mathcal{T}$ denotes TBox and comprises of a set of general inclusion axioms and $\mathcal{A}$ stands for ABox and consists of a set of instance assertions. The signature of an ontology is defined as a set of all concept names (C_N), role names (R_N) and individuals (I_N) which are included its knowledge base. The semantic of a DL is defined by an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}}$ is a non-empty set of individuals and $\cdot^{\mathcal{I}}$ is a function which maps each $C \in C_N$ to $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, each $R \in R_N$ to $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ and each $a \in I_N$ to an $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. An interpretation $\mathcal{I}$ satisfies a TBox axiom $C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, satisfies an ABox assertion $C(a)$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$ and an ABox assertion $R(x, y)$ iff $\langle x^{\mathcal{I}}, y^{\mathcal{I}} \rangle \in R^{\mathcal{I}}$. An interpretation $\mathcal{I}$ is a model of a knowledge base Ψ if it satisfies every TBox axiom and ABox assertion of Ψ. A knowledge base is consistent iff it has a model. A concept C is *satisfiable* if there is a model $\mathcal{I}$ for Ψ such that $C^{\mathcal{I}} \neq \emptyset$.

Tableau algorithms [3] are well known methods for deciding satisfiability of concepts in a DL knowledge base by constructing a model for them. These models includes a set of nodes corresponding to individuals and a set of edges corresponding to roles. Given a set of nodes that have been initialized by a set of formulas, tableau algorithms repeatedly apply a set of expansion rules and construct new nodes, edges or decompose the formulas to the new flatter ones. The construction terminates either when none of the rules can be applied anymore, or when a contradiction, *clash*, shows up in the node's assigned formulas.

In this paper, for revising distributed ontologies, we use the notion of labeled tableau from [20]. A labeled tableau is a tableau whose constituent formulas have corresponding labels. For a node x, $\mathcal{L}(x)$ denotes the set of assigned formulas to the node. Each axiom ($\mathtt{A} \sqsubseteq \mathtt{B}$) in the TBox, has an associated label l. For creating a tableau model for a knowledge base, for all nodes x, we assign $\mathcal{L}(x)$=$\mathcal{L}(x) \cap \bigcup_{all\ axioms\ in\ TBox} (\neg \mathtt{A} \sqcup \mathtt{B})^{l_i}$. A labeled formula is in the form of $(C)^l$, where l is the labels of a set of axioms. A labeled branch is a branch whose nodes have labeled formulas.

In their seminal works [2], Alchourron, Gardenfors and Makinson provide a set of eight postulates for the revision of deductively closed knowledge sets. They argue that these postulates should be complied by every reasonable revision operator. These postulates need a clear definition of the negation of sentences. For description logics, the negation of an ABox axiom like $A(a)$ is clearly defined as $\neg A(a)$. However as pointed out in [8], the negation of TBox axioms like $A \sqsubseteq B$ is not precisely defined nor universally agreed among the DL community. In order to encounter this issue, we use the modified version of AGM postulates from [16] for revising deductively closed theories shown below:

R1 $\psi \circ_{do} \mu \models \mu$

R2 if $\psi \cup \mu$ is consistent, then $\psi \circ_{do} \mu \equiv \psi \cup \mu$

R3 if μ is consistent, $\psi \circ_{do} \mu$ is also consistent.

R4 if $\psi \equiv \psi'$ and $\mu \equiv \mu'$, then $\psi \circ_{do} \mu \equiv \psi' \circ_{do} \mu'$

R5 $(\psi \circ_{do} \mu) \cup \mu' \models \psi \circ_{do} (\mu \cup \mu')$

R6 if $(\psi \circ_{do} \mu) \cup \mu'$ is satisfiable then $\psi \circ_{do} (\mu \cup \mu') \models (\psi \circ_{do} \mu) \cup \mu'$

where given a knowledge set ψ, modification μ and ϕ, $\psi \circ \mu$ denotes the revision of ψ by μ by the revision operator $\circ$.

3. OVERVIEW OF OUR APPROACH

In order to illustrate the semantics of knowledge change in distributed DL knowledge bases, we address our main principles in the following three subsections. First we illustrate our characterization of distributed DL knowledge bases and the kind of knowledge change we deal with, in this paper. Second we explain our distinction between local and external symbols of a knowledge base when applying a modification and resolving consequential inconsistencies. Third, we elucidate our revision approach instead of an update one for modifying knowledge bases.

