Tableau-based Revision for Expressive Description Logics

Thinh Dong Chan Le Duc Myriam Lamolle

LIASD - IUT de Montreuil, Université Paris 8, France

{dong,leduc,lamolle}@iut.univ-paris8.fr

Abstract

In this paper we present a tableau algorithm for building completion graphs of an ontology expressed in the description logic $\mathcal{SHIQ}$ with individuals. Based on a distance defined over completion graphs, we introduce a revision operation applied to a $\mathcal{SHIQ}$ ontology with a set of new semantic constraints. This revision operation computes the completion graphs that a revised ontology should admit. However, there does not always exist an ontology expressible in $\mathcal{SHIQ}$ from which a tableau algorithm generates exactly a given set of completion graphs. This leads us to introduce the notion of *upper approximation ontology* from which a tableau algorithm can generate the smallest set of completion graphs including a given set of completion graphs. This notion allows us to design an algorithm for constructing a revised ontology from an initial ontology with a set of new semantic constraints. We also implement the proposed algorithms with optimizations and report some experimental results to show that a model-based approach to revision of expressive ontologies is practicable.

1 Introduction

Formalisms based on Description Logics (DLs) such as OWL are widely used to represent ontologies encapsulated in semantics-based applications. An interesting feature of ontologies expressed in DLs (called DL ontologies) is to support automated inference services which allow ontology designers to detect eventual errors and allow users to entail new knowledge from ontologies with help of a reasoner. However, ontologies are not static but evolve over time. When changing ontologies, we are confronted with the problem of dealing with inconsistencies since new knowledge may contradict what exists in the ontology. The problem of revising a DL ontology is closely related to the problem of belief revision which has been widely discussed in the literature. Among early works on belief revision, Alchourrón, Gärdenfors and Makinson (AGM) [4] introduced intuitive and plausible constraints (namely AGM postulates) which should be satisfied by any rational belief revision operator. Existing belief revision approaches can be classified into *syntax-based* and *model-based (semantic)* approaches [18].

Syntax-based approaches manipulate directly syntactical entities such as formulas occurring in a knowledge base (KB). To take into account a new formula in preserving consistency, these approaches try to identify other formulas which should be removed. A main advantage of syntax-based approaches is to allow for distinguishing between the relevance of different formulas [2]. For instance, one can affect a lower priority to formulas that can change and a higher priority to those that would be "protected". The main issues are that the procedures resulting from these approaches heavily depend on the syntax of knowledge bases. Despite these issues, there have been some syntax-based belief revision operations developed for revising a DL ontology [10, 15].

Contrary to syntax-based approaches, semantic approaches investigate and manipulate models of ontologies rather than their syntactical entities. The main issues in adapting semantic approaches to DL ontologies are how to define a distance between models and how to compute a revised ontology from the models selected according to the defined distance. In addition, other problems may arise from dealing with models of DL ontologies. First, DL ontologies have infinitely many models which make impossible to construct directly a revised ontology from models. Second, models of a DL ontology have usually infinite complex structures, which may require a complex definition of distance between two models. Despite these problems, there have been several attempts to adapt classical model-based revision approaches to DL ontologies [9, 19, 21, 20, 22].

In this paper, we propose a new model-based approach for revising ontologies in $\mathcal{SHIQ}$ with individuals. A preliminary result of the present work for revising ontologies in $\mathcal{SHIQ}$ without individuals was published at LPAR forum [17]. We base the construction of our revision procedure on the following points : (i) using completion graphs generated by a novel tableau algorithm to characterize the semantics of a $\mathcal{SHIQ}$ ontology. A completion graph for an ontology O consists of nodes and edges which are respectively labelled by sets of concepts and roles from the signature of O in such a way that each axiom from O is satisfied in each node and edge. This algorithm must build a set of completion graphs, denoted $\mathsf{FM}(O)$, for an ontology O by considering all intrinsic non-deterministic cases instead of building one completion graph as existing tableau algorithms do ; (ii) defining a distance over a set of completion graphs for addressing the principle of minimal change. Given an ontology O' containing new axioms which should be taken into account when revising, this distance can help to choose completion graphs from $\mathsf{FM}(O')$ that are semantically closest to those in $\mathsf{FM}(O)$. A revised ontology of O by O' should admit the chosen completion graphs as models ; (iii) introducing the notion of approximation ontology to overcome inexpressibility issue. Our revision procedure returns an approximation ontology that is expressible in $\mathcal{SHIQ}$ and admits a given set of completion forests.

To illustrate the idea behind the construction, we consider the following running example.

Example 1 *Given an ontology* UNI *consisting of the following axioms et assertion :* α_1 : $\mathsf{Professor} \sqsubseteq \mathsf{Researcher} \sqcup \mathsf{Expert}$ *(Professors are researchers or experts),* α_2 : $\mathsf{Professor} \sqsubseteq \exists \mathsf{supervises}.\mathsf{Student}$ *(A professor supervises at least a student),* α_3 : $\mathsf{Professor} \sqsubseteq (\geq 2\ \mathsf{teaches}.\mathsf{Course})$ *(A professor teaches at least two courses), and* β : $\mathsf{Professor}(\mathsf{Alex})$ *(Alex is a professor).*

Assume that researchers and experts do not supervise any student. We add to UNI *the following axioms which express these semantic constraints :* (δ_1) : $\mathsf{Researcher} \sqsubseteq \forall \mathsf{supervises}.(\neg \mathsf{Student})$, (δ_2) : $\mathsf{Expert} \sqsubseteq \forall \mathsf{supervises}.(\neg \mathsf{Student})$.

However, the presence of δ_1 *and* δ_2 *will make* UNI *inconsistent, and this requires a revision to maintain consistency of* UNI*. One of the ways of revision is to remove a number of axioms from* UNI*. Intuitively, we can eliminate the axiom* α_1 *or* α_2 *to maintain consistency of* UNI*. In this case, if* α_1 *is chosen to remove then the obtained ontology* $\widehat{O} = \{\alpha_2, \alpha_3, \beta, \delta_1, \delta_2\}$ *is consistent. However, the knowledge "Professors are experts" in* α_1 *does not contradict the ontology* $\widehat{O}$ *but it has been removed together with* α_1*. In other words, the goal should be to build a new ontology* O^* *which is "compatible" with the axioms from* UNI *such that* O^* *is semantically as close as possible to* UNI*. We can check that the completion graphs* $\mathcal{F}_1$, $\mathcal{F}_2$ *in Figure 1 yield models of* UNI*. Similarly, the two completion graphs in Figure 2 yield models of* $\{\delta_1, \delta_2\}$.

If we define a distance between completion graphs based on structural similarity, it would be plausible to say that $\mathcal{F}'_1$ *is closer to* $\mathcal{F}_1$ *and* $\mathcal{F}_2$ *than* $\mathcal{F}'_2$*. Therefore, a revised ontology* O^* *should admit* $\mathcal{F}'_1$ *rather than* $\mathcal{F}'_2$*. Indeed, this intuition will be confirmed in Section 5 where we present a procedure for computing the revised ontology.*

The present paper is organized as follows. Section 2 describes the DL $\mathcal{SHIQ}$. In Section 3, we present a novel tableau algorithm for building a set of completion forests which represents all models of a $\mathcal{SHIQ}$ ontology. Section 4 introduces a revision operation which satisfies all revision postulates reformulated for DL ontologies. Based on the defined revision operation, we introduce in Section 5 the notion of upper approximation which allows us to propose a procedure for computing a revised ontology expressible in $\mathcal{SHIQ}$ from a set of completion forests. Section 6 describes some techniques for optimizing our procedure. We also describe an implementation of our algorithm and report some experimental results in Section 7. Finally, we summarize our work in Section 8.

2 Preliminaries

We begin by presenting the syntax and the semantics of $\mathcal{SHIQ}$. Let $\mathbf{R}$ be a non-empty set of *role names* and $\mathbf{R}_+ \subseteq \mathbf{R}$ be a set of transitive role names. We use $\mathbf{R}_\mathsf{I} = \{R^- \mid R \in \mathbf{R}\}$ to denote a set of inverse roles. Each element of $\mathbf{R} \cup \mathbf{R}_\mathsf{I}$ is called a $\mathcal{SHIQ}$-role. To simplify notations for nested inverse roles, we define a function $\mathsf{Inv}(S) = R^-$ if $S = R$; and $\mathsf{Inv}(S) = R$ if $S = R^-$ where $R \in \mathbf{R}$. A *role inclusion axiom* is of the form $R \sqsubseteq S$ for two (possibly inverse) $\mathcal{SHIQ}$-roles R and S. A *role hierarchy* $\mathcal{R}$ is a finite set of role inclusion axioms. A sub-role relation $\underline{\sqsubseteq}$ is defined as the transitive-reflexive closure of $\sqsubseteq$ on $\mathcal{R}^+ = \mathcal{R} \cup \{\mathsf{Inv}(R) \sqsubseteq \mathsf{Inv}(S) \mid R \sqsubseteq S \in \mathcal{R}\}$. We define a function $\mathsf{Trans}(R)$ which returns true iff R is a transitive role. More precisely, $\mathsf{Trans}(R) = \mathit{true}$ iff $R \in \mathbf{R}_+$ or $\mathsf{Inv}(R) \in \mathbf{R}_+$. A role R is called *simple* with respect to (w.r.t.) $\mathcal{R}^+$ iff $R \notin \mathbf{R}_+$ and, for any $R' \underline{\sqsubseteq} R$, R' is also a simple role. An interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ consists of a non-empty set $\Delta^\mathcal{I}$ (*domain*) and a function $\cdot^\mathcal{I}$ which maps each role name to a subset of $\Delta^\mathcal{I} \times \Delta^\mathcal{I}$ such that $R^{-\mathcal{I}} = \{\langle x, y \rangle \in \Delta^\mathcal{I} \times \Delta^\mathcal{I} \mid \langle y, x \rangle \in R^\mathcal{I}\}$ for all $R \in \mathbf{R}$, and $\langle x, z \rangle \in S^\mathcal{I}, \langle z, y \rangle \in S^\mathcal{I}$ implies $\langle x, y \rangle \in S^\mathcal{I}$ for each $S \in \mathbf{R}_+$. An interpretation $\mathcal{I}$ is a model of $\mathcal{R}$, written $\mathcal{I} \models \mathcal{R}$, if $R^\mathcal{I} \subseteq S^\mathcal{I}$ for each $R \sqsubseteq S \in \mathcal{R}$.

