Artificial Intelligence Research and Development
R. Alquézar et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-643-0-269

Similarity for Attribute-value Representations in Fuzzy Description Logics

Àngel GARCÍA-CERDAÑA [a,1], Eva ARMENGOL [a], Pilar DELLUNDE [a,b]
[a] *Artificial Intelligence Research Institute (IIIA - CSIC)*
[b] *Universitat Autònoma de Barcelona*

Abstract. In this paper we explore the possibility of introducing the equality symbol in the languages of Fuzzy Description Logics (FDLs) interpreted as a similarity relation. The goal is twofold: dealing with attribute-value representations at the domain objects level, and integrating the treatment of similarities inside the description languages and their corresponding knowledge bases.

Keywords. Equality, Similarity Relation, Description Logics, Fuzzy Description Logics, Attribute-value Representation

Introduction

Similarity has been a central issue for decades in different disciplines, ranging from philosophy (Leibniz's Principle of the Identity of Indiscernibles) and psychology (Tversky's stimuli judged similarity) to natural sciences (taxonomy) and mathematics (geometric similarity). Also in artificial intelligence similarity plays an important role since the reasoning by analogy is behind some of the early machine learning methods. Particularly, case-based reasoning methods are based on the principle that "similar problems have similar solutions" where the notion of similarity has a capital importance (see [14]).

The aim of this paper is to analyze the role of similarity in the context of *Description Logics* (DLs). These formalisms are knowledge representation languages built on the basis of classical logic. DLs allow the creation of knowledge bases and provide ways to reason on the contents of these bases [1]. The vocabulary of DLs consists of *concepts*, which denote sets of individuals, and *roles*, which denote binary relations among individuals. From atomic concepts and roles and by means of *constructors*, DL systems allow us to build complex descriptions of both concepts and roles. These complex descriptions are used to describe a domain through a knowledge base (KB). One of the main issues of DLs is the fact that the statements in the KB can be identified with formulas in first order logic or an extension of it; therefore we can use reasoning to obtain implicit knowledge from the explicit knowledge in the KB. *Fuzzy Description Logics* (FDLs) are natural extensions of DLs for dealing with vague concepts, commonly present in real applications

[1]Corresponding Author: Àngel García-Cerdaña, Institut d'Investigació en Intel.ligència Artificial, Campus de la UAB, 08193 Bellaterra, Barcelona, Spain; E-mail: angel@iiia.csic.es

(see [16] for a survey). Hájek [12] proposed to deal with FDLs taking as basis t-norm based fuzzy logics with the aim of enriching the expressive possibilities in FDLs and to capitalize on recent developments in the field of mathematical fuzzy logic (see [10]). We are also interested in how to deal with similarity inside FDLs. Although similarity has been widely studied in fuzzy logics [18,11,9,6], until now its use in FDLs has deserved little attention. For a historical overview on fuzzy similarity relations see [17].

Concerning knowledge representation in artificial intelligence, it is common to represent objects as sets of pairs attribute-value. Therefore it should be interesting to analyze description logic languages capturing this kind of representation. In this paper we make a preliminary study of the requirements needed to expand a description logic in order to accomplish the following issues: a) to be able to represent objects as attribute-value pairs; b) to express similarity between objects, between concepts and between relations; c) to express the basic similarity logical properties (reflexivity, symmetry, etc.); d) to be able to define similarities using formulas of the description language; and e) to provide the language the needed skills to reason with similarities using both classical and fuzzy concepts and roles.

The paper is organized as follows. Section 1 discusses how to deal with atributte-value representation in DLs. Section 2 contains our approach introducing similarity inside DLs. Section 3 considers several definitions of similarity between concepts. Last section is devoted to conclusions and future work.

1. Attribute-value Representations in Description Logics

In artificial intelligence, domain objects are commonly represented using sets of attribute-value pairs. For instance, let us suppose that a person is described by the following attributes: name, age, hair, and weight. Notice that the values that each attribute can take belong to different domains. Thus, both age and weight are *numerical*, however we can consider that the former is a natural number whereas the later is a rational number; name is a *string*; and hair can take values into a finite set (sometimes this is called an *enumerated* type), for instance $\{blonde, white, black, brown\}$. In this context, the *global* similarity between two objects has to be seen as an aggregation of the *local* similarities of the attributes describing them (see [15] and [7] for a collection of similarity and aggregation measures, respectively). One of our goals is to analyze whether or not a basic description language as, for instance $\mathcal{ALC}$, is the more appropriate framework to express objects represented as attribute-value vectors and, if not, which could be the necessary extensions to capture the attribute-value representation of domain objects. We will use a running example to motivate the necessity of looking for such language expansions: the Little Robots data set (Fig. 1). This application domain is a free adaptation of the Monks data set from the Machine Learning Repository of the Irvine's University (http://archive.ics.uci.edu/ml/). Robots of the data set are described by 5 attributes: body, head, has-tie, happy, holding. Attributes body and head are of the enumerated type *Forms* taking values in the set $\{round, octagon, square\}$; the attribute holding is of the enumerated type *Tools* and takes values from the set $\{balloon, flag, sword, flower, ax\}$; and the attributes has-tie and happy are *boolean*.

