REASONING WITH THE FUZZY DESCRIPTION LOGIC $f_Z\mathcal{SI}$

Jidi Zhao[1], Harold Boley[2] and Weichang Du[1]
[1]*Faculty of Computer Science, University of New Brunswick, 540 Windsor Street, Fredericton, Canada*
[2]*Institute for Information Technology, National Research Council of Canada, 46 Dineen Drive, Fredericton, Canada*

Keywords: Semantic web, Uncertain knowledge, Description logic, Fuzzy logic, Linear programming.

Abstract: While applications in different areas have shown the necessity of dealing with uncertain knowledge, Semantic Web techniques based on standard Description Logics do not have such a capability. Motivated by this discrepancy, we introduce an expressive fuzzy description logic, $f_Z\mathcal{SI}$, which extends the classic Description Logic $\mathcal{SI}$ to deal with uncertain knowledge about concepts and roles as well as instances of concepts and roles. In the family of Fuzzy Logics it is semantically based on Zadeh Logic, which naturally interprets uncertain knowledge about concepts and roles as fuzzy sets and fuzzy relations, and interprets uncertain knowledge about instances as elements with degrees of membership. The paper focuses on several reasoning methods for the main reasoning problems in $f_Z\mathcal{SI}$, including consistency checking, instance range entailment, and f-retrieval problems.

1 INTRODUCTION

The Semantic Web initiative aims at creating an extension to the current World Wide Web by developing logic-based standards and technologies that enable machines to understand the information on the Web, so that they can support richer knowledge inference and automate the performance of various tasks for human beings (Berners-Lee et al., 2001).

The current W3C standard for Semantic Web ontology languages, Web Ontology Language (OWL), is designed for use by applications that need to process the content of information instead of just presenting information to humans (McGuinness and van Harmelen, 2004). It facilitates greater machine interpretability of Web content than that supported by other Web languages such as XML, RDF, and RDF Schema (RDFS). This ability of OWL is enabled by its underlying knowledge representation formalism of Description Logics (DLs). DLs (Baader et al., 2003; Horrocks and Sattler, 1999) 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. 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 axioms allowed in their terminologies. The basic propositionally closed DL is $\mathcal{ALC}$ in which the letters $\mathcal{AL}$ stand for attributive language and the letter $\mathcal{C}$ for complement (negation of arbitrary concepts). Besides $\mathcal{ALC}$, other letters are used to indicate various DL extensions. For example, in the Description Logic $\mathcal{SI}$ (Horrocks and Sattler, 1999), $\mathcal{S}$ is used for $\mathcal{ALC}$ extended with transitive roles (R^+), and I for inverse roles. DLs have a model-theoretic semantics, which is defined by interpreting concepts as sets of individuals and roles as sets of pairs of individuals. An interpretation I is a pair $I = (\Delta^I, \cdot^I)$ consisting of a domain Δ^I which is a non empty set and of an interpretation function $\cdot^I$ which maps each individual x into an element of Δ^I ($x \in \Delta^I$), each concept C into a subset of Δ^I ($C^I \subseteq \Delta^I$) and each role R into a subset of $\Delta^I \times \Delta^I$ ($R^I \subseteq \Delta^I \times \Delta^I$). The semantics of complex concept and role descriptions can be found in (Baader et al., 2003; Horrocks and Sattler, 1999). Furthermore, a knowledge base (KB) in DLs consists of two parts: the terminological box (TBox T) and the assertional box (ABox A).

Uncertainty is an intrinsic feature of real-world knowledge, which is also reflected in the World Wide Web and the Semantic Web. Many concepts needed in knowledge modeling lack well-defined boundaries or, precisely defined criteria. Examples are the concepts of young, tall, and cold. The *Uncertainty Reason-*

Zhao J., Boley H. and Du W..
REASONING WITH THE FUZZY DESCRIPTION LOGIC fZSI.
DOI: 10.5220/0003054700210030
In *Proceedings of the International Conference on Fuzzy Computation and 2nd International Conference on Neural Computation* (ICFC-2010), pages 21-30
ISBN: 978-989-8425-32-4

ing for the World Wide Web (URW3) Incubator Group defined the challenge of representing and reasoning with uncertain information on the Web. According to the latest URW3 draft report, uncertainty is a term intended to encompass different forms of uncertain knowledge, including incompleteness, inconclusiveness, vagueness, ambiguity, and others (Laskey et al., 2008). The need to model and reason with uncertainty has been found in many different Semantic Web contexts, such as matchmaking in Web services (Martin-Recuerda and Robertson, 2005), classification of genes in bioinformatics (Stevens et al., 2007), multimedia annotation (Stamou et al., 2006), and ontology learning (Haase and Völker, 2005).

Fuzzy Set Theory was first introduced by Zadeh (Zadeh, 1965) as an extension to the classic notion of a set to capture 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 classic predicate logic. Formally, a fuzzy set A with respect to a set of elements Ω (also called a universe) is characterized by a membership function $\mu_A(x)$ which assigns a value in the real unit interval [0,1] to each element x in Ω ($x \in \Omega$), notated as $\mu_A : \Omega \rightarrow [0,1]$. $\mu_A(x)$, often written as $A(x)$, gives the degree of an element x belonging to the set A. Such degrees can be computed based on a membership function. A fuzzy relation R over two fuzzy sets A and B is similarly defined by a function $R : \Omega \times \Omega \rightarrow [0,1]$.

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

To deal with the 'crisp limitation' of classic DLs, considerable work has been carried out on integrating uncertain knowledge into DLs in recent years. The current literature generally follows two approaches. One is Probabilistic Logic based on Probability Theory; for example the work in (Jaeger, 1994; Koller et al., 1997; Lukasiewicz, 2008). The other is Fuzzy Logic and Fuzzy Sets; for example the work in (Yen, 1991; Straccia, 2001; Zhao and Boley, 2010). A review and comparison of these works can be found in (Zhao, 2010). We presented a Norm-Parameterized Fuzzy Description Logic $f\mathcal{ALCN}$ and addressed the consistency checking problem in (Zhao and Boley, 2010). We use $f_{\mathfrak{N}}\mathfrak{DL}$ to denote a Fuzzy Description Logic $f\mathfrak{DL}$ with norm parameter $\mathfrak{N}$. Omitting the index $\mathfrak{N}$ means the $f\mathfrak{DL}$ is norm-parameterized. In the current paper, we follow the Fuzzy Sets and Fuzzy Logic approach and present the fuzzy Description Logic $f_Z\mathcal{SI}$. We call this fuzzy Description Logic $f_Z\mathcal{SI}$ as $\mathcal{SI}$ is the underlying Description Logic and Z fixes the norms to Zadeh Logic. This paper is different from previous work due to the following features. First, the underlying classic DL $\mathcal{SI}$ is a more expressive Description Logic which deals with fuzzy transitive roles and fuzzy inverse roles. Second, we combine Description Logic, Fuzzy Logic, and Linear Programming methods in the reasoning procedure. Last but not least, $f_Z\mathcal{SI}$ supports both fuzzy axioms and fuzzy assertions for uncertain knowledge representation and reasoning.

2 THE FUZZY DL $f_Z\mathcal{SI}$

$f_Z\mathcal{SI}$ extends the $f_Z\mathcal{ALCN}$ DL with inverse roles, and transitive roles but excludes number restrictions. Due to space limitations, we refer interested readers to (Zhao and Boley, 2010) for the syntax and semantics of complex concept descriptions as well as axioms and assertions for $f_Z\mathcal{ALC}$ by specializing the t-norm to min and the s-norm to max. Here we simply list them in Tables 1 and 2, and then explain the expressiveness beyond $f_Z\mathcal{ALC}$. A fuzzy knowledge base in $f_Z\mathcal{SI}$ consists of two parts: the fuzzy terminological box consisting of a finite set of fuzzy axioms (TBox $\mathcal{T}$) and the fuzzy assertional box consisting of a finite set of fuzzy assertions (ABox $\mathcal{A}$). As shown in Table 2, a fuzzy axiom or fuzzy assertion is of the form $\alpha\ [l,u]$ with $0 \le l \le u \le 1$, which is equivalent to the two inequalities $\alpha \ge l$ and $\alpha \le u$. In what follows, we use these expressions as needed.