Let $\mathbf{C}$ be a non-empty set of *concept names*. The set of $\mathcal{SHIQ}$-concepts is inductively defined as the smallest set containing all C in $\mathbf{C}$, $\top$, $C \sqcap D$, $C \sqcup D$, $\neg C$, $\exists R.C$, $\forall R.C$, $(\leq n S.C)$ and $(\geq n S.C)$ where n is a positive integer, C and D are $\mathcal{SHIQ}$-concepts, R is a $\mathcal{SHIQ}$-role and S is a simple role w.r.t. a role hierarchy. We write $\bot$ for $\neg\top$. The interpretation function $\cdot^\mathcal{I}$ of an interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$

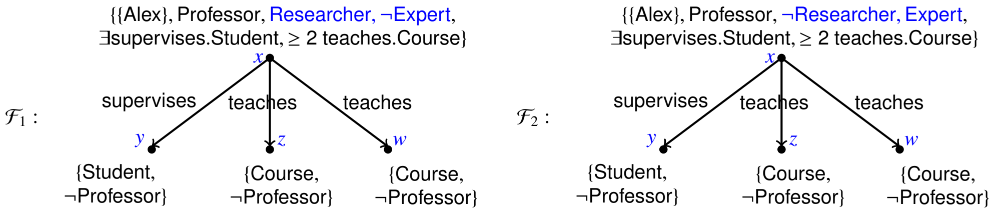


FIGURE 1 – Completion graphs yielding models of UNI

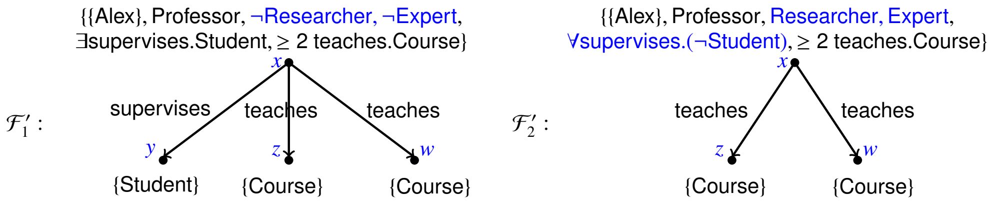


FIGURE 2 – Completion graphs yielding models of $\{\delta_1, \delta_2\}$

maps each concept name to a subset of $\Delta^{\mathcal{I}}$ such that $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$, $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$, $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$, $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \backslash C^{\mathcal{I}}$, $(\exists R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}}, \langle x, y\rangle \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$, $(\forall R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \forall y \in \Delta^{\mathcal{I}}, \langle x, y\rangle \in R^{\mathcal{I}} \Rightarrow y \in C^{\mathcal{I}}\}$, $(\geq n\, S.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid |\{y \in C^{\mathcal{I}} \mid \langle x, y\rangle \in S^{\mathcal{I}}\}| \geq n\}$, $(\leq n\, S.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid |\{y \in C^{\mathcal{I}} \mid \langle x, y\rangle \in S^{\mathcal{I}}\}| \leq n\}$ where $|S|$ stands for the cardinality of a set S. An axiom $C \sqsubseteq D$ is called a general concept inclusion (GCI) where C, D are (possibly complex) $\mathcal{SHIQ}$-concepts, and a finite set of GCIs is called a terminology $\mathcal{T}$. An interpretation $\mathcal{I}$ satisfies a GCI $C \sqsubseteq D$, written $\mathcal{I} \models (C \sqsubseteq D)$, if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. $\mathcal{I}$ is a model of $\mathcal{T}$, written $\mathcal{I} \models \mathcal{T}$, if $\mathcal{I}$ satisfies each GCI in $\mathcal{T}$.

Let $\mathbf{I}$ be a set of individual names. An assertion is of the form $C(a)$, $R(a, b)$, or $a \neq b$ for $a, b \in \mathbf{I}$, a $\mathcal{SHIQ}$-role R and a $\mathcal{SHIQ}$-concept C. An ABox consists of a finite set of assertions. For an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, an element $x \in \Delta^{\mathcal{I}}$ is called an instance of a concept C iff $x \in C^{\mathcal{I}}$. For ABoxes, the function $\cdot^{\mathcal{I}}$ of $\mathcal{I}$ maps each individual $a \in \mathbf{I}$ to some element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. An interpretation $\mathcal{I}$ satisfies an assertion $C(a)$ (resp. $R(a, b)$, and $a \neq b$) iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$ (resp. $\langle a^{\mathcal{I}}, b^{\mathcal{I}}\rangle \in R^{\mathcal{I}}$, and $a^{\mathcal{I}} \neq b^{\mathcal{I}}$). $\mathcal{I}$ satisfies an ABox $\mathcal{A}$ if it satisfies each assertion in $\mathcal{A}$. Such an interpretation is called a *model* of $\mathcal{A}$, denoted by $\mathcal{I} \models \mathcal{A}$.

We use $O = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ to denote a $\mathcal{SHIQ}$ *ontology*, where $\mathcal{T}$ is a $\mathcal{SHIQ}$ terminology, $\mathcal{R}$ is a $\mathcal{SHIQ}$ role hierarchy, and $\mathcal{A}$ is an ABox. An ontology $O = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ is said to be consistent if there is a model $\mathcal{I}$ of $\mathcal{T}$, $\mathcal{R}$ and $\mathcal{A}$, i.e., $\mathcal{I} \models \mathcal{T}$, $\mathcal{I} \models \mathcal{R}$ and $\mathcal{I} \models \mathcal{A}$. Additionally, we use $\mathsf{Mod}(O)$ to denote all the models, and $\mathcal{S}(O) = \mathbf{R} \cup \mathbf{C} \cup \mathbf{I}$ to denote the signature of an ontology O.

For the ease of construction, we assume all concepts to be in *negation normal form* (NNF), i.e., negation occurs only in front of concept names. Any $\mathcal{SHIQ}$-concept can be transformed to an equivalent one in NNF by using De Morgan's laws and the duality between concepts [14]. For a concept C, we use $\mathsf{nnf}(C)$ and $\dot{\neg} C$ to denote respectively the NNF of C and $\neg C$. The function $\mathsf{nnf}(C)$ can be computed in polynomial time in the size of C [7]. In the remaining of this section, we introduce some notations which will be used in the next sections.

Definition 1 (Subconcepts) *Let $O = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ be a $\mathcal{SHIQ}$ ontology with $\mathcal{S}(O) = \mathbf{R} \cup \mathbf{C} \cup \mathbf{I}$. A set $\mathsf{sub}(O)$ is inductively defined as follows :*

$$\mathsf{sub}(O) = \mathsf{sub}(\mathcal{T}) \cup \mathsf{sub}(\mathcal{A}) \cup \{\dot{\neg} C \mid C \in \mathsf{sub}(\mathcal{T}) \cup \mathsf{sub}(\mathcal{A})\}$$
$$\mathsf{sub}(\mathcal{T}) = \bigcup_{C \sqsubseteq D \in \mathcal{T}} \mathsf{sub}(\mathsf{nnf}(\neg C \sqcup D))$$
$$\mathsf{sub}(\mathcal{A}) = \{\mathsf{sub}(\mathsf{nnf}(C)) \mid C(a) \in \mathcal{A}\}$$

$$\mathsf{sub}(C) = \begin{cases} \{C, \neg C\} & \text{if } C \in \mathbf{C} \\ \mathsf{sub}(E) \cup \mathsf{sub}(F) & \text{if } C \in \{E \sqcap F, E \sqcup F\} \\ \{C\} \cup \{\exists R'.E \mid R \sqsubseteq^{*} R'\} \cup \mathsf{sub}(E) & \text{if } C = \exists R.E \\ \{C\} \cup \{\forall R'.E \mid R \sqsubseteq^{*} R'\} \cup \mathsf{sub}(E) & \text{if } C = \forall R.E \\ \{C\} \cup \{\geq nR'.E \mid R \sqsubseteq^{*} R'\} \cup \mathsf{sub}(E) & \\ & \text{if } C = (\geq nR.E) \\ \{C\} \cup \mathsf{sub}(E) & \text{if } C = (\leq nR.E) \end{cases}$$

Note that $\mathsf{sub}(O)$ contains no disjunctions or conjunctions since they are replaced with their disjuncts and conjuncts.

To characterize the semantics of an ontology we need to explore all intrinsic non-determinism arising from disjunctions and numbering restrictions when constructing a completion graph for the ontology. For this reason, we introduce a function $\mathsf{Flat}(C)$ which makes explicit all disjunctions at top-level of a concept C (i.e. those that do not appear in the filler of a universal, existential, numbering restrictions occurring in C).

