Reasoning with F-ALCI Over Lattices

George Voutsadakis\textsuperscript{1,2}, Giora Slutzki\textsuperscript{1}, and Vasant Honavar\textsuperscript{1}

\textsuperscript{1} Department of Computer Science, Iowa State University, Ames, IA 50011
\textsuperscript{2} School of Mathematics and Computer Science, Lake Superior State University, Sault Ste. Marie, MI 49783

Abstract. The fully contextualized, federated semantic web language F-ALCI is generalized to allow reasoning over arbitrary certainty lattices. These are complete distributive lattices with a negation operation which is order-reversing and involutive. The resulting language, denoted LF-ALCI, apart from supporting fully contextualized modular reasoning, encompasses reasoning over structures with a wide variety of orderings, including fuzzy reasoning. The work takes after similar work of Sraccia, who pioneered reasoning over lattices for the description logic ALC.

1 Introduction

This paper is a contribution to the ongoing efforts to endow the world wide web with machine interpretable content and machine interoperable resources and services, thus transforming it into the semantic web \cite{4}. Knowledge representation and knowledge acquisition in the semantic web are aimed to be performed, at least partially, by machines and, thus, have to be machine-friendly. The most common platform for this machine-oriented knowledge representation are ontologies. They provide both a syntactic and a semantic framework for reasoning with resources and relations between them. Because in a typical web application many agents contribute parts of an ontology that are often partially overlapping, a significant effort in the area of ontologies focuses on what are called modular or federated ontologies \cite{6,8,12,19,29,2,3}. These are ontologies with multiple modules. Each of the modules is typically constructed independently of other modules and possibly stored in a different machine. The semantics of a modular ontology allows for a smooth interaction between the overlapping parts of these various independently developed modules.

For constructing ontologies, the most commonly used languages are those that form decidable fragments of first-order logic; they are termed description logics and the reader is referred to the introduction \cite{1} for more details. On the other hand, to support modularization, several modular ontology formalisms have been introduced and explored. Examples include distributed description logics (DDL)\cite{6,8}, E-Connections \cite{12}, semantic importing \cite{19}, semantic binding \cite{29}, and package-based description logics (P-DL) \cite{2,3}. In all these approaches, several constructs are provided for sharing of knowledge across ontology modules. An alternative approach to knowledge reuse relies on a particular
notion of modularity of ontologies based on the notion of conservative extensions [9, 11, 10], which allows ontology modules to be interpreted using standard semantics by requiring that they share the same interpretation domain.

This paper focuses on a specific kind of description logic and a specific kind of modular ontology language. We will combine various features of both languages in order to create a novel modular ontology language that will allow us to reason about uncertain or imprecise knowledge on the semantic web.

The description logic that our language will be based on is the language $L\text{-}ALCI$. Its syntax coincides with the syntax of the well-known description logic $ALCI$, which allows forming negations, conjunctions, disjunctions, universal and existential role quantifications of concepts and role inversions. $L\text{-}ALCI$ and $ALCI$ differ from each other with respect to their semantics. Whereas reasoning in $ALCI$ is based on boolean interpretations and, therefore, can accommodate certain (true or false) knowledge, the semantics of $L\text{-}ALCI$ provides for reasoning with uncertain information. More precisely, the concept expressions of $L\text{-}ALCI$ are interpreted in a complete distributed lattice $L$ of certainty values having a negation operation. A description logic along these lines was introduced by Straccia in [24], whose paper was the inspiration for considering this framework. Knowledge management of uncertain or imprecise information has been considered before, for instance in [13, 16, 17] using probability theory, [15] using possibility theory, [20, 21] using many-valued logics and [5, 14, 22, 23, 26, 28] using fuzzy logic. This list of references is indicative of some of the efforts spent towards handling uncertainty and it is not meant to be exhaustive.

The modular ontology language that we will be basing our investigations on is the language $F\text{-}ALCI$, which was introduced and studied in some detail by the authors in [27]. It is a modular ontology language whose main feature is that all its logical connectives are contextualized, i.e., are interpreted locally and their interpretations are then propagated using image-domain relations that relate individuals in different interpretation domains. This language is related to P-DLs, which were previously considered in [2, 3]. P-DLs have only logical negation as a contextualized connective. Furthermore, the semantics in the two platforms are different. P-DL semantics imposes more restrictions on the image-domain relations resulting in a very tight relationship between the overlapping elements of the interpretation domains. Consequently, P-DL semantics is less flexible.

This paper aims at combining the expressive power of $F\text{-}ALCI$ with the idea drawn from $L\text{-}ALCI$ of allowing interpretations to vary over arbitrary uncertainty lattices. In this way, given an uncertainty lattice $L$, a new modular ontology language $LF\text{-}ALCI$ is obtained. Its syntax is identical with the syntax of $F\text{-}ALCI$. Its semantics, however, allows reasoning with $F\text{-}ALCI$ concepts and $F\text{-}ALCI$ subsumptions using lattice-theoretic tools. More precisely, membership of an element in a concept extension or of a pair of elements in a role extension is not just true or false but is, instead, assigned a certainty value drawn from the given certainty lattice $L$. Based on this basic assignments and various recursive rules involving both the available contextualized logical connectives
and the image-domain relations, all memberships in complex concept expressions assume specific certainty values. Apart from formulating this framework, we also present a reduction from \( \text{LF-ALCI} \) to \( \text{L-ALCI} \). This reduction allows us to draw conclusions on various computational aspects of \( \text{LF-ALCI} \) from corresponding statements known about \( \text{L-ALCI} \). For instance, Straccia [24] has shown that, under appropriate restrictions on the structure of subsumptions and on the certainty lattice \( L \), satisfiability of an ABox in the language \( \text{L-ALC} \) is PSPACE-complete with respect to the joint cardinality of the ABox and the lattice. Our result shows that, under the same restrictions as Straccia’s on \( L \), satisfiability in the language \( \text{LF-ALCI} \) is of the same complexity as satisfiability of an acyclic TBox in \( \text{L-ALCI} \). The transformation of an acyclic TBox to an ABox, however, may be of exponential length in general. It is still open whether the tracing technique [25] may be used to transform an acyclic \( \text{L-ALCI} \) TBox to an \( \text{L-ALCI} \) ABox. In that case, under the same restrictions as Straccia’s on \( L \), satisfiability in the language \( \text{LF-ALCI} \) would also be PSPACE-complete.

In summary, the main contribution of the paper is the study of a modular ontology language that incorporates contextualized and uncertain reasoning. Whereas modularity and contextualization, on the one hand, and uncertainty, on the other, have been studied separately before, to the best of our knowledge, it is the first time that they are being studied in a common framework. Moreover, for acyclic terminological knowledge, the combination of these features does not increase the computational complexity of reasoning.