In classic DLs, a role R is symmetric iff for all $x, y \in \triangle^I$, $(Inv(R))^I(y,x) = R^I(x,y)$, where the role function $Inv(R)$ defines the inverse of a role. The same property holds for a fuzzy symmetric role. For example, the role *hasPart* is the inverse of the role *isPartOf*.

In classic DLs, a role R is transitive iff for all $x, y, z \in \triangle^I$, $R^I(x,y)$ and $R^I(y,z)$ imply $R^I(x,z)$. While in Fuzzy Logic, a fuzzy role R is transitive iff for all

Table 1: Syntax and semantics of $f_Z\mathcal{SI}$ constructors.

Constructor	Syntax	Semantics
top concept	$\top$	$\top^I = 1$
bottom concept	$\bot$	$\bot^I = 0$
atomic negation	$\neg A$	$(\neg A)^I(x) = 1 - A^I(x)$
concept conjunction	$C \sqcap D$	$(C \sqcap D)^I = \min(C^I(x), D^I(x))$
concept disjunction	$C \sqcup D$	$(C \sqcup D)^I = \max(C^I(x), D^I(x))$
exists restriction	$\exists R.C$	$(\exists R.C)^I(x) = \sup_{y\in\Delta^I}\{\min(R^I(x,y), C^I(y))\}$
value restriction	$\forall R.C$	$(\forall R.C)^I(x) = \inf_{y\in\Delta^I}\{\max(1 - R^I(x,y), C^I(y))\}$
inverse role	$Inv(R)$	$(Inv(R))^I(y,x) = R^I(x,y)$

Table 2: Syntax and semantics of $f_Z\mathcal{SI}$ axioms.

Axioms	Syntax	Semantics
concept inclusion	$A \sqsubseteq C$	$\forall x \in \Delta^I, A^I(x) \leq C^I(x)$
concept definition	$A \equiv C$	$\forall x \in \Delta^I, A^I(x) = C^I(x)$
concept implication	$A \rightarrow C\ [l,u]$	$\forall x \in \Delta^I, C^I(x) \in \min(A^I(x), [l,u])$
transitive role	$Trans(R)$	$R^I(a,c) \geq \sup_{b\in\triangle^I}\min(R^I(a,b), R^I(b,c))$
concept assertion	$C(a)\ [l,u]$	$l \leq C^I(a) \leq u$
role assertion	$R(a,b)\ [l,u]$	$l \leq R^I(a,b) \leq u$
individual inequality	$a \neq b$	$a^I \neq b^I$

$x,y,z \in \triangle^I$, it satisfies the following inequality (Díaz et al., 2010):

$$R^I(x,z) \geq \sup_{y\in\triangle^I} t(R^I(x,y), R^I(y,z)) \tag{1}$$

where $t(\eta,\theta)$ denotes a general t-norm. Thus, in the case of Zadeh Logic, a transitive role satisfies:

$$R^I(x,z) \geq \sup_{y\in\triangle^I} \min(R^I(x,y), R^I(y,z)) \tag{2}$$

In order to make the following explanations easier, we introduce the role function $Trans(R)$ which specifies that R is transitive or $Inv(R)$ is transitive.

Now, we use some mathematical properties of Zadeh Logic to show that the following property is satisfied by a role value restriction $\forall R.C$ with $Trans(R)$.

Lemma 1. *Under Zadeh Logic, if* $(\forall R.C)^I(x) \geq l$ *(*$l \in [0,1]$*) and* R *is transitive, then* $(\forall R.(\forall R.C))^I(x) \geq l$ *holds.*

Proof. $(\forall R.C)^I(x) \geq l$

$\xrightarrow{Definition\ of\ semantics}$ $\inf_{z\in\Delta^I}\{\max(\neg R^I(x,z), C^I(z))\} \geq l$

$\xrightarrow{Equation\ 1}$ $\inf_{z\in\Delta^I}\inf_{y\in\Delta^I}\{\max(\neg(\min(R^I(x,y), R^I(y,z))), C^I(z))\} \geq l$

$\xrightarrow{De\ Morgan's\ Law}$ $\inf_{z\in\Delta^I}\inf_{y\in\Delta^I}\{\max(\max(\neg R^I(x,y), \neg R^I(y,z)), C^I(z))\} \geq l$

$\xrightarrow{Associativity}$ $\inf_{z\in\Delta^I}\inf_{y\in\Delta^I}\{\max(\neg R^I(x,y), \max(\neg R^I(y,z), C^I(z)))\} \geq l$

$\xrightarrow{Commutativity}$ $\inf_{y\in\Delta^I}\{\max(\neg R^I(x,y), \inf_{z\in\Delta^I}\max(\neg R^I(y,z), C^I(z)))\} \geq l$

$\xrightarrow{Definition\ of\ semantics}$ $\inf_{y\in\Delta^I}\{\max(\neg R^I(x,y), (\forall R.C)^I(y))\} \geq l$

$\xrightarrow{Definition\ of\ semantics}$ $(\forall R.(\forall R.C))^I(x) \geq l$

□

However, in the cases of $\leq$, we cannot derive such a property for $(\forall R.C)(x)$ and $Trans(R)$.

Under Zadeh Logic, by applying the semantics of $\exists R.C$ and negation, it is easy to see that the following equivalence rules hold:

$\forall a,b \in \Delta^I$,

$$\neg\neg C \equiv C\ , \tag{3}$$

$$\neg\exists R.C \equiv \forall R.\neg C\ , \tag{4}$$

$$\neg\forall R.C \equiv \exists R.\neg C \tag{5}$$

Then, $(\exists R.C)^I(x) \leq u$

$\xrightarrow{Monotonicity}$ $\neg((\exists R.C)^I(x)) \geq 1-u$

$\xrightarrow{Equilvalence\ 4}$ $(\forall R.(\neg C))^I(x)) \geq 1-u$

$\xrightarrow{Lemma\ 1}$ $(\forall R.(\forall R.(\neg C)))^I(x)) \geq 1-u$

$\xrightarrow{Monotonicity}$ $\neg(\forall R.(\forall R.(\neg C))^I(x)) \leq u$

$\xrightarrow{Equilvalence\ 5}$ $(\exists R.\neg(\forall R.(\neg C)))^I(x) \leq u$

$\xrightarrow{Equilvalence\ 5\ and\ 3}$ $(\exists R.(\exists R.C))^I(x) \leq u$

Therefore, the following property is satisfied with respect to a role exists restriction $\exists R.C$ and $Trans(R)$. Such a property cannot be inferred from the cases of $\geq$.

Lemma 2. *Under Zadeh Logic, if* $(\exists R.C)^I(x) \leq u$ *and* R *is transitive, then* $(\exists R.(\exists R.C))^I(x) \leq u$ *holds.*

Although we can show that such properties also hold under Product Logic and other logics, we neglect it here, as it is out of scope. We will soon see that these properties will be embodied in the fuzzy completion rules for the $f_Z\mathcal{SI}$ reasoning algorithm.

3 REASONING ALGORITHM FOR BUILDING A FUZZY TABLEAU OF $f_Z\mathcal{SI}$

The reasoning algorithm that we will present is a fuzzy extension to the tableau method and tests the consistency of a knowledge base $KB = <\mathcal{T}, \mathcal{A}>$ by trying to construct a model of *KB*. A model of *KB* in our Fuzzy Description Logic $f_Z\mathcal{SI}$ is a fuzzy interpretation $I = (\triangle^I, \cdot^I)$. Similar to the classic DL, such a model has the shape of a forest, i.e., a collection of trees, with nodes corresponding to individuals, root nodes corresponding to named individuals, and edges corresponding to roles between individuals. Each node has a node label $\mathcal{L}$(individual), but different from classic DLs, each node in a $f_Z\mathcal{SI}$ tableau is labeled with a set of $f_Z\mathcal{SI}$-concepts. Each element in the set consists of a pair of elements $\{concept, constraint\}$. The sets for all nodes are restricted to subsets of $sub(\mathcal{A})$, where $sub(\mathcal{A})$ is the set of sub-concepts of concepts that appear within an ABox $\mathcal{A}$. Furthermore, each edge is associated with an edge label $\mathcal{L}(\text{individual}_1, \text{individual}_2)$ which consists of a pair of elements $\{role, constraint\}$.