Firstly, let us recall some definitions relative to description languages. Their formulas (concept or role descriptions) are built from *atomic concepts* (A) and *atomic roles*

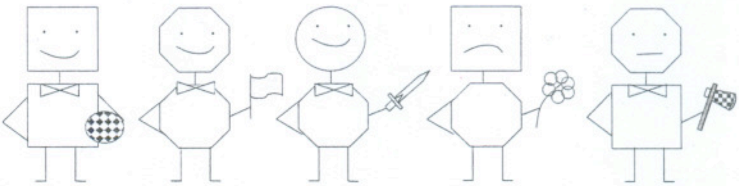

Figure 1. Some little robots of our data set.

(R) by means of *constructors*. In order to define a formal semantics for the description formulas we consider *interpretations*. An interpretation $\mathcal{I}$ is a pair $\langle \Delta^{\mathcal{I}}, (.)^{\mathcal{I}} \rangle$, where $\Delta^{\mathcal{I}}$ is a non-empty set (the interpretation domain), and $(.)^{\mathcal{I}}$ is a map, which assigns to every atomic concept A a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and to every atomic role R a binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. For instance, in the Robots data set, we take as atomic concepts the following ones: Object, Wears-tie, FriendlyObject, Body, Head, Tool, Balloon, Flag, Sword, Flower, and Ax; and the atomic role hasObject. Table 1 shows the name, syntax and semantics for each constructor of the language $\mathcal{ALC}$. Thus, for instance, the interpretation domain $\Delta^{\mathcal{I}}$ is the set of robots in Fig. 1 and the concept Happy $\sqcap$ Wears-tie is interpreted as the intersection of objects that both are happy and wear a tie.

Commonly, a knowledge base in classical DLs is formed by a TBox and an ABox. In its more general form a TBox is a finite set of *(general) concept inclusion axioms*, which are expressions of the form $C \sqsubseteq D$ (see example in Fig. 2). An interpretation $\mathcal{I}$ satisfies the axiom $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. The expression $C \equiv D$ denotes equality, that is, $C^{\mathcal{I}} = D^{\mathcal{I}}$. An interpretation which satisfies a TBox $\mathcal{T}$ is said to be a *model* of $\mathcal{T}$. An ABox for $\mathcal{ALC}$ is a finite set of formal expressions of the form $C(a)$ (*concept assertion* axiom) and $R(a, b)$ (*role assertion* axiom). For instance, the description of robot r_1 shown in the right part of Fig. 2 should be placed in the ABox. The semantics of an ABox is given by extending the interpretation $\mathcal{I}$ mapping each individual name a to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. This mapping has to respect the *unique name assumption*: if a and b are distinct individual names, then $a^{\mathcal{I}} \neq b^{\mathcal{I}}$, i.e., each name uniquely corresponds to an object. The interpretation $\mathcal{I}$ *satisfies* the axiom $C(a)$ if $a^{\mathcal{I}} \in C^{\mathcal{I}}$, and it *satisfies* $R(a, b)$ if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$. An interpretation $\mathcal{I}$ which satisfies an assertion α is said a *model* of α. We will say that $\mathcal{I}$ satisfies α *with respect to* a TBox $\mathcal{T}$ if in addition to being a model of α, it is a model of $\mathcal{T}$.

Table 1. Concept Constructors for $\mathcal{ALC}$.