2 A Quick Review of \( \text{L-ALCI} \) and \( \text{LF-ALCI} \)

2.1 \( \text{L-ALCI} \) Syntax and Semantics

Let \( L = (L, \leq) \) be a complete distributive lattice, with \( L \) its universe and \( \leq \) the partial ordering of \( L \). Denote by \( \land \) and \( \lor \), as usual, the meet and join operations, respectively, induced by \( \leq \) and by 1 and 0 its top and bottom elements. This lattice is perceived as a lattice of “certainty” values into which the expressions of the language \( \text{F-ALC} \) will be interpreted. To accommodate negation, we assume that \( L \) is also equipped with a \textit{negation}, i.e., an anti-monotone involutive unary operation \( \sim \) with respect to \( \leq \). More explicitly, this means that, for all \( a, b \in L \),

- \( a \leq b \) implies \( \sim b \leq \sim a \) and
- \( \sim \sim a = a \).

The term \textit{certainty lattice} is used to refer to the structure \( L = (L, \leq, \sim) \). Note that in such a lattice, the De Morgan Laws hold. Examples of certainty lattices, many of which have been widely used in various contexts and for various forms of reasoning in AI, are provided in [24]. Some of them are:

- \textbf{ Classical 0-1:} The 2-element Boolean algebra \( L_{\{0,1\}} \), where 0 denotes falsity and 1 truth;
- \textbf{ Fuzzy:} The real unit interval \( L_{[0,1]} \), with negation \( \sim \alpha = 1 - \alpha \), for all \( \alpha \in [0,1] \).
- **Four-Valued**: Belnap's $\mathcal{FOU} \mathcal{R}$, with four values $f, t, u, i$, satisfying $f \leq u \leq t$ and $f \leq i \leq t$. Negation is given by $\neg f = t$ and $\neg u = u, \neg i = i$. Intuitively, $u$ stands for unknown and $i$ for inconsistency.

- **Many-Valued**: This is the lattice $\mathbb{L}_n$ with $n$ values $0, \frac{1}{n-1}, \ldots, \frac{n-2}{n-1}, 1$ with the ordinary ordering. Negation is as in $\mathbb{L}_{[0,1]}$.

The syntax of the language $\mathcal{L} \hspace{0.15cm} \mathcal{ALCI}$ is identical to the syntax of the well-known language $\mathcal{ALCI}$. More precisely, we have a collection $\mathcal{C}$ of concept names and a collection $\mathcal{R}$ or role names. Then the set $\hat{\mathcal{R}}$ or roles is the set $\hat{\mathcal{R}} = \mathcal{R} \cup \mathcal{R}^\sim$, where $\mathcal{R}^\sim = \{ R^\sim : R \in \mathcal{R} \}$. The set $\hat{\mathcal{C}}$ of concepts is defined recursively using the following syntax rules for constructing new concepts:

$$A \in \mathcal{C}, \top, \bot, \neg C, C \sqcap D, C \sqcup D, \exists R.C, \forall R.C,$$

for all $C, D \in \hat{\mathcal{C}}$ and all $R \in \hat{\mathcal{R}}$. A (subsumption) formula is an expression of the form $C \sqsubseteq D$, with $C, D \in \hat{\mathcal{C}}$. An ontology (also known as a knowledge base) or, for the purposes of this paper, as a TBox is a finite set of formulas. The $\mathcal{ALCI}$-semantics interprets all logical connectives in the usual way (see Chapter 2 of [1]). This notion of an $\mathcal{ALCI}$-interpretation is generalized to obtain the notion of an $\mathcal{L} \hspace{0.15cm} \mathcal{ALCI}$-interpretation, which allows reasoning with uncertain and/or imprecise information based on the certainty values provided by the lattice $\mathbb{L}$. The definition is essentially that of [24].

An $\mathcal{L} \hspace{0.15cm} \mathcal{ALCI}$-interpretation, or simply $\mathcal{L}$-interpretation, is a pair $\mathcal{I} = \langle \Delta^\mathcal{I}, \mathcal{I} \rangle$, where $\Delta^\mathcal{I}$ is a nonempty set, the domain of the interpretation, and $\mathcal{I}$ is an interpretation function mapping

- a concept name $C$ into a function $C^\mathcal{I} : \Delta^\mathcal{I} \to \mathbb{L}$
- a role name $R$ into a function $R^\mathcal{I} : \Delta^\mathcal{I} \times \Delta^\mathcal{I} \to \mathbb{L}$.

For all $a, b \in \Delta^\mathcal{I}$, $C^\mathcal{I}(a)$ and $R^\mathcal{I}(a, b)$ are supposed to provide the degree of certainty of $a$ being an instance of the concept $C$ and of $(a, b)$ being an instance of the role $R$, respectively, under the interpretation $\mathcal{I}$. The interpretation function $\mathcal{I}$ extends to arbitrary roles and concepts by using the following rules recursively, for all $a, b \in \Delta^\mathcal{I}$:

$$
\begin{align*}
R^\mathcal{I}(a, b) & = R^\mathcal{I}(b, a) \\
\top^\mathcal{I}(a) & = 1 \\
\bot^\mathcal{I}(a) & = 0 \\
(\neg C)^\mathcal{I}(a) & = \neg C^\mathcal{I}(a) \\
(C \sqcap D)^\mathcal{I}(a) & = C^\mathcal{I}(a) \land D^\mathcal{I}(a) \\
(C \sqcup D)^\mathcal{I}(a) & = C^\mathcal{I}(a) \lor D^\mathcal{I}(a) \\
(\forall R.C)^\mathcal{I}(a) & = \bigwedge_{b \in \Delta^\mathcal{I}} (\neg R^\mathcal{I}(a, b) \lor C^\mathcal{I}(b)) \\
(\exists R.C)^\mathcal{I}(a) & = \bigvee_{b \in \Delta^\mathcal{I}} (R^\mathcal{I}(a, b) \land C^\mathcal{I}(b)).
\end{align*}
$$

Given a formula $C \sqsubseteq D$ and an interpretation $\mathcal{I}$, $\mathcal{I}$ satisfies $C \sqsubseteq D$ or $\mathcal{I}$ is a model of $C \sqsubseteq D$ if, for all $a \in \Delta^\mathcal{I}$, we have $C^\mathcal{I}(a) \leq D^\mathcal{I}(a)$. An interpretation $\mathcal{I}$ satisfies a knowledge base $T$ or is a model of $T$ if it is a model of every formula $\tau \in T$. 

4
Given a specific collection \( D \subseteq C \) of concept names, an \( L \)-interpretation \( I \) \( D \)-satisfies a knowledge base \( T \) or is a \( D \)-model of \( T \) if it is a model of every formula \( \tau \in T \), such that \( D^I : \Delta^I \to \{0, 1\} \), for all \( D \in D \). In other words a \( D \)-model interprets the concept names in \( D \) as subsets of \( \Delta^I \) in the ordinary way.