3.1 Distributed Knowledge Base Modification

Numerous formalisms have been proposed for representing distributed and modular DL ontologies [5, 1]. Despite their different approaches for integration and reasoning, they mostly share the canonical feature that an ontology in a distributed representation comprises of local and external symbols. External symbols are concepts, roles or individuals that are imported from other ontologies. Formally, for a DL ontology ψ, $\mathrm{Sig}(\psi)$ is equal to the disjoint union of its external signature, denoted by $\mathrm{Ext}(\psi)$, and local signature, represented by $\mathrm{Loc}(\psi)$. As a motivating example, consider a 'Tourism' ontology, which models the tourism attractions of North American destinations. Assume that it imports two concepts `Small-City` and `Polluted-City` from the'City' ontology, an external DL knowledge base, which provides information about countries, cities and postal addresses in North America. Hence, $\mathrm{Ext}(\psi) = \{$`Small-City`, `Polluted-City`$\}$. Suppose that the 'Tourism' knowledge base be as follows:

`Small-City` $\sqsubseteq$ ∃hasSightseeing.`ForSeniors-Sightseeing`
`ForSeniors-Sightseeing` $\sqsubseteq$ ∀isCloseTo.¬Polluted-City
hasSightseeing=Inv(isCloseTo)

Accordingly, a `Small-City` has at least one sightseeing attraction for seniors. In addition, senior sightseeing attractions are not close to any `Polluted-City`.

Based on the above knowledge base, it can be observed that the Tourism ontology entails a subsumption between its external elements, `Small-City` $\sqsubseteq \neg$`Polluted-City`. This relationship may not hold in the City ontology where these symbols are imported from. Cuenca Grau and Kutz [11] argue that producing new subsumptions between imported symbols makes the integration of ontologies *'unsafe'*. In the context of knowledge revision, the following example explicates an instance of potential challenges in such *'unsafe'* knowledge bases. Assume that the relationship:
`Small-City` $\sqsubseteq \neg$`Polluted-City`
has not been asserted in the City ontology. In addition, assume that the City ontology expands with the ABox assertions:
`Small-City`(Wayne), `Polluted-City`(Wayne).

Obviously, this expansion is a monotonic change for the City ontology. However, it makes the Tourism ontology inconsistent. The notion of conservative extension is employed in [10] to put some limitations on the definition of modular ontologies in order to prevent unsafe integrations.

Alternatively, in this paper we do not put any limitations on the definition of distributed ontologies. Based on our assumption, each DL knowledge base may include axioms and assertions that comprise of a free combination of local and external symbols. We provide semantics for revising such DL-based ontologies. A DL knowledge base is revised by a set of axioms and assertions called *modifications*. Since in this paper, we deal with distributed knowledge change, we consider those modifications that originate from expansion, revision or contradiction of external ontologies. Definition 3.1 gives a formal definition for modifications.

DEFINITION 3.1. *Let ψ be a knowledge base of a DL-based ontology and μ be a set of axioms and assertions. μ is a modification for ψ iff:*

- *μ is consistent;*
- $\mathrm{Sig}(\mu) \subseteq \mathrm{Ext}(\psi)$.

We denote all implications of a knowledge base ψ by $\mathrm{Cn}(\psi)$. $\mathrm{Cn}(\psi)^{Loc}$ is a subset of $\mathrm{Cn}(\psi)$ whose signature is local. From these definitions, its not hard to see that:

COROLLARY 3.1. *Let ψ be a DL knowledge base of a DL-based ontology and μ be a modification on it. ψ is consistent with $Cn^{Loc}(\psi)$.*

3.2 External vs. Local Knowledge Change

Given a DL knowledge base ψ and a modification μ, the question is how to revise ψ by μ, considering the distinction between external and local axioms. We use the terms of *knowledge* and *belief* for distinguishing different parts of a knowledge base. Accordingly, the content of a knowledge base can be prioritized as following:

- $\mathrm{Cn}^{Loc}(\psi)$ is the *knowledge* of ψ about its local symbols. It must not change after applying a modification that has come from an external source. For instance, in the Tourism example, the axiom hasSightseeing=Inv(isCloseTo) is the knowledge of the Tourism ontology about its local roles and must not change when new information such as 'Wayne is both a small and polluted city' has been asserted.
- $\mathrm{Cn}(\psi) - Cn^{Loc}(\psi)$ is the *belief* of ψ about the relationship between external or between internal and external symbols. This belief can be updated by applying a modification but must experience a *minimal change*. In the Tourism example, the axiom `Small-City` $\sqsubseteq \neg$`Polluted-City` is the belief of the Tourism ontology about its external symbols. However, this belief can be revised by observing the new fact that Wayne is a city that is both small and polluted.

The first item in the above list points out to what we name the principle of *persistency* for local knowledge. The following postulate captures this principle.

