NRC Publications Archive Archives des publications du CNRC

Knowledge Representation and Consistency Checking in Norm-Parameterized Fuzzy Description Logic

Zhao, Jidi; Boley, Harold; Du, Weichang

This publication could be one of several versions: author's original, accepted manuscript or the publisher's version. /
La version de cette publication peut être l'une des suivantes : la version prépublication de l'auteur, la version acceptée du manuscrit ou la version de l'éditeur.

Publisher's version / Version de l'éditeur:

*Proceedings of the International Conference on Intelligent Computing (ICIC 2009),
2009-09-19*

NRC Publications Record / Notice d'Archives des publications de CNRC:

<https://nrc-publications.canada.ca/eng/view/object/?id=5ba5ec6d-6935-439a-a278-6f1f0449b827>
<https://publications-cnrc.canada.ca/fra/voir/objet/?id=5ba5ec6d-6935-439a-a278-6f1f0449b827>

Access and use of this website and the material on it are subject to the Terms and Conditions set forth at

<https://nrc-publications.canada.ca/eng/copyright>

READ THESE TERMS AND CONDITIONS CAREFULLY BEFORE USING THIS WEBSITE.

L'accès à ce site Web et l'utilisation de son contenu sont assujettis aux conditions présentées dans le site

<https://publications-cnrc.canada.ca/fra/droits>

LISEZ CES CONDITIONS ATTENTIVEMENT AVANT D'UTILISER CE SITE WEB.

Questions? Contact the NRC Publications Archive team at

PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca. If you wish to email the authors directly, please see the first page of the publication for their contact information.

Vous avez des questions? Nous pouvons vous aider. Pour communiquer directement avec un auteur, consultez la première page de la revue dans laquelle son article a été publié afin de trouver ses coordonnées. Si vous n'arrivez pas à les repérer, communiquez avec nous à PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca.

Knowledge Representation and Consistency Checking in Norm-Parameterized Fuzzy Description Logic

Jidi Zhao¹, Harold Boley², and Weichang Du¹

¹Faculty of Computer Science,
University of New Brunswick, Fredericton, Canada
{Judy.Zhao, wdu} AT unb.ca

²Institute for Information Technology, National Research Council of Canada
Fredericton, NB, E3B 9W4 Canada
Harold.Boley AT nrc.gc.ca

Abstract. This paper has its motivation in the occurrence of uncertain knowledge in different application areas, and introduces an expressive fuzzy description logic that extends classical description logics to many-valued logics. We represent, and reason with, uncertain knowledge in the description logic *ALCHIN* extended by an interval-based, norm-parameterized Fuzzy Logic. First, the syntax and the semantics of the proposed fuzzy description logic are addressed. Then the paper presents an algorithm for consistency checking of knowledge bases in the proposed language.

1 Introduction

The Semantic Web is an evolving extension of the World Wide Web in which the semantics of the information and services available on the Web are formally described, making it possible for machines to better understand and satisfy requests of people. The Web ontology language (OWL), built upon the logic-based knowledge formalisms known as Description Logics (DLs), has enabled machine interpretation of such semantics and become a W3C recommendation for ontology representation.

Description Logics (DLs) [1] are a family of logic-based formalisms designed to represent and reason about the conceptual knowledge of arbitrary domains. Elementary descriptions of DLs are atomic concepts and atomic roles. Complex concept descriptions and role descriptions can be built from the elementary descriptions according to construction rules. As a notational convention, in this paper we will use a, b, x for individuals, A for an atomic concept, C and D for concept descriptions, R and P for atomic roles. Different description languages of DLs are distinguished by the kind of concept and role constructors (such as conjunction, disjunction, and exists restriction) allowed in their description language and the kinds of axiom allowed in their terminologies. For example, the *ALCHIN* DL [5] extends the well known *ALC* DL with inverse roles, role hierarchies (inclusion axioms), and number restrictions.

Standard DLs in use today are only suitable for representing and reasoning with crisp concepts, properties and constructors, but are not capable of dealing with uncertainty. To overcome this deficiency, considerable work has been carried out in integrating uncertain knowledge into DLs in the last decade. The approaches based on Fuzzy Set Theory and Fuzzy Logic, such as [7][8][6][2], have proved very promising in various Semantic Web applications.

Fuzzy Set Theory was first introduced by Zadeh [9] as an extension to the classical notion of a set to capture the inherent vagueness (the lack of crisp boundaries of sets). Fuzzy Logic is a form of multi-valued logic derived from Fuzzy Set Theory to deal with reasoning that is approximate rather than precise. In Fuzzy Logic, the degree of truth of a statement can range between 0 and 1, and is not constrained to the two truth values $\{0, 1\}$ or $\{false, true\}$ as in classical predicate logic. Formally, a fuzzy set X with respect to a set of elements Ω (also called a universe) is characterized by a membership function $\mu(x)$ which assigns a value in the real unit interval $[0, 1]$ to each element x in X ($x \in X$), notated as $X : \Omega \rightarrow [0, 1]$. $\mu(x)$ gives us a degree of an element x belonging to a set X . Such degrees can be computed based on some specific membership function. A fuzzy relation R over two fuzzy sets X_1 and X_2 is defined by a function $R : \Omega \times \Omega \rightarrow [0, 1]$.

Fuzzy Logic extends the Boolean operations defined on crisp sets and relations in the context of fuzzy sets and fuzzy relations. These operations, such as complement, union, and intersection, are interpreted as mathematical functions over the unit interval $[0, 1]$. The mathematical functions for fuzzy intersection are usually called t-norms ($t(\eta, \theta)$); those for fuzzy union are called s-norms ($s(\eta, \theta)$); and those for the fuzzy set complement are called negations ($\neg\eta$); here η, θ define the truth degrees of sets and relations and can range between 0 and 1. These functions usually satisfy certain mathematical properties. The most widely known operations in the Fuzzy Logic family of Zadeh Logic, Lukasiewicz Logic, Product Logic, and Gödel Logic, are summarized in Table 1.