**Example:** Suppose that we are dealing with a knowledge base \( T \) consisting of information about the employees of a certain university. This knowledge base uses \( ALCI \) as the underlying language and contains three concept names \( \text{Faculty}, \text{Highly-Paid-Faculty}, \text{Productive-Faculty} \). Clearly, if the language is to be interpreted in any certainty lattice, the extension of the concept \( \text{Faculty} \) should be a \( \{0, 1\} \)-interpretation, whereas the extensions of the other two concept names could be arbitrary evaluations in this lattice. Thus, to answer a query concerning faculty of the university, e.g., to get the certainty values of the set of all faculty that are both highly-paid and productive, we would like to evaluate the extension of \( \text{Highly-Paid-Faculty} \sqcap \text{Productive-Faculty} \) in the knowledge base \( T \), that includes the axioms

\[
\text{Highly-Paid-Faculty} \sqsubseteq \text{Faculty} \\
\text{Productive-Faculty} \sqsubseteq \text{Faculty},
\]

under all interpretations that \( \{\text{Faculty}\} \)-satisfy \( T \). If there were other concept names in \( T \) that should be given “crisp” interpretations, they should be added in this latter set. ■

### 2.2 The Federated Language \( F-ALCI \)

The fully contextualized federated extension of \( ALCI \), denoted by \( F-ALCI \), was introduced in [27].

Suppose a directed acyclic graph \( G = (V, E) \), with \( V = \{1, 2, \ldots, n\} \), is given. For technical reasons, a loop is added on each vertex of \( G \). For every node \( i \in V \), the signature of the \( i \)-language includes a set \( C_i \) of \( i \)-concept names and a set \( R_i \) of \( i \)-role names. We assume that all sets of names are pairwise disjoint. Out of these, a set of \( i \)-concepts \( \hat{C}_i \) and a set of \( i \)-roles \( \hat{R}_i \) are built.

**Definition 1 (Roles and Concepts)** The set of \( i \)-roles or \( i \)-role expressions \( \hat{R}_i \) consists of expressions of the form \( R, R^{-} \), with \( R \in R_j, (j, i) \in E \).

The set of \( i \)-concepts or \( i \)-concept expressions \( \hat{C}_i \), on the other hand, is defined recursively as follows:

\[
A \in \hat{C}_j, \top_j, \bot_j, \neg_j C, C \sqcap_j D, C \sqcup_j D, \exists_j R.C, \forall_j R.C,
\]

where \( (j, i) \in E, C, D \in \hat{C}_i \cap \hat{C}_j \) and \( R \in \hat{R}_i \cap \hat{R}_j \).

Using the concepts and roles of \( F-ALCI \), we define its formulas, as follows:

For any \( i \in V \), the \( i \)-formulas are expressions of the form \( C \sqsubseteq D \), with \( C, D \in \hat{C}_i \). An \( F-ALCI \)-ontology or \( F-ALCI \)-knowledge base is a collection
\[ T = \{ T_i \}_{i \in V}, \text{ where } T_i \text{ is a finite set of } i \text{-formulas, } i \in V. \] The \( T_i \)'s are referred to as the modules of the ontology \( T \).

An F-\( \mathcal{ALC}T \)-interpretation \( I = (\{ T_i \}_{i \in V}, \{ r_{ij} \}_{(i,j) \in E}) \) consists of a family \( T_i = (\Delta^i, \cdot^i), i \in V, \) of local interpretations, together with a family of image domain relations \( r_{ij} \subseteq \Delta^i \times \Delta^j, (i, j) \in E, \) such that \( r_{ii} = \text{id}_{\Delta^i}, \) for all \( i \in V. \)

**Notation:** For a binary relation \( r \subseteq \Delta^i \times \Delta^j, X \subseteq \Delta^i \) and \( S \subseteq \Delta^i \times \Delta^j, \) we set
\[
\begin{align*}
  r(X) & := \{ y \in \Delta^j : (\exists x \in X)((x, y) \in r) \}, \\
  r(S) & := \{(z, w) \in \Delta^j \times \Delta^j : (\exists(x, y) \in S)((x, z), (y, w) \in r)\}.
\end{align*}
\]

A local interpretation function \( \cdot^i \) interprets \( i \)-role names and \( i \)-concept names, as well as \( \bot \) and \( T \), as follows:

- \( C_i^i \subseteq \Delta^i \), for all \( C \subseteq C_i \),
- \( R_i^i \subseteq \Delta^i \times \Delta^i \), for all \( R \subseteq R_i \),
- \( \top_i^i = \Delta^i \), \( \bot_i^i = \emptyset \).

On the other hand, \( r_i^j \) interprets \( j \)-concept names and \( j \)-role names, for \( (j, i) \in E \), sometimes referred to as imported concept and role names, respectively, using the following rules:

- \( C_i^j \subseteq \Delta^j \), for all \( C \subseteq C_j \cap \hat{C}_i \),
- \( R_i^j \subseteq \Delta^j \times \Delta^j \), for all \( R \subseteq R_j \cap \hat{R}_i \),
- \( \top_i^i = r_i^j(\Delta^j), \top_i^i = \emptyset \).

The recursive features of the local interpretation function \( \cdot^i \) are as follows:

- \( R^{-i} = R_i^{-i} \), for all \( R \subseteq R_i \),
- \( (C \cap D)^i = r_i^j(C^j \cap D^j) \),
- \( (C \cup D)^i = r_i^j(C^j \cup D^j) \),
- \( (\exists f C)^i = r_i^j((x \in \Delta^j : (\exists y)((x, y) \in R^j \text{ and } y \in C^j))) \),
- \( (\forall j R.C)^i = r_i^j((x \in \Delta^j : (\forall y)((x, y) \notin R^j \text{ or } y \notin C^j))) \).