DEFINITION 3.2. *Let ψ be a DL knowledge base, α an axiom or assertion, μ be a modification according to Definition 3.1 and $\psi \circ \mu$ denote revising ψ by μ.*

(DO) if $\psi \models \alpha$ and $\mathrm{Sig}(\alpha) \subseteq \mathrm{Loc}(\psi)$, then $\psi \circ \mu \models \alpha$

Informally stated, the principle of persistency for local knowledge prevents the local knowledge of ψ to change once an inconsistent modification is received.

3.3 Revision vs. Update Distributed DL Knowledge Bases

'Update' and 'revision' are two different approaches for changing a knowledge base that have been distinguished in the literature [13] and their fundamental differences have been analyzed. From existing work on applying modification in description logics, [6, 7] choose an update approach. Accordingly, they propose a semantic that updates each possible interpretation of the knowledge base. Here and for changing distributed DL knowledge bases, we follow a revision approach. This means that for changing a knowledge base ψ based on an incoming modification μ, we consider methods that select those models of μ that are closest to the models of ψ. The intuition behind this approach for distributed ontologies is as follow: The models of ψ are the possible worlds that give semantics to the local and external symbols. Even though the models of ψ have their own interpretation of external elements, the ontologies where these external symbols originate from and from whom these elements have been imported may expose new information that change the perception of ψ. Observably, the possible interpretations of ψ, which had interpreted external elements very differently from what the new incoming knowledge is specifying, must be eliminated. For instance, in the case of the Tourism Example, the modification μ changes the perception of the Tourism ontology about small cities and polluted cities, i.e., a small city can be a polluted city as well.

In the following section we illuminate the semantic of distributed DL knowledge base revision.

4. SEMANTICS OF REVISION

In order to define appropriate semantics for the revision of ontologies in a distributed representation, we first define the notion of distance between interpretations of a knowledge base. We assume that all interpretations have the same domain and interpret individual names to the same elements of the domain.

DEFINITION 4.1. *For a knowledge base ψ, let $P_i, 1 \leq i \leq n$, denotes all classes and roles in Sig(ψ) and $d \in Mod(\psi)$ be a model for ψ. Furthermore, allow for two given sets A and B, $A \ominus B$ abbreviate $(A \cup B) \setminus (A \cap B)$. For an interpretation $\mathcal{I}$ of ψ, we define*

$$Dist(\mathcal{I}, d) = \begin{cases} \infty & if\ CONDITION^* \\ \langle |P_1^{\mathcal{I}} \ominus P_1^d|, ..., |P_n^{\mathcal{I}} \ominus P_n^d| \rangle & otherwise. \end{cases}$$

where $CONDITION^*$ is $(\exists P \in \mathrm{Loc}(\psi)\ :\ |P^{\mathcal{I}} \ominus P^d| \neq 0)$

Example 4.1 shows a case of the distance between two interpretations of a knowledge base.

EXAMPLE 4.1. *Let $\psi = \{A \sqsubseteq \neg B\}$ and $\triangle = \{\alpha, \beta, \rho\}$. Consider two interpretations $\mathcal{I}$ and $\mathcal{J}$ such that $A^{\mathcal{I}} = \{\alpha\}$, $B^{\mathcal{I}} = \{\rho, \beta\}$ and $A^{\mathcal{J}} = \{\alpha\}$, $B^{\mathcal{J}} = \{\alpha\}$. According to Definition 4.1, $Dist(\mathcal{I}, \mathcal{J}) = \langle 0, 3 \rangle$*

Let ψ be a knowledge base and μ a modification for it, we define $\mathrm{Mod}_{cl}(\mu, \psi)$ to be the set of all models of μ such as $\mathcal{I}$, such that there exists a $\mathcal{J} \in Mod(\psi)$ where $\mathrm{Dist}(\mathcal{I}, \mathcal{J}) \neq \infty$. The following theorems show that $\mathrm{Mod}_{cl}(\mu, \psi)$ is not empty.

THEOREM 4.1. *$Mod_{cl}(\mu, \psi)$ is not an empty set.*

PROOF. We make a model for μ whose distance to a given model of ψ is not ∞. Let $\mathcal{J} \in Mod(\psi)$ and $\mathcal{I} \in Mod(\mu)$. Let interpretation $\mathcal{K}$ for ψ be defined as $\forall P \in Loc(\psi), P^{\mathcal{K}} = P^{\mathcal{J}}$ and $\forall P \in Sig(\mu), P^{\mathcal{K}} = P^{\mathcal{I}}$. Since $Loc(\psi)$ and $Sig(\mu)$ are disjoint (Definition 3.1), the interpretation of a concept or a role is specified clearly. $\mathcal{K}$ is an interpretation for μ whose distance to $\mathcal{J}$ is not ∞. $\square$

For revising an ontology by a modification that originates from external resources, we exploit the above definitions for selecting the models of the modification that have the least distances to the models of the knowledge base. Theorem 4.1 helps to define a revision operator that respects the principle of persistency for local knowledge. In other words, the revision operator select those models of the modification that do not change the interpretation of local symbols.