In (Zhao and Boley, 2010), we explained the TBox processing procedure which consists of some preprocessing steps to deal with the fuzzy TBox before applying the reasoning algorithm. Those steps are applicable to the $f_Z\mathcal{SI}$ knowledge base. Therefore, we can assume all concepts C occurring in *KB* to be in *negation normal form* (NNF) and we only deal with unfoldable TBox after those preprocessing steps. However, due to the properties of a $f_Z\mathcal{SI}$ knowledge base, the TBox processing procedure should include a couple of other steps. First, the TBox processing procedure transforms all the assertions in the fuzzy ABox and the fuzzy implication axioms in the fuzzy TBox with the form $\alpha_0\ [l,u]$ into two expressions: $\alpha_0 \geq l$ and $\alpha_0 \leq u$. In order to keep our presentation simple and compact, in what follows, we use a general form $\alpha\ op\ n$ where $op \in \{\geq, \leq\}$ and $n \in [0,1]$ whenever applicable. Second, an $f_Z\mathcal{SI}$ knowledge base may contain transitive role axioms and inverse roles. We know that if a role R is transitive, the inverse role of R is also transitive. Therefore, for each pair of $Trans(R)$ and $Inv(R)$, the procedure should also add an axiom $Trans(Inv(R))$. After the application of the TBox processing procedure, in what follows, we only have to consider a knowledge base in $f_Z\mathcal{SI}$ only consists of fuzzy ABox assertions, a set of transitive role axioms, and a finite set of fuzzy implication axioms.

Next, we first present the definitions of fuzzy tableau, clash, and clash-free, and then prove the relation between the consistency of a fuzzy knowledge base $KB = <\mathcal{T}, \mathcal{A}>$ and the existence of a fuzzy tableau $\mathfrak{T}$ for *KB*.

Definition 1. *If $KB = <\mathcal{T}, \mathcal{A}>$ is an $f_Z\mathcal{SI}$ knowledge base, $\mathbf{R}_\mathcal{A}$ is the set of roles occurring in $\mathcal{A}$, together with their $Inv(R)$s, a fuzzy tableau $\mathfrak{T}$ for KB is defined to be a quadruple $(S, \mathcal{L}, \varepsilon, \mathcal{C})$ such that: S is a set of individuals, $\mathcal{L} : S \times sub(\mathcal{A}) \rightarrow [0,1]$ maps each individual and a concept which is a subset of $sub(\mathcal{A})$ to the membership degree of the individual to that concept, $\varepsilon : \mathbf{R}_\mathcal{A} \times S \times S \rightarrow [0,1]$ maps each role in $R_\mathcal{A}$ and a pair of individuals to the membership degree of the pair to the role, and $\mathcal{C}$ is a set of constraints must be satisfied. For all $x, y \in S$, $A, C, D \in sub(\mathcal{A})$, $R \in \mathbf{R}_\mathcal{A}$ and $n \in [0,1]$, it holds that:*

1. *For any $x \in S$, $\{x : \bot = 0\}$ and $\{x : \top = 1\} \in \mathcal{L}(x)$.*
2. *If $\{x : \neg(A)\ \text{op}\ n\} \in \mathcal{L}(x)$, then $\{x : A\ \text{op}\ 1-n\} \in \mathcal{L}(x)$.*
3. *If $\{x : C \sqcap D \geq n\} \in \mathcal{L}(x)$, then $\{x : C \geq n\} \in \mathcal{L}(x)$ and $\{x : D \geq n\} \in \mathcal{L}(x)$.*
4. *If $\{x : C \sqcup D \leq n\} \in \mathcal{L}(x)$, then $\{x : C \leq n\} \in \mathcal{L}(x)$ and $\{x : D \leq n\} \in \mathcal{L}(x)$.*
5. *If $\{x : C \sqcap D \leq n\} \in \mathcal{L}(x)$, then $\{x : C \leq n_1\} \in \mathcal{L}(x)$, $\{x : D \leq n_2\} \in \mathcal{L}(x)$, and $n = \min(n_1, n_2)$ for some n_1, n_2.*
6. *If $\{x : C \sqcup D \geq n\} \in \mathcal{L}(x)$, then $\{x : C \geq n_1\} \in \mathcal{L}(x)$, $\{x : D \geq n_2\} \in \mathcal{L}(x)$, and $n = \max(n_1, n_2)$ for some n_1, n_2.*
7. *If $\{x : \exists R.C \geq n\} \in \mathcal{L}(x)$, then there exists $y \in S$ such that $\{<x,y>: R \geq n\} \in \varepsilon(R)$ and $\{y : C \geq n\} \in \mathcal{L}(y)$.*
8. *If $\{x : \forall R.C \leq n\} \in \mathcal{L}(x)$, then there exists $y \in S$ such that $\{<x,y>: R \geq 1-n\} \in \varepsilon(R)$ and $\{y : C \leq n\} \in \mathcal{L}(y)$.*
9. *If $\{x : \exists R.C \leq n\} \in \mathcal{L}(x)$, then $\{<x,y>: R \leq n_1\} \in \varepsilon(R)$, $\{y : C \leq n_2\} \in \mathcal{L}(y)$, and $n = \min(n_1, n_2)$ for some n_1, n_2.*
10. *If $\{x : \forall R.C \geq n\} \in \mathcal{L}(x)$, then $\{<x,y>: R \leq 1-n_1\} \in \varepsilon(R)$, $\{y : C \geq n_2\} \in \mathcal{L}(y)$, and $n = \max(1-n_1, n_2)$ for some n_1, n_2.*
11. *$\{<x,y>: R\ \text{op}\ n\} \in \varepsilon(R)$ iff $\{<y,x>: Inv(R)\ \text{op}\ n\} \in \varepsilon(R)$.*
12. *If $\{x : \forall R.C \geq n\} \in \mathcal{L}(x)$ and $Trans(R)$, then $\{<x,y>: R \leq 1-n_1\} \in \varepsilon(R)$, $\{y : \forall R.C \geq n_2\} \in \mathcal{L}(y)$, and $n = \max(1-n_1, n_2)$ for some n_1, n_2.*
13. *If $\{x : \exists R.C \leq n\} \in \mathcal{L}(x)$ and $Trans(R)$, then $\{<x,y>: R \leq n_1\} \in \varepsilon(R)$, $\{y : \exists R.C \leq n_2\} \in \mathcal{L}(y)$, and $n = \min(n_1, n_2)$ for some n_1, n_2.*
14. *If $\{A \rightarrow C \geq n\} \in \mathcal{T}$ and $\{x : A \geq n_1\} \in \mathcal{L}(x)$, then $\{x : C \geq n_2\} \in \mathcal{L}(x)$ and $n_2 = \min(n, n_1)$, for any $x \in S$.*

15. *If $\{A \to C \leq n\} \in \mathcal{T}$ and $\{x : A \leq n_1\} \in \mathcal{L}(x)$, then $\{x : C \leq n_2\} \in \mathcal{L}(x)$ and $n_2 = \min(n, n_1)$, for any $x \in S$.*

In (Zhao and Boley, 2010), we defined the semantics $(C \sqcap D)^I$ as $t(C^I(x), D^I(x))$ for various t-norms. For the case of Zadeh Logic, we have that if $(C \sqcap D)^I(x) \geq n$, then $C^I(x) = n_C$, $D^I(x) = n_D$, and $min(n_C, n_D) \geq n$. In this definition, we can draw a further conclusion based on the properties of the min norm that $C^I(x) = n_C \geq n$ and $D^I(x) = n_D \geq n$. Similar extensions are conducted on other $f_Z\mathcal{SI}$ concepts and roles.

Definition 2. *Let $\mathcal{A}$ be an extended $f_Z\mathcal{SI}$ ABox, $\mathcal{A}$ contains a clash if only if one of the following situations occurs:*

1. $\{\perp(a) \neq 0\} \subseteq \mathcal{A}$
2. $\{\top(a) \neq 1\} \subseteq \mathcal{A}$
3. $\{\alpha \leq n_1, \alpha \geq n_2\} \subseteq \mathcal{A}$ *and* $n_1 < n_2$
4. *there is no solution for the constraint system of inequations $\mathcal{C}$*

$\mathcal{A}$ is called clash-free if it does not contain any clash.