For all \( i \in V, \) \( i \)-satisfiability, denoted by \( \models_i \), is defined by \( I \models_i C \subseteq D \) if \( C_i^i \subseteq D_i^i \). Given a knowledge base \( T = \{ T_i \}_{i \in V} \), the interpretation \( I \) is a model of \( T \), written \( I \models T \), if \( I \models_i \tau \), for every \( \tau \in T_i \). Moreover, \( I \) is a model of \( T \), written \( T \models I \), whenever \( I \models_i T_i \), for every \( i \in V. \)

Let \( w \in V. \) Define \( G_w = (V_w, E_w) \) to be the subgraph of \( G \) induced by those vertices in \( G \) from which \( w \) is reachable and \( T_w^* := \{ T_i \}_{i \in V_w} \). We say that an F-\( \mathcal{ALC}T \)-ontology \( T = \{ T_i \}_{i \in V} \) is consistent as witnessed by a module \( T_w \) if \( T_w \) has a model \( I = (\{ I_i \}_{i \in V_w}, \{ r_{ij} \}_{(i,j) \in E_w}) \), such that \( \Delta^w \neq \emptyset \). A concept \( C \) is satisfiable as witnessed by \( T_w \) if there is a model \( I \) of \( T_w \), such that \( C^w \neq \emptyset \). A concept subsumption \( C \subseteq D \) is valid as witnessed by \( T_w \), denoted by \( C \subseteq D \) if, for every model \( I \) of \( T_w \), \( C^w \subseteq D^w \). An alternative notation for \( C \subseteq D \) is \( T^w_w \models |w C \subseteq D. \)
3 LF-\textit{ALCI}: Reasoning with F-\textit{ALCI} over Lattices

We now proceed to describe the syntax and semantics of LF-\textit{ALCI}, which extends the modular ontology language F-\textit{ALCI} to support reasoning over arbitrary certainty lattices.

Since the syntax of LF-\textit{ALCI} is identical with the syntax of F-\textit{ALCI}, which was reviewed in the previous section, we concentrate here on the semantics. As contrasted with the language F-\textit{ALCI}, the novel feature of the new language is its semantics which allows its expressions to be interpreted as arbitrary values in the certainty lattice \( \mathbf{L} \), rather than just as “true” (1) or “false” (0).

**Definition 2** An interpretation \( \mathcal{I} = \langle \mathcal{I}_i \rangle_{i \in \mathcal{V}} \), \( \{ r_{ij} \}_{(i,j) \in \mathcal{E}} \) consists of a family \( \mathcal{I}_i = (\Delta_i^n, ^i) \), \( i \in \mathcal{V} \), of local interpretations, together with a family of image domain relations \( r_{ij} : \Delta_i^n \times \Delta_j^n \to L \), \( i, j \in \mathcal{V} \), such that, for all \( i \in \mathcal{V} \),

\[
r_{ii}(a, b) = \begin{cases} 1, & \text{if } a = b \\ 0, & \text{otherwise} \end{cases}
\]

A local interpretation function \(^i\) interprets \( i\)-role names and \( i\)-concept names, as well as \( \bot_i \) and \( \top_i \), as follows:

- \( C^i : \Delta_i^n \to L \), for all \( C \in \mathcal{C}_i \),
- \( R^i : \Delta_i^n \times \Delta_i^n \to L \), for all \( R \in \mathcal{R}_i \),
- \( \top^i_i : \Delta_i^n \to L \) is the function \( \top^i_i(a) = 1 \), for all \( a \in \Delta_i^n \),
- \( \bot^i_i : \Delta_i^n \to L \) is the function \( \bot^i_i(a) = 0 \), for all \( a \in \Delta_i^n \).

The interpretations of imported role names and imported concept names are computed by the following rules, for all \( a, b \in \Delta_i^n \):

- \( C^i(a) = \bigvee_{c \in \Delta_i^n} (C^j(c) \land r_{ji}(c, a)) \), for all \( C \in \mathcal{C}_j \cap \widehat{\mathcal{C}}_i \),
- \( R^i(a, b) = \bigvee_{c, d \in \Delta_i^n} (R^j(c, d) \land r_{ji}(c, a) \land r_{ji}(d, b)) \), for all \( R \in \mathcal{R}_j \cap \widehat{\mathcal{R}}_i \),
- \( \top^j_j(a) = \bigvee_{c \in \Delta_j^n} (\top^j_j(c) \land r_{ji}(c, a)) \) = \( \bigvee_{c \in \Delta_i^n} r_{ji}(c, a) \),
- \( \bot^j_j(a) = 0 \).

**Example:** Assume that \( \mathbf{L} = \mathbf{L}_{[0,1]} \). Suppose that we are dealing with the knowledge base \( T \) containing information about certain products. It consists of two modules \( T_1 \) and \( T_2 \) describing different but related varieties of products that only partially match (like, e.g., cameras and camcorders). In this context, if the knowledge base is to compute the certainty value of product \( A \) in the domain \( \Delta^2 \) being in the extension of the concept \textbf{Expensive}, defined in module \( T_1 \), (see Figure 1), it would have to perform the following computation:

\[
\textbf{Expensive}^2(A) = (\textbf{Expensive}^1(a) \land r_{12}(a, A)) \lor (\textbf{Expensive}^1(b) \land r_{12}(b, A)) \\
= \left( \frac{3}{6} \land \frac{3}{4} \right) \lor \left( \frac{3}{4} \land \frac{1}{4} \right) \\
= \frac{3}{6} \lor \frac{3}{4} \\
= \frac{4}{8} \lor \frac{3}{8} = \frac{7}{8}.
\]

The conclusion is that product \( A \) in \( \Delta^2 \) is expensive with certainty degree \( \frac{7}{8} \). ■

The recursive features of the local interpretation function \(^i\) are given, for all \( a, b \in \Delta_i^n \), by:
Fig. 1. An Interpretation over $L_{[0,1]}$. 

- $R^{-1}(a, b) = R^i(b, a)$, for all $R \in R_i$.
- $(\neg_j C)^i(a) = \bigvee_{c \in A_i} (\neg C^j(c) \wedge r_{ji}(c, a))$
- $(C \cap_j D)^i(a) = \bigvee_{c \in A_i} (C^j(c) \wedge D^i(c) \wedge r_{ji}(c, a))$
- $(C \cup_j D)^i(a) = \bigvee_{c \in A_i} ((C^j(c) \vee D^i(c)) \wedge r_{ji}(c, a))$
- $(\exists_j R.C)^i(a) = \bigvee_{c \in A_i} (R^j(c, d) \wedge C^i(d) \wedge r_{ji}(c, a))$
- $(\forall_j R.C)^i(a) = \bigvee_{c \in A_i} (\bigwedge_{d \in A_i} (\neg R^j(c, d) \vee C^i(d)) \wedge r_{ji}(c, a))$

**Example:** Assume that $L = L_{[0,1]}$. We illustrate the application of the recursive $\exists$-rule by computing $(\exists_1 R.C)^2(a)$ in the interpretation of a federated knowledge base $T$, that consists of two modules $T_1$ and $T_2$, which is depicted in Figure 2. The leftmost column of numbers gives the certainty value of membership in $C^1$. The second column provides the certainty values of membership in $R^1$ and the values between the two rectangular boxes are the certainty values for membership in the image domain relation $r_{12}$. The computation goes as follows:

$$(\exists_1 R.C)^2(a) = \left[ (R^1(c_1, d_1) \wedge C^1(d_1)) \vee (R^1(c_2, d_2) \wedge C^1(d_2)) \wedge r_{12}(c_1, a) \right] \vee \left[ (R^1(c_2, d_2) \wedge C^1(d_2)) \wedge R^i(c_3, a) \right] \vee \left[ (R^1(c_1, d_1) \wedge C^1(d_1)) \wedge r_{12}(c_2, a) \right]$$

$$= \left[ \left( \frac{1}{3} \wedge \frac{3}{4} \right) \vee \left( \frac{3}{8} \wedge \frac{3}{4} \right) \right] \vee \left[ \left( \frac{1}{2} \wedge \frac{1}{4} \right) \vee \left( \frac{5}{8} \wedge \frac{3}{4} \right) \right]$$

$$= \left( \frac{1}{2} \right) \vee \left( \frac{5}{8} \wedge \frac{3}{4} \right)$$

$$= \frac{3}{4}.$$

Thus $a$ is an element in $(\exists_1 R.C)^2$ with certainty value $\frac{3}{4}$. 