NAME	SYNTAX	SEMANTICS
Top	$\top$	$\Delta^{\mathcal{I}}$
Bottom	$\bot$	$\emptyset$
Intersection	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
Union	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
Complementation	$\neg C$	$\Delta^{\mathcal{I}} \backslash C^{\mathcal{I}}$
Value restriction	$\forall R.C$	$\{a \in \Delta^{\mathcal{I}} : \{b \in \Delta^{\mathcal{I}} : \langle a, b \rangle \in R^{\mathcal{I}}\} \subseteq C^{\mathcal{I}}\}$
Existential quantification	$\exists R.C$	$\{a \in \Delta^{\mathcal{I}} : \{b \in \Delta^{\mathcal{I}} : \langle a, b \rangle \in R^{\mathcal{I}}\} \cap C^{\mathcal{I}} \neq \emptyset\}$

Body $\sqcup$ Head $\sqcup$ Tool	$\sqsubseteq$	Object
Body $\sqcap$ Head	$\sqsubseteq$	$\bot$
Tool $\sqcap$ Body	$\sqsubseteq$	$\bot$
Tool $\sqcap$ Head	$\sqsubseteq$	$\bot$
FriendlyObject	$\sqsubseteq$	Tool
UnfriendlyObject	$\equiv$	Tool $\sqcap$ $\neg$FriendlyObject
Flower	$\sqsubseteq$	FriendlyObject
Balloon	$\sqsubseteq$	FriendlyObject
Flag	$\sqsubseteq$	FriendlyObject
Sword	$\sqsubseteq$	UnfriendlyObject
Ax	$\sqsubseteq$	UnfriendlyObject

Object(r_1)
hasObject(r_1, b_1)
hasObject(r_1, h_1)
hasObject(r_1, o_1)
Body(b_1)
Head(h_1)
Balloon(o_1)

Figure 2. Example of TBox and description of the robot r_1 in the ABox.

The left part of Fig. 2 shows a possible TBox for the Robots data set represented using only constructors from $\mathcal{ALC}$. This TBox states that there are three kinds of objects: body, head and tool; and that there are two kinds of tools: friendly objects and unfriendly objects. The right part of Fig. 2 shows the description of the robot r_1 contained in the ABox. From the atomic concepts above, we can define in the TBox a *robot* as follows:

$$\text{Robot} \equiv \text{Object} \,\&\, (\exists\ \text{hasObject.Body}) \,\&\, (\exists\ \text{hasObject.Head}) \,\&\, (\exists\ \text{hasObject.Tool})$$

Notice that robot r_1 is described by means of boolean predicates. For instance, to say that the robot holds a balloon it is necessary to say that it has an object o_1 and that o_1 is a balloon; and therefore to define a boolean predicate for each one of the possible values that the attribute can take. The use of enumerated types in attribute-value representations avoids this need of defining a boolean predicate for each value of an attribute.

A natural and elegant way to handle enumerated types in DLs is by using the *concrete domains* proposed by Baader and Hanschke in [2]. Formally, a *concrete domain* $\mathcal{D}$ consists of a set $\Delta^{\mathcal{D}}$ (the universe of $\mathcal{D}$) and a set of predicate names *pred*($\mathcal{D}$). Each $P \in pred(\mathcal{D})$ is associated with an arity n and an n-ary relation $P^{\mathcal{D}} \subseteq (\Delta^{\mathcal{D}})^n$. $\mathcal{ALC}(\mathcal{D})$ is obtained from $\mathcal{ALC}$ by introducing *abstract features* (roles interpreted as functional relations), *concrete features* (a new syntactic type that is interpreted as a partial function from the domain $\Delta^{\mathcal{I}}$ into the concrete domain $\Delta^{\mathcal{D}}$) and finally, by adding the concept constructor called *existential predicate restriction* $\exists(u_1, \ldots, u_n).P$, where P is an n-ary predicate of the concrete domain and $u_1, \ldots, u_n$ are feature chains. Every chain u_i is of the form $f_1 \circ \cdots \circ f_m \circ g$, where $f_1 \ldots f_m$ are abstract features and g is a concrete feature. The semantics of each u_i is given by the composition of the partial functions interpreting their components, that is, $u^{\mathcal{I}}(a) = g^{\mathcal{D}}(f_m^{\mathcal{I}}(\ldots f_1^{\mathcal{I}}(a) \ldots))$. The interpretation $(\exists(u_1, \ldots, u_n).P)^{\mathcal{I}}$ of this new constructor is the set of elements $a \in \Delta^{\mathcal{I}}$ such that there exist $r_1, \ldots, r_n \in \Delta^{\mathcal{D}}$ in such a way that $u_1^{\mathcal{I}}(a) = r_1, \ldots, u_n^{\mathcal{I}}(a) = r_n$ and $(r_1, \ldots, r_n) \in P^{\mathcal{D}}$.