Lemma 3. *An $f_Z\mathcal{SI}$ knowledge base $KB =< \mathcal{T}, \mathcal{A} >$ is consistent iff there exists a clash-free fuzzy tableau for KB.*

Proof. For the if direction, if $\mathfrak{T}$=$(S, \mathcal{L}, \varepsilon)$ is a clash-free fuzzy tableau for a fuzzy knowledge base KB, a fuzzy interpretation I=$(\Delta^I, \cdot^I)$ can be constructed as:

$\Delta^I = S$

$\top^I$=$\{x : \top = 1\} \in \mathcal{L}(x)$ for any x in S

$\perp^I$=$\{x : \perp = 0\} \in \mathcal{L}(x)$ for any x in S

A^I=$\{x : A\ op\ n\} \in \mathcal{L}(x)$ for all concept names A in $sub(\mathcal{A})$

$$R^I = \begin{cases} \varepsilon(R)^+ & if\ Trans(R) \\ \varepsilon(R) & otherwise \end{cases}$$

where $\varepsilon(R)^+$ denotes the fuzzy sup-min transitive closure of $\varepsilon(R)$ (Lee, 2001; Mitsuishi and Bancerek, 2003).

To prove that I is a model of KB, we show by induction on the structure of concepts that, if $\{x : E\ op\ n\} \in \mathcal{L}(x)$, then $E^I(x)\ op\ n$ for any x in S. Without loss of generality, we only show in the following the cases with $\{x : E \geq n\} \in \mathcal{L}(x)$.

1. If E the $\top$ or $\perp$ concept, and $\{x : \perp = 0\}$ and $\{x : \top = 1\} \in \mathcal{L}(x)$, then by definition, $\top^I(x) = 1$ or $\perp^I(x) = 0$.
2. If E is a concept name other than $\top$ and $\perp$, and $\{x : E \geq n\} \in \mathcal{L}(x)$, then $E^I(x) \geq n$ by definition.
3. If $E = \neg(C)$ and $\{x : \neg(C) \geq n\} \in \mathcal{L}(x)$, then $\{x : C \leq 1 - n\} \in \mathcal{L}(x)$ (due to Property 2 in Definition 1), so we have $C^I(x) \leq 1 - n$ by induction. Hence, $(\neg C)^I(x) \geq 1 - (1 - n) = n$, i.e., $E^I(x) \geq n$.
4. If $E = (C_1 \sqcap C_2)$ and $\{x : C_1 \sqcap C_2 \geq n\} \in \mathcal{L}(x)$, then $\{x : C_1 \geq n\} \in \mathcal{L}(x)$ and $\{x : C_2 \geq n\} \in \mathcal{L}(x)$, so by induction $(C_1)^I(x) \geq n$ and $(C_2)^I(x) \geq n$. Hence, $(C_1 \sqcap C_2)^I(x) = min((C_1)^I(x), (C_2)^I(x)) \geq n$.
5. If $E = (C_1 \sqcup C_2)$ and $\{x : C_1 \sqcup C_2 \geq n\} \in \mathcal{L}(x)$, since the tableau is clash free, we can find some n_1, n_2 so that $\{x : C_1 \geq n_1\} \in \mathcal{L}(x)$, $\{x : C_2 \geq n_2\} \in \mathcal{L}(x)$ and $n = \max(n_1, n_2)$. By induction $(C_1)^I(x) \geq n_1$, $(C_2)^I(x) \geq n_2$. Hence, $(C_1 \sqcap C_2)^I(x) = max((C_1)^I(x), (C_2)^I(x)) \geq n$.
6. If $E = (\exists S.C)$ and $\{x : \exists S.C \geq n\} \in \mathcal{L}(x)$, then there exists $y \in S$ such that $\{< x, y >: S \geq n\} \in \varepsilon(S)$ and $\{y : C \geq n\} \in \mathcal{L}(y)$, so by induction $S^I(x, y) \geq n$ and $C^I(y) \geq n$. Hence $(\exists S.C)^I(x) = sup_{y \in \Delta^I} min(S^I(x, y), C^I(y)) \geq n$.
7. If $E = (\forall S.C)$, $\{x : \forall S.C \geq n\} \in \mathcal{L}(x)$, and $S^I(x, y) = m$, then it would be either of the following two cases.
   - $\{< x, y >: S = m\} \in \varepsilon(S)$: if $m > 1 - n$, then $\{y : C \geq n\} \in \mathcal{L}(y)$ (due to Property 10 in Definition 1), so we have $S^I(x, y) > 1 - n$ and $C^I(y) \geq n$, hence, $(\forall S.C)^I(x) = inf_{y \in \Delta^I} max(1 - S^I(x, y), C^I(y)) \geq n$; if $m <= 1 - n$, then $1 - S^I(x, y) = 1 - m \geq n$, hence $(\forall S.C)^I(x) = inf_{y \in \Delta^I} max(1 - S^I(x, y), C^I(y)) \geq n$.
   - $\{< x, y >: S = m\} \notin \varepsilon(S)$ and there exist l paths $(l \geq 1)$ such that in each path, $\{< x, x_{l1} >: S = m_{l1}\} \in \varepsilon(S)$, $\{< x_{l1}, x_{l2} >: S = m_{l2}\} \in \varepsilon(S)$, $\cdots$, $\{< x_{ln}, y >: S = m_{l(n+1)}\} \in \varepsilon(S)$ and $Trans(R)$. Thus, the truth degree of $< x, y >$ to the transitive closure of S, m, would be equal to the supremum value among all the minimum values of each path. In this case: if $m > 1 - n$, then there exists at least one path k, $\{< x, x_{k1} >: S = m_{k1}\} \in \varepsilon(S)$, $\{< x_{k1}, x_{k2} >: S = m_{k2}\} \in \varepsilon(S)$, $\cdots$, $\{< x_{kn}, y >: S = m_{k(n+1)}\} \in \varepsilon(S)$, we have $m_{ki} > m > 1 - n$ $(1 \leq i \leq (n+1))$ (as m is the minimum value of the path), $\{x_{ki} : (\forall S.C) \geq n\} \in \mathcal{L}(x_{ki})$ $(1 \leq i \leq n)$, and $\{y : (\forall S.C) \geq n\} \in \mathcal{L}(y)$ (due to Property 12 in Definition 1), so, inducted from $\{< x, x_{k1} >: S > 1 - n\} \in \varepsilon(S)$ and $\{x_{k1} : (\forall S.C) \geq n\} \in \mathcal{L}(x_{k1})$, we have $S^I(x, x_{k1}) > 1 - n$ and $(\forall S.C)^I(x_{k1}) \geq n$, and thus $C^I(x_{k1}) \geq n$, hence $(\forall S.C)^I(x) = inf_{y \in \Delta^I} max(1 - S^I(x, x_{k1}), C^I(x_{k1})) \geq n$; if $m <= 1 - n$, then we have $max(1 - m, C^I(y)) \geq n$, hence $(\forall S.C)^I(x) \geq n$.

The cases for the $\leq$ inequalities can be proved in a similar way.

For the converse, if I=$(\Delta^I, \cdot^I)$ is a model of $\mathcal{A}$, then a fuzzy tableau $\mathfrak{T}$=$(S, \mathcal{L}, \varepsilon)$ can be defined as:

$S = \Delta^I$

$\varepsilon(R) = R^I$

$\mathcal{L}(x) = \{x : C\ op\ n\}$ for all $x \in S$ and $C \in sub(\mathcal{A})$

To prove that $\mathfrak{T}$ is a fuzzy tableau of KB, we show that, based on I, all the properties in Definition 1 are satisfied.