For all $i \in V$, $i$-satisfiability, denoted by $|=i$, is defined by $I |=i C \sqsubseteq D$ if, for all $a \in A^i$, $C^i(a) \leq D^i(a)$. Given a TBox $T = \{ T_i \}_{i \in V}$, the interpretation $I$ is a **model** of $T_i$, written $I |=i T_i$, if $I |=i \tau$, for every $\tau \in T_i$. Moreover, $I$ is a **model** of $T$, written $I |= T$, whenever $I |=i T_i$, for every $i \in V$.

Let $w \in V$. Define $G_w = (V_w, E_w)$ to be the subgraph of $G$ induced by those vertices in $G$ from which $w$ is reachable and $T_w := \{ T_i \}_{i \in V_w}$. We say that an **LF-ALCI-ontology** $T = \{ T_i \}_{i \in V}$ is **consistent as witnessed by a module** $T_w$. 

8
if $T_w$ has a model $\mathcal{I} = \langle \{I_i\}_{i \in V_w}, \{r_{ij}\}_{(i,j) \in E_w} \rangle$, such that $\top_w$ is not identically 0 (as a function). A concept $C$ is **satisfiable as witnessed by** $T_w$ if there is a model $\mathcal{I}$ of $T_w$, such that $C_w$ is not the zero function. A concept subsumption $C \sqsubseteq D$ is **valid as witnessed by** $T_w$, denoted by $C \sqsubseteq_w D$, if, for every model $\mathcal{I}$ of $T_w$, $C_w(a) \leq D_w(a)$, for all $a \in \Delta^w$. An alternative notation for $C \sqsubseteq_w D$ is $T_w \models_w C \sqsubseteq D$.

4 A Reduction from LF-\textit{ALCI} to L-\textit{ALCI}

A **reduction** $\mathcal{R}$ from an LF-\textit{ALCI} KB $\Sigma_d = \{T_i\}$ to an L-\textit{ALCI} KB $\Sigma := \mathcal{R}(\Sigma_d)$ follows along the same lines of a corresponding reduction from F-\textit{ALCI} to \textit{ALCI} presented in [27] and is obtained as follows:

The signature of $\Sigma$ is the union of the local signatures of the modules together with a global top $\top$, a global bottom $\bot$, local top concepts $\top_i$, for all $i \in V$, and, finally, a collection of new role names $\{R_{ij}\}_{(i,j) \in E}$, i.e.,

$$\operatorname{Sig}(\Sigma) = \bigcup_i (\mathcal{C}_i \cup R_i) \cup \{\top, \bot\} \cup \{\top_i : 1 \leq i \leq n\} \cup \{R_{ij} : (i,j) \in E\}.$$ 

Moreover, various axioms derived from the structure of $\Sigma_d$ are added to $\Sigma$.

- For each $C \in \mathcal{C}_i$, $C \sqsubseteq \top_i$ is added to $\Sigma$.
- For each $R \in R_i$, $\top_i$ is stipulated to be the domain and range of $R$, i.e., $\top \sqsubseteq \forall R \top_i$ and $\top_i \sqsubseteq \forall R \top_i$ are added to $\Sigma$.
- For each new role name $R_{ij}$, $\top_i$ is stipulated to be its domain and $\top_j$ to be its range, i.e., $\top_i \sqsubseteq \forall R_{ij} \top_i$ and $\top_i \sqsubseteq \forall R_{ij} \top_j$ are added to $\Sigma$.
- For each $C \sqsubseteq D \in T_i$, $\#_i(C) \sqsubseteq \#_i(D)$ is added to $\Sigma$, where $\#_i$ is a function from $\tilde{C}_i$ to the set of L-\textit{ALCI}-concepts. The precise definition of $\#_i$ is given below.
The mapping \( \#_i(C) \) serves to maintain the compatibility of the concept domains. It is defined by induction on the structure of \( C \in \hat{C}_i \):

- \( \#_i(C) = C \), if \( C \in C_i \);
- \( \#_i(C) = \exists R_{i,j} \#_j(C) \), if \( C \in C_j \cap \hat{C}_i \);
- \( \#_i(\neg D) = \exists R_{i,j}^{-1}(\neg \#_j(D) \cap I_j) \);
- \( \#_i(D \uplus j E) = \exists R_{i,j} \exists R_{i,j}^{-1}(\#_j(D) \uplus \#_j(E)) \), where \( \uplus = \cap \) or \( \uplus = \cup \);
- \( \#_i(\exists \exists R.D) = \exists R_{i,j} \exists R_{i,\hat{k}} \exists R_{i,j}^{-1}(\exists R.R_{k,j} \#_j(D))) \), for \( R \in R_k \cup R_{\hat{k}} \);
- \( \#_i(\forall \exists R.D) = \exists R_{i,j} \exists R_{i,\hat{k}} \exists R_{i,j}^{-1}(\forall R.R_{k,j} \#_j(D))) \), for \( R \in R_k \cup R_{\hat{k}} \).

In the next section we show that the reduction \( R \) is sound and complete in the sense that, if the local top concept \( T_w \) in \( R(\Sigma_d) \) is \( \{ T_i, \bot_i : i \in V \} \)-satisfiable in an \( L-ALCI \)-model of \( R(\Sigma_d) \), then \( \Sigma_d \) itself is consistent as witnessed by \( T_w \) and conversely, \( \{ T_i, \bot_i : i \in V \} \)-satisfiability will be referred to in the sequel as tb-satisfiability (top, bottom satisfiability) and a corresponding model termed a tb-model.

5 Soundness and Completeness of the Reduction \( R \)

In this section we present the main result of the paper, viz. that the soundness and completeness proofs can be carried out in the case of an interpretation into a general complete distributive lattice with negation, rather than just the classical Boolean interpretation.