Table 1. Fuzzy Operations

	Zadeh Logic	Lukasiewicz Logic	Product Logic	Gödel Logic
t-norm ($t(\eta, \theta)$)	$\min(\eta, \theta)$	$\max(\eta + \theta - 1, 0)$	$\eta \cdot \theta$	$\min(\alpha, \beta)$
s-norm ($s(\eta, \theta)$)	$\max(\eta, \theta)$	$\min(\eta + \theta, 1)$	$\eta + \theta - \eta \cdot \theta$	$\max(\eta, \beta)$
negation ($\neg\eta$)	$1 - \eta$	$1 - \eta$	if $\eta=0$ then 1 else 0	if $\eta=0$ then 1 else 0

In existing fuzzy extensions to DLs, most work is based on Zadeh logic (e.g. [6][7]) and some work is based on Product logic (e.g. [2]). In this paper, we extend our work in [10], and propose a general, parameterized semantics for t-norms and s-norms as well as allowing Lukasiewicz negation in the fuzzy description logic, called *fALCHIN*, so that the interpretation of complex concept descriptions can cover different types of operations in the parameterized Fuzzy Logic family. Another effort considering different logic operations in Fuzzy Logic is due to

Hájek [4], with the underlying DL language being $\mathcal{ALC}$. Furthermore, unlike other approaches based on Fuzzy Logic, which only deal with crisp subsumption of fuzzy concepts, our fuzzy description logic deals with fuzzy subsumption of fuzzy concepts and addresses its semantics. We argue that fuzzy subsumption of fuzzy concepts permits more adequate modeling of the uncertain knowledge existing in real world applications. The notion of fuzzy subsumption was first proposed in [8] and used in the forms $\geq l$ and $\leq u$, where $l, u \in [0, 1]$, but it was left unsolved how to do reasoning on fuzzy knowledge bases. In this paper, we also define a unified interval form $[l, u]$ and present a reasoning algorithm.

2 A Norm-Parameterized Fuzzy Description Logic

2.1 Syntax and Semantics of $f\mathcal{ALCHIN}$

Concept descriptions in $f\mathcal{ALCHIN}$ are formed based on the following syntax:

$$C \rightarrow \top | \perp | A | \neg A | \neg C | C \sqcap D | C \sqcup D | \exists R.C | \forall R.C | \geq nR | \leq nR$$

We can see that the syntax of this fuzzy description logic is identical to that of the standard description logic $\mathcal{ALCHIN}$. The difference is that the concepts and roles in $f\mathcal{ALCHIN}$ are defined as fuzzy concepts (i.e. fuzzy sets) and fuzzy roles (i.e. fuzzy relations).

Similar to standard DLs, the semantics of the proposed $f\mathcal{ALCHIN}$ is based on the notion of interpretation. Classical interpretations are extended to the notion of fuzzy interpretations by using membership functions that range over the interval $[0, 1]$. A fuzzy interpretation I is still a pair $I = (\Delta^I, \cdot^I)$ consisting of a domain Δ^I which is a non-empty set and of a fuzzy interpretation function $\cdot^I$ which maps each individual x into an element of Δ^I ($x \in \Delta^I$), each concept C into a membership function of $C^I : \Delta^I \rightarrow [0, 1]$, and each atomic role R into a membership function of $R^I : \Delta^I \times \Delta^I \rightarrow [0, 1]$.

The semantics of the top concept $\top$ is the greatest element in the domain Δ^I , that is, $\top^I = 1$ ($\forall x \in \Delta^I$). Note that, in standard DLs, the top concept $\top \equiv A \sqcup \neg A$, while in $f\mathcal{ALCHIN}$, $\top \neq A \sqcup \neg A$. As shown in Table 1, after applying the s-norms on $A \sqcup \neg A$, the result is no longer always equal to 1, so cannot be used as $\top$. Thus, we explicitly define the top concept, stating that the truth degree of x in $\top$ is 1. Similarly, the bottom concept $\perp$ is the least element in the domain, defined as $\perp^I = 0$ ($\forall x \in \Delta^I$).

In the Fuzzy Logic family, there exist two kinds of negation: Lukasiewicz negation and Gödel negation. We do not see much practical interest in the latter negation, and thus prefer to interpret the concept negation $\neg C$ based on Lukasiewicz negation, that is, $(\neg C)^I(x) = 1 - C^I(x)$. For example, if we have that the statement “John is young” has a truth value of greater than or equal to 0.8 ($Young(John) [0.8, 1]$), then the statement “John is not young” is written as $\neg Young(John) = \neg[0.8, 1] = [0, 0.2]$.

The interpretation of concept conjunction is defined by a t-norm as $(C \sqcap D)^I(x) = t(C^I(x), D^I(x))$ ($\forall x \in \Delta^I$). For example, if we have both $Young(John) [0.8, 1]$ and $Tall(John) [0.7, 1]$, and assume the product function in Product Logic is chosen as the t-norm, then the degree of truth that John is both young and tall is $(Young \sqcap Tall)(John) = t_P([0.8, 1], [0.7, 1]) = [0.56, 1]$.

The interpretation of concept disjunction is defined by an s-norm as $(C \sqcup D)^I(x) = s(C^I(x), D^I(x)) (\forall x \in \Delta^I)$. For example, if we have $Young(John) [0.8, 1]$ and $Tall(John) [0.7, 1]$, and the s-norm is the maximum function as in Zadeh logic or Gödel logic, then the degree of truth that John is young or tall is $(Young \sqcup Tall)(John) = \max([0.8, 1], [0.7, 1]) = [0.8, 1]$.