Definition 2 (Flattening) *Let C be a $\mathcal{SHIQ}$ concept. We define a function* $\mathsf{Flat}(C)$ *which returns a set of subsets of* $\mathsf{sub}(C)$ *as follows :*

1. *If C is a concept name or C is an existential, universal, number restriction, we define* $\mathsf{Flat}(C) = \{\{C\}\}$*;*
2. *If $C = E \sqcup F$, we define* $\mathsf{Flat}(C) = \mathsf{Flat}(E) \cup \mathsf{Flat}(F)$*;*
3. *If $C = E \sqcap F$, we define* $\mathsf{Flat}(C) = \{W \cup W' \mid W \in \mathsf{Flat}(E), W' \in \mathsf{Flat}(F)\}$

Applying the item 3 in Definition 2 to a concept C may make $\mathsf{Flat}(C)$ increase exponentially. For instance, $\mathsf{Flat}((A_1 \sqcup B_1) \sqcap (A_2 \sqcup B_2)) = \{W \cup W' \mid W \in \{\{A_1\}, \{B_1\}\}, W' \in \{\{A_2\}, \{B_2\}\}\} = \{\{A_1, A_2\}, \{A_1, B_2\}, \{B_1, A_2\}, \{B_1, B_2\}\}$. More general, if $C = (A_1 \sqcup B1) \sqcap \cdots \sqcap (A_n \sqcup B_n)$, $\mathsf{Flat}(C)$ contains 2^n elements.

3 Novel Tableau-based algorithm

In this section we introduce a tableau-based algorithm for generating a finite set of completion graphs representing the infinite set of all models of a $\mathcal{SHIQ}$ ontology. Horrocks and colleagues [12] have proposed a tableau algorithm for checking consistency of a $\mathcal{SHIQ}$ ontology and have shown that there always exists a finite *completion forest* iff the ontology is consistent. This algorithm attempts to construct a completion forest, returns "YES" if it succeeds in building such a completion forest and "NO" if it fails after considering all possibly non-deterministic cases. To be able to characterize the semantics of an ontology, we need rather a set of completion forests which describes different models resulting from non-deterministic logical constructors than one completion forest. For this purpose, we adapt the tableau algorithm by Horrocks and colleagues [12] in such a way that it would explore all intrinsic non-deterministic cases.

Definition 3 (Completion forest) *Let $O = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ be a $\mathcal{SHIQ}$ ontology. A* completion forest *$\mathcal{F}$ for O is a tuple $\mathcal{F} = (G, T\langle\widehat{x_1}\rangle, \ldots, T\langle\widehat{x_n}\rangle)$ where*

• *$G = (\mathbf{V}, \mathbf{E}, \mathbf{L})$ is a directed graph with $\mathbf{V}$ a set of root nodes, $\mathbf{E}$ a set of edges connecting root nodes and $\mathbf{L}$ a labelling function which associates to each node $x \in \mathbf{V}$ a set $\mathbf{L}(x) \subseteq \mathsf{sub}(O)$ and to each edge $\langle\widehat{x}, \widehat{y}\rangle \in \mathbf{E}$ a set $\mathbf{L}(\langle\widehat{x}, \widehat{y}\rangle) \subseteq \mathbf{R} \cup \mathbf{R}_{\mathsf{I}}$. A node $\widehat{y} \in \mathbf{V}$ is called an R-neighbor of $\widehat{x} \in \mathbf{V}$ if $R \in \mathbf{L}(\langle\widehat{x}, \widehat{y}\rangle)$ or $\mathsf{Inv}(R) \in \mathbf{L}(\langle\widehat{y}, \widehat{x}\rangle)$.*

• *Each $T\langle\widehat{x_i}\rangle = (V_i, E_i, L_i)$ ($1 \le i \le n$) is a tree rooted by $\widehat{x_i}$ belonging to G (i.e. $\widehat{x_i} \in \mathbf{V}$), V_i a set of nodes, E_i a set of edges, and L_i a labelling function which associates to each node $x \in V_i$ a set $L_i(x) \subseteq \mathsf{sub}(O)$ and to each edge $\langle x, y\rangle \in E_i$ a set $L_i(\langle x, y\rangle) \subseteq \mathbf{R} \cup \mathbf{R}_{\mathsf{I}}$.*

If two nodes $x, y \in V_i$ (of some tree $T\langle\widehat{x_i}\rangle = (V_i, E_i, L_i)$) connected by an edge $\langle x, y\rangle \in E_i$, then y is called a successor *of x, and x is called a* predecessor *of y;* ancestor *is the transitive closure of* predecessor. *A node y is called an R-successor of x if, for some role R' with $R' \underline{\circledast} R$, $R' \in L(\langle x, y\rangle)$; x is called an R-predecessor of y, if y is an R-successor of x. A node y is called an R-neighbor of x if y is an R-successor or x is an $\mathsf{Inv}(R)$-successor of y.*

A node $x \in V_i$ is called blocked *by a node $y \in V_i$ if x is not a root node and it has ancestors x', y and y' such that (i) y is not a root node, (ii) x is a successor of x' and y is a successor of y', (iii) $L(x) = L(y)$, $L(x') = L(y')$, and (iv) $L(\langle x', x\rangle) = L(\langle y', y\rangle)$.*

Furthermore, there are an inequality relation $\not\doteq$ and an equality relation $\doteq$ defined over nodes in $\mathcal{F}$. In addition, $\mathcal{F}$ is said to contain a clash *if (i) there is some node x in $\mathcal{F}$ such that either $\{A, \neg A\} \subseteq L(x)$ for some concept name $A \in \mathbf{C}$, or (ii) $(\le nS.C) \in L(x)$ and there are $(n+1)$ S-neighbors $y_1, \cdots, y_{n+1}$ of x with $y_i \not\doteq y_j$ and $X \subseteq L(y_i)$ for some $X \in \mathsf{Flat}(C)$ and all $1 \le i < j \le (n+1)$.*

Based on the Horrocks and colleagues' work [12], we design a tableau algorithm for building a completion forest by applying the expansion rules in Figure 3. There are two main differences between the rules in Figure 3 and those presented in a standard tableau algorithm : (i) the absence of conjunction and disjunction rules. According to Definition 2, applying the function Flat to a concept freshly added to the label of a node removes all conjunctions and disjunctions at top-level from that concept; (ii) the presence of the sat-rule (sat stands for *saturate*). A choice of a subset $S \subseteq \mathsf{sub}(O)$ in the sat-rule must include all flattened concepts from GCI axioms (such as those added by the $\sqsubseteq$-rule [12]). In addition, the sat-rule adds to each node label either C or $\dot{\neg}C$ for each $C \in \mathsf{sub}(O)$. These behaviors may lead to an exponential blow-up but it is needed for constructing an approximation ontology from a set of completion forests and a set of subconcepts (Section 5). An optimization of this rule will be proposed and discussed in Section 6.

For an input ontology $O = \{\mathcal{T}, \mathcal{R}, \mathcal{A}\}$, our tableau algorithm starts by initializing a completion forest $\mathcal{F}$ with only root nodes and edges between them. This part of $\mathcal{F}$ represents individuals and assertions defined in $\mathcal{A}$. The algorithm applies the rules from Figure 3 to each node until no rule is applicable to any node. In this case, $\mathcal{F}$ is called *complete*. If $\mathcal{F}$ contains no clash it is called *clash-free*.

Lemma 1 ***(Soundness and completeness).*** *Let $O = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ be a $\mathcal{SHIQ}$ ontology.*

1. *The tableau algorithm with the expansion rules in Figure 3 terminates.*
2. *If the tableau algorithm with the expansion rules in Figure 3 yields a complete and clash-free completion forest from the input ontology O then O is consistent;*
3. *If O is consistent then the tableau algorithm with the*

$\exists$-rule : if (1) $\exists S.C \in L(x)$, x is not blocked, and (2) x has no S-neighbor y s.t. $X \subseteq L(y)$ for some $X \in \mathsf{Flat}(C)$, then create a new node y with $L(\langle x, y\rangle) \leftarrow \{S\}$ and $L(y) \leftarrow X$ for some $X \in \mathsf{Flat}(C)$.

$\forall$-rule : if (1) $\forall S.C \in L(x)$, and (2) there is an S-neighbor y of x s.t. $X \nsubseteq L(y)$ for all $X \in \mathsf{Flat}(C)$, then $L(y) \leftarrow L(y) \cup X$ for some $X \in \mathsf{Flat}(C)$.

$\forall_+$-rule : if (1) $\forall S.C \in L(x)$, (2) there is an R with $\mathsf{Trans}(R)$ s.t. $R \sqsubseteq^* S$, and (3) there is an R-neighbor y of x s.t. $\forall R.C \notin L(y)$, then $L(y) \leftarrow L(y) \cup \{\forall R.C\}$.

$\geq$-rule : if 1. $(\geq nS.C) \in L(x)$, x is not blocked, and (2) x has no n S-neighbors $y_1, \cdots, y_n$ such that $X \subseteq L(y_i)$ for some $X \in \mathsf{Flat}(C)$ and $y_i \not\doteq y_j$ for $0 \leq i < j \leq n$, then (i) create n new nodes $y_1, \cdots, y_n$ with $L(\langle x, y_i\rangle) \leftarrow \{S\}$,
(ii) $L(y_i) \leftarrow X$ for some $X \in \mathsf{Flat}(C)$ and $y_i \not\doteq y_j$ for $1 \leq i < j \leq n$.