5.1 Soundness

**Definition 3** Let \( \Sigma_d = \{ T_i \}_{i \in V} \) be an \( LF-ALCI \) KB and \( I = (\Delta^T, \mathcal{T}) \) an interpretation of the \( L-ALCI \) ontology \( R(\Sigma_d) \). Construct an interpretation \( \mathcal{V}(I) = \{ \{ T_i \}_{i \in V}, \{ r_{i,j} \}_{(i,j) \in E} \} \) for \( \Sigma_d \) as follows:

- \( \Delta^i = \{ a \in \Delta^T : \epsilon_i T_i(a) > 0 \} \), for all \( i \in V \);
- \( C^i(a) = C^T(a) \), for all \( a \in \Delta^i \) and every \( C \in C_i \);
- \( R^i(a, b) = R^T(a, b) \), for all \( a, b \in \Delta^i \) and every \( R \in R_i \);
- \( r_{i,j}(a, b) = R^j_{i,j}(a, b) \), for all \( a, b \in \Delta^i \), \( b \in \Delta^j \) and every \( (i, j) \in E \).

We start with an easy technical lemma that shows, roughly speaking, that the image of the interpretation of a concept \( C \) under the interpretation of one of the new role names \( R_{i,j} \) is equal to the interpretation of the concept \( \exists R_{i,j}^{-1} C \) in the same model. This lemma is preparatory in dealing with the various cases involved in the definition of the translation function \( \#_i \).

**Lemma 4** Let \( \Sigma_d \) be an \( LF-ALCI \) KB and \( I = (\Delta^T, \mathcal{T}) \) an interpretation for \( R(\Sigma_d) \). Then, for every concept \( C \in \hat{C}_i \),

\[
(\exists R_{i,j}^{-1} C)^T(a) = \bigvee_{c \in \Delta^T} (C^T(c) \land R^j_{i,j}(c, a)), \text{ for all } a \in \Delta^T.
\]
Proof: We do indeed have

\[
\begin{align*}
(\exists R_{ij}^c, C)^T &= \bigvee_{c \in \Delta^T} (R_{ij}^T (a, c) \land C^T (c)) \quad \text{(by the definition of } T) \\
&= \bigvee_{c \in \Delta^T} (C^T (c) \land R_{ij}^T (c, a)) \quad \text{(by the definition of } R_{ij}^T).
\end{align*}
\]

\[
\]

Next, we present another technical lemma which supplies the precise value of the interpretation of the concept \(\forall R_{kj}^c (\forall R (\forall R_{kj}^c, #_j (C)))\) in terms of the translation \(#_j (C)\) of a concept \(C \in \hat{C}\) and the role name \(R \in \mathcal{R}_k\). This lemma will help us deal with the universal quantification case involved in the recursive definition of the translation function \(#_1\).

**Lemma 5** Let \(\Sigma_d\) be an LF-ALC\(\mathcal{T}\) KB and \(\mathcal{I} = \langle \Delta^T, T \rangle\) an interpretation for \(\mathcal{R}(\Sigma_d)\). Then, for all \(D \in \hat{C}\), \(R \in \mathcal{R}_k\) and for all \(c \in \Delta^T\),

\[
\begin{align*}
(\forall R_{kj}^c (\forall R (\forall R_{kj}^c, #_j (D))))^T (c) &= \bigwedge_{f \in \Delta^T} \left( \sim \bigvee_{d, e \in \Delta^T} (R_{kj}^T (d, c) \land R^T (d, e) \land R_{kj}^T (e, f)) \lor #_j (D)^T (f) \right).
\end{align*}
\]

(2)

**Proof:**

Using the definition of \(T\) three times, distributivity twice and De Morgan’s Laws twice, we get

\[
\begin{align*}
(\forall R_{kj}^c (\forall R (\forall R_{kj}^c, #_j (D))))^T (c) &= \bigwedge_{d \in \Delta^T} \left( \sim R_{kj}^T (c, d) \lor (\forall R (\forall R_{kj}^c, #_j (D)))^T (d) \right) \\
&= \bigwedge_{d \in \Delta^T} \left( \sim R_{kj}^T (d, c) \lor \bigwedge_{e \in \Delta^T} \left( \sim R^T (d, e) \lor (\forall R_{kj}^c, #_j (D))^T (e) \right) \right) \\
&= \bigwedge_{d \in \Delta^T} \left( \sim R_{kj}^T (d, c) \lor \bigwedge_{e \in \Delta^T} \left( \sim R^T (d, e) \lor \left( \bigwedge_{f \in \Delta^T} (\sim R_{kj}^T (e, f) \lor #_j (D)^T (f)) \right) \right) \right) \\
&= \bigwedge_{d \in \Delta^T} \left( \bigwedge_{e \in \Delta^T} \left( \sim R_{kj}^T (d, c) \lor R^T (d, e) \lor \left( \bigwedge_{f \in \Delta^T} (\sim R_{kj}^T (e, f) \lor #_j (D)^T (f)) \right) \right) \right) \\
&= \bigwedge_{f \in \Delta^T} \left( \bigwedge_{d, e \in \Delta^T} \left( \sim R_{kj}^T (d, c) \land R^T (d, e) \land \left( \bigwedge_{f \in \Delta^T} (\sim R_{kj}^T (e, f) \lor #_j (D)^T (f)) \right) \right) \right) \\
&= \bigwedge_{f \in \Delta^T} \left( \bigwedge_{d, e \in \Delta^T} \left( \sim R_{kj}^T (d, c) \land R^T (d, e) \land #_j (D)^T (f) \right) \right).
\end{align*}
\]

To relate the interpretation \(\mathcal{I}\) with its federated counterpart \(\mathcal{F}(\mathcal{I})\), we need to establish a correspondence between the interpretation of the translation \(#_1 (C)\) of a concept \(C \in \hat{C}\) under \(\mathcal{I}\) and that of the concept \(C\) under \(\mathcal{F}(\mathcal{I})\). This relationship is explored in the following lemma.

**Lemma 6** Let \(\Sigma_d\) be an LF-ALC\(\mathcal{T}\) KB, \(\mathcal{I} = \langle \Delta^T, T \rangle\) a tb-interpretation for \(\mathcal{R}(\Sigma_d)\) and \(\mathcal{F}(\mathcal{I}) = \{ \mathcal{I}_i \}_{i \in V}, \{ r_{ij} \}_{(i, j) \in E}\), with \(\mathcal{I}_i = \langle \Delta^i, T^i \rangle\). Then, for all \(i \in V\), all \(C \in \hat{C}\) and all \(a \in \Delta^i\), \(#_i (C)^T (a) = C^i (a)\).
The proof is by structural induction on $C$ and will be omitted.

The following is the soundness theorem for the reduction $\mathcal{R}$.

**Theorem 7 (Soundness)** Let $\Sigma_d$ be an LF-$\mathcal{ALC}$ KB, and $T_w$ a module of $\Sigma_d$. If $T_w$ is tb-satisfiable with respect to $\mathcal{R}(T^*_w)$, then $\Sigma_d$ is consistent as witnessed by $T_w$.