For a knowledge base ψ and a modification μ, the distance of interpretations in $Mod_{cl}(\mu, \psi)$ from models of ψ is in the form of an n-tuple. We say an n-tuple $T = \langle t_1, ..., t_n \rangle$ is less equal than the other n-tuple $T' = \langle t'_1, ..., t'_n \rangle$ and denote it as $T \leq T'$, iff $\forall_{1 \leq i \leq n} t_i \leq t'_i$. Similarly, $T < T'$, iff $\forall_{1 \leq i \leq n} t_i < t'_i$. Note that there may exist some interpretations like $\mathcal{I} \in Mod_{cl}(\psi, \mu)$, whose distance to a model of ψ like $\mathcal{J}$ includes an ∞ element. That happens when either $\mathcal{I}$ or $\mathcal{J}$ assign infinite elements of the domain to external classes or roles. In such cases and for comparing two n-tuples, we consider that all integer values are less than ∞.

DEFINITION 4.2. *Let ψ be a knowledge base being be revised by a modification μ. We define* DiffSet *as a set of tuples as follows:*

DiffSet$(\mu, \psi) = \{T = \langle t_1, ..., t_n \rangle | \ T = Dist(\mathcal{I}, \mathcal{J})$*, where* $\mathcal{I} \in Mod_{cl}(\psi, \mu)$ *and* $\mathcal{J} \in Mod(\psi)\}$.

Furthermore, $Dist(\mathcal{I}, \mathcal{J}) \in$ DiffSet(μ, ψ) is $<_{do}$ minimal iff $\nexists\ (\mathcal{I}' \ and \ \mathcal{J}'), (\mathcal{I}' \in Mod_{cl}(\psi, \mu), \mathcal{J}' \in Mod(\psi))$ such that $Dist(\mathcal{I}', \mathcal{J}') < Dist(\mathcal{I}, \mathcal{J})$.

Based on Definition 4.2, we define the revision operator for distributed ontologies as follow:

DEFINITION 4.3. *Let ψ be a knowledge base being revised by a modification μ. The revision operator $\circ_{do}$ is defined as follows (do stands for 'Distributed Ontologies'):*

$Mod(\psi \circ_{do} \mu) = \{\mathcal{J} \in Mod_{cl}(\mu, \psi) | \ \exists \mathcal{I} \in Mod(\psi) \ s.t.$
$Dist(\mathcal{J}, \mathcal{I}) \ is \ <_{do} \ minimal \ in$ DiffSet$(\mu, \psi)\}$.

The introduced operator respects the principle of persistency for local knowledge:

THEOREM 4.2. *$\circ_{do}$ respects the only condition (DO) for the principle of persistency for local knowledge (Definition 3.2).*

The following theorem shows how the operator respects AGM postulates.

THEOREM 4.3. *Let μ and μ' be two modifications for the knowledge base ψ. $\circ_{do}$ satisfies (R1)-(R5).*

PROOF. According to Definition 4.3, all models of $\psi \circ_{do} \mu$ are in $Mod_{cl}(\mu, \psi)$, consequently $\circ_{do}$ satisfies R1. For R2, consider $\mathcal{I} \in Mod(\mu) \cap Mod(\psi)$. Since $\psi \cup \mu$ is consistent, $Mod(\mu) \cap Mod(\psi)$ is not an empty set. $Dist(I, I) = \langle 0, 0, .., 0 \rangle$. It is not ∞ so $\mathcal{I}$ will be in $Mod_cl(\psi, \mu)$. In addition, zero tuple is the $<_{do}$ minimal in $DiffSet(\psi, \mu)$. So $\mathcal{I}$ is in $Mod(\psi \circ_{do} \mu)$. Observably, $\circ_{do}$ satisfies R2. According to Theorem 4.1, if μ is consistent then $Mod_{cl}(\mu, \psi)$ is not empty. Hence, $\circ_{do}$ satisfies R3. R4 is obvious in light of this fact that two equivalent knowledge bases have the same set of models. For R5, assume that $\mathcal{I} \in Mod((\psi \circ_{do} \mu) \cup \mu')$. Accordingly, $\mathcal{I} \in Mod_{cl}(\mu, \psi)$ and $\mathcal{I} \in Mod(\mu')$ and there is an interpretation $\mathcal{J}$ in $Mod(\psi)$ such that $Dist(\mathcal{I}, \mathcal{J})$ is $\leq_{do}$ minimum in $DiffSet(\psi, \mu)$. Since $Dist(\mathcal{I}, \mathcal{J})$ is not ∞, $\mathcal{I}$ is $Mod_{cl}(\mu', \psi)$ as well. $Mod(\mu \cup \mu')$ is a subset of $Mod(\mu)$ and therefore $DiffSet(\mu \cup \mu', \psi)$ is a subset of $DiffSet(\mu, \psi)$. Observably, $Dist(I, J)$ is the $\leq_{do}$ minimum in $DiffSet(\psi, \mu \cup \mu')$. Consequently, $\mathcal{I} \in Mod_{cl}(\mu \cup \mu', \psi)$ and for J, $Dist(\mathcal{I}, \mathcal{J})$ is the $\leq_{do}$ minimum in $DiffSet(\psi, \mu \cup \mu')$ and hence $\mathcal{I} \in Mod(\psi \circ_{do} \mu \cup \mu')$.