The semantics of exists restriction $\exists R.C$ is the result of viewing $\exists R.C$ as the open first order formula $\exists y.R(x, y) \wedge C(y)$ and the existential quantifier $\exists$ as a disjunction over the domain, defined as $\sup$ (supremum or least upper bound). Therefore, we define $(\exists R.C)^I(x) = \sup_{y \in \Delta^I} \{t(R^I(x, y), C^I(y))\}$.

A value restriction $\forall R.C$ is viewed as an implication of the form $\forall y \in \Delta^I, R^I(x, y) \rightarrow C^I(y)$. As proposed by Hájek [3], we interpret $\forall$ as $\inf$ (infimum or greatest lower bound). Furthermore, in classical logic, $a \rightarrow b$ is a shorthand for $\neg a \vee b$; we can thus interpret $\rightarrow$ as the Kleene-Dienes implication, and finally get its semantics as $(\forall R.C)^I(x) = \inf_{y \in \Delta^I} \{s(\neg R^I(x, y), C^I(y))\}$.

A fuzzy at-least restriction is of the form $\geq nR$, whose semantics

$$(\geq nR)^I(x) = \sup_{y_1, \dots, y_n \in \Delta^I, y_i \neq y_j, 1 \leq i < j \leq n} t_{i=1}^n \{R^I(x, y_i)\}$$

is derived from its first order reformulation

$$\exists y_1, \dots, y_n. \bigwedge_{i=1}^n R(x, y_i) \wedge \bigwedge_{1 \leq i < j \leq n} y_i \neq y_j.$$

The semantics states that there are at least n distinct individuals $(y_1, \dots, y_n)$ all of which satisfy $R(x, y_i)$ to some degree.

Furthermore, as suggested by [8], we define the semantics of a fuzzy at-most restriction as

$$\begin{aligned} (\leq nR)^I(x) &= \neg(\geq (n+1)R)^I(x) \\ &= \inf_{y_1, \dots, y_{n+1} \in \Delta^I, y_i \neq y_j, 1 \leq i < j \leq n+1} s_{i=1}^{n+1} \{\neg R^I(x, y_i)\} \end{aligned}$$

The semantics states that for $n+1$ role assertions $R(x, y_i)$ ($1 \leq i \leq n+1$) that can be formed, at least one of them satisfies $\neg R(x, y_i)$ to some degree. More intuitively, we can directly view the semantics of a fuzzy at-most number restriction as there being at most n unique individuals $(y_1, \dots, y_n)$ that satisfy $R(x, y_i)$ to some degree. For example, $(\leq 2R)(a) [0.8, 1]$ means that there are at most two role assertions about the individual a : $R(a, b_1)$ and $R(a, b_2)$ ($b_1 \neq b_2$). Assuming $x_{R(a, b_1)}$ is the truth degree of $R(a, b_1)$ and $x_{R(a, b_2)}$ is the truth degree of $R(a, b_2)$, then both $x_{R(a, b_1)} = [0.8, 1]$ and $x_{R(a, b_2)} = [0.8, 1]$ hold.

2.2 Knowledge Bases in $f\mathcal{ALCHIN}$

A fuzzy knowledge base $KB = \langle T, A \rangle$ in $f\mathcal{ALCHIN}$ consists of two parts: the terminological box (TBox T) and the assertional box (ABox A). The TBox has several kinds of axioms. A fuzzy concept inclusion axiom has the form of $C \sqsubseteq D [l, u]$ with $0 \leq l \leq u \leq 1$, which describes that the subsumption degree of truth between concepts C and D is from l to u .

For example, the axiom

$$Professor \sqsubseteq (\exists publishes. Journalpaper \sqcap \exists teaches. Graduatecourse) [0.8, 1]$$

states that the concept *professor* is subsumed by entities that *publish* journal papers and *teach* graduate courses with a truth degree of at least 0.8.

The FOL translation of a fuzzy concept inclusion axiom $C \sqsubseteq D$ has the form $\forall x. C(x) \rightarrow D(x)$; under the Kleene-Dienes implication, we thus define its semantics as

$$(C \sqsubseteq D)^I(x) = \inf_{x \in \Delta^I} C^I(x) \rightarrow D^I(x) = \inf_{x \in \Delta^I} \{s(\neg C^I(x), D^I(x))\}.$$

We also consider a fuzzy concept equivalence axiom of the form $C \equiv D [l, u]$ with the semantics as $C^I = D^I [l, u]$. In fact, concept equations ($C \equiv D [l, u]$) are interchangeable with a pair of fuzzy concept inclusion axioms ($C \sqsubseteq D [l, u]$ and $D \sqsubseteq C [l, u]$).

A fuzzy role inclusion axiom has the form $R \sqsubseteq P [l, u]$. Similarly to concept inclusion, the semantics of a role inclusion axiom $R \sqsubseteq P$ is defined as

$$(R \sqsubseteq P)^I(x, y) = \inf_{x, y \in \Delta^I} \{s(\neg R^I(x, y), P^I(x, y))\}.$$

A fuzzy role equivalence axiom is an expression of the form $R \equiv P [l, u]$, i.e., a pair of fuzzy role inclusion axioms ($R \sqsubseteq P [l, u]$ and $P \sqsubseteq R [l, u]$).

There are three kinds of assertions in the ABox: concept individual, role individual, and individual inequality. A fuzzy concept assertion and a fuzzy role assertion are of the form $C(a) [l, u]$ and the form $R(a, b) [l, u]$, respectively. The semantics of assertions is interpreted as the assertion $C(a) [l, u]$ (resp. $R(a, b) [l, u]$) being satisfied by I iff $a^I \in C^I = [l, u]$ (resp. $(a^I, b^I) \in R^I = [l, u]$). An individual inequality in the ABox is identical to standard DLs and has the crisp form $a \neq b$ for a pair of individuals a and b , with the natural semantics.