$\leq$-rule : if (1) $(\leq nS.C) \in L(x)$, (2) x has $n+1$ S-neighbors $y_0, \ldots, y_n$ s.t. $X \subseteq L(y_i)$ for some $X \in \mathsf{Flat}(C)$, (3) there are two S-neighbors y, z of x with $X_1 \subseteq L(y)$, $X_2 \subseteq L(z)$ for some $X_1, X_2 \in \mathsf{Flat}(C)$, y is not an ancestor of z, and not $y \not\doteq z$, then (i) $L(z) \leftarrow L(z) \cup L(y)$ and $L(\langle x, y\rangle) \leftarrow \emptyset$;
(ii) if z is an ancestor of x then $L(\langle z, x\rangle) \leftarrow L(\langle z, x\rangle) \cup \{\mathsf{Inv}(R) \mid R \in L(\langle x, y\rangle)\}$, else $L(\langle x, z\rangle) \leftarrow L(\langle x, z\rangle) \cup L(\langle x, y\rangle)$;
(iii) add $u \not\doteq z$ for all u such that $u \not\doteq y$.

$\leq_r$-rule : if (1) $(\leq nS.C) \in L(x)$, (2) x has $n+1$ S-neighbors $y_0, \cdots, y_n$ s.t. $X \subseteq L(y_i)$ for some $X \in \mathsf{Flat}(C)$,
(3) $y_i \not\doteq y_j$ does not hold for some $0 \leq i < j \leq n$ where y_i, y_j are root nodes,
then (i) $L(y_i) \leftarrow L(y_i) \cup L(y_j)$,
(ii) for all edges $\langle y_j, w\rangle$: if the edge $\langle y_i, w\rangle$ does not exist, create it with $L(\langle y_i, w\rangle) = \emptyset$;
set $L(\langle y_i, w\rangle) \leftarrow L(\langle y_i, w\rangle) \cup L(\langle y_j, w\rangle)$,
(iii) for all edges $\langle w, y_j\rangle$: if the edge $\langle w, y_i\rangle$ does not exist, create it with $L(\langle w, y_i\rangle) = \emptyset$;
set $L(\langle w, y_i\rangle) \leftarrow L(\langle w, y_i\rangle) \cup L(\langle w, y_j\rangle)$,
(iv) set $L(y_j) \leftarrow \emptyset$ and remove all edges to/from y_j, (v) set $u \not\doteq y_i$ for all u with $u \not\doteq y_j$, et (vi) set $y_j \doteq y_i$.

---

sat-rule : if sat-rule has never been applied to x then (i) choose a subset $S \subseteq \mathsf{sub}(O)$ such that $L(x) \cup \bigcup_{X \in \mathsf{Flat}(\mathsf{nnf}(\neg C \sqcup D)), C \sqsubseteq D \in \mathcal{T}} X \subseteq S$, and (ii) set $L(x) \leftarrow S \cup \bar{S}$ where $\bar{S} = \{\dot{\neg} C \mid C \in \mathsf{sub}(O) \setminus S\}$.

FIGURE 3 – Expansion rules for $\mathcal{SHIQ}$

expansion rules in Figure 3 yields a complete and clash-free completion forest from the input ontology O.

The tableau algorithm can build a completion forest whose depth is bounded by an exponential function in the size of O due to the blocking condition. Given a $\mathcal{SHIQ}$ ontology O, a complete and clash-free completion forest $\mathcal{F}$ built by running the tableau algorithm with the expansion rules in Figure 3 over O is called a *forest-like model*. According to soundness of the tableau algorithm (Lemma 1), one can devise by unraveling a model from a complete and clash-free completion forest $\mathcal{F}$, denoted $\widehat{\mathcal{I}}(\mathcal{F})$. Given an axiom $C \sqsubseteq D$, define $\widehat{\mathcal{I}}(\mathcal{F}) \models (C \sqsubseteq D)$ if $C^{\widehat{\mathcal{I}}(\mathcal{F})} \subseteq D^{\widehat{\mathcal{I}}(\mathcal{F})}$. Given an assertion $C(a)$ (or $R(a, b)$), define $\widehat{\mathcal{I}}(\mathcal{F}) \models C(a)$ (resp. $\widehat{\mathcal{I}}(\mathcal{F}) \models (R(a, b))$) if $a^{\widehat{\mathcal{I}}(\mathcal{F})} \in C^{\widehat{\mathcal{I}}(\mathcal{F})}$ (resp. $\langle a^{\widehat{\mathcal{I}}(\mathcal{F})}, b^{\widehat{\mathcal{I}}(\mathcal{F})}\rangle \in R^{\widehat{\mathcal{I}}(\mathcal{F})}$). Conversely, according to completeness of the tableau algorithm (Lemma 1), it can build a complete and clash-free completion forest $\mathcal{F}$ from a model $\mathcal{I} \in \mathsf{Mod}(O)$, denoted $\mathcal{F}(\mathcal{I})$. This remark allows us to introduce the following notation.

Notation 1 *Let O be a $\mathcal{SHIQ}$ ontology.*

- Let $\mathcal{F}$ a forest-like model constructed by the tableau algorithm which takes O as input. We note $\widehat{\mathcal{I}}(\mathcal{F})$ the model of O obtained by unraveling $\mathcal{F}$.

- Let $\mathcal{I}$ a model in $\mathsf{Mod}(O)$. We note $\mathcal{F}(\mathcal{I})$ the forest-like model of O constructed by the tableau algorithm from $\mathcal{I}$.

Contrary to standard tableau algorithms which terminate when a forest-like model is found, we design a new tableau algorithm which has to consider all non-deterministic cases and build all forest-like models for an ontology O. We use $\mathsf{FM}(O)$ to denote the set of all forest-like models built by running the new tableau algorithm over a $\mathcal{SHIQ}$ ontology O. Given a set of concepts sub, we define $\mathsf{FM}(O, \mathsf{sub})$ to be the set of all forest-like models built by running the tableau algorithm on O such that the sat-rule operates on $\mathsf{sub}(O) \cup \mathsf{sub}$ (i.e. it chooses a subset $S \subseteq \mathsf{sub}(O) \cup \mathsf{sub}$). In particular, given an ontology O' the set $\mathsf{FM}(O, \mathsf{sub}(O'))$ can be built by running the new tableau algorithm over O with the sat-rule operating on $\mathsf{sub}(O) \cup \mathsf{sub}(O')$. In addition, we need to import to O roles which occur in role assertions from O'. This leads to add to the label of each root edge $\langle \widehat{x}, \widehat{y}\rangle$ of each forest $\mathcal{F} \in \mathsf{FM}(O, \mathsf{sub}(O'))$ a subset of roles

$S_{\mathcal{R}} \subseteq \mathbf{R}_{O'}$ where $\mathbf{R}_{O'}$ is the set of all roles occurring in O' with their inverse. This new behavior can be formalized as follows :

- $\mathsf{sat}_{\mathcal{R}}$-rule : for each root edge $\langle \widehat{x}, \widehat{y} \rangle \in \mathbf{E}$ with $G = (\mathbf{V}, \mathbf{E}, \mathbf{L})$, $\mathcal{F} = (G, T\langle \widehat{x_1} \rangle, \cdots, T\langle \widehat{x_n} \rangle)$ and $\mathcal{F} \in \mathsf{FM}(O, \mathsf{sub}(O'))$, we set $L(\langle \widehat{x}, \widehat{y} \rangle) \leftarrow L(\langle \widehat{x}, \widehat{y} \rangle) \cup S_{\mathcal{R}}$ for some $S_{\mathcal{R}} \subseteq \mathbf{R}_{O'}$.

The construction of $\mathsf{FM}(O, \mathsf{sub}(O'))$ allows one to import the signature of an ontology O' to O when building completion forests for O. Note that we do not import any new semantic constraint from O' when building $\mathsf{FM}(O, \mathsf{sub}(O'))$. What we really perform in this construction is to import into O concepts written in the signature of O'. This importation may extend $\mathsf{FM}(O)$ with new completion forests but never changes consistency of O.

We now use the notation recently introduced to formulate the following properties on $\mathsf{FM}(O)$ which characterizes the semantics of an ontology O.

Corollary 1 *Let O and O' be two consistent $\mathcal{SHIQ}$ ontologies. Let α be a concept axiom or assertion written in $\mathcal{S}(O)$. $\widehat{\mathcal{I}}(\mathcal{F}) \models \alpha$ for each forest-like model $\mathcal{F} \in \mathsf{FM}(O, \mathsf{sub}(\alpha))$ iff $\mathcal{I} \models \alpha$ for each model $\mathcal{I} \in \mathsf{Mod}(O)$.*

The Corollary 1 affirms the semantic equivalence between $\mathsf{Mod}(O)$ and $\mathsf{FM}(O)$ in the sense that each axiom/assertion which is satisfied by $\mathsf{Mod}(O)$ is satisfied by $\mathsf{FM}(O)$, and conversely. This result allows us to replace a possibly infinite set $\mathsf{Mod}(O)$ with a finite set $\mathsf{FM}(O)$ in constructions presented in the next sections.

4 Revision Operation

The main goal of the present section is to define a revision operation which allows for revising a consistent ontology O by axioms from another consistent ontology O', and however $O \cup O'$ is inconsistent. Such a revision operation returns a set of completion forests of which a revised ontology should admit in order to take into account new knowledge from O' and to be semantically as close as possible to O. These properties on revision operation are captured by the AGM postulates rephrased for DL ontologies [9]. To reach this goal, we need to define a distance between two completion forests which yields a total pre-order over them and allows one to talk about similarity between two ontologies. This distance is an extension of that defined over completion trees [17].