1. T satisfies Property 1 - 12, 14, and 15 as a direct consequence of the semantics of $f_Z\mathcal{SI}$ concepts. For example, let $\{x : C \sqcap D \geq n\} \in \mathcal{L}(x)$, the semantics of fuzzy concept conjunction implies that $(C \sqcap D)^I(x) = min(C^I(x), D^I(x)) \geq n$, thus we have $D^I(x) \geq n$ and $D^I(x) \geq n$, that is, $\{x : C \geq n\} \in \mathcal{L}(x)$ and $\{x : D \geq n\} \in \mathcal{L}(x)$, hence Property 3 is satisfied. For similar reasons, other properties hold.
2. Property 12 of Definition 1 is satisfied as a result of the semantics and the properties of transitive roles and value restrictions that have been investigated in Section 2. Hence, if $(\forall R.C)^I(x) \geq n$, $Trans(R)$ then $(\forall R.(\forall R.C))^I(x) \geq n$, thus $R^I(s,t) \leq 1 - n_1$, $(\forall R.C)^I(t) \geq n_2$ and $n = \max(1 - n_1, n_2)$ hold.
3. Similarly, Property 13 is satisfied as a result of the semantics and the properties of transitive roles and role exists restrictions.

□

From Lemma 3, an algorithm that constructs a fuzzy tableau for an $f_Z\mathcal{SI}$ knowledge base can be used as a decision procedure for the consistency checking problem.

Similar to the tableau algorithm presented by Horrocks et al. (Horrocks et al., 2000), our algorithm works on building a fuzzy tableau for an $f_Z\mathcal{SI}$ knowledge base which may be a completion-forest since the ABox might contain several named individuals with arbitrary edges connecting them. Each node x is labeled with a set $\mathcal{L}(x) = \{\{x : C_1\ op\ n_1\}, \cdots, \{x : C_m\ op\ n_m\}\}$ $(m \geq 1)$ and a constraint set $\mathcal{C}(x) = \{\{x_{C_1}\ op\ n_1\}, \cdot, \{x_{C_m}\ op\ n_m\}\}$, where $C_i \in sub(\mathcal{A})$, $x_{C_i}, n_i \in [0,1]$, $1 \leq i \leq m$, and $op \in \{\geq, \leq\}$. Each edge $< x, y >$ is labeled with a set $\mathcal{L}(x,y) = \{[x,y] : R\ op\ n\}$ and a constraint in the set $\mathcal{C}(x,y) = \{x_R\ op\ n\}$, where R are roles occurring in $\mathcal{A}$.

We adapt the conjugation concept in (Straccia, 2001) to represent pairs of fuzzy assertions that form a contradiction. Let α be a $\mathcal{SI}$ assertion, two fuzzy assertions ($\alpha \geq n_1$ and $\alpha \leq n_2$) conjugate with each other if $n_1 > n_2$. For a given fuzzy assertion, its conjugated assertion is not unique, and in fact, infinite. For example, both $\{[x,y] : R \leq 0.5\}$ and $\{[x,y] : R \leq 0.3\}$ conjugate with the fuzzy assertion $\{[x,y] : R \geq 0.6\}$.

Let us recall some notations used in (Horrocks and Sattler, 1999). If nodes x and y are connected by an edge $< x, y >$ with $\{R\ op\ n\} \in \mathcal{L}(x,y)$, then y is called an *$R_n$-successor* of x and x is called an *R_n-predecessor* of y. *Ancestor* is the transitive closure of *predecessor*. If y is an *R_n-successor* or an $(Inv(R))_n$-*predecessor* of x, then y is called an *R_n-neighbor* of x. An expressive DL such as $f_Z\mathcal{SI}$ which allows transitive roles and inverse roles may lead to nontermination as the fuzzy completion rules can introduce new concepts that are the same size as the decomposed concept. Our algorithm for the consistency checking of an $f_Z\mathcal{SI}$ knowledge base follows the *dynamic blocking* presented in (Horrocks and Sattler, 1999) and uses it to guarantee the termination of the reasoning algorithm. In *dynamic blocking*, blocked nodes are allowed to be dynamically established and broken as the expansion progresses, and expand role value restriction and role exists restriction concepts. This dynamic blocking strategy is crucial in the presence of inverse roles since information might be propagated up the completion-forest and affect other branches. For example, consider the nodes x, y and z, the edges $< x, y >$ and $< x, z >$. Suppose x blocks y. In the presence of inverse roles it is possible that z adds information to node x, although z is a successor of x. In that case the block on y must be broken. A node x is *blocked* if for some ancestor y, y is blocked or $\mathcal{L}(x) = \mathcal{L}(y)$. *Dynamic blocking* uses the notions of *directly blocked* and *indirectly blocked* nodes. If a blocked node x's predecessor is blocked, x is called *indirectly blocked*. A blocked node x is called *directly blocked* if it has a unique ancestor y such that $\mathcal{L}(x) = \mathcal{L}(y)$.

Now, for an expanded $f_Z\mathcal{SI}$ ABox $\mathcal{A}$ with a set of transitive role axioms and a set of fuzzy implication axioms, the algorithm initializes a forest to contain (1)root nodes, for each individual x occurring in $\mathcal{A}$, the root node x is labeled with $\mathcal{L}(x) = \{x : C\ op\ n\}$ and $\mathcal{C}(x) = \{x_C\ op\ n\}$ for each assertion of the form $C(x)\ op\ n$ in $\mathcal{A}$, and (2)edges, each edge $\langle x, y \rangle$ corresponds to an assertion $R(x,y)\ op\ n$ in $\mathcal{A}$ with R be an atomic role or an inverse role and is labeled with $\mathcal{L}(x,y) = \{[x,y] : R\ op\ n\}$ and $\mathcal{C}(x,y) = \{x_R\ op\ n\}$. If an assertion is of the form $Inv(P)(x,y)\ op\ n$, the corresponding edge is also labeled with $\mathcal{L}(x,y) = \{[y,x] : P\ op\ n\}$ and $\mathcal{C}(x,y) = \{x_P\ op\ n\}$. The completion forest is then expanded by repeatedly applying the following fuzzy completion rules in Table 3. The completion forest is complete when a clash is detected, or none of the fuzzy completion rules are applicable.