**Proof:**
Suppose that $T_w$ is tb-satisfiable with respect to $\mathcal{R}(T^*_w)$. Then $\mathcal{R}(T^*_w)$ has a model $\mathcal{I} = (\Delta^T, \mathcal{T})$, such that $\Delta^T_w$ is not identically 0. Our goal is to show that $F(\mathcal{I}) = \langle \{I_i\}_{i \in V_w},\{r_{ij}\}_{(i,j) \in E_w} \rangle$ is a model of $T^*_w$, such that $\Delta^w \neq \emptyset$.

Clearly, we have $\Delta^w = \{a \in \Delta^T : \Delta^T_w(a) \neq 0\} \neq \emptyset$, by the hypothesis. So it suffices to show that $F(\mathcal{I})$ is a model of the federated ontology $T_w$, i.e., that it satisfies $I_i \models T_i$, for every $i \in V_w$. Suppose that $C \subseteq D \subseteq T_i$. By the construction of $\mathcal{R}(T^*_w)$ and the fact that $\mathcal{I} \models \mathcal{R}(T^*_w)$, we must have, for all $a \in \Delta^T$, $\#_i(C)^T(a) \leq \#_i(D)^T(a)$, whence, by Lemma 6, for all $a \in \Delta^T$, $C^i(a) \leq D^i(a)$, showing that $F(\mathcal{I}) \models T^*_w$. □

### 5.2 Completeness of the Reduction
We turn now to the proof of the completeness of the reduction $\mathcal{R}$. Informally speaking, it will be shown that, if an LF-$\mathcal{ALC}$ KB $\Sigma_d$ is consistent as witnessed by a module $T_w$, then the corresponding local top concept $T_w$ in $\Sigma = \mathcal{R}(\Sigma_d)$ is satisfiable.

**Definition 8** Suppose that $\Sigma_d$ is an LF-$\mathcal{ALC}$ KB and that $\mathcal{I}_d = \langle \{I_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ is a model of $\Sigma_d$. Construct a tb-interpretation $\mathcal{I} := G(\mathcal{I}_d) = (\Delta^T, \mathcal{T})$ of $\mathcal{R}(\Sigma_d)$ as follows:

- $\Delta^T = \bigcup_{i \in V} \Delta^i$;
- $I^T_i(a) = \begin{cases} 1, & \text{if } a \in \Delta^i \\ 0, & \text{otherwise}, \text{ for every } a \in \Delta^i, i \in V; \end{cases}$
- $C^T(a) = \begin{cases} C^i(a), & \text{if } a \in \Delta^i \\ 0, & \text{otherwise}, \text{ for every } a \in \Delta^T, C \in C_i; \end{cases}$
- $R^T(a,b) = \begin{cases} R^i(a,b), & \text{if } a,b \in \Delta^i \\ 0, & \text{otherwise}, \text{ for every } a,b \in \Delta^T, R \in R_i; \end{cases}$
- $R^T_{ij}(a,b) = \begin{cases} r_{ij}(a,b), & \text{if } (a,b) \in \Delta^i \times \Delta^j \\ 0, & \text{otherwise}. \end{cases}$

To relate the federated interpretation $\mathcal{I}_d$ with its single module-counterpart $\mathcal{I} := G(\mathcal{I}_d)$, we need to establish a correspondence between the interpretation of the translation $\#_i(C)$ of a concept $C \in C_i$ under $F(\mathcal{I}_d)$ and that of the concept $C$ under $\mathcal{I}$. Such a correspondence is established in the following lemma.

**Lemma 9** Let $\Sigma_d$ be an LF-$\mathcal{ALC}$ KB, $\mathcal{I}_d = \langle \{I_i\}_{i \in V}, \{r_{ij}\}_{(i,j) \in E} \rangle$ a model of $\Sigma_d$ and set $\mathcal{I} := G(\mathcal{I}_d) = (\Delta^T, \mathcal{T})$. Then, for all $i \in V$, $C \in C_i$ and $a \in \Delta^i$, $\#_i(C)^T(a) = C^i(a)$.  

12
Proof:

This will follow directly from Lemma 6 once it is shown that $\mathcal{I}_d = \mathcal{F}(G(I_d))$. We have, using the full model names to keep notation clear,

- For all $i \in V$, $\Delta^{\mathcal{F}(G(I_d))}_i = \{ a \in \Delta^{G(I_d)}_i : \top^G(I_d)_i(a) \neq 0 \} = \Delta^d$,
- For every $C \in C_i$ and all $a \in \Delta^i$, $\mathcal{F}^{\mathcal{F}(G(I_d))}_i(a) = C^{\mathcal{F}(G(I_d))}_i(a) = C^d(a)$.
- For all $R \in \mathcal{R}_i$, $a, b \in \Delta^i$, we get $R^{\mathcal{F}(G(I_d))}_i(a, b) = R^G(I_d)_i(a, b) = R^d(a, b)$.
- For all $(i, j) \in E$, $a, b \in \Delta^i$, $r_{ij}^G(I_d)(a, b) = r_{ij}^d(a, b)$, where the superscripts of $r_{ij}$ specify the model of which they are part.

Therefore, we do indeed have $\mathcal{I}_d = \mathcal{F}(G(I_d))$. By Lemma 6, for all $a \in \Delta^i$, $\#_i(C)^d(a) = \#_i(C)^{\mathcal{F}(G(I_d))}_i(a) = C^{\mathcal{F}(G(I_d))}_i(a) = C^d(a)$.

The main goal of this section is to show that the converse of Theorem 7 also holds.

Theorem 10 (Completeness) Let $\Sigma_d = \{ T_i \}_{i \in V}$ be an $\mathbf{L}F$-ALC$\mathcal{T}$ ontology. If $\Sigma_d$ is consistent as witnessed by a module $T_w$, then $\top_w$ is tb-satisfiable with respect to $\mathcal{R}(T^*_w)$.

Proof:

Suppose that $\Sigma_d$ is consistent as witnessed by $T_w$. Thus, it has a model $\mathcal{I}_d = \{ \{ T_i \}_{i \in V}, \{ r_{ij} \}_{(i, j) \in E} \}$, such that $\Delta^w \neq \emptyset$. We proceed to show that $\mathcal{I} := G(I_d)$ is a tb-model of $\mathcal{R}(T^*_w)$, such that $\top^\mathcal{I}$ is not identically 0.

Since, by hypothesis, $\Delta^w \neq \emptyset$, there exists $a \in \Delta^w$. Thus, by the definition of $\top^\mathcal{I}$, $\top^\mathcal{I}(a) = 1$ and, therefore, $\top^\mathcal{I}$ is not identically 0.