3 Consistency Checking

The most important reasoning task for an *fALCHIN* KB, as for any DL, is **consistency checking**, which refers to determining whether the KB is consistent or not. Here we first give formal definitions for *fALCHIN* consistency, generalizing those for a standard DL KB consistency.

Definition 1 (Complete). Let A_i^ε be an extended ABox obtained by applications of the completion rules (given in Definition 7). An extended ABox A_i^ε is complete if no more completion rule can be applied to A_i^ε . $\square$

Definition 2 (Solution). Let $\mathcal{C}$ be the constraint set obtained by applications of the completion rules. Let $Var(\mathcal{C})$ be the set of variables occurring in the constraint set $\mathcal{C}$. If the system of inequations in $\mathcal{C}$ is solvable, the result of the constraint set, i.e. the mapping $\Phi : Var(\mathcal{C}) \rightarrow [0, 1]$, is a solution. $\square$

Definition 3 (Clash). We say there is a clash in the extended ABox A_i^ε iff one of the following situations occurs:

1. $\{\perp(a) [l, u]\} \subseteq A_i^\varepsilon$ and $0 < l \leq u$
2. $\{\top(a) [l, u]\} \subseteq A_i^\varepsilon$ and $l \leq u < 1$
3. $\{A(a) [l_1, u_1], A(a) [l_2, u_2]\} \subseteq A_i^\varepsilon$ and $(u_1 < l_2 \text{ or } u_2 < l_1)$
4. $\{R(a, b) [l_1, u_1], R(a, b) [l_2, u_2]\} \subseteq A_i^\varepsilon$ and $(u_1 < l_2 \text{ or } u_2 < l_1)$
5. $\{(\leq nR)(a) [l, u]\} \cup \{R(a, b_i) [l_i, u_i] | 1 \leq i \leq n+1\} \cup \{b_i \neq b_j | 1 \leq i < j \leq n+1\} \subseteq A_i^\varepsilon$ and $([l_i, u_i] \subseteq [l, u] | 1 \leq i \leq n+1)$ $\square$

For example, if a knowledge base contains both assertions $Tall(John) [0, 0.2]$ and $Tall(John) [0.7, 1]$, since $0.2 < 0.7$, the third clash trigger will detect this as an inconsistency.

Definition 4 (Model). Let A_i^ε be the extended ABox obtained by applications of the completion rules. Let $I = (\Delta^I, \cdot^I)$ be a fuzzy interpretation, $\Phi : Var(\mathcal{C}) \rightarrow [0, 1]$ be a solution, x_α be the variable representing the truth

degree of assertion α . The pair $\langle I, \Phi \rangle$ is a model of the extended ABox A_i^ε if both of the following hold:

1. for each assertion $(C(a) \ x_{C(a)}) \in A_i^\varepsilon$, $(C^I(a) = x_{C(a)}) \in \Phi$.
2. for each assertion $(R(a, b) \ x_{R(a, b)}) \in A_i^\varepsilon$, $(R^I(a, b) = x_{R(a, b)}) \in \Phi$. $\square$

Definition 5 (Consistency). Let $KB = \langle T, A \rangle$ be an *fALCHIN* knowledge base where T is the TBox and A is the ABox. Let $I = (\Delta^I, \cdot^I)$ be a fuzzy interpretation, and $\Phi : Var(\mathcal{C}) \rightarrow [0, 1]$ be a solution. If there exists a model $\langle I, \Phi \rangle$ for the extended ABox resulting from $KB = \langle T, A \rangle$, we say the knowledge base $KB = \langle T, A \rangle$ is consistent. If there is no such model, we call the knowledge base inconsistent. $\square$

Note that the consistency checking of an *fALCHIN* knowledge base is a crisp ‘yes/no’ decision.

As in standard DLs, the reasoning procedure for *fALCHIN* tries to construct a model over the KB. Such a model has the shape of a forest, a collection of trees, with nodes corresponding to individuals, root nodes corresponding to named individuals, and edges corresponding to roles between individuals. Each node is associated with a node label, $\mathcal{L}(\text{individual})$. But unlike in standard DLs where a node is labeled only with concepts, each node in *fALCHIN* is associated with a label that consists of two components: the concept assertions for this individual and the corresponding constraints. Furthermore, each edge is associated with an edge label, $\mathcal{L}(\text{individual1}, \text{individual2})$ which consists of two components: the role assertions between the two individuals and the corresponding constraints, instead of simply being labeled with roles as in standard DLs.

The tableau algorithm consists of the following steps.

1. Replace each axiom of the form $C \equiv D [l, u]$ with the following two subsumption axioms: $C \sqsubseteq D [l, u]$ and $D \sqsubseteq C [l, u]$. Replace each axiom of the form $R \equiv P [l, u]$ with the following two subsumption axioms: $R \sqsubseteq P [l, u]$ and $P \sqsubseteq R [l, u]$. Transform every axiom in the TBox into its normal form. That is, replace each axiom of the form $C \sqsubseteq D [l, u]$ with $\top \sqsubseteq \neg C \sqcup D [l, u]$, $R \sqsubseteq P [l, u]$ with $\top \sqsubseteq \neg R \sqcup P [l, u]$.
2. Augment the ABox A with respect to the TBox T . That is, for each individual a in A and each axiom $\top \sqsubseteq \neg C \sqcup D [l, u]$ in T , add $(\neg C \sqcup D)(a) [l, u]$ to A ; for each role assertion $R(a, b)$ and each axiom $\top \sqsubseteq \neg R \sqcup P [l, u]$ in T , add $(\neg R \sqcup P)(a, b) [l, u]$ to A . The resulting ABox after finishing this step is called the initial extended ABox, denoted by A_0^ε . Initialize the set of constraints $\mathcal{C}_0$ to be the empty set.
3. Apply completion rules. Through applying these rules in arbitrary order, derived assertions are added to the extended ABox A_i^ε . At the same time, corresponding constraints that denote the semantics of the assertions are added to the constraint set $\mathcal{C}_j$ in the form of linear/non-linear inequations. This process stops when either A_i^ε contains a clash or no further rule can be applied to A_i^ε . If A_i^ε contains a clash, the knowledge base is inconsistent. If the complete ABox obtained this way does not contain a direct clash, the reasoning procedure continues to solve the inequations in the constraint set

$\mathcal{C}_j$. If the system of these inequations is unsolvable, the knowledge base is inconsistent, and consistent otherwise. The values returned from the system of inequations, if solvable, serve as the truth degrees of the entailment problem.