Definition 4 (Isomorphism) *Let $\mathcal{F}$=$(G, T\langle \widehat{x_1} \rangle, \ldots, T\langle \widehat{x_n} \rangle)$ and $\mathcal{F}' = (G', T\langle \widehat{x'_1} \rangle, \ldots, T\langle \widehat{x'_n} \rangle)$ be two forest-like models with $G = (\mathbf{V}, \mathbf{E}, \mathbf{L})$, $G' = (\mathbf{V}', \mathbf{E}', \mathbf{L}')$, $T\langle \widehat{x_i} \rangle = \langle V_i, L_i, E_i \rangle$ and $T\langle \widehat{x'_j} \rangle = \langle V'_j, L'_j, E'_j \rangle$ $(1 \le i, j \le n)$. Let $\mathcal{V} = \mathbf{V} \cup V_1 \cup \cdots \cup V_n$ and $\mathcal{V}' = \mathbf{V}' \cup V'_1 \cup \cdots \cup V'_n$. Let $\mathcal{E} = \mathbf{E} \cup E_1 \cup \cdots \cup E_n$ and $\mathcal{E}' = \mathbf{E}' \cup E'_1 \cup \cdots \cup E'_n$. We use $\mathsf{succ}(x)$ to denote the set of successors of a node x in a tree $T\langle \widehat{x_i} \rangle$ or $T\langle \widehat{x'_j} \rangle$ with $1 \le i, j \le n$.*

- *$T\langle \widehat{x_i} \rangle$ and $T\langle \widehat{x'_j} \rangle$ are* isomorphic *for $1 \le i, j \le n$ if there is a bijection π from V_i to V'_j such that (i) $\pi(\widehat{x_i}) = \widehat{x'_j}$; and (ii) for each node $x \in V_i$, we have $\pi(x') \in \mathsf{succ}(\pi(x))$ for each $x' \in \mathsf{succ}(x)$.*

- *$\mathcal{F}$ and $\mathcal{F}'$ are isomorphic if there is a bijection π from $\mathcal{V}$ to $\mathcal{V}'$ such that (i) $\pi(\widehat{x_i}) = \widehat{x'_j}$ for each $\widehat{x_i} \in \mathbf{V}$, (ii) for each $T\langle \widehat{x_i} \rangle \in \mathcal{F}$, two trees $T\langle \widehat{x_i} \rangle$ and $T\langle \pi(\widehat{x_i}) \rangle$ are* isomorphic. *In this case, we say that π is an isomorphism between $\mathcal{F}$ and $\mathcal{F}'$.*

Note that if there exists a bijection π between two trees $T\langle \widehat{x_i} \rangle$ and $T\langle \widehat{x'_j} \rangle$ as described in Definition 4 then the restriction of π to $\mathsf{succ}(x)$, denoted $\pi|_{\mathsf{succ}(x)}$, is a bijection from $\mathsf{succ}(x)$ to $\mathsf{succ}(\pi(x))$.

Remark 1 *Let $\mathcal{F} \in \mathsf{FM}(O, \mathsf{sub}(O'))$ and $\mathcal{F}' \in \mathsf{FM}(O', \mathsf{sub}(O))$ two forest-like models with the sets of root nodes $\mathbf{V}$ and $\mathbf{V}'$. It is needed to import all individuals from O (included in $\mathsf{sub}(O)$) to O' and reversely when revising O by new axioms from O'. Therefore, for each individual a there are a unique tree $T\langle \widehat{x_i} \rangle$ of $\mathcal{F}$ and a unique tree $T\langle \widehat{x'_j} \rangle$ of $\mathcal{F}'$ such that $a \in L(\widehat{x_i}) \cap L(\widehat{x'_j})$. This implies that there is a bijection φ from $\mathbf{V}$ to $\mathbf{V}'$ such that $\varphi(\widehat{x_i}) = \widehat{x'_j}$ iff $a \in L(\widehat{x_i}) \cap L(\widehat{x'_j})$ for some individual a.*

Since the notion of isomorphism refers only to the structure of completion forests, we can always obtain such an isomorphism between two any completion forests by adding empty nodes and edges to these completion forests. This is similar to what we have made between two completion trees [17]. In the following, we introduce a distance between two isomorphic completion forests.

Definition 5 (Distance) *Let $\mathcal{F}$=$(G, T\langle \widehat{x_1} \rangle, \ldots, T\langle \widehat{x_n} \rangle)$ and $\mathcal{F}' = (G', T\langle \widehat{x'_1} \rangle, \ldots, T\langle \widehat{x'_n} \rangle)$ two forest-like models with $G = (\mathbf{V}, \mathbf{E}, \mathbf{L})$, $G' = (\mathbf{V}', \mathbf{E}', \mathbf{L}')$, $T\langle \widehat{x_i} \rangle = \langle V_i, L_i, E_i \rangle$ and $T\langle \widehat{x'_j} \rangle = \langle V'_j, L'_j, E'_j \rangle$ for $1 \le i, j \le n$. Let φ be a bijection from $\mathbf{V}$ to $\mathbf{V}'$ such that $\varphi(\widehat{x_i}) = \widehat{x'_j}$ iff there is some individual a satisfying $a \in \mathbf{L}(\widehat{x_i}) \cap \mathbf{L}'(\widehat{x'_j})$. The distance between $\mathcal{F}$ and $\mathcal{F}'$, denoted $d(\mathcal{F}, \mathcal{F}')$, is defined as follows :*

$$d(\mathcal{F}, \mathcal{F}') = \sum_{i=1}^{n} d(T\langle \widehat{x_i} \rangle, T\langle \varphi(\widehat{x_i}) \rangle) + \max_{\langle x, y \rangle \in \mathbf{E}} (|L(\langle x, y \rangle) \vartriangle L'(\langle \varphi(x), \varphi(y) \rangle)|)$$

where $d(T, T') = \min_{\pi \in \Pi(T, T')} \{ \max_{\langle x, y \rangle \in E} (|L(x) \vartriangle L'(\pi(x))| + |L(\langle x, y \rangle) \vartriangle L'(\langle \pi(x), \pi(y) \rangle)| + |L(y) \vartriangle L'(\pi(y))|) \}$

with $S \vartriangle S' = (S \cup S') \setminus (S \cap S')$ for any two sets S and S', and $\Pi(T, T')$ is the set of all isomorphisms between two *trees T and T'.*

We have defined a distance over forest-like models whose sets of root nodes should be associated by a bijection φ according to Remark 1. As any distance, $d(\mathcal{F}, \mathcal{F}')$ should allow one to measure the difference between two forests $\mathcal{F}$ and $\mathcal{F}'$—that is—it must satisfy identity, symmetry and triangle inequality properties. For this purpose, we use an operator, namely max, which represents the greatest difference between two triples (composed of an edge and two nodes) associated by an isomorphism π between $\mathcal{F}$ and $\mathcal{F}'$. This operation max allows us to ensure the identity property but it is not sufficient for guaranteeing the triangle inequality property $d(\mathcal{F}, \mathcal{F}') \leq d(\mathcal{F}, \mathcal{F}'') + d(\mathcal{F}'', \mathcal{F}')$. For this reason, we must use a further operator, namely min, which allows for choosing an isomorphism π from all isomorphisms between $\mathcal{F}$ and $\mathcal{F}'$ (one of which gets involved to determine $d(\mathcal{F}, \mathcal{F}'') + d(\mathcal{F}'', \mathcal{F}')$) such that the greatest difference between triples associated by π is smallest.

Lemma 2 *The function $d(\mathcal{F}, \mathcal{F}')$ in Definition 5 satisfies identity, symmetry and triangle inequality properties.*

To show that the distance in Definition 5 yields a total pre-order over a set of isomorphic completion forests, we define a relation "$\mathcal{F} \leq \mathcal{F}'$" over isomorphic completion forests including a forest $\mathcal{F}_0$ containing only empty labels as follows : $\mathcal{F} \leq \mathcal{F}'$ iff $d(\mathcal{F}_0, \mathcal{F}) \leq d(\mathcal{F}_0, \mathcal{F}')$.

Lemma 3 *The relation "$\leq$" is a total pre-order over isomorphic completion forests.*

All of the above notions provide sufficiently elements to define a revision operation for a $\mathcal{SHIQ}$ ontology O by another ontology O'.

Definition 6 (Revision Operation) *Let O and O' be two consistent $\mathcal{SHIQ}$ ontologies. A set of forest-like models of the revision of O by O', denoted $\mathsf{FM}(O, O')$, is defined as follows :*

$$\begin{aligned}\mathsf{FM}(O, O') = \{\mathcal{F} \in \mathsf{FM}(O', \mathsf{sub}(O)) \mid \exists \mathcal{F}_0 \in \mathsf{FM}(O, \mathsf{sub}(O')),\\ \forall \mathcal{F}' \in \mathsf{FM}(O', \mathsf{sub}(O)), \mathcal{F}'' \in \mathsf{FM}(O, \mathsf{sub}(O')) :\\ d(\mathcal{F}, \mathcal{F}_0) \leq d(\mathcal{F}', \mathcal{F}'')\}\end{aligned}$$

Intuitively, among the forest-like models in $\mathsf{FM}(O', \mathsf{sub}(O))$, $\mathsf{FM}(O, O')$ retains only those which are closest to forest-like models from $\mathsf{FM}(O, \mathsf{sub}(O'))$ thanks to the distance $d(\mathcal{F}_1, \mathcal{F}_2)$ that characterizes the difference between $\mathcal{F}_1$ and $\mathcal{F}_2$.