Table 3: Fuzzy Completion Rules for f_ZSI

f_ZSI **Fuzzy Completion Rules**
$\neg_{\geq}$**-rule** **Condition**: $\{x : (\neg A) \geq n\} \in \mathcal{L}(x)$ and $\{x : A \leq 1-n\} \notin \mathcal{L}(x)$ **Action**: $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{\{x : A \leq 1-n\}\}$ and $\mathcal{C}(x) \longrightarrow \mathcal{C}(x) \cup \{x_A \leq (1-n)\}$
$\neg_{\leq}$**-rule** **Condition**: $\{x : (\neg A) \leq n\} \in \mathcal{L}(x)$ and $\{x : A \geq 1-n\} \notin \mathcal{L}(x)$ **Action**: $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{\{x : A \geq 1-n\}\}$ and $\mathcal{C}(x) \longrightarrow \mathcal{C}(x) \cup \{x_A \geq (1-n)\}$
$\sqcap_{\geq}$**-rule** **Condition**: $\{x : (C_1 \sqcap C_2) \geq n\} \in \mathcal{L}(x)$, x is not indirectly blocked, and $\{\{x : C_1 \geq n\}, \{x : C_2 \geq n\}\} \nsubseteq \mathcal{L}(x)$ **Action**: $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{\{x : C_1 \geq n\}, \{x : C_2 \geq n\}\}$ and $\mathcal{C}(x) \longrightarrow \mathcal{C}(x) \cup \{x_{C_1} \geq n, x_{C_2} \geq n\}$
$\sqcap_{\leq}$**-rule** **Condition**: $\{x : (C_1 \sqcap C_2) \leq n\} \in \mathcal{L}(x)$, x is not indirectly blocked, and $\{\{x : C_1 \leq n\}, \{x : C_2 \leq n\}\} \cap \mathcal{L}(x) = \varnothing$ **Action**: $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{\{x : C_1 \leq x_1\}, \{x : C_2 \leq x_2\}\}$ and $\mathcal{C}(x) \longrightarrow \mathcal{C}(x) \cup \{x_{C_1} \leq x_1, x_{C_2} \leq x_2, x_1 + x_2 = 1 + n, x_1 \geq y, x_2 \geq 1 - y, y \in \{0,1\}, x_1 \in [0,1], x_2 \in [0,1]\}$
$\sqcup_{\geq}$**-rule** **Condition**: $\{x : (C_1 \sqcup C_2) \geq n\} \in \mathcal{L}(x)$, x is not indirectly blocked, and $\{\{x : C_1 \geq n\}, \{x : C_2 \geq n\}\} \cap \mathcal{L}(x) = \varnothing$ **Action**: $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{\{x : C_1 \geq x_1\}, \{x : C_2 \geq x_2\}\}$ and $\mathcal{C}(x) \longrightarrow \mathcal{C}(x) \cup \{x_{C_1} \geq x_1, x_{C_2} \geq x_2, x_1 + x_2 = n, x_1 \leq y, x_2 \leq 1 - y, y \in \{0,1\}, x_1 \in [0,1], x_2 \in [0,1]\}$
$\sqcup_{\leq}$**-rule** **Condition**: $\{x : (C_1 \sqcup C_2) \leq n\} \in \mathcal{L}(x)$, x is not indirectly blocked, and $\{\{x : C_1 \leq n\}, \{x : C_2 \leq n\}\} \nsubseteq \mathcal{L}(x)$ **Action**: $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{\{x : C_1 \leq n\}, \{x : C_2 \leq n\}\}$ and $\mathcal{C}(x) \longrightarrow \mathcal{C}(x) \cup \{x_{C_1} \leq n, x_{C_2} \leq n\}$
$\exists_{\geq}$**-rule** **Condition**: $\{x : (\exists R.C) \geq n\} \in \mathcal{L}(x)$, x is not blocked, and x has no R_n-neighbor y **Action**: create a new node y with $\mathcal{L}(x,y) = \{\{[x,y] : R \geq n\}\}$, $\mathcal{L}(y) = \{\{y : C \geq n\}\}$, $\mathcal{C}(x,y) = \{x_R \geq n\}$, and $\mathcal{C}(y) = \{x_C \geq n\}$
$\exists_{\leq}$**-rule** **Condition**: $\{x : (\exists R.C) \leq n\} \in \mathcal{L}(x)$, x is not indirectly blocked, and x has an R_{n1_R}-neighbor y with $\{[x,y] : R\ op\ n1\} \in \mathcal{L}(x,y)$ and $\{y : C \leq n\} \notin \mathcal{L}(y)$. **Action**: $\mathcal{L}(y) \longrightarrow \mathcal{L}(y) \cup \{\{y : C \leq n\}\}$, if $\{[x,y] : R\ op\ n1\}$ conjugates with $\{[x,y] : R \leq n\}$, then $\mathcal{C}(y) \longrightarrow \mathcal{C}(y) \cup \{x_C \leq n\}$, else $\mathcal{C}(y) \longrightarrow \mathcal{C}(y) \cup \{x_C \leq n, n_1 > n\}$
$\forall_{\geq}$**-rule** **Condition**: $\{x : (\forall R.C) \geq n\} \in \mathcal{L}(x)$, x is not indirectly blocked and x has an R_{n1_R}-neighbor y with $\{y : C \geq n\} \notin \mathcal{L}(y)$ **Action**: $\mathcal{L}(y) \longrightarrow \mathcal{L}(y) \cup \{\{y : C \geq n\}\}$, if $\{[x,y] : R\ op\ n1\}$ conjugates with $\{[x,y] : R \leq (1-n)\}$, then $\mathcal{C}(y) \longrightarrow \mathcal{C}(y) \cup \{x_C \geq n\}$, else $\mathcal{C}(y) \longrightarrow \mathcal{C}(y) \cup \{x_C \geq n, n1 > 1-n\}$
$\forall_{\leq}$**-rule** **Condition**: $\{x : (\forall R.C) \leq n\} \in \mathcal{L}(x)$, x is not blocked, x has no R_n-neighbor y, and $\{y : C \leq n\} \in \mathcal{L}(y)$ **Action**: create a new node y with $\mathcal{L}(x,y) = \{\{[x,y] : R \geq (1-n)\}\}$, $\mathcal{L}(y) = \{\{y : C \leq n\}\}$, $\mathcal{C}(x,y) = \{x_R \geq (1-n)\}$, and $\mathcal{C}(y) = \{x_C \leq n\}$
$\exists_{\leq,+}$**-rule** **Condition**: $\{x : (\exists R.C) \leq n\} \in \mathcal{L}(x)$, $Trans(R)$, x is not indirectly blocked, and x has an R_{n1_R}-neighbor y with $\{y : (\exists R.C) \leq n\} \notin \mathcal{L}(y)$ **Action**: $\mathcal{L}(y) \longrightarrow \mathcal{L}(y) \cup \{\{y : (\exists R.C) \leq n\}\}$, if $\{[x,y] : R\ op\ n1\}$ conjugates with $\{[x,y] : R \leq n\}$, then $\mathcal{C}(y) \longrightarrow \mathcal{C}(y) \cup \{x_{\exists R.C} \leq n\}$, else $\mathcal{C}(y) \longrightarrow \mathcal{C}(y) \cup \{x_{\exists R.C} \leq n, n1 > n\}$
$\forall_{\geq,+}$**-rule** **Condition**: $\{x : (\forall R.C) \geq n\} \in \mathcal{L}(x)$, $Trans(R)$, x is not indirectly blocked, and x has an R_{n1_R}-neighbor y with $\{y : (\forall R.C) \geq n\} \notin \mathcal{L}(y)$ **Action**: $\mathcal{L}(y) \longrightarrow \mathcal{L}(y) \cup \{\{y : (\forall R.C) \geq n\}\}$, if $\{[x,y] : R\ op\ n1\}$ conjugates with $\{[x,y] : R \leq (1-n)\}$, then $\mathcal{C}(y) \longrightarrow \mathcal{C}(y) \cup \{x_{\forall R.C} \geq n\}$, else $\mathcal{C}(y) \longrightarrow \mathcal{C}(y) \cup \{x_{\forall R.C} \geq n, n1 > 1-n\}$
$\rightarrow_{\geq}$**-rule** **Condition**: $\{A \rightarrow C \geq n\} \in \mathcal{T}$, $\{x : A \geq n_1\} \in \mathcal{L}(x)$ **Action**: $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{\{x : D \geq n_2\}\}$ and $\mathcal{C}(x) \longrightarrow \mathcal{C}(x) \cup \{x_D \geq n_2, n_2 = \min(n, n_1)\}$
$\rightarrow_{\leq}$**-rule** **Condition**: $\{A \rightarrow C \leq n\} \in \mathcal{T}$, $\{x : A \leq n_1\} \in \mathcal{L}(x)$ **Action**: $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{\{x : D \leq n_2\}\}$ and $\mathcal{C}(x) \longrightarrow \mathcal{C}(x) \cup \{x_D \leq n_2, n_2 = \min(n, n_1)\}$

The algorithm stops when a clash occurs; *KB* is consistent iff the completion rules can be applied in such a way that they yield a complete and clash-free completion forest, and *KB* is inconsistent otherwise.

Example 1. *Consider a fuzzy knowledge base* $KB = \{\ CP \rightarrow \exists hP.CP\ [0.5,1],\ CP(P002)\ [0.6,1],\ (\exists hP.CP)(P002)\ [0,0.4]\}$ *where we abbreviate the concept* CancerPatient *and the role* hasFirstDegreeRelatives *by CP and hP, respectively. The knowledge base describes that the truth degree for a first-degree relative of a cancer patient also being a cancer patient is greater than or equal to 0.5. Person* P002 *is a cancer patient with certainty greater than 0.6 and the possibility that one of* P002*'s first-degree relative is also a cancer patient is less than or equal to 0.4. The query is that whether KB is consistent or not.*

First, because of the fuzzy concept implication axiom, $\{\exists hP.CP(P002)\ [0.5,1]\}$ *is added to* $\mathcal{A}$. *Next, we can initialize the fuzzy tableau by creating a node* P002 *and label it with* $\mathcal{L}(P002) = \{\{P002 : CP \geq 0.6\}, \{P002 : \exists hP.CP \geq 0.5\}, \{P002 : \exists hP.CP \leq 0.4\}\}$ *and* $\mathcal{C}(P002) = \{x_{CP} \geq 0.6, x_{\exists hP.CP} \geq 0.5, x_{\exists hP.CP} \leq 0.4\}\}$. *Since both* $\{P002 : \exists hP.CP \geq 0.5\}$ *and* $\{P002 : \exists hP.CP \leq 0.4\}$ *are contained in the fuzzy tableau, the reasoning algorithm obviously detects a clash. Therefore, it stops the application of any fuzzy completion rule and returns that KB is inconsistent.*

Next, let us look at an example for the $\forall_{\geq,+}$-rule.