□

5. AN ALGORITHM FOR REVISING DISTRIBUTED ONTOLOGIES

In this section, we present an algorithm for revising ontologies in a distributed representation based on the introduced operator and semantics in the previous sections.

Algorithm *Revision*
Input: ψ : the knowledge base of the ontology that is going to be revised, μ: the knowledge base of the modification, F : { all individuals like ρ : where $\rho \in Sig(\mu)$}
Output: Ψ: a set of revised knowledge bases
if $\psi \cup \mu$ is consistent
  then return $\Psi = \{\psi\}$
$\mathbb{T}$ = getLabeledTableau (ψ, F)
$\hat{\mathbb{T}}$ =getTableau(μ, F)
for each labeled branch $\mathcal{B} \in \mathbb{T}$
  for each branch $\hat{\mathcal{B}} \in \hat{\mathbb{T}}$
    $\iota = \ getMerge(\mathcal{B}, \hat{\mathcal{B}})$
    $\Lambda(\mathcal{B}, \hat{\mathcal{B}}, \rho)$=$getConflictingLabeledConcepts(\iota)$
$\Upsilon = \{\lambda \in \Lambda : |\lambda|$ is minimum$\}$
for each $\lambda(\mathcal{B}, \hat{\mathcal{B}}), \rho) \in \Upsilon$
  $\psi_i = \psi$ such that all $C^{label} \in \Lambda$ is replaced by $C^{label} \sqcap \neg\rho$ in the axioms correspond to the '*label*'
$\Psi = \Psi \cup \psi_i$

Algorithm *Revision* provides a method for revising $\mathcal{ALC}$ ontologies given a modification ontology which is comprised of a set of assertions such as C(α), where C is an atomic concept. We call this kind of modification a *plain modification*.

The algorithm operates as follows. For a given knowledge base ψ, $\mathbb{T}$ is initialized by a set of labeled branches which are initialized by a set of nodes corresponding to each individual in F. Similarly, $\hat{\mathbb{T}}$ is initialized by set of labeled branches according to the μ knowledge base and the F individuals. Afterwards, by merging each pair of branches in $\mathbb{T}$ and $\hat{\mathbb{T}}$, the algorithm creates a new branch ι. The merging function creates $|F|$ nodes and for each node, assigns the union of labeled formulas from each branch of $\mathbb{T}$ and $\hat{\mathbb{T}}$ to it. Given this new branch, all contradicting formulas are assigned to Λ. Since the modification is not consistent with the ontology, Λ is not an empty set. In the next step, the smallest number of formulas that must be removed to make a ι branch without clashes are recognized, called Υ. The last step is to make a new consistent knowledge base by removing the individuals from the set of contradicting concepts and creating a new knowledge base.

The following example shows the process of Algorithm 1.

EXAMPLE 5.1. *Assume the ontology $\psi = \{(\#1)$`I:A`$\sqsubseteq$`B`$,(\#2)$`B`$\sqsubseteq \neg$`I:C`$\}$ is going to be revised by the modification $\mu = \{$`I:A`(α), `I:C`$(\alpha)\}$. Obviously, this modification is not consistent with the ψ ontology because it violates the inferred axiom `I:A` $\sqsubseteq \neg$ `I:C`. Labeled tableau $\mathbb{T}$ for ψ comprises of three branches each of which has a node corresponding to α as follows:*

$\mathbb{T}$ branches	*Nodes*
$\mathcal{B}1$	$node_\alpha$=$\{(\neg I : A)^{\#1}, (\neg B)^{\#2}\}$
$\mathcal{B}2$	$node_\alpha$=$\{(\neg I : A)^{\#1}, (\neg I : C)^{\#2}\}$
$\mathcal{B}3$	$node_\alpha$=$\{(B)^{\#1}, (\neg I : C)^{\#2}\}$