Clearly, if $C \in C_i$, then, for all $a \in \Delta^2$,

$$C^\mathcal{I}(a) = \begin{cases} C^i(a), & \text{if } a \in \Delta^i \\ 0, & \text{otherwise} \end{cases} \leq \begin{cases} 1, & \text{if } a \in \Delta^i \\ 0, & \text{otherwise} \end{cases} = \top^\mathcal{I}(a),$$

whence $C \subseteq T_i$ holds in $\mathcal{I}$.

To see that $\top \subseteq \forall R^\mathcal{I}$. $\top_i$ holds in $\mathcal{I}$, we must show that, for all $a \in \Delta^2$,

$$\top^\mathcal{I}(a) \leq \bigwedge_{c \in \Delta^2} \sim R^\mathcal{I}(c, a) \lor \top^\mathcal{I}(c).$$

In turn, it suffices to show that, for all $c \in \Delta^2$, $\sim R^\mathcal{I}(c, a) \lor \top^\mathcal{I}(c) = 1$. In fact, if $c \notin \Delta^i$, then $R^\mathcal{I}(c, a) = 0$, whence $\sim R^\mathcal{I}(c, a) = 1$ and, thus, $\sim R^\mathcal{I}(c, a) \lor \top^\mathcal{I}(c) = 1$. If, on the other hand, $c \in \Delta^i$, then $\top^\mathcal{I}(c) = 1$, whence, again, $\sim R^\mathcal{I}(c, a) \lor \top^\mathcal{I}(c) = 1$. The fact that $\mathcal{I} \models \top \subseteq \forall R. \top_i$ may be shown similarly. Also along the same lines follow the proofs that the two concept inclusion axioms $\top \subseteq \forall R_{ij}. \top_i$ and $\top \subseteq \forall R_{ij}. \top_j$ are valid in $\mathcal{I}$.

Finally, suppose that $\#_i(C) \subseteq \#_i(D)$ is in $\mathcal{R}(\Sigma_d)$. Then $C \subseteq D \in T_i$ and, since $\mathcal{I}_d \models \Sigma_d$, we must have, for all $a \in \Delta^i$, $C^i(a) \leq D^i(a)$. Therefore, by Lemma 9, for all $a \in \Delta^2$, $\#_i(C)^d(a) \leq \#_i(D)^d(a)$, which shows that $\mathcal{I} \models \#_i(C) \subseteq \#_i(D)$. Thus, if $\mathcal{I}_d \models T^*_w$, we must have that $G(I_d) \models \mathcal{R}(T^*_w)$. This concludes the proof that, if $\Sigma_d$ is consistent as witnessed by a package $T^*_w$, then $\top_w$ is tb-satisfiable with respect to $\mathcal{R}(T^*_w)$.

By combining Theorems 7 and 10 we get the following
Theorem 11 (Soundness and Completeness) Suppose that $\Sigma_d = \{T_i\}_{i \in V}$ is an LF-ALCI ontology. $\Sigma_d$ is consistent as witnessed by a module $T_w$ if and only if $\top^w$ is $tb$-satisfiable with respect to $\Re(T^w)$.

5.3 Complexity

Straccia [24] provides a tableau-style calculus for deciding satisfiability in the language $L$-ALC under some restrictions on both the form of the TBox and the certainty lattice $L$.

In the TBox subsumptions are restricted to two kinds of axioms: concept specializations and concept definitions. A concept specialization is an axiom of the form $A \sqsubseteq C$ and a concept definition an axiom of the form $A = C$, where, in both cases, $A$ is a concept name and $C$ an arbitrary concept expression. In the TBox no concept name appears more than once on the left hand side of an axiom and no cyclic definitions are allowed.

The certainty lattice $L$ is assumed to be ps-safe (polynomial space safe). This requires that it be safe, which, roughly, means that

- the set of minimal pairs $D_L(c) = \min\{(a, b) \in L \times L : a \lor b \geq c\}$, with the order $(a, b) \leq (d, e)$ iff $a \leq d$ and $b \leq e$, is finite, for all $c \in L$;
- a certain decision problem concerning the inconsistency of a set of constraints is decidable;

and, moreover, that some polynomial boundedness conditions hold.

Straccia defines the combined complexity to be the complexity with respect to the sum of the cardinalities of the knowledge base and of the certainty lattice. Using a tableau-like calculus, he shows that satisfiability of an $L$-ALC knowledge base, having a TBox that satisfies the restrictions listed above, is PSPACE-complete with respect to the combined complexity of a ps-safe lattice, provided that the technique of transforming the TBox into an ABox [18] does not cause exponential blow up [24, 18]. It is not difficult to see that, as in the case of ALCI versus $ACL$, Straccia’s algorithm may be extended to accommodate inverse roles, i.e., to the language $L$-ALCI, while preserving the complexity.

To be able to use Straccia’s result to obtain PSPACE complexity for the consistency problem for LF-ALCI, apart from imposing similar restrictions to Straccia’s on the certainty lattice and the form of our local TBoxes, it is necessary that transforming the $L$-ALCI TBox resulting from the reduction $\Re$ into an ABox does not cause exponential blowup. Since this cannot be generally ensured, we define a federated LF-ALCI terminological knowledge base $\Sigma_d = \{T_i\}_{i \in V}$ to be tame if satisfiability of $\Re(\Sigma_d)$ can be reduced to that of an ABox by the technique of [18] in polynomial space.

Assuming that the certainty lattice we reason with is ps-safe and that the terminological axioms used in each module satisfy the same restrictions imposed by Straccia in [24] and that, furthermore, our federated knowledge base is tame, Straccia’s result (Proposition 4 of [24]), combined with Theorem 11, imply that satisfiability in $L$-ALCI is also PSPACE-complete with respect to the combined complexity of the ps-safe lattice.
Summary

In this paper we have introduced a new modular ontology language, LF-ALCI, that allows uncertain or imprecise fully contextualized reasoning in a federated setting. An LF-ALCI ontology consists of multiple ontology modules each of which can be viewed as an ALCI ontology. The interpretations of memberships in concept and role extensions are taking values in a certainty lattice L. This is a complete distributive lattice with a negation operation. Concept and role names can be shared by “importing” relations among modules, which are also interpreted in an uncertain fashion.

A reduction is provided from the federated language LF-ALCI to the language L-ALCI. This language is very similar to one considered in [24]. Using techniques very similar to the ones employed in [24], it may be shown that, under certain restrictions on L and the form of subsumptions allowed in the knowledge base, satisfiability in L-ALCI is PSPACE-complete with respect to the joint cardinality of the knowledge base and the lattice. Our reduction entails the PSPACE-completeness of the satisfiability problem in LF-ALCI subject to the same restrictions on the lattice L and the form of subsumptions allowed in the local modules.

References