Example 2. *Consider there are two assertions in a fuzzy knowledge base:* $(\forall hasFriend.Student)(John)\ [0.75,1]$ *and* $hasFriend(John,Mary)\ [0.7,1]$ *where hasFriend is a transitive role.*

Following the preprocessing steps, we have $\{John : (\forall hasFriend.Student) \geq 0.75\} \in \mathcal{L}(John)$ *and* $\{[John,Mary] : hasFriend \geq 0.7\}$. *Since* $\{[John,Mary] : hasFriend \geq 0.7\}$ *conjugates with* $\{[John,Mary] : hasFriend \leq 0.25\}$, *the* $\forall_{\geq,+}$*-rule is applicable, thus* $\{Mary : (\forall hasFriend.Student) \geq 0.75\}$ *is added to* $\mathcal{L}(Mary)$.

We can see from Table 3 that all these fuzzy completion rules are based on the properties and the semantics of $f_Z\mathcal{SI}$ concepts. Notice that since we assume all concepts to be in their negation normal form, the fuzzy concept negation rule only applies to concept names.

Let us take a second look at the $\sqcup_{\geq}$-rule and the $\sqcap_{\leq}$-rule. The $\sqcup_{\geq}$-rule generates several new constraints $\{x_1 + x_2 = n, x_1 \leq y, x_2 \leq 1-y, y \in \{0,1\}, x_1 \in [0,1], x_2 \in [0,1]\}$. We can see that y is an integer variable with value of 0 or 1. When $y = 0$, we have $x_1 = 0$, $x_2 = n$, and thus $\{x_{C_1} \geq 0, x_{C_2} \geq n\}$; while $y = 1$, we have $x_1 = n$, $x_2 = 0$, and thus $\{x_{C_1} \geq n, x_{C_2} \geq 0\}$. These two cases are actually representing the or-branch of the concept disjunction rule in classic DL. That is, the $\{0,1\}$ integer variable y enable the simulation of or-branching. Furthermore, by the introduction of the variable y, we also transform the non-linear constraint $max(x_1,x_2) \geq n$ into a set of linear constraints. Similar conclusions can be drawn on the $\sqcap_{\leq}$-rule. Now we can see that all the fuzzy completion rules in Table 3 generate only linear constraints, therefore, the resulted constraint set for any node or edge is a linear constraint set. Such a property makes it possible for the reasoning algorithm to call some external *Linear Programming* solver to solve the constraint set.

Here is another example to explain how the reasoning algorithm determines the consistency of a knowledge base.

Example 3. *Consider the following fuzzy knowledge base* $KB = \{Trans(R),\ C(a)\ [0.7,1],\ D(b)\ [0.8,1],\ R(a,b)\ [0.6,1],\ R(b,c)\ [0.7,1],\ (\exists Inv(R).C \sqcap \exists Inv(R).D)(c)\ [0,0.5]\}$. *We want to check the consistency of the knowledge base.*

With $Trans(R)$ *and* $Inv(R)$, *we have* $Trans(Inv(R))$. *The fuzzy tableau is initialized as shown in Figure 1.*

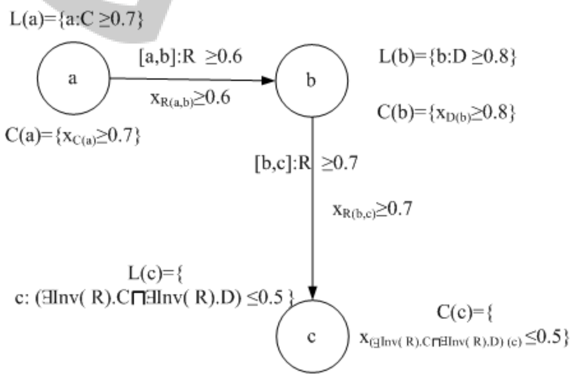

Figure 1: The initial fuzzy tableau of example 3.

Next, since $\{c : (\exists Inv(R).C \sqcap \exists Inv(R).D) \leq 0.5\} \in \mathcal{L}(c)$, *the* $\sqcap_{\leq}$*-rule is triggered, the reasoning algorithm adds* $\{c : (\exists Inv(R).C) \leq x_1\}$ *and* $\{c : (\exists Inv(R).D) \leq x_2\}$ *to* $\mathcal{L}(c)$, *adds* $\{x_{(\exists Inv(R).C)(c)} \leq x_1, x_{(\exists Inv(R).D)(c)} \leq x_2, x_1 + x_2 = 1 + 0.5, x_1 \geq y, x_2 \geq 1-y, y \in \{0,1\}, x_1 \in [0,1], x_2 \in [0,1]\}$ *to* $\mathcal{C}(c)$.

Next, since $\{c : (\exists Inv(R).C) \leq x_1\} \in \mathcal{L}(c)$, $\{c : (\exists Inv(R).D) \leq x_2\} \in \mathcal{L}(c)$, *and we have* $[b,c] : R \geq 0.7$, *the* $\exists_{\leq}$*-rule is applicable, thus the reasoning algorithm adds* $\{b : C \leq x_1\}$ *and* $\{b : D \leq x_2\}$ *to* $\mathcal{L}(b)$, *adds* $\{x_{C(b)} \leq x_1, x_{D(b)} \leq x_2, x_1 < 0.7, x_2 < 0.7\}$ *to* $\mathcal{C}(b)$. *Note that the constraints* $x_1 < 0.7$ *and* $x_2 < 0.7$ *are added because of conjugation.*

Next, since $\{c : (\exists Inv(R).C) \leq x_1\} \in \mathcal{L}(c)$, $\{c : (\exists Inv(R).D) \leq x_2\} \in \mathcal{L}(c)$, *we have* $[b,c] : R \geq 0.7$ *and* $Trans(Inv(R))$, *the* $\exists_{\leq,+}$*-rule is also applicable, thus the reasoning algorithm adds* $\{b : (\exists Inv(R).C) \leq x_1\}$ *and* $\{b : (\exists Inv(R).D) \leq x_2\}$ *to* $\mathcal{L}(b)$, *adds* $\{x_{(\exists Inv(R).C)(b)} \leq x_1, x_{(\exists Inv(R).D)(b)} \leq x_2\}$ *to* $\mathcal{C}(b)$.

Next, since $\{b : (\exists Inv(R).C) \leq x_1\} \in \mathcal{L}(b)$, $\{b : (\exists Inv(R).D) \leq x_2\} \in \mathcal{L}(b)$, *and we have* $[a,b] : R \geq 0.6$, *the* $\exists_{\leq}$*-rule is also applicable, thus the reasoning algorithm adds* $\{a : C \leq x_1\}$ *and* $\{a : D \leq x_2\}$ *to* $\mathcal{L}(a)$, *adds* $\{x_{C(b)} \leq x_1, x_{D(b)} \leq x_2, x_1 < 0.6, x_2 < 0.6\}$ *to* $\mathcal{C}(a)$.