Let us to illustrate the fact that a language as $\mathcal{ALC}(\mathcal{D})$ is appropriated to deal with objects represented as vectors of attribute values. For instance, suppose that we want to define an *homogeneous robot* as a robot having the same shape of both the head and the body. Using the constructor introduced above this definition can be done as follows:

$$\text{Homogeneous-Robot} \equiv \exists(\text{body} \circ \text{has-form}, \text{head} \circ \text{has-form}).=$$

where the symbol $=$ is interpreted as the usual crisp equality. The interpretation of the abstract features body and head is the following: $\mathsf{body}^{\mathcal{I}} : \mathsf{Robot}^{\mathcal{I}} \to \mathsf{Body}^{\mathcal{I}}$; $\mathsf{head}^{\mathcal{I}} : \mathsf{Robot}^{\mathcal{I}} \to \mathsf{Head}^{\mathcal{I}}$; and the interpretation of the concrete feature has-form is the following: $\mathsf{has\text{-}form}^{\mathcal{I}} : \mathsf{Body}^{\mathcal{I}} \cup \mathsf{Head}^{\mathcal{I}} \to Forms = \{round, octagon, square\}$. If b_1 and h_1 are respectively the body and the head of the robot r_1 we have $\mathsf{body}^{\mathcal{I}}(r_1) = b_1$, $\mathsf{head}^{\mathcal{I}}(r_1) = h_1$, $\mathsf{has\text{-}form}^{\mathcal{I}}(b_1) = square$ and $\mathsf{has\text{-}form}^{\mathcal{I}}(h_1) = square$. Thus, $\mathsf{body}^{\mathcal{I}} \circ \mathsf{has\text{-}form}^{\mathcal{I}}(r_1) = square$.

2. Similarity in Fuzzy Description Logics

In classical mathematics, identity is understood as "numerical identity" or "sameness". That is, an object can only be related by the identity relation with itself. Classical first order mathematical logic inherits this conception of identity interpreting the identity symbol always as the diagonal of the cartesian product of the elements of the domain. The identity symbol is regarded as a logical symbol (in the same way as connectives or quantifiers). When description languages are regarded as fragments of predicate logic, the same conception prevails. Thus, when in some extensions of $\mathcal{ALC}$, given a concept C, the *identity role* $id(C)$ is defined, its interpretation is $\{(a, a) : a \in C^{\mathcal{I}}\}$.

Similarity between mathematical objects is often formalized by means of equivalence or congruence relations. However, in other domains such as psychology, for instance Tversky's stimuli judged similarity [20], its formalization enjoys different properties. Our aim is to introduce in the language roles interpreted as similarities (possibly with different properties, depending of the domain) and to endow DL languages with the appropriate tools to reason with them. For this purpose we need, on the one hand, to express basic properties of similarities, such as reflexivity or symmetry. On the other hand, we would like to allow the language to define object similarities in terms of previously defined value similarities for the object attributes (for instance, in our running example, to define when two robots are similar in terms of the form of their bodies and heads or the objects they hold). To satisfy these requirements we propose to extend the KBs for our description languages in the following way: 1) to allow similarity assertions in the ABox of the form $a \approx b$, and 2) to include a *Similarity Box* (SBox) in the KB. In the SBox we could declare properties of similarities and define new roles, also interpreted as similarities. For this purpose we take $\mathcal{SROIQ}(D)$ [13] that is a DL language providing regular complex role inclusion axioms of the form $w \dot{\sqsubseteq} R$, where w is a finite string of roles (regularity prevents a role hierarchy from containing cyclic dependencies). In $\mathcal{SROIQ}(D)$ there is a RBox that includes role assertions stating, for instance, that a certain role must be interpreted as a reflexive relation or that two (possibly inverse) roles are to be interpreted as disjoint relations. The SBox we propose is a particular case of the RBox of $\mathcal{SROIQ}(D)$ restricted to sentences about similarities. Table 2 shows the kind of statements contained into the SBox.

We believe that our approach can be appropriated for both classical and fuzzy frameworks. As Ruspini suggests in [18], the degree of similarity between two objects A and B may be seen as the degree of truth of the vague proposition "A is similar to B". Thus, similarity among objects can be seen as a phenomena essentially fuzzy. Because of that we need to extend $\mathcal{SROIQ}(D)$ to the fuzzy framework and take fuzzy interpretations of the SBox.

Table 2. Similarity Box Axioms.