Example 2 *Consider again the ontology* UNI *of Example 1. As for simplification, assume that O is an ontology obtained by adding into* UNI *the axioms from Table 1, and O' consists of δ_1, δ_2. By applying the new tableau algorithm over O, the set $\mathsf{FM}(O, \mathsf{sub}(O'))$ contains 3 forest-like models $\mathcal{F}_1, \mathcal{F}_2$, and $\mathcal{F}_3$ in Figure 4. By running the new tableau algorithm over O', we obtain a forest-like model $\mathcal{F}_1' \in \mathsf{FM}(O', \mathsf{sub}(O))$ illustrated in Figure 4 among other forest-like models. By applying the distance formula introduced in Definition 5, we obtain $d(\mathcal{F}_1', \mathcal{F}_3) = 4$ and $d(\mathcal{F}_1', \mathcal{F}_1) = d(\mathcal{F}_1', \mathcal{F}_2) = 2$. According to Definition 8, $\mathsf{FM}(O, O')$ contains a unique forest-like model $\mathcal{F}_1'$. Note that $d(\mathcal{F}_x', \mathcal{F}_y) > 2$ for all $\mathcal{F}_y \in \mathsf{FM}(O, \mathsf{sub}(O'))$ and $\mathcal{F}_x' \in \mathsf{FM}(O', \mathsf{sub}(O))$ with $\mathcal{F}_x' \neq \mathcal{F}_1'$.*

As mentioned in Section 1, our goal is to propose a revision operation that ensures the principle of minimal change introduced by Alchourrón, Gärdenfors and Makinson [4] as postulates in belief revision framework. Katsuno and Mendelzon [11] have rephrased these postulates for propositional knowledge bases and shown that the existence of a total pre-order over models of a propositional knowledge base is equivalent to the satisfaction of the postulates. Inspired from Katsuno and Mendelzon's work [11], we rephrase the postulates in our setting as follows.

(P1) $\widehat{\mathcal{I}}(\mathcal{F}) \models \alpha$ for each forest-like model $\mathcal{F} \in \mathsf{FM}(O, O')$ and each axiom $\alpha \in O'$.
(P2) If $\mathsf{FM}(O, \mathsf{sub}(O')) \cap \mathsf{FM}(O', \mathsf{sub}(O)) \neq \emptyset$ then $\mathsf{FM}(O, O') = \mathsf{FM}(O, \mathsf{sub}(O')) \cap \mathsf{FM}(O', \mathsf{sub}(O))$.
(P3) If O' is consistent then $\mathsf{FM}(O, O') \neq \emptyset$.
(P4) If $\mathsf{FM}(O_1, \mathsf{sub}(O_1')) = \mathsf{FM}(O_2, \mathsf{sub}(O_2'))$ and $\mathsf{FM}(O_1', \mathsf{sub}(O_1)) = \mathsf{FM}(O_2', \mathsf{sub}(O_2))$ then $\mathsf{FM}(O_1, O_1') = \mathsf{FM}(O_2, O_2')$.
(P5) $\mathsf{FM}(O, O') \cap \mathsf{FM}(O'', \mathsf{sub}(O) \cup \mathsf{sub}(O')) \subseteq \mathsf{FM}(O, O' \cup O'')$.
(P6) If $\mathsf{FM}(O, O') \cap \mathsf{FM}(O'', \mathsf{sub}(O) \cup \mathsf{sub}(O')) \neq \emptyset$ then $\mathsf{FM}(O, O' \cup O'') \subseteq \mathsf{FM}(O, O') \cap \mathsf{FM}(O'', \mathsf{sub}(O) \cup \mathsf{sub}(O'))$.

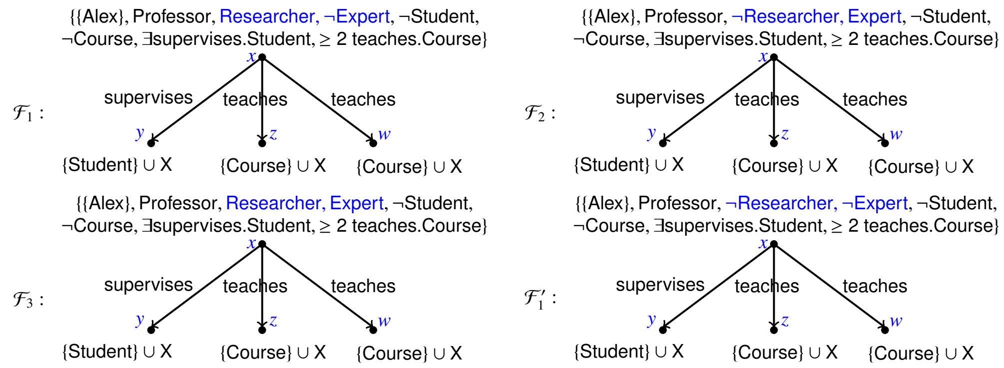
Intuitively, **(P1)** guarantees that all axioms from O' can be inferred from the revised ontology. **(P2)** says that the initial ontology O is not changed if $O \cup O$ is consistent. **(P3)** is a condition preventing a revision from introducing unwarranted inconsistency. **(P4)** says that the revision should be independent of the syntax of ontologies. In fact, if we replace O_1 and O_1' with O_2 and O_2' such that they admit the same forest-like models then revision ontologies admit the same forest-like models as well. **(P5)** and **(P6)** can ensure the principle of minimal change.

Theorem 1 *The revision operation $\mathsf{FM}(O, O')$ described in Definition 6 satisfies the postulates **(P1)-(P6)**.*

The equivalence between the existence of a total pre-order over forest-like models and the satisfaction of **(P1)-(P6)** also holds in our setting. Indeed, $\mathsf{FM}(O, O')$ in Definition 6 retains only forest-like models from $\mathsf{FM}(O', \mathsf{sub}(O))$ which are closest to forest-like models from $\mathsf{FM}(O, \mathsf{sub}(O'))$ according to the distance between completion forests. This distance infers the total pre-order "$\leq$" over forest-like models. This observation allows us to get straightforwardly the result saying that the postulates imply a total pre-order over forest-like models since we consider only models such as forest-like models over which a total pre-order exists already. In addition, the clause that a total pre-order

α_4 : ¬Professor ⊑ ∀supervises.(¬Student) ⊓ (≤ 1 teaches.Course)	Someone who is not a professor cannot supervise any student and does not teach more than a course
α_5 : Student⊥Course, Student⊥Professor, Student⊥Researcher, Student⊥Expert	A student is not a course nor a professor nor a researcher nor an expert (note that $A \perp B$ is equivalent to $A \sqsubseteq \neg B$)
α_6 : Course⊥Professor, Course⊥Researcher, Course⊥Expert	A course is not a professor nor a researcher nor an expert

TABLE 1 – Axioms added into the ontology UNI



where X = {¬Professor, ¬Expert, ¬Researcher, ∀supervises.(¬Student), (≤ 1 teaches.Course)}

FIGURE 4 – Completion forests yielding models of UNI ($\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$) and of $\{\delta_1, \delta_2\}$ ($\mathcal{F}'_1$)

over forest-like models implies the postulates is proved by Theorem 1. Therefore, the principle of minimal change is also ensured in our revision.

5 Computing The Revised Ontology

In this section, we present a procedure for constructing a $\mathcal{SHIQ}$ ontology O^* that admits at least forest-like models in $\mathsf{FM}(O, O')$. It has turned out [8] that there may not exist a DL-lite ontology which admits exactly a given set of models. It is also the case for $\mathcal{SHIQ}$ ontologies. To address this issue, we are borrowing the notion of *maximal approximation* from De Giacomo and colleagues' work [8] to define *upper approximation ontology* in our setting as follows.

Definition 7 (Upper approximation) *Let O and O' be two consistent $\mathcal{SHIQ}$ ontologies with revision operation* $\mathsf{FM}(O, O')$. *We use $\mathcal{S}(O'')$ to denote the signature of an ontology O''. An ontology O^* is an* upper approximation *from* $\mathsf{FM}(O, O')$ *if (i) $\mathcal{S}(O^*) \subseteq \mathcal{S}(O) \cup \mathcal{S}(O')$; (ii)* $\mathsf{FM}(O, O') \subseteq \mathsf{FM}(O^*)$*; (iii) There does not exist any ontology O'' such that* $\mathsf{FM}(O, O') \subseteq \mathsf{FM}(O'') \subset \mathsf{FM}(O^*)$.

Definition 7 characterizes the approximation ontology we should build such that it admits all models in $\mathsf{FM}(O, O')$. An interesting point is that if such an upper approximation exists it is unique up to semantic equivalence. We show that such an upper approximation in Definition 7 actually exists and propose a procedure to build it.

