Uncertainty Treatment in the Rule Interchange Format: From Encoding to Extension

Jidi Zhao[1], Harold Boley[2]

[1] Faculty of Computer Science, University of New Brunswick, Fredericton, NB, E3B 5AC, Canada
Judy.Zhao AT unb.ca

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

Abstract. The Rule Interchange Format (RIF) is an emerging W3C format that allows rules to be exchanged between rule systems. Uncertainty is an intrinsic feature of real world knowledge, hence it is important to take it into account when building logic rule formalisms. However, the set of truth values in the Basic Logic Dialect (RIF-BLD) currently consists of only two values (t and f). In this paper, we first present two techniques of encoding uncertain knowledge and its fuzzy semantics in RIF-BLD presentation syntax. We then propose an extension leading to an Uncertainty Rule Dialect (RIF-URD) to support a direct representation of uncertain knowledge. In addition, rules in Logic Programs (LP) are often used in combination with the other widely-used knowledge representation formalism of the Semantic Web, namely Description Logics (DL), in order to provide greater expressive power. To prepare DL as well as LP extensions, we present a fuzzy extension to Description Logic Programs (DLP), called Fuzzy DLP, and discuss its mapping to RIF. Such a formalism not only combines DL with LP, as in DLP, but also supports uncertain knowledge representation.

1. Introduction

Description Logics (DL) and Logic Programs (LP) are the two main categories of knowledge representation formalisms for the Semantic Web, both of which are based on subsets of first-order logic [1]. DL and LP cover different but overlapping areas of knowledge representation. They are complementary to some degree; for example, DL cannot express LP's n-ary function applications (complex terms) while LP cannot express DL's disjunctions (in the head). Combining DL with LP in order to "build rules on top of ontologies" or, "build ontologies on top of rules" has become an emerging topic for various applications of the Semantic Web. It is therefore important to research the combination of DL and LP with different strategies. There have been various achievements in this area, including several proposed combination frameworks [2-6]. As a minimal approach in this area, the Description Logic Program (DLP) 'intersection' of DL and LP has been studied, along with mappings from DL to LP [2]. Both [3] and [5] studied the combination of standard Datalog inference procedures with intermediate *ALC* DL satisfiability checking.

On the other hand, as evidenced by Fuzzy RuleML [7] and W3C's Uncertainty Reasoning for the World Wide Web (URW3) Incubator Group [8], handling uncertain knowledge is becoming a critical research direction for the (Semantic) Web. For example, many concepts needed in business ontology modeling lack well-defined boundaries or, precisely defined criteria of relationships with other concepts. To take care of these knowledge representation needs, different approaches for integrating uncertain knowledge into traditional rule languages and DL languages have been studied [1, 9-17].

The Rule Interchange Format (RIF) is being developed by W3C's Rule Interchange Format (RIF) Working Group to support the exchange of rules between rule systems [18]. In particular, the Basic Logic Dialect (RIF-BLD) [19] corresponds to the language of definite Horn rules with equality and a standard first-order semantics. While RIF's Framework for Logic-based Dialects (RIF-FLD) [20] permits multi-valued logics, the current version of RIF-BLD instantiates RIF-FLD with the set of truth values consisting of only two values, t and f, hence is not designed for expressing uncertain knowledge.

According to the final report from the URW3 Incubator group, uncertainty is a term intended to include different types of uncertain knowledge, including incompleteness, vagueness, ambiguity, randomness, and inconsistency [8]. Mathematical theories for representing uncertain knowledge include, but are not limited to, Probability, Fuzzy Sets, Belief Functions, Random Sets, Rough Sets, and combinations of several models (Hybrid). The uncertain knowledge representations and interpretations discussed in this paper are limited to Fuzzy set theory and Fuzzy Logic (a multi-valued logic based on Fuzzy set theory); other types of uncertainty will be studied in future work.