The completion rules are a set of consistency-preserving transformation rules. Each rule either detects a clash or derives one or more assertions and constraints. In the reasoning procedure, the application of some completion rules, including the role existential restrictions and at-least number restrictions, may lead to nontermination. Therefore, we have to find some blocking strategy to ensure the termination of the reasoning procedure.

Definition 6 (Blocking). Let a, b be anonymous individuals in the extended ABox A_i^ε , $A_i^\varepsilon(a)$ (respectively, $A_i^\varepsilon(b)$) be all the assertions in A_i^ε that are related to a (respectively, b), $\mathcal{C}_j(a)$ (respectively, $\mathcal{C}_j(b)$) be all the constraints in $\mathcal{C}_j$ that are related to a (respectively, b), $\mathcal{L}(a) = \langle A_i^\varepsilon(a), \mathcal{C}_j(a) \rangle$ and $\mathcal{L}(b) = \langle A_i^\varepsilon(b), \mathcal{C}_j(b) \rangle$ be the node labels for a and b . An individual b is said to be *blocked* by its ancestor a if $\mathcal{L}(b) \subseteq \mathcal{L}(a)$, where ' $\subseteq$ ' applies to both label components. $\square$

Definition 7 (Completion Rules). Let T be the TBox in normal form, A_i^ε be an extended ABox which is initialized to be the initial extended ABox A_0^ε , and $\mathcal{C}_0$ be the initial constraint set. Let Γ be either a range of truth degrees or the variable x_α denoting the truth degree of an assertion α . The completion rules are as follows:

1. **The concept assertion rule**

Condition:

A_i^ε contains $A(a) \Gamma$, $\{x_{A(a)} = \Gamma\} \not\subseteq \mathcal{C}_j$, and Γ is not the variable $x_{A(a)}$.

Action: $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{x_{A(a)} = \Gamma\}$; if $\{x_{\neg A(a)} = \neg \Gamma\} \not\subseteq \mathcal{C}_j$, then $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{x_{\neg A(a)} = \neg \Gamma\}$.

2. **The role assertion rule**

Condition:

A_i^ε contains $R(a, b) \Gamma$, $\{x_{R(a, b)} = \Gamma\} \not\subseteq \mathcal{C}_j$, and Γ is not the variable $x_{R(a, b)}$.

Action: $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{x_{R(a, b)} = \Gamma\}$; if $\{x_{\neg R(a, b)} = \neg \Gamma\} \not\subseteq \mathcal{C}_j$, then $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{x_{\neg R(a, b)} = \neg \Gamma\}$.

3. **The concept negation rule**

Condition: A_i^ε contains $\neg A(a) \Gamma$.

Action:

If Γ is not the variable $x_{\neg A(a)}$ and A_i^ε does not contain $A(a) \neg \Gamma$, then $A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{A(a) \neg \Gamma\}$. If Γ is the variable $x_{\neg A(a)}$ and $\{x_{\neg A(a)} = 1 - x_{A(a)}\} \not\subseteq \mathcal{C}_j$, then $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{x_{\neg A(a)} = 1 - x_{A(a)}\}$.

4. **The role negation rule**

Condition: A_i^ε contains $\neg R(a, b) \Gamma$.

Action:

If Γ is not the variable $x_{\neg R(a, b)}$ and A_i^ε does not contain $R(a, b) \neg \Gamma$, then $A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{R(a, b) \neg \Gamma\}$. If Γ is the variable $x_{\neg R(a, b)}$ and $\{x_{\neg R(a, b)} = 1 - x_{R(a, b)}\} \not\subseteq \mathcal{C}_j$, then $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{x_{\neg R(a, b)} = 1 - x_{R(a, b)}\}$.

5. **The concept conjunction rule**

Condition:

A_i^ε contains $(C \sqcap D)(a) \Gamma$, but it does not contain both $C(a) x_{C(a)}$ and

$D(a) \ x_{D(a)}$. $\mathcal{C}_j$ does not contain $t(x_{C(a)}, x_{D(a)}) = \Gamma$.

Action:

$A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{C(a) \ x_{C(a)}, D(a) \ x_{D(a)}\}$, $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{t(x_{C(a)}, x_{D(a)}) = \Gamma\}$.

6. **The role conjunction rule**

Condition:

A_i^ε contains $(R \sqcap P)(a, b) \ \Gamma$, but $\mathcal{C}_j$ does not contain $t(x_{R(a,b)}, x_{P(a,b)}) = \Gamma$.

Action:

$\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{t(x_{R(a,b)}, x_{P(a,b)}) = \Gamma\}$. If Γ is not the variable $x_{(R \sqcap P)(a,b)}$, $\mathcal{C}_{j+1} = \mathcal{C}_{j+1} \cup \{x_{(R \sqcap P)(a,b)} = \Gamma\}$. If A_i does not contain $R(a, b) \ x_{R(a,b)}$, $A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{R(a, b) \ x_{R(a,b)}\}$. If A_i does not contain $P(a, b) \ x_{P(a,b)}$, $A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{P(a, b) \ x_{P(a,b)}\}$.