Definition 8 (Revised ontology) *Let $O = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ and $O' = (\mathcal{T}', \mathcal{R}', \mathcal{A}')$ be two consistent $\mathcal{SHIQ}$ ontologies with* $\mathsf{FM}(O, O') = \{\mathcal{F}_1, \cdots, \mathcal{F}_n\}$ *for $1 \leq i \leq n$. For each $\mathcal{F}_i = (G_i, T\langle\widehat{x_1}\rangle, \ldots, T\langle\widehat{x_m}\rangle)$ with $G_i = (\mathbf{V}_i, \mathbf{E}_i, \mathbf{L}_i)$ and $T\langle\widehat{x_j}\rangle = \langle V_j, L_j, E_j\rangle$ ($1 \leq j \leq m$), let $\mathcal{V}_i = \mathbf{V}_i \cup V_1 \cup \cdots \cup V_m$. A revised ontology $O^* = (\widehat{\mathcal{T}}, \widehat{\mathcal{R}}, \widehat{\mathcal{A}})$ of O by O' is defined as follows : $\widehat{\mathcal{R}} := \mathcal{R}'$, $\widehat{\mathcal{T}} := \mathcal{T}' \cup \{\top \sqsubseteq \bigsqcup_{1 \leq i \leq n} (\bigsqcup_{x \in \mathcal{V}_i} (\bigsqcap_{C \in L_i(x)} C))\}$, and $\widehat{\mathcal{A}}$ contains a set of assertions as follows :*

$$\{C(x) \in \mathcal{A}'\} \cup \{R(x,y) \in \mathcal{A}'\} \ \cup$$
$$\{C(x) \in \mathcal{A} \mid X \subseteq L_i(x), X \in \mathsf{Flat}(C), 1 \leq i \leq n\} \ \cup$$
$$\{R(x,y) \in \mathcal{A} \mid R \in L_i(\langle x,y \rangle), 1 \leq i \leq n\} \ \cup$$
$$\{x \neq y \mid x, y \in \mathbf{I} \cup \mathbf{I}', x \neq y \in \mathcal{F}_i, 1 \leq i \leq n\}.$$

The construction of revised ontology O^* retains all concept and role axioms as well as assertions from O'. It also adds to O^* a new concept axiom which is built literally from $\mathsf{FM}(O, O')$. When building a forest-like model $\mathcal{F}$ by running the tableau algorithm on O^*, this concept axiom forces to choose a node x from a forest-like model $\mathcal{F}_i \in \mathsf{FM}(O, O')$ to add its label $L(x)$ to the current node of $\mathcal{F}$. Apart from the forest-like models in $\mathsf{FM}(O, O')$, the tableau algorithm may build a forest-like model whose nodes have labels coming from different forest-like models from $\mathsf{FM}(O, O')$. This is why $\mathsf{FM}(O^*)$ may be larger than $\mathsf{FM}(O, O')$.

Note that the concept axiom built from $\mathsf{FM}(O, O')$ allows for capturing semantic parts of only role and concept axioms from O which should be propagated to O^*. To transfer semantic parts of the assertions from O to O^*, it is needed to determine assertions from O which remain to be satisfied in forest-like models from $\mathsf{FM}(O, O')$. This means that there may be some assertion from O which cannot be propagated to O^*. For example, O contains assertions $R(a,b)$, $S(a,c)$ while O' contains axioms $\top \sqsubseteq \forall R.\bot$ and $\top \sqsubseteq \forall S.\top$. By construction, each forest-like model for O' has an empty edge between a and b but a is an S-neighbor of c. This implies that $S(a,c)$ but not $R(a,b)$ will be added to O^*.

Example 3 *To continue Example 2, we construct from* $\mathsf{FM}(O, O')$ *an ontology* O^* *which admits a unique forest-like model* $\mathcal{F}_1'$ *according to Definition 8. Thus,* O^* *contains the following axioms :* $\mathsf{Expert} \sqsubseteq \forall \mathsf{supervises}.(\neg\mathsf{Student})$, $\mathsf{Researcher} \sqsubseteq \forall \mathsf{supervises}.(\neg\mathsf{Student})$ *(from* O'*), and*

$\top \sqsubseteq (\mathsf{Professor} \sqcap \neg\mathsf{Researcher} \sqcap \mathsf{Expert} \sqcap \neg\mathsf{Course} \sqcap \neg\mathsf{Student} \sqcap \exists\mathsf{supervises}.\mathsf{Student} \sqcap (\geq 2\ \mathsf{teaches}.\mathsf{Course}) \sqcup (\mathsf{Professor} \sqcap \mathsf{Researcher} \sqcap \neg\mathsf{Expert} \sqcap \neg\mathsf{Course} \sqcap \neg\mathsf{Student} \sqcap \exists\mathsf{supervises}.\mathsf{Student} \sqcap (\geq 2\ \mathsf{teaches}.\mathsf{Course}) \sqcup (\mathsf{Student} \sqcap \forall\mathsf{supervises}.(\neg\mathsf{Student}) \sqcap (\leq 1\ \mathsf{teaches}.\mathsf{Course}) \sqcap \neg\mathsf{Course} \sqcap \neg\mathsf{Professor} \sqcap \neg\mathsf{Researcher} \sqcap \neg\mathsf{Expert}) \sqcup (\mathsf{Course} \sqcap \forall\mathsf{supervises}.(\neg\mathsf{Student}) \sqcap (\leq 1\ \mathsf{teaches}.\mathsf{Course}) \sqcap \neg\mathsf{Student} \sqcap \neg\mathsf{Professor} \sqcap \neg\mathsf{Researcher} \sqcap \neg\mathsf{Expert})$, $\mathsf{Professor}(\mathsf{Alex})$.

We can now formulate an important result which affirms that the revised ontology O^* defined for two given ontologies O and O' according to Definition 8 is an upper approximation from $\mathsf{FM}(O, O')$. Our argument relies heavily on the specific behavior of the sat-rule and the particularity of the concept axiom added to O^*. In fact, if one knows the result of application of the sat-rule (i.e. the subset S chosen from $\mathsf{sub}(O)$) to each node of a forest-like model, she knows also the whole forest-like model.

Theorem 2 *Let* O *and* O' *be two consistent* $\mathcal{SHIQ}$ *ontologies. The revised ontology* O^* *of* O *by* O' *is an upper approximation from* $\mathsf{FM}(O, O')$*. Additionally, the size of* O^* *is bounded by a triple exponential function in the size of* O *and* O'*.*

6 Optimizations

So far we have showed that the size of revised ontology is bounded by a triple exponential function in the size of initial ontology. This high complexity is not surprising and arises mainly from the following sources : (i) the characterization of the ontology semantics by using forest-like models obtained from exploring all non-deterministic branches ; (ii) the computation of the distance between two forest-like models may be exponential in the size of forest-like models, and (iii) the construction of an upper approximation ontology forces the tableau algorithm to use the sat-rule which considers exhaustively non-deterministic cases. We present optimization techniques to reduce the complexity arisen from the mentioned sources.

6.1 Computing distance between two forests

According to the formula of the distance between two forest-like models (Definition 5), there is a unique isomorphism between two root nodes of two forest-like models. Therefore, it suffices to investigate optimization of distance computation between two tree-like structures. We present an algorithm for computing the distance $d(T\langle x_0\rangle, T\langle z_0\rangle)$ that runs in time polynomial in the size of two trees $T\langle x_0\rangle$ and $T\langle z_0\rangle$. For lack of space, we only give the main ideas of the algorithm which are founded on the following observations :

(i) Given an isomorphism π, we denote $h(\pi) = \max_{\langle x,y\rangle \in E_1}(|L_1(x) \vartriangle L_2(\pi(x))| + |L_1(\langle x,y\rangle) \vartriangle L_2(\langle \pi(x), \pi(y)\rangle)| + |L_1(y) \vartriangle L_2(\pi(y))|)$.

There are at most $O(\ell)$ different values of $h(\pi)$ where ℓ is the maximum size of O and O'. In fact, by construction we have $|L(x)| \leq O(\ell)$ and $|L(\langle x,y\rangle)| \leq O(\ell)$ for each node x and edge $\langle x,y\rangle$ of trees. This allows us to partition $\Pi(T\langle x_0\rangle, T\langle z_0\rangle)$ into groups each of which corresponds to a value $v_i \in \Delta$ where $v_{i-1} > v_i$ for all $2 \leq i \leq m$.

(ii) For each value $v_i \in \Delta$ from the greatest to the smallest value, it is possible to determine polynomially whether there exists an isomorphism $\pi \in \Pi(T\langle x_0\rangle, T\langle z_0\rangle)$ such that $v_i > h(\pi)$. If there does not exist such an isomorphism π, we obtain $d(T\langle x_0\rangle, T\langle z_0\rangle) = v_i$. Otherwise, the algorithm considers the value v_{i+1}.

6.2 Constructing $\mathsf{FM}(O, O')$

According to Definition 6, $\mathsf{FM}(O, O')$ is built from $\mathsf{FM}(O, \mathsf{sub}(O'))$ and $\mathsf{FM}(O', \mathsf{sub}(O))$ where O is much larger than O'. As described in Section 3, the tableau algorithm must find all completion forests for O to build $\mathsf{FM}(O, \mathsf{sub}(O'))$. This construction involves at least two sources of complexity : (i) exponential blow-up arising from disjunction and numbering restrictions occurring in O, (ii) exponential blow-up arising from behavior of the sat-rule.

To address the first source of complexity, we use various optimization techniques in the literature such as basic and binary absorptions [13, 1]. However, this complexity belongs to intrinsicness of our characterization of ontology semantics since we need a model for each intrinsic non-deterministic case. Therefore, the construction of $\mathsf{FM}(O, \mathsf{sub}(O'))$ is as complex as answering YES to a query such as $O \models C \sqsubseteq D$ since a reasoner must consider all non-deterministic cases.

To address the second source of complexity, we perform the construction $\mathsf{FM}(O, O')$ in several stages : (i) Constructing $\mathsf{FM}(O, \mathsf{sub}(O'))$ without sat-rule. (ii) Applying indirectly the sat-rule by using absorption techniques to saturate the labels of nodes and edges in completion forests from $\mathsf{FM}(O, \mathsf{sub}(O'))$. (iii) Constructing $\mathsf{FM}(O', \mathsf{sub}(O))$ by propagating node and edge labels from completion forests in $\mathsf{FM}(O, \mathsf{sub}(O'))$. (iv) Choosing completion forests from $\mathsf{FM}(O', \mathsf{sub}(O))$ to build $\mathsf{FM}(O, O')$ by computing distance between completion forests from $\mathsf{FM}(O, \mathsf{sub}(O'))$ and $\mathsf{FM}(O', \mathsf{sub}(O))$.