The main contributions of this paper are: (1) two techniques of encoding uncertain information in RIF as well as an uncertainty extension to RIF; (2) an extension of DLP to Fuzzy DLP and the mapping of Fuzzy DLP to RIF. Two earlier uncertainty extensions to the combination of DL and LP that we can expand on are [21] and [22]. While our approach allows DL atoms in the head of hybrid rules and DL subsumption axioms in hybrid rules, the approach of [21] excludes them. Our approach deals with fuzzy subsumption of fuzzy concepts of the form $C \sqsubseteq D \quad = c$ whereas [22] deals with crisp subsumption of fuzzy concepts of the form $C \sqsubseteq D$. Also, we do not limit hybrid knowledge bases to the intersection of (fuzzy) DL and (fuzzy) LP. We extend [22] and study the decidable union of DL and LP. In this paper, we only consider the Horn logic subset of LP.

The rest of this paper is organized as follows. Section 2 reviews earlier work on the interoperation between DL and LP in the intersection of these two formalisms (known as DLP) and represents the DL-LP mappings in RIF. Section 3 addresses the syntax and semantics of fuzzy Logic Programs, and then presents two techniques of bringing uncertainty into the current version of RIF presentation syntax (hence its semantics and XML syntax), using encodings as RIF functions and RIF predicates. Section 4 adapts the definition of the set of truth values in RIF-FLD for the purpose of representing uncertain knowledge directly, and proposes the new Uncertainty Rule Dialect (RIF-URD), extending RIF-BLD. Section 5 extends DLP to Fuzzy DLP, supporting mappings between fuzzy DL and fuzzy LP, and gives representations of Fuzzy DLP in RIF and RIF-URD. Finally, Section 6 summarizes our main results and gives an outlook on future research.

2. Description Logic Programs and Their Representation in RIF

In this section, we summarize the work on Description Logic Programs (DLP) [2] and then show how to represent the mappings between DL and LP in RIF presentation syntax.

The paper [2] studied the intersection between the leading Semantic Web approaches to rules in LP and ontologies in DL, and showed how to interoperate between DL and LP in the intersection known as DLP. A DLP knowledge base consists of axioms of the following kinds: $\underline{C} \sqsubseteq \underline{D}$, $\underline{C} \equiv \underline{D}$, $\top \sqsubseteq \forall \underline{R}.\underline{C}$,

$\top \sqsubseteq \forall \underline{R}^{-}.\underline{C}$, $\underline{R} \sqsubseteq \underline{P}$, $\underline{P} \equiv \underline{R}$, $\underline{P} \equiv \underline{R}^{-}$, $\underline{R}^{+} \sqsubseteq \underline{R}$, $\underline{C}(a)$ and $\underline{R}(a,b)$, where $\underline{C},\underline{D}$ are concepts, $\top$ is the universal concept, $\underline{P},\underline{R}$ are roles, $\underline{R}^{-}$ and $\underline{R}^{+}$ are the inverse role and the transitive role of $\underline{R}$, respectively, and a,b are individuals.

In RIF presentation syntax, the quantifiers Exists and Forall are made explicit, rules are written with a ":-" infix, variables start with a "?" prefix, and whitespace is used as a separator.

Table 1 summarizes the mappings in [2] between DL and LP in the DLP intersection, and shows its representation in RIF. In Table 1, C, D, C_1, C_2 are atomic concepts, P, R, R_1, R_2 are atomic roles, R^{-} and R^{+} are the inverse role and the transitive role of R , respectively, and T, a, b are defined as above. Note that in DLP, a complex concept expression which is a disjunction (e.g. $C_1 \sqcup C_2$) or an existential (e.g. $\exists R.C$) is not allowed in the right side of a concept subsumption axiom.