7. **The concept disjunction rule**

Condition: A_i^ε contains $(C \sqcup D)(a) \ \Gamma$, but neither $C(a) \ x_{C(a)}$ nor $D(a) \ x_{D(a)}$. $\mathcal{C}_j$ does not contain $s(x_{C(a)}, x_{D(a)}) = \Gamma$.

Action:

$A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{C(a) \ x_{C(a)}, D(a) \ x_{D(a)}\}$, $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{s(x_{C(a)}, x_{D(a)}) = \Gamma\}$.

8. **The role disjunction rule**

Condition: A_i^ε contains $(R \sqcup P)(a, b) \ \Gamma$, but neither $R(a, b) \ x_{R(a,b)}$ nor $P(a, b) \ x_{P(a,b)}$. $\mathcal{C}_j$ does not contain $s(x_{R(a,b)}, x_{P(a,b)}) = \Gamma$.

Action: $A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{R(a, b) \ x_{R(a,b)}, P(a, b) \ x_{P(a,b)}\}$, $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{s(x_{R(a,b)}, x_{P(a,b)}) = \Gamma\}$.

9. **The inverse role rule**

Condition: A_i^ε contains $R^-(a, b) \ \Gamma$, but it does not contain $R(b, a) \ \Gamma$.

Action: $A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{R(b, a) \ x_{R(b,a)}\}$, $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{x_{R(b,a)} = \Gamma\}$.

10. **The exists restriction rule**

Condition: A_i^ε contains $(\exists R.C)(a) \ \Gamma$, and a is not blocked.

Action:

If there is no individual name b such that $C(b) \ x_{C(b)}$ and $R(a, b) \ x_{R(a,b)}$ are in A_i^ε , and $\mathcal{C}_j$ does not contain $t(x_{C(b)}, x_{R(a,b)}) = x_{(\exists R.C)(a)}$, then $A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{C(b) \ x_{C(b)}, R(a, b) \ x_{R(a,b)}\}$, $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{t(x_{C(b)}, x_{R(a,b)}) = x_{(\exists R.C)(a)}\}$, for each axiom $\top \sqsubseteq \neg C \sqcup D \ [l, u]$ in the TBox, add $A_{i+1}^\varepsilon = A_{i+1}^\varepsilon \cup \{(\neg C \sqcup D)(b) \ [l, u]\}$.

If there exists an individual name b such that $C(b) \ x_{C(b)}$ and $R(a, b) \ x_{R(a,b)}$ are in A_i^ε , but $\mathcal{C}_j$ does not contain $t(x_{C(b)}, x_{R(a,b)}) = x_{(\exists R.C)(a)}$, then $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{t(x_{C(b)}, x_{R(a,b)}) = x_{(\exists R.C)(a)}\}$;

If Γ is not the variable $x_{((\exists R.C)(a))}$, then if there exists $x_{((\exists R.C)(a))} = \Gamma'$ in $\mathcal{C}_j$, then $\mathcal{C}_{j+1} = \mathcal{C}_{j+1} \setminus \{x_{((\exists R.C)(a))} = \Gamma'\} \cup \{x_{((\exists R.C)(a))} = \sup(\Gamma, \Gamma')\}$, else add $\mathcal{C}_{j+1} = \mathcal{C}_{j+1} \cup \{x_{(\exists R.C)(a)} = \Gamma\}$.

11. **The value restriction rule**

Condition:

A_i^ε contains $(\forall R.C)(a) \ \Gamma$ and $R(a, b) \ \Gamma'$, but it does not contain $C(b) \ x_{C(b)}$. $\mathcal{C}_j$ does not contain $s(x_{C(b)}, x_{\neg R(a,b)}) = x_{(\forall R.C)(a)}$.

Action:

$A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{C(b) \ x_{C(b)}\}$, $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{s(x_{C(b)}, x_{\neg R(a,b)}) = x_{(\forall R.C)(a)}\}$.

If Γ is not the variable $x_{((\forall R.C)(a))}$, then if there exists $x_{((\forall R.C)(a))} = \Gamma'$ in $\mathcal{C}_j$,

add $\mathcal{C}_{j+1} = \mathcal{C}_{j+1} \setminus \{x_{((\forall R.C)(a))} = \Gamma'\} \cup \{x_{((\forall R.C)(a))} = \inf(\Gamma, \Gamma')\}$, otherwise,
 add $\mathcal{C}_{j+1} = \mathcal{C}_{j+1} \cup \{x_{((\forall R.C)(a))} = \Gamma\}$.

12. **The at-least rule**

Condition:

A_i^ε contains $(\geq nR)(a)$ Γ , a is not blocked, and there are no individual names $b_1, \dots, b_n$ such that $R(a, b_i)$ Γ_i ($1 \leq i \leq n$) are contained in A_i^ε .

Action:

$A_{i+1}^\varepsilon = A_i^\varepsilon \cup \{R(a, b_i) \mid 1 \leq i \leq n\}$,
 $\mathcal{C}_{j+1} = \mathcal{C}_j \cup \{t(x_{R(a, b_1)}, \dots, x_{R(a, b_n)}) = \Gamma\}$.

13. **The at-most rule**

Condition:

A_i^ε contains $n + 1$ distinguished individual names $b_1, \dots, b_{n+1}$ such that $(\leq nR)(a)$ Γ and $R(a, b_i)$ Γ_i ($1 \leq i \leq n + 1$) are contained in A_i^ε , $b_i \neq b_j$ is not in A_i^ε for some $i \neq j$, and if Γ is not the variable $x_{\leq nR(a)}$ and for any i ($1 \leq i \leq n + 1$), Γ_i is not the variable $x_{R(a, b_i)}$, $\Gamma_i \subseteq \Gamma$ holds.