The labeled tableau for μ ($\hat{\mathbb{T}}$) has just one branch:

$\hat{\mathbb{T}}$ branches	*Nodes*
$\hat{\mathcal{B}}1$	$node_\alpha$=$\{(I : A), (I : C)\}$

The next step is to merge these branches and make a new set of branches:

ι *branches*	*Nodes*
$\iota 1$	$node_\alpha$={$(\neg I:A)^{\#1}, (\neg B)^{\#2}$}, $(I:A), (I:C)$
$\iota 2$	$node_\alpha$={$(\neg I:A)^{\#1}, (\neg I:C)^{\#2}$}, $(I:A), (I:C)$
$\iota 3$	$node_\alpha$={$(B)^{\#1}, (\neg I:C)^{\#2}$}, $(I:A), (I:C)$

$\iota 1$ and $\iota 3$ have one contradicting labeled formula, $(\neg I : A)^{\#1}$ and $(\neg I : C)^{\#2}$ respectively. $\iota 2$ has two contradicting formulas, so it will not be assigned to Υ. Two resulting knowledge bases are as shown below:

ψ_1	{I:A $\sqcap \neg\{\alpha\} \sqsubseteq$ B, B $\sqsubseteq \neg$ I:C}
ψ_2	{I:A$\sqsubseteq$ B,B$\sqsubseteq \neg$I:C $\sqcup \{\alpha\}$}

In the following we show the correctness and completeness of Algorithm 1 with regards to the revision operator introduced in the previous sections.

THEOREM 5.1. *For the $\mathcal{ALC}$ knowledge base ψ, a plain modification μ, let Ψ be the set of revised knowledge bases obtained from Algorithm Revision. Furthermore, let* Mod*(Ψ) be the set of all models of all ontologies $\psi \in \Psi$. Then* Mod*(Ψ)$\subseteq$* Mod*($\psi \circ_{do} \mu$). (In this therom, we have made unique name assumption)*

For proving this theorem, we first provide another theorem as follows:

THEOREM 5.2. *Let ψ be a knowledge base being revised by a modification μ. Let $\langle I, J\rangle$ be a pair of interpretations such that $\mathcal{I} \in Mod_{cl}(\mu, \psi)$ and $\mathcal{J} \in Mod(\psi)$ and $Dist(I, J)$ be the $\leq_{do}$ minimum in the $DiffSet(\psi, \mu)$. Let $\urcorner(I, J) = \{\{P_1^{\mathcal{I}} \ominus P_1^J\}, ..., \{P_n^{\mathcal{I}} \ominus P_n^J\}\}$ for all $P_i \in Sig(\psi)$. All members of each set in the $\urcorner$ multi set are asserted individuals in μ.*

PROOF. We proceed by assuming by contradiction that there exists some individuals in $\{P_i^{\mathcal{I}} \ominus P_i^J\}$ that are not in $Sig(\mu)$. We make a new model $\mathcal{I}$' for μ such that for all ρ that are not asserted in μ, $\rho^{I'} \in P_i^{I'}$ if $\rho^J \in P_i^J$ and for all ρ that are asserted in μ, $\rho^{I'} = \rho^I$. Since for some individuals ρ, $\rho \in \{P_i^{\mathcal{I}} \ominus P_i^J\}$ and $\rho \notin \{P_i^{\mathcal{I}'} \ominus P_i^J\}$, $Dist(I', J) < Dist(I, J)$, so $Dist(I, J)$ is not the minimum member of $DiffSet(\psi, \mu)$. □

PROOF. for Theorem 5.1
Assume $\mathcal{I} \in Mod(Psi)$. From the algorithm it is obvious that $I \in Mod(\mu)$. Now we prove that there is an interpretation $\mathcal{J}$ in $Mod(\psi)$, such that $Dist(I, J)$ is the $\leq_{do}$ minimum in the $DiffSet(\psi, \mu)$. We proceed by assuming by contradiction that there exists an interpretation $I' \in Mod(\mu)$ and J' in $Mod(\psi)$ such that $Dist(I', J') < Dist(I, J)$. Let $\urcorner$ be a multi set equal to $\{\{P_1^{\mathcal{I}} \ominus P_1^J\}, ..., \{P_n^{\mathcal{I}} \ominus P_n^J\}\}$ for all $P_i \in Sig(\psi)$. The Algorithm creates different knowledge bases based on new labeled branches (and corresponding models) where their difference to the original branches (models) is lack of individuals in the interpretation of various concepts. These individuals are those for which the labeled nodes in tableaus had been initialized. Therefore, all members of each set in the $\urcorner$ multi set are asserted in μ. Moreover, let $\urcorner' = \{\{P_1^{\mathcal{I}'} \ominus P_1^{J'}\}, ..., \{P_n^{\mathcal{I}'} \ominus P_n^{J'}\}\}$ for all $P_i \in Sig(\psi)$. According to Theorem 5.2, all members of each set in the $\urcorner'$ multi set are also asserted in μ.