Table 1. Mapping between LP and DL

LP syntax	DL syntax	RIF
$D(x) \leftarrow C(x)$	$C \sqsubseteq D$	Forall ?x (D(?x) :- C(?x))
$D(x) \leftarrow C(x),$ $C(x) \leftarrow D(x)$	$C \equiv D$	Forall ?x (D(?x) :- C(?x)) Forall ?x (C(?x) :- D(?x))
$R(x,y) \wedge C(y)$	$\exists R.C$	Forall ?x (Exists ?y (And(R(?x ?y) C(?y))))
$C(y) \leftarrow R(x,y)$	$\top \sqsubseteq \forall R.C$	Forall ?x ?y (C(?y) :- R(?x ?y))
$C(x) \leftarrow R(x,y)$	$\top \sqsubseteq \forall R^{-}.C$	Forall ?x ?y (C(?x) :- R(?x ?y))
$C(a)$	$C(a)$	C(a)
$R(a,b)$	$R(a,b)$	R(a b)
$R(x,y) \leftarrow P(x,y),$ $P(x,y) \leftarrow R(x,y)$	$P \equiv R$	Forall ?x ?y (R(?x ?y) :- P(?x ?y)) Forall ?x ?y (P(?x ?y) :- R(?x ?y))
$R(x,y) \leftarrow P(y,x),$ $P(y,x) \leftarrow R(x,y)$	$P \equiv R^{-}$	Forall ?x ?y (R(?x ?y) :- P(?y ?x)) Forall ?x ?y (P(?y ?x) :- R(?x ?y))
$R(x,z) \leftarrow R(x,y), R(y,z)$	$R^{+} \sqsubseteq R$	Forall ?x ?y ?z (R(?x ?z) :- And(R(?x ?y) R(?y ?z)))
$P(x,y) \leftarrow R(x,y)$	$R \sqsubseteq P$	Forall ?x ?y (P(?x ?y) :- R(?x ?y))
$D(x) \leftarrow C_1(x) \wedge C_2(x)$	$C_1 \sqcap C_2 \sqsubseteq D$	Forall ?x (D(?x) :- And(C1(?x) C2(?x)))
$P(x,y) \leftarrow R_1(x,y) \wedge R_2(x,y)$	$R_1 \sqcap R_2 \sqsubseteq P$	Forall ?x ?y (P(?x ?y) :- And(R1(?x ?y) R2(?x ?y)))

3. Encoding Uncertainty in RIF

Fuzzy set theory was introduced in [23] as an extension of the classical notion of sets to capture the inherent vagueness (the lack of crisp boundaries) of real-world sets. Formally, a fuzzy set A with respect to a set of elements X (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 $x \in X$. $\mu_A(x)$ gives the degree to which an element x belongs to the set A . 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, t and f , as in classic predicate logic [24]. Such degrees can be computed based on various specific membership functions, for example, a trapezoidal function.

In this section, we first present the syntax and semantics for fuzzy Logic Programs based on Fuzzy Sets and Fuzzy Logic [23] and on previous work on fuzzy LP [15, 16, 25], and then propose two techniques of encoding the semantics of uncertain knowledge based on Fuzzy Logic in the presentation syntax of RIF-BLD using BLD functions and BLD predicates respectively.

3.1. Fuzzy Logic Programs

Rules in van Emden's formalism for fuzzy LP have the syntactic form

$$H \leftarrow_c B_1, \cdots, B_n \qquad (\mathbf{1})$$

where H, B_i are atoms, $n \geq 0$, and the factor c is a real number in the interval (0,1] [15]. For $n = 0$, such fuzzy rules degenerate to fuzzy facts.

The fuzzy LP language proposed by [16, 25] is a generalization of van Emden's work [15]. Rules are constructed from an implication ($\leftarrow$) with a corresponding t-norm adjunction operator (f_1), and another t-norm operator denoted by f_2. A t-norm is a generalization to the many-valued setting of the conjunction connective. In their setting, a rule is of the form $H \leftarrow_{f_1} f_2(B_1, \cdots, B_n)\ with-cf\ c$, where the confidence factor c is a real number in the unit interval [0,1