Action:

For each pair b_i, b_j such that $j > i$ and $b_i \neq b_j$ is not in A_i^ε , the ABox A_{i+1}^ε is obtained from A_i^ε and the constraint set $\mathcal{C}_{j+1}$ is obtained from $\mathcal{C}_j$ by replacing each occurrence of b_j by b_i , and if Γ_i is the variable $x_{R(a, b_i)}$, $\mathcal{C}_{j+1} = \mathcal{C}_{j+1} \cup \{x_{R(a, b_i)} = \Gamma\}$. $\square$

4 Examples

In this section, we use two examples to explain the reasoning procedure presented in the above sections.

Example 1. In order to explain the importance of the blocking strategy, we consider a fuzzy knowledge base as follows, where we abbreviate the concept *CancerPatient* and the role *hasParent* by *CP* and *hP*, respectively.

$$T = \{CP \sqsubseteq \exists hP.CP [0.5, 1]\}, A = \{CP(P002) [0, 0.5]\}$$

The reasoning task is to check the consistency of this knowledge base. Assume we choose Zadeh Logic. First, we transform the axiom into its normal form $T = \{\top \sqsubseteq \neg CP \sqcup \exists hP.CP [0.5, 1]\}$.

After augmenting the ABox, we can initialize the extended ABox as $A_0^\varepsilon = \{CP(P002) [0, 0.5], (\neg CP \sqcup \exists hP.CP)(P002) [0.5, 1]\}$ and the initial constraint set as $\mathcal{C}_0 = \{\}$.

Next, we apply the relevant completion rules. For compactness, we use Table 2 to show the procedure steps.

The last row in the table shows the application of blocking. The fully expanded tree is shown in Figure 1. The notation in this figure is the same as in Table 2, where unlabeled dashed arrows connect an individual to its alternative branches when applying the concept disjunction rule. This illustrates how Table 2 leads to $A_4^\varepsilon(c) = A_2^\varepsilon(b)$, $A_5^\varepsilon(1)(c) = A_3^\varepsilon(1)(b)$, $A_5^\varepsilon(2)(c) = A_3^\varepsilon(2)(b)$, and $\mathcal{C}_9(C) = \mathcal{C}_6(b)$. Therefore, $\mathcal{L}(c) \subseteq \mathcal{L}(b)$, and the individual c is blocked by its ancestor b . We then stop applying the role existential restriction rule to A_5^ε and $\mathcal{C}_9$ at this point.

Table 2. Reasoning Procedure Applied to Example 1

$\mathcal{C}_1 = \{x_{CP(P002)} = [0, 0.5], x_{\neg CP(P002)} = [0.5, 1]\}$	The concept assertion rule
$A_1^\varepsilon(1) = A_0^\varepsilon \cup \{\neg CP(P002) \ x_{\neg CP(P002)}\}, A_1^\varepsilon(2) = A_0^\varepsilon \cup \{\exists hP.CP(P002) \ x_{\exists hP.CP(P002)}\}$	The concept disjunction rule
$\mathcal{C}_2 = \mathcal{C}_1 \cup \{\max(x_{\neg CP(P002)}, x_{\exists hP.CP(P002)}) = [0.5, 1]\}$	
$\mathcal{C}_3 = \mathcal{C}_2 \cup \{x_{\neg CP(P002)} = 1 - x_{CP(P002)}\}$	The concept negation rule
$A_2^\varepsilon = A_1^\varepsilon(2) \cup \{hP(P002, b) \ x_{hP(P002, b)}, CP(b) \ x_{CP(b)}, (\neg CP \sqcup \exists hP.CP)(b) \ [0.5, 1]\}$	The exists restriction rule
$\mathcal{C}_4 = \mathcal{C}_3 \cup \{\min(x_{CP(b)}, x_{hP(P002, b)}) = x_{\exists hP.CP(P002)}\}$	
$A_3^\varepsilon(1) = A_2^\varepsilon \cup \{\neg CP(b) \ x_{\neg CP(b)}\}, A_3^\varepsilon(2) = A_2^\varepsilon \cup \{\exists hP.CP(b) \ x_{\exists hP.CP(b)}\}$	
$\mathcal{C}_5 = \mathcal{C}_4 \cup \{\max(x_{\neg CP(b)}, x_{\exists hP.CP(b)}) = [0.5, 1]\}$	The concept disjunction rule
$\mathcal{C}_6 = \mathcal{C}_5 \cup \{x_{\neg CP(b)} = 1 - x_{CP(b)}\}$	The concept negation rule
$A_4^\varepsilon = A_3^\varepsilon(2) \cup \{CP(c) \ x_{CP(c)}, hP(b, c) \ x_{hP(b, c)}, (\neg CP \sqcup \exists hP.CP)(c) \ [0.5, 1]\}$	The exists restriction rule
$\mathcal{C}_7 = \mathcal{C}_6 \cup \{\min(x_{CP(c)}, x_{hP(b, c)}) = x_{\exists hP.CP(b)}\}$	
$A_5^\varepsilon(1) = A_4^\varepsilon \cup \{\neg CP(c) \ x_{\neg CP(c)}\}, A_5^\varepsilon(2) = A_4^\varepsilon \cup \{\exists hP.CP(c) \ x_{\exists hP.CP(c)}\}$	
$\mathcal{C}_8 = \mathcal{C}_7 \cup \{\max(x_{\neg CP(c)}, x_{\exists hP.CP(c)}) = [0.5, 1]\}$	The concept disjunction rule
$\mathcal{C}_9 = \mathcal{C}_8 \cup \{x_{\neg CP(c)} = 1 - x_{CP(c)}\}$	The concept negation rule
$\{\mathcal{L}(c) = \langle A_5^\varepsilon(c), \mathcal{C}_9(c) \rangle\} \subseteq \{\mathcal{L}(b) = \langle A_5^\varepsilon(b), \mathcal{C}_9(b) \rangle\}$	Blocking