NAME	**SYNTAX**	
Role equality	$R \equiv S$	
Role similarity	$R \approx S$	
Reflexivity of $\approx$	$Id(\top) \sqsubseteq \approx$	
Symmetry of $\approx$	$\approx \equiv (\approx)^-$	
Transitivity of $\approx$	$\approx \circ \approx \sqsubseteq \approx$	
$\approx$ is a congruence	$\forall \approx .C \sqsubseteq C$	C a concept
	$\approx \circ R \sqsubseteq R$	R a role

The extension of DLs to the fuzzy framework consists in interpreting the atomic concepts as fuzzy sets and the atomic roles as fuzzy binary relations. Thus, a *fuzzy interpretation* for a description language is also a pair $\mathcal{I} = \langle \Delta^{\mathcal{I}}, (.)^{\mathcal{I}} \rangle$, where $\Delta^{\mathcal{I}}$ is a *crisp* non-empty set but in this case, the map $(.)^{\mathcal{I}}$ assigns, to every atomic concept A, a function $A^{\mathcal{I}} : \Delta^{\mathcal{I}} \to [0,1]$, i.e., a *fuzzy set* on $\Delta^{\mathcal{I}}$; and it assigns, to every atomic role R, a function $R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \to [0,1]$, i.e., a *fuzzy binary relation* on $\Delta^{\mathcal{I}}$. Notice that this definition generalizes the classical notion of interpretation (see Section 1) to the fuzzy setting.[2] The map $(.)^{\mathcal{I}}$ can be inductively extended to other concept constructors by using a continuous t-norm $*$, the Łukasiewicz negation $N(x) = 1 - x$, the dual t-conorm of $*$ w.r.t. this negation, and the residuum $\to_*$ of $*$. Thus, the algebra of truth values is $[0,1]_* = \langle [0,1], \leq, *, \to_*, N, 0, 1 \rangle$. For instance, in the case of $\mathcal{ALC}$, the interpretation can be defined by the following fuzzy sets (cf. [10]):

$$
\begin{array}{rcl}
\bot^{\mathcal{I}}(a) & = & 0 \\
\top^{\mathcal{I}}(a) & = & 1 \\
(\neg C)^{\mathcal{I}}(a) & = & 1 - C^{\mathcal{I}}(a) \\
(C \sqcap D)^{\mathcal{I}}(a) & = & C^{\mathcal{I}}(a) * D^{\mathcal{I}}(a) \\
(C \sqcup D)^{\mathcal{I}}(a) & = & 1 - [(\neg C)^{\mathcal{I}}(a) * (\neg D)^{\mathcal{I}}(a)] \\
(\forall R.C)^{\mathcal{I}}(a) & = & \inf\{R^{\mathcal{I}}(a,b) \to_* C^{\mathcal{I}}(b) : b \in M\} \\
(\exists R.C)^{\mathcal{I}}(a) & = & \sup\{R^{\mathcal{I}}(a,b) * C^{\mathcal{I}}(b) : b \in M\}
\end{array}
$$

A *fuzzy* KB is composed of a *fuzzy* TBox and a *fuzzy* ABox. An axiom of a fuzzy KB is an expression of the following forms $\langle \alpha, \geq r \rangle$, $\langle \alpha, = r \rangle$, or $\langle \alpha, \leq r \rangle$, where $r \in [0,1]$, and α may be either $C \sqsubseteq D$ (if it is a fuzzy TBox axiom) or $C(a)$ o $R(a,b)$ (if it is a fuzzy ABox axiom). An example of fuzzy TBox axiom is $\langle \mathsf{Sword} \sqsubseteq \mathsf{FriendlyObject}, = 0.25 \rangle$ stating that a sword is a friendly object to a degree 0.25. An example of ABox axiom is $\langle \mathsf{Happy}(r_5), = 0.5 \rangle$ stating that robot r_5 is happy to a degree 0.5. We denote by $\models$ the satisfiability relation. Given a fuzzy interpretation $\mathcal{I}$, the semantics for these fuzzy axioms is the following:

$$
\begin{array}{lll}
\mathcal{I} \models \langle \alpha, \geq r \rangle & \text{iff} & \alpha^{\mathcal{I}} \geq r \\
\mathcal{I} \models \langle \alpha, = r \rangle & \text{iff} & \alpha^{\mathcal{I}} = r \\
\mathcal{I} \models \langle \alpha, \leq r \rangle & \text{iff} & \alpha^{\mathcal{I}} \leq r
\end{array}
$$

[2] A crisp set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ (resp. a crisp binary relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$) can be equivalently seen as a function $A^{\mathcal{I}} : \Delta^{\mathcal{I}} \to \{0,1\}$ (resp. $R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \to \{0,1\}$), that is, as the *characteristic function* associated to the set $A^{\mathcal{I}}$ (resp. $R^{\mathcal{I}}$) w.r.t. the universe $\Delta^{\mathcal{I}}$ (resp. $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$).