7 Implementation and Experiments

We have implemented a revision engine as prototype, called OntoRev, which is based on the algorithms and definitions described in the previous sections. Similarly to DL reasoners such as HermiT [16], Pellet [6], FaCT++ [5], we have implemented in OntoRev various optimization techniques such as absorption, core/anywhere blockings. For instance, we have used basic and binary absorptions [13, 1] to reduce non-deterministic cases arising from disjunction. We have also applied the core blocking technique [3] beside the pairwise blocking technique to scale down the size of completion forests. Differently from existing tableau reasoners, we need to explore all intrinsic non-deterministic cases involved in ontologies to construct all completion forests. As a consequence, we always consider worst-case scenarios where all forests would be built to represent the semantics of an ontology. In the current version of OntoRev, some optimization techniques such as pruning of backtracking points for dealing with intrinsic non-determinism have not been implemented. The lack of implementations of advanced optimizations may slow down OntoRev when it runs on ontologies containing a numerous amount of non-determinism.

We have carried experiments on ontologies `GALEN`, `PIZZA` and `TRAINING` which are modified for simplication. The reason for this choice is that `PIZZA` is a small ontology with a numerous amount of non-determinism arising from disjunctions while `GALEN` would force a tableau algorithm to build completion forests with a sizeable depth. `TRAINING` has resulted from a FUI research project on e-learning [1] which involves a revision engine within its plateform.

We present in Table 2 the ontology characteristics used for tests in revision of an initial ontology by a revising ontology, and in Table 3 the obtained results. We have run all tests on a DELL with 8 Intel 3.4GHz Processors and 32Gb RAM under Ubuntu. As mention in Section 6.2, the construction of $\mathsf{FM}(O, \mathsf{sub}(O'))$ without sat-rule allows us to explore only intrinsic non-deterministic

1. http ://www.omendo.com/plateforme-learning-cafe

Initial Ontology	Ontology characteristics					
				Axioms		
	Concepts	Roles	Assertions	Inclusion	Equivalence	Disjointness
`GALEN_1`	2748	413	2	3238	699	2
`GALEN_2`	2748	413	1	3239	699	1
`PIZZA`	99	5	2	259	8	398
`TRAINING`	451	95	2	442	0	79

Revising Ontology	Ontology characteristics					
				Axioms		
	Concepts	Roles	Assertions	Inclusion	Equivalence	Disjointness
`REV_GALEN_1`	3	2	0	2	0	0
`REV_GALEN_2`	2	1	0	1	0	0
`REV_PIZZA`	4	1	0	2	0	0
`REV_TRAINING`	2	2	1	1	0	0

Table 2 – Ontologies for experiments with characteristics

Ontology	Revision result				
	$\lvert\mathsf{FM}(O, \mathsf{sub}(O'))\rvert$	$\lvert\mathsf{FM}(O, O')\rvert$	Tree depth	N^0 of disjunctions	Times (sec.)
`REV_GALEN_1`	1	1	3	11	3
`REV_GALEN_2`	1	1	6	17	4
`REV_PIZZA`	4096	4096	2	18	165
`REV_TRAINING`	2	2	1	4	2

Table 3 – Results of experiments

cases in ontologies. The set $\mathsf{FM}(O, \mathsf{sub}(O'))$ can help to construct $\mathsf{FM}(O', \mathsf{sub}(O))$ since it contains forests candidates which have minimal distances to $\mathsf{FM}(O, \mathsf{sub}(O'))$. Therefore, the size of $\mathsf{FM}(O, O')$ is greater than or equal to that of $\mathsf{FM}(O, \mathsf{sub}(O'))$. In addition, there may exist completion forests in $\mathsf{FM}(O, O')$ which are equivalent. This explains why the number of disjunctions (N^0 of disjunctions in Table 3) in the axiom of the resulting ontology O^* is small. Moreover, we can also restore axioms from initial ontology by checking whether a concept name occurs in a node label of a completion forest and, if that is not the case, we add directly to the resulting ontology the axioms transformed by absorption.

8 Conclusion

We have presented in this paper a model-based approach for revising a $\mathcal{SHIQ}$ ontology with individuals. An interesting feature of our approach is to introduce finite structures, namely *completion forests*, for characterizing the semantics of a $\mathcal{SHIQ}$ ontology. Semantic distance between expressive ontologies can now be translated onto a distance between completion forests. This feature is crucial to define a revision operation which ensures minimal change. Thanks to that distance we are able to determine a set of completion forests that a revised ontology should admit. To deal with inexpressiveness issue, we have introduced the notion of upper approximation ontology. Finally, we have also proposed optimization techniques for addressing two main sources of complexity, and the presented algorithms have been implemented and tested on various $\mathcal{SHIQ}$ ontologies.

Références

[1] Andreas Steigmiller, Thorsten Liebig et Birte Glimm: *Konclude : System description*. J. Web Sem., 27 :78–85, 2014.

[2] Bernhard Nebel: *Belief Revision and Default Reasoning : Syntax-Based Approaches*. Dans *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91). Cambridge, MA, USA, April 22-25, 1991.*, pages 417–428, 1991.

[3] Birte Glimm, Ian Horrocks et Boris Motik: *Optimized Description Logic Reasoning via Core Blocking*. Dans *Proceedings of the 5th International Conference on Automated Reasoning*, IJCAR'10, pages 457–471, 2010.

[4] Carlos Alchourrón, Peter Gärdenfors et David Makinson: *On the logic of theory change : Partial meet contraction and revision functions*. Journal of symbolic Logic, 50 :510–530, 1985.

[5] Dmitry Tsarkov et Ian Horrocks: *FaCT++ Description Logic Reasoner : System Description*. Dans *Proceedings of the Third International Joint Conference on Automated Reasoning*, IJCAR'06, pages 292–297, 2006.

[6] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur et Yarden Katz: *Pellet : A practical OWL-DL reasoner*. Web Semantics : Science, Services and Agents on the World Wide Web, 5(2) :51 – 53, 2007.

[7] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi et Peter F. Patel-Schneider (rédacteurs): *The Description Logic Handbook : Theory, Implementation, and Applications*. Cambridge University Press, 2007.

[8] Giuseppe De Giacomo, Maurizio Lenzerini, Antonella Poggi et Riccardo Rosati: *On the Approximation of Instance Level Update and Erasure in Description Logics*. Dans *Proceedings of AAAI*, pages 403–408, 2007.

[9] Guilin Qi et Jianfeng Du: *Model-based Revision Operators for Terminologies in Description Logics*. Dans *Proceedings of the 21st International Jont Conference on Artifical Intelligence*, IJCAI'09, pages 891–897, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.

[10] Guilin Qi, Weiru Liu et David A. Bell: *Knowledge Base Revision in Description Logics*. Dans *European Conference on Logics in Artificial Inteligence*, pages 386–398, 2006.

[11] Hirofumi Katsuno et Alberto O. Mendelzon: *Propositional knowledge base revision and minimal change*. Artificial Intelligence, 53(3) :263–294, 1991.

[12] Horrocks, Ian, Ulrike Sattler et Stephan Tobies: *Reasoning with Individuals for the Description Logic* SHIQ, pages 482–496. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000, ISBN 978-3-540-45101-3.

[13] Ian Horrocks et Ulrike Sattler: *Optimised reasoning for $\mathcal{SHIQ}$*. Dans *Proceedings of the 15th European Conference on Artificial Intelligence*, pages 277–281. Springer-Verlag, 2002.

[14] Ian Horrocks, Ulrike Sattler et Stephan Tobies: *Practical reasoning for expressive description logics*. Dans *Proceedings of LPAR*. Springer, 1999.

[15] Marcio Moretto Ribeiro et Renata Wassermann: *Base revision in description logics - preliminary results*. Proc. of IWOD, pages 69–82, 2007.

[16] Rob Shearer, Boris Motik et Ian Horrocks: *HermiT : A Highly-Efficient OWL Reasoner*. Dans Ruttenberg, Alan, Ulrile Sattler et Cathy Dolbear (rédacteurs) : *Proc. of the 5th Int. Workshop on OWL : Experiences and Directions (OWLED 2008 EU)*, Karlsruhe, Germany, October 26–27 2008.

[17] Thinh Dong, Chan Le Duc, Philippe Bonnot et Myriam Lamolle: *Tableau-Based Revision over* SHIQ *TBoxes*. Dans *Logic for Programming, Artificial Intelligence, and Reasoning - 20th International Conference, LPAR-20 2015, Suva, Fiji, November 24-28, 2015, Proceedings*, pages 575–590, 2015.

[18] Thomas Eiter et Georg Gottlob: *On the Complexity of Propositional Knowledge Base Revision, Updates, and Counterfactuals*. Artificial Intelligence, 57 :227–270, 1992.

[19] Zhe Wang, Kewen Wang et Rodney Topor: *A New Approach to Knowledge Base Revision in DL-Lite*. Proc. of 24th AAAI, pages 369–374, 2010.

[20] Zhe Wang, Kewen Wang, Zhiqiang Zhuang et Guilin Qi: *Instance-driven Ontology Evolution in DL-lite*. Dans *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 1656–1662. AAAI Press, 2015, ISBN 0-262-51129-0.

[21] Zhiqiang Zhuang, Zhe Wang, Kewen Wang et Guilin Qi: *Contraction and Revision over DL-Lite TBoxes*. Dans *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, pages 1149–1155. AAAI Press, 2014.

[22] Zhiqiang Zhuang, Zhe Wang, Kewen Wang et Guilin Qi: *DL-Lite Contraction and Revision*. Journal of Artificial Intelligence Research, 56 :329–378, 2016.