Next, we rewrite the constraint set into the following form:

$$\text{subject to } \begin{cases} 0.5 \leq x_{CP(P002)} \leq 1 \\ 0 \leq x_{\neg CP(P002)} \leq 0.5 \\ 0.5 \leq \max(x_{\neg CP(P002)}, x_{\exists hP.CP(P002)}) \leq 1 \\ x_{\neg CP(P002)} = 1 - x_{CP(P002)} \\ \min(x_{hP(P002, b)}, x_{CP(b)}) = x_{\exists hP.CP(P002)} \\ 0.5 \leq \max(x_{\neg CP(b)}, x_{\exists hP.CP(b)}) \leq 1 \\ x_{\neg CP(b)} = 1 - x_{CP(b)} \\ \min(x_{hP(b, c)}, x_{CP(c)}) = x_{\exists hP.CP(b)} \end{cases}$$

Using a linear/non-linear programming solver, e.g., the GLPK solver in our experiments (<http://www.gnu.org/software/glpk/>), it is easy to show that the constraint set is solvable. Thus, the interpretation includes $\Delta^I = \{P002, b\}$ united with $\{CP^I(P002) [0, 0.5], CP^I(b) [0, 1], hP^I(P002, b) [0, 1]\}$.

Figure 2 shows the model constructed for Example 1. The dotted arrow here indicates the blocking.

Therefore, we can draw the conclusion that the knowledge base is consistent.

Example 2. To explain the at-most role restriction rule and inverse role rule, let us consider the following simple fuzzy knowledge base:

$$\{(\leq 1R)(a) [0.7, 1], R(a, b_1) [0.8, 1], R^-(b_2, a) [0.7, 1]\}$$

By applying the inverse role rule, we add $R(a, b_2) [0.7, 1]$ to the extended ABox. By applying the at-most role restriction rule, we have

Fig. 1. Expanded Tree for Example 1

Fig. 2. A Model of Example 1

$$\begin{aligned}
 \mathcal{A}_1^\varepsilon &= \{(\leq 1R)(a) [0.7, 1], R(a, b_1) [0.8, 1], R^-(b_1, a) [0.7, 1], R(a, b_1) \ x_{R(a, b_1)}\} \\
 \mathcal{C}_1 &= \{x_{(\leq 1R)(a)} = [0.7, 1], x_{R(a, b_1)} = [0.8, 1], x_{R^-(b_1, a)} = [0.7, 1], x_{R(a, b_1)} = [0.7, 1]\}
 \end{aligned}$$

Using the GLPK solver, it can be shown that this constraint set is solvable. Therefore, we can draw the conclusion that the knowledge base is consistent.

But if we had an extra assertion $b_1 \neq b_2$ in the knowledge base, then the at-most role restriction rule would not be applicable, and the knowledge base would be inconsistent because it contained a clash.

5 Conclusion and Future Work

In this paper we propose an extension to Description Logics based on Fuzzy Set Theory and Fuzzy Logic. The syntax and semantics of the proposed norm-parameterized Fuzzy Description Logic $fALCHIN$ are explained in detail. We then address the reasoning task of consistency checking on $fALCHIN$ knowledge bases. A reasoning procedure that always terminates and its completion rules are presented. As the examples suggest, the computational complexity of the reasoning procedure is usually high, which surfaces even for a relatively small knowledge base. Therefore, one of the main practical directions for future work is to investigate the core of the reasoning procedure and to develop strategies for reducing the computational complexity. Description Logics constitute a family of knowledge formalisms with different expressiveness. Our fuzzy Description Logic $fALCHIN$ extends fuzzy ALC taking into account inverse roles, role inclusion axioms, and number restrictions. For reasons of simplicity, it does not yet include $SHIN$ -style transitive roles, nor nominals (i.e. collections of individuals) or datatypes. Future work will consider fuzzy extensions to these more expressive description languages.

References

1. BAADER, F., CALVANESE, D., MCGUINNESS, D., NARDI, D., AND PATEL-SCHNEIDER, P. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, Cambridge, MA, 2003.
2. BOBILLO, F., AND STRACCIA, U. A fuzzy description logic with product t-norm. In *Proceedings of the IEEE International Conference on Fuzzy Systems (Fuzz IEEE-07)* (2007), IEEE Computer Society, pp. 652C–657.
3. HÁJEK, P. *Metamathematics of fuzzy logic*. Kluwer, 1998.
4. HÁJEK, P. Making fuzzy description logics more expressive. *Fuzzy Sets Syst.* 154, 1 (2005), 1–15.
5. NETO, A. G. S. S. Tableau algorithm for alchin consistency checking.
6. STOILLOS, G., STAMOU, G., PAN, J., TZOUVARAS, V., AND HORROCKS, I. Reasoning with very expressive fuzzy description logics. *Journal of Artificial Intelligence Research* 30 (2007), 273–320.
7. STRACCIA, U. A fuzzy description logic. In *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI’98)* (1998), pp. 594–599.
8. STRACCIA, U. Towards a fuzzy description logic for the semantic web (preliminary report). In *2nd European Semantic Web Conference (ESWC-05)* (2005), Lecture Notes in Computer Science, Springer Verlag, pp. 167–181.
9. ZADEH, L. A. Fuzzy sets. *Information and Control* 8, 3 (1965), 338–353.
10. ZHAO, J., BOLEY, H., AND DU, W. Expressing Vague Knowledge in the Fuzzy Description Logic $fALCHIN$. In *Proceedings of the Sixth Annual Research Exposition*, <http://www.cs.unb.ca/itc/ResearchExpo/Expo2009-proceeding.pdf> (2009), pp. 72–81.