Now the fuzzy tableau is shown in Figure 2. Together with the default variable constraints, the reasoning algorithm forms the following constraint set:

$$\text{subject to} \begin{cases} x_{C(a)} \geq 0.7, x_{D(b)} \geq 0.8 \\ x_{R(a,b)} \geq 0.6, x_{R(b,c)} \geq 0.7 \\ x_{(\exists Inv(R).C \sqcap \exists Inv(R).D)(c)} \leq 0.5 \\ x_{(\exists Inv(R).C)(c)} \leq x_1, x_{(\exists Inv(R).D)(c)} \leq x_2 \\ x_1 + x_2 = 1 + 0.5 \\ x_1 \geq y, x_2 \geq 1 - y \\ x_{C(b)} \leq x_1, x_{D(b)} \leq x_2 \\ x_1 < 0.7, x_2 < 0.7 \\ x_{C(a)} \leq x_1, x_{D(a)} \leq x_2 \\ x_1 < 0.6, x_2 < 0.6 \\ x_{(\exists Inv(R).C)(b)} \leq x_1, x_{(\exists Inv(R).D)(b)} \leq x_2 \\ x_{C(a)}, x_{D(b)}, x_{R(a,b)}, x_{R(b,c)} \in [0,1] \\ x_{(\exists Inv(R).C \sqcap \exists Inv(R).D)(c)} \in [0,1] \\ x_{(\exists Inv(R).C)(c)}, x_{(\exists Inv(R).D)(c)} \in [0,1] \\ y \in \{0,1\} \\ x_1, x_2, x_{C(b)}, x_{D(b)} \in [0,1] \\ x_{(\exists Inv(R).C)(b)}, x_{(\exists Inv(R).D)(b)} \in [0,1] \end{cases}$$

Using a Linear Programming *solver, e.g., the GLPK solver (GLPK, 2008), it is easy to show that the constraint set is unsolvable. Therefore, the fuzzy knowledge base is inconsistent.*

Through this example, it is shown that the consistency check of a knowledge base can be reduced to a problem of constraints solving in linear programming. The constraints solving can be processed either at the end of the reasoning procedure when no further fuzzy completion rules are applicable, or after each application of a completion rule. The advantage of the later case is that, in some situations, the computation effort could be saved when the constraints solver can identify unsolvable constraints sets earlier in the reasoning process. However, in other situations, since calling an external solver is time consuming, frequent calls will severely affect the overall performance. In the former case, we only have to call the external solver once. In

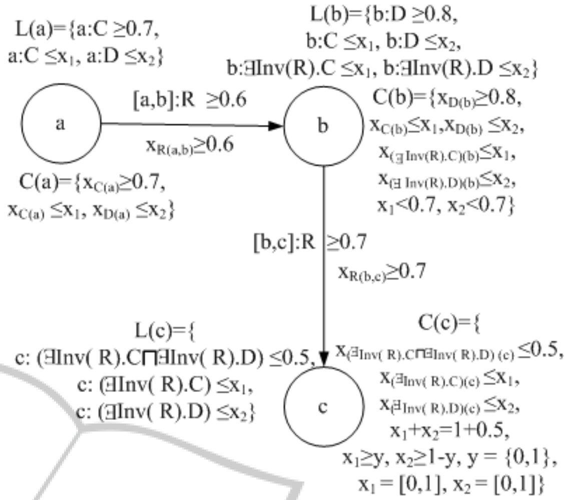

Figure 2: The extended fuzzy tableau of example 3.

addition, we can apply some optimization strategies such as trivial clash detection and individual groups to improve the performance.

It is well known that there is always the trade-off issue between the expressive power of a DL and its computational complexity. The more expressive a DL is, the higher its computational complexity. Horrocks et. al presented an optimized version of the tableau algorithm for classic $\mathcal{SI}$ in (Horrocks et al., 1998), which generates completion trees whose depth is polynomially bounded by the size of $sub(\mathcal{A})$. It is an interesting problem to investigate the applicability of the optimization to the fuzzy case.

4 CONCLUSIONS

In this paper, we address the fuzzy instance entailment problem with respect to a fuzzy knowledge base and then present a fuzzy extension to the expressive Description Logic $\mathcal{SI}$ based on Zadeh Logic and the residual R-implication.

For real-world applications where a knowledge base is considered as a means to store information (both precise and imprecise) about individuals, usually more complex inferences other than consistency checking are required. For example, users may want to pose a query like “Given a knowledge base, what’s the certainty of an assertion?”. Another kind of query can be “How many individuals belong to a given concept description with a confidence greater than 0.5, and what are they?” We describe the former query as an instance range entailment problem and the later as an f-retrieval problem. However, due to space limitations, the reasoning methods for these problems are omitted in this paper.

A prototype reasoner using SWI-Prolog and GLPK has been under implementation based on the $\mathcal{ALC}$ reasoner ALCAS (Spencer, 2006). It currently supports functionalities to check consistency, fuzzy instance entailment and f-retrieval of a fuzzy $f_Z\mathcal{ALC}$ knowledge base. Part of our ongoing work considers further development of the reasoner to support other reasoning problems as well as more expressivity in the fuzzy knowledge base.

As we pointed out in Section 2, the properties for transitive roles and value restrictions also hold under Product Logic. Therefore, another direction of future work is to investigate the reasoning algorithms for expressive fuzzy Description Logics under norms from other logics in the family of Fuzzy Logics.

REFERENCES

Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., and Patel-Schneider, P. F. (2003). *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, Cambridge, MA.

Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web. *Scientific American*, 284(5):34–44.

Díaz, S., De Baets, B., and Montes, S. (2010). General results on the decomposition of transitive fuzzy relations. *Fuzzy Optimization and Decision Making*, 9(1):1–29.

GLPK (2008). GNU linear programming kit. Technical Report http://gnuwin32.sourceforge.net/packages/ glpk.htm.

Haase, P. and Völker, J. (2005). Ontology learning and reasoning - dealing with uncertainty and inconsistency. In *Proceedings of Uncertainty Reasoning for the Semantic Web*, pages 45–55.

Horrocks, I. and Sattler, U. (1999). A description logic with transitive and inverse roles and role hierarchies. *J. of Logic and Computation*, 9(3):385–410.

Horrocks, I., Sattler, U., and Tobies, S. (1998). A PSPACE-algorithm for deciding $\mathcal{ALCI}_{R^+}$-satisfiability. LTCS-Report 98-08, LuFg Theoretical Computer Science, RWTH Aachen, Germany.

Horrocks, I., Sattler, U., and Tobies, S. (2000). Reasoning with individuals for the description logic $\mathcal{SHIQ}$. In McAllester, D., editor, *Proc. of the 17th Int. Conf. on Automated Deduction (CADE 2000)*, volume 1831 of *Lecture Notes in Computer Science*, pages 482–496. Springer.

Jaeger, M. (1994). Probabilistic reasoning in terminological logics. In *Proc. of the 4th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR94)*, pages 305–316.

Koller, D., Levy, A., and Pfeffer, A. (1997). P-classic: A tractable probabilistic description logic. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97)*, pages 390–397.

Laskey, K. J., Laskey, K. B., Costa, P. C. G., Kokar, M. M., Martin, T., and Lukasiewicz, T. (05 March, 2008). W3C incubator group report. Technical Report http://www.w3.org/2005/Incubator/urw3, W3C.

Lee, H.-S. (2001). An optimal algorithm for computing the maxmin transitive closure of a fuzzy similarity matrix. *Fuzzy Sets and Systems*, 123(1):129–136.

Lukasiewicz, T. (2008). Fuzzy description logic programs under the answer set semantics for the semantic web. *Fundamenta Informaticae*, 82(3):289–310.

Martin-Recuerda, F. and Robertson, D. (2005). Discovery and uncertainty in semantic web services. In *Proceedings of Uncertainty Reasoning for the Semantic Web*, page 188.

McGuinness, D. L. and van Harmelen, F. (2004). Owl web ontology language overview. http://www.w3.org/TR/owl-features/.

Mitsuishi, T. and Bancerek, G. (2003). Transitive closure of fuzzy relations. *Journal of Formalized Mathematics*, 15.

Spencer, B. (2006). ALCAS: An ALC Reasoner for CAS. http://www.cs.unb.ca/ bspencer/cs6795swt/alcas.prolog.

Stamou, G., van Ossenbruggen, J., Pan, J. Z., and Schreiber, G. (2006). Multimedia annotations on the semantic web. *IEEE MultiMedia*, 13:86–90.

Stevens, R., Aranguren, M. E., Wolstencroft, K., Sattlera, U., Drummond, N., Horridge, M., and Rectora, A. (2007). Using owl to model biological knowledge. *International Journal of Human-Computer Studies*, 65(7):583–594.

Straccia, U. (2001). Reasoning within fuzzy description logics. *Journal of Artificial Intelligence Research*, 14:137–166.

Yen, J. (1991). Generalizing term subsumption languages to fuzzy logic. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI'91)*, pages 472–477.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, 8(3):338–353.

Zhao, J. (2010). *Uncertainty and Rule Extensions to Description Logics and Semantic Web Ontologies*, chapter 1, pages 1–22. Advances in Semantic Computing. Technomathematics Research Foundation. accepted.

Zhao, J. and Boley, H. (2010). *Knowledge Representation and Reasoning in Norm-Parameterized Fuzzy Description Logics*. Canadian Semantic Web: Technologies and Applications. Springer.