First, we show that all ρ in each set of the multi set $\urcorner$, (we show this notion by $\rho \in \urcorner$), also exist in $\urcorner'$. Consider a $\rho \in \urcorner$ that is not in $\urcorner'$. It means that the assertions of μ about ρ is consistent with ψ. So in the algorithm process, there must exist at least one combination of branches for ρ that is consistent without removing any formula. So, The algorithm must select that combination and ρ will not show up in $\urcorner$.

Now, we prove that $Dist(I', J') \nless Dist(I, J)$. For each $\rho \in \urcorner$, let $Diff(\rho, I, J)$ be the set of all concepts like $C \in Sig(\psi)$ such that $\rho^I \in \{C^I \oplus C^J\}$. Since by assumption, $Dist(I', J') < Dist(I, J)$, there should be at least one ρ such that $|Diff(\rho, I, J)| > |Diff(\rho, I', J')|$. However in the algorithm, we minimize $Diff(\rho, I, J)$ by finding the minimum number of contradicting concepts.

□

THEOREM 5.3. *For the $\mathcal{ALC}$ knowledge base ψ, a plain modification μ, let Ψ be the set of revised knowledge bases obtained from Algorithm Revision. Furthermore, let* Mod*(Ψ) be the set of all models of all ontologies $\psi \in \Psi$. Then* Mod*($\psi \circ_{do} \mu$) $\subseteq$* Mod*(Ψ).*

PROOF. Assume $\mathcal{I} \in Mod(\psi \circ_{do} \mu)$. There exist an interpretation $\mathcal{J} \in Mod(\psi)$ such that $Dist(\mathcal{I}, \mathcal{J})$ is the $\leq_{do}$ minimum. In the labeled tableau for ψ, we select a combination of the labeled branches for each $\rho \in Sig(\mu)$ such that the combined branch does not violate $\mathcal{J}$. Based on $Dist(\mathcal{I}, \mathcal{J})$, we make a knowledge base by removing those individuals that are/are not in a concept according to $\mathcal{I}$ but are/are not based on $\mathcal{J}$. By removing, we mean replacing concept C by $C \sqcap \neg\{\rho\}$ in corresponding axioms. Since $Dist(\mathcal{I}, \mathcal{J})$ is minimum, the replacement will remove minimum number of individuals from concepts. So the resulting knowledge base is in Ψ and $\mathcal{I}$ is an interpretation of it. □

6. RELATED WORK

One of most influential work in the belief revision community is the set of AGM postulates that should be satisfied by any rational revision operator for belief bases. [9] attempts to generalize these postulates for description logics and OWL ontologies. The main challenge is to give a definition for the negation of DL axioms. In [8], the authors formalize different ways for defining inconsistency and incoherency in description logics and accordingly, give a definition for the negation of DL axioms. In this paper and for the introduced revision operator, we exploit an equivalent set of postulates that are not dependent on the negation of DL axiomss. Hence, we do not face the challenges discussed in [9, 8].

There are a few efforts in literature to resolve inconsistencies in ontologies by pinpointing the subset of the TBox that is responsible for incoherency. In [20], two different algorithms are introduced for the calculation of Minimal Unsatisfiability Preserving Sub-terminologies(MUPS) and Minimal Incoherence- Preserving Sub-TBoxes (MIPS). [12] describe a framework for selecting and removing erroneous axioms from MUPS to fix unsatisfiable concepts. Using and generalizing the notion of MIPS, [19] introduces a revision operator for terminologies. The introduced revision operator for ontologies in this paper distinguishes these works in the sense that it considers both TBox and ABox in the revision process.

In [17], updating ontology ABoxes has been studied. In that paper, only knowledge bases with just ABox assertions are considered. [6] analyzes instance level update in Description Logics and specially in DL-Lite. Contrary to our approach for *revising* distributed ontologies, their work focuses on *updating* monolithic knowledge bases. Furthermore, [21] is among the few proposals that investigate the notion of change and update in modular ontologies. It gives a definition for modular ontologies through which change propagation is defined. Accordingly, a modular ontology comprises of a set of self-containment ontology modules that are connected by conjunctive queries. The result of conjunctive mapping queries are computed off-line and are inserted into the ontology module at *compilation* time. After any *harmful* change in an ontology module, all connected modules should be recompiled. The main focus of the paper is on finding harmful changes that may result in a recompilation for others. However, in our approach, the revision method starts when a modification form an external ontology comes in and results in inconsistencies.