In a language expanded with similarity roles we propose to introduce in the ABox expressions as for instance $\langle a \approx b, \geq r \rangle$ interpreted as assessing that the similarity degree between a and b is greater or equal than r. Concerning the SBox we want to include axioms expressing properties of fuzzy equalities. Hájek [11] proposes the following axioms for similarities in fuzzy predicate languages:

1. *(Reflexivity)* $\forall x\, x \approx x$
2. *(Symmetry)* $\forall x \forall y (x \approx y \rightarrow y \approx x)$
3. *(Transitivity)* $\forall x \forall y \forall z ((x \approx y \,\&\, y \approx z) \rightarrow x \approx z)$
4. For each n-ary predicate P,
   $\forall x_1 \dots x_n \forall y_1 \dots y_n ((x_1 \approx y_1 \& \dots \& x_n \approx y_n) \rightarrow (P(x_1, \dots, x_n) \leftrightarrow P(y_1, \dots, y_n)))$

Axioms 1-3 are called *Similarity Axioms* and the axiom 4 is called *Congruence Axiom*. These axioms can be expressed in the SBox using the ones of Table 2 with a truth degree equal to 1. In an SBox we can also define similarity of two objects with respect to their attributes. For this purpose we work with fuzzy concrete domains as in [3]. However, we try to give a more general account by dealing with a logic that can be translated in a natural way into a fragment of a many-sorted fuzzy predicate logic (see Hájek [11]).

We define a *fuzzy concrete domain* $\mathcal{D}$ as a set $\Delta^{\mathcal{D}}$ and a set of predicate names *pred*($\mathcal{D}$). Each $P \in$ *pred*($\mathcal{D}$) is interpreted as a function $P^{\mathcal{D}} : (\Delta^{\mathcal{D}})^n \rightarrow [0,1]$. Working with fuzzy concrete domains we can use predicate restrictions to define new roles in our SBox. More precisely, given two feature chains u and v and a predicate name $P \in$ *pred*($\mathcal{D}$), we can introduce the complex role $\exists(u,v).P$ interpreted as $(\exists(u,v).P)^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0,1]$, where for every $a, b \in \Delta^{\mathcal{I}}$, $(\exists(u,v).P)^{\mathcal{I}}(a,b) = P^{\mathcal{D}}(u^{\mathcal{I}}(a), v^{\mathcal{I}}(b))$. For instance, let us suppose that we are interested on defining the similarity between two robots in terms of the similarity between their bodies and their heads. First we define a fuzzy concrete domain, including a fuzzy similarity $\approx_f^{\mathcal{D}}$ in $\Delta^{\mathcal{D}} = \mathit{Forms} = \{$round, square, octagon$\}$. Then we can define a similarity $\approx_\rho$ between two robots in the following way:

$$\approx_\rho \equiv \exists(\text{body} \circ \text{has-form}, \text{body} \circ \text{has-form}).\approx_f \sqcap \exists(\text{head} \circ \text{has-form}, \text{head} \circ \text{has-form}).\approx_f$$

In other words, two robots are similar when they have both bodies with similar forms and heads with similar forms. Notice that this kind of expressions is very appropriated when domain objects are described as sets of attribute-value pairs, since they make easy the comparison of attributes between the two objects. In addition, because the similarity between shapes is fuzzy (for instance, we could assess the similarity between a round and an octagon as 0.8), we can assess the global similarity between two objects as the aggregation of the fuzzy similarities of the attributes describing those objects. In this direction, we plan as future work to analyze the aggregation of fuzzy similarities from the logical point of view.