7. CONCLUSION

In this paper, we have provided precise definitions and semantics for distributed ontology revision. We introduced a new principle, persistency of local knowledge, for revising knowledge bases that include external symbols. We introduced a revision operator that respects this principle as well as most AGM postulates. Furthermore, we proposed an algorithm for revising $\mathcal{ALC}$ knowledge bases based on plain modifications. We showed that the algorithm is complete and correct with respect to the introduced semantics.

Despite the defined covering semantics of the proposed revision operator, the introduced algorithm is limited to $\mathcal{ALC}$ knowledge bases and plain modifications. We are extending our work to support more expressive description logic ontologies as well as more complicated modifications.

8. REFERENCES

[1] *On the formalization of interface-based modular ontologies under the closed-world assumption*, Tech. report, 2008.

[2] C.E. Alchourron, P. Gärdenfors, and D. Makinson, *On the logic of theory change: Partial meet contraction and revision functions*, Journal of symbolic logic (1985), 510–530.

[3] F. Baader and U. Sattler, *An overview of tableau algorithms for description logics*, Studia Logica **69** (2001), no. 1, 5–40.

[4] S. Benferhat, S. Kaci, D. Le Berre, and M.A. Williams, *Weakening conflicting information for iterated revision and knowledge integration*, Artificial Intelligence **153** (2004), no. 1, 339–371.

[5] Alexander Borgida and Luciano Serafini, *Distributed description logics: Assimilating information from peer sources.*, J. Data Semantics **1** (2003), 153–184.

[6] G. De Giacomo, M. Lenzerini, A. Poggi, and R. Rosati, *On the update of description logic ontologies at the instance level*, PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, vol. 21, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006, p. 1271.

[7] ———, *On the approximation of instance level update and erasure in description logics*, PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, vol. 22, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007, p. 403.

[8] G. Flouris, Z. Huang, J.Z. Pan, D. Plexousakis, and H. Wache, *Inconsistencies, negations and changes in ontologies*, Proceedings of the National Conference on Artificial Intelligence, vol. 21, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006, p. 1295.

[9] G. Flouris, D. Plexousakis, and G. Antoniou, *On applying the AGM theory to DLs and OWL*, Lecture notes in computer science **3729** (2005), 216.

[10] B.C. Grau, I. Horrocks, Y. Kazakov, and U. Sattler, *A logical framework for modularity of ontologies*, Proc. IJCAI, 2007, pp. 298–304.

[11] Bernardo Cuenca Grau and Oliver Kutz, *Modular ontology languages revisited*, Proceedings of the IJCAI'07 Workshop on Semantic Web for Collaborative Knowledge Acquisition,, 2007.

[12] A. Kalyanpur, B. Parsia, and E. Sirin, *Repairing unsatisfiable concepts in owl ontologies*, 3rd European Semantic Web Conference, ESWC 2006, 2006.

[13] H. Katsuno and A.O. Mendelzon, *On the difference between updating a knowledge base and revising it*, In Proceedings of the Second International Conference on the Principles of Knowledge Representation and Reasoning (KR91, 1991.

[14] ———, *Propositional knowledge base revision and minimal change*, Artificial Intelligence **52** (1991), no. 3, 263–294.

[15] Oliver Kutz, Carsten Lutz, Frank Wolter, and Michael Zakharyaschev, *E-connections of abstract description systems*, Artif. Intell. **156** (2004), no. 1, 1–73.

[16] D. Lehmann, M. Magidor, and K. Schlechta, *Distance semantics for belief revision*, Journal of Symbolic Logic (2001), 295–317.

[17] H. Liu, C. Lutz, M. Milicic, and F. Wolter, *Updating description logic aboxes*, International Conference of Principles of Knowledge Representation and Reasoning (KR), 2006.

[18] T. Meyer, K. Lee, and R. Booth, *Knowledge integration for description logics*, PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, vol. 20, Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005, p. 645.

[19] Guilin Qi, Peter Haase, Zhisheng Huang, Qiu Ji, Jeff Z. Pan, and Johanna Völker, *A kernel revision operator for terminologies – algorithms and evaluation*, ISWC '08: Proceedings of the 7th International Conference on The Semantic Web (Berlin, Heidelberg), Springer-Verlag, 2008, pp. 419–434.

[20] S. Schlobach, Z. Huang, R. Cornet, and F. van Harmelen, *Debugging incoherent terminologies*, Journal of Automated Reasoning **39** (2007), no. 3, 317–349.

[21] Heiner Stuckenschmidt and Michel C. A. Klein, *Integrity and change in modular ontologies*, IJCAI, 2003, pp. 900–908.