A different approach is that token by Bobillo and Straccia [4] that present a fuzzy rough extension of $\mathcal{SROIQ}(\mathcal{D})$. The key idea in rough set theory is the approximation of a vague concept by means of a pair of concepts. Usually, this approximation is based on an equivalence relation between elements of the domain. In [4] they extend this idea using fuzzy similarity relations instead of equivalence relations, giving raise to fuzzy rough sets. We work also in a fuzzy extension of $\mathcal{SROIQ}(\mathcal{D})$, but our approach differs from theirs in different aspects. They focus in the combination of fuzzy DLs with fuzzy rough sets, while we are mainly interested in the integration of the treatment of similarities inside fuzzy DLs able to deal with attribute-value representations. Bobillo and Straccia use a fixed set of similarities in order to introduce the semantics for the upper and lower approximation constructors. For them, similarities are always reflexive, symmetric and transitive fuzzy relations. On the contrary, we are interested in the possibility of using fuzzy DL's in contexts in which some of these conditions (symmetry, transitivity...) are not needed meanwhile possibly other properties are required. Thus we need to analyze the expressive power of fuzzy $\mathcal{SROIQ}(\mathcal{D})$ and see how this language allows us to define similarities using axioms in the SBox. In the future we plan to combine both approaches.

3. Similarity between concepts

Once the similarity between objects has been established, it should be also interesting to analyze how to define similarity between concepts. There are several works that we want to analyze since although they take a logical approach different than ours, we think that some of their ideas can be included in our framework. For instance, Sheremet et al. in [19] propose an integration of logic-based and similarity-based approaches in classical DLs. They use concept constructors such as "in the r-neighborhood" of C' where r is a positive rational number; or the operator $C \leftleftarrows D$ which is interpreted by the set of all points in the similarity space that are closer to the instances of C than to the instances of D. For example, it can be used to model statements like 'X resembles C more than D'. In our formalism we can also express both, a notion of *neighbourhood of a concept* and a notion of *comparative* similarity between concepts (as in Sheremet et al. [19]). Given a concept C, and a similarity role $\approx$, by using existential quantification, we define the concept $\exists \approx .C$ interpreted as a fuzzy set in the following way: for every $d \in \Delta^{\mathcal{I}}$,

$$(\exists \approx .C)^{\mathcal{I}}(d) = \sup\{(b \approx^{\mathcal{I}} d) * C^{\mathcal{I}}(b) : b \in \Delta^{\mathcal{I}}\}$$

Thus, given a rational number r, the r-neighbourhood of concept C is the set of all $d \in \Delta^{\mathcal{I}}$ such that $(\exists \approx .C)^{\mathcal{I}}(d) \geqslant r$. Analogously, let C and D be two concepts. By using an implication concept constructor in the language interpreted as the residuum of a t-norm (see [12]), we define the concept $\exists \approx .C \rightarrow \exists \approx .D$ interpreted as a fuzzy set as follows: for every $d \in \Delta^{\mathcal{I}}$,

$$(\exists \approx .C \rightarrow \exists \approx .D)^{\mathcal{I}}(d) = (\exists \approx .C)^{\mathcal{I}}(d) \rightarrow_* (\exists \approx .D)^{\mathcal{I}}(d)$$

Now, the elements $d \in \Delta^{\mathcal{I}}$ which are more similar (or equally similar) to D than to C are those for which $(\exists \approx .C \rightarrow \exists \approx .D)^{\mathcal{I}}(d) = 1$.

Similarity in DLs has also been studied by Borgida et al. [5] and D'Amato et al. [8] among others, focusing on similarity measures between DL concepts. D'Amato et al. take as starting point the idea that measures for estimating concept similarity have to be able to appropriately consider concept semantics in order to correctly assess their similarity value. In accordance with this goal the authors propose a set of properties that a semantic similarity measure should have, analyze different extensional-based and intensional-based similarity measures proposed in the literature, and show that these approaches lack some of the needed properties. Finally, they define a measure for complex descriptions in some DL languages that is compliant with all of these criteria.

4. Concluding Remarks and Future Work

This work is a preliminary step on the direction of defining similarities inside a fuzzy description logic. In the paper we propose an extension of fuzzy DLs consisting on defining a SBox that includes definitions and properties of similarities. We provide some examples of use of that SBox in both the crisp and fuzzy framework. However, there are several interesting issues that have not been addressed in this paper and that will be the focus of future research.

On the one hand, we want to study further the notion of similarity between concepts. A possibility should be: since we allow similarity assertions about objects in the ABox it seems natural to define that two concepts C and D are similar when each object satisfying C is similar to some object satisfying D. This in fact can be done in terms of a notion of approximate subsumption of C in D: when every a satisfying C is similar to some b satisfying D. Then C and D are similar simply when C is approximately subsumed by D and vice-versa. In the future we plan to explore different notions of similarity between concepts.

On the other hand, another interesting topic is the study of the reasoning tasks in DLs involving similarities. In the fuzzy context it is useful to deal with arguments involving fuzzy concepts. For instance we could speak of the 'degree of homogeneity' of a robot and then guarantee that if robots r_1 and r_2 have a similarity degree $\geq s$ and the homogeneity degree of r_1 is $\geq s'$, then the homogeneity degree of r_2 is $\geq k$, where k is obtained from s and s' using the t-norm $*$.

Finally, we would like to obtain decidability and complexity results for the FDLs dealing with similarities we propose.

Acknowledgments

The authors thank Lluís Godo and the anonymous reviewers their useful comments for the improvement of the paper. Research partially funded by the Spanish MICINN projects MULOG2 (TIN 2007-68005-C04-01/04), ARINF (TIN2009-14704-C03-03), CONSOLIDER (CSD2007-0022), and ESF Eurocores-LogICCC/MICINN (FFI2008-03126-E/FILO), and the grants 2009-SGR-1433/ 1434 from the Generalitat de Catalunya.

References

[1] F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, and P.F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, New York, NY, USA, 2003.

[2] F. Baader and Ph. Hanschke. A scheme for integrating concrete domains into concept languages. In *Proceedings of IJCAI'91*, pages 452–457, San Francisco, CA, USA, 1991. Morgan Kaufmann Publishers Inc.

[3] F. Bobillo, M. Delgado, J. Gómez-Romero, and U. Straccia. Fuzzy description logics under Gödel semantics. *International Journal of Approximate Reasoning*, 50(3):494–514, 2009.

[4] F. Bobillo and U. Straccia. Supporting Fuzzy Rough Sets in Fuzzy Description Logics. In C. Sossai and G. Chemello, editors, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, volume 5590 of *Lecture Notes in Computer Science*, pages 676–687. Springer, 2009.

[5] A. Borgida, T. Walsh, and H. Hirsh. Towards measuring similarity in description logics. In *Procs. of the International Description Logics Workshop, Vol. 147 of CEUR*, 2005.

[6] R. Bělohlávek. *Fuzzy relational systems: foundations and principles*. Number 20 in IFSR International Series on Systems Science and Engineering. Kluwer Academic, 2002.

[7] T. Calvo, G. Mayor, and R. Mesiar, editors. *Aggregation operators: new trends and applications*. Physica-Verlag GmbH, Heidelberg, Germany, 2002.

[8] C. D'Amato, S. Staab, and N. Fanizzi. On the influence of description logics ontologies on conceptual similarity. In *EKAW '08: Proceedings of the 16th international conference on Knowledge Engineering*, pages 48–63, Berlin, Heidelberg, 2008. Springer-Verlag.

[9] F. Esteva, P. Garcia, and L. Godo. *Similarity-based reasoning*, volume 57 of *Stud. Fuzziness Soft Comput.*, pages 367–396. Physica, Heidelberg, 2000.

[10] À. García-Cerdaña, E. Armengol, and F. Esteva. Fuzzy description logics and t-norm based fuzzy logics. *International Journal of Approximate Reasoning*, 51(6):632–655, 2010.

[11] P. Hájek. *Metamathematics of Fuzzy Logic*, volume 4 of *Trends in Logic. Studia Logica Library*. Kluwer Academic Publishers, Dordrecht, 1998.

[12] P. Hájek. Making fuzzy description logic more general. *Fuzzy Sets and Systems*, 154(1):1–15, 2005.

[13] I. Horrocks, O. Kutz, and U. Sattler. The even more irresistible SROIQ. In *KR*, pages 57–67. AAAI Press, 2006.

[14] E. Hüllermeier. *Case-Based Approximate Reasoning*. Springer-Verlag, 2007.

[15] T. W. Liao, Z. Zhang, and C. Mount. Similarity measures for retrieval in case-based reasoning systems. *Applied Artificial Intelligence*, 12(4):267–288, 1998.

[16] T. Lukasiewicz and U. Straccia. Managing uncertainty and vagueness in description logics for the semantic web. *Journal of Web Semantics*, 6(4):291–308, 2008.

[17] J. Recasens. *Indistinguishability Operators. Modelling Fuzzy Equalities and Fuzzy Equivalence Relations*. Springer, to appear.

[18] E. H. Ruspini. On the semantics of fuzzy logics. *International Journal of Approximate Reasoning*, 5:45–88, 1991.

[19] M. Sheremet, D. Tishkovsky, F. Wolter, and M. Zakharyaschev. A logic for concepts and similarity. *Journal of Logic and Computation*, 17(3):415–452, 2007.

[20] A. Tversky. Features of similarity. *Psychological Review*, 84(4):327–352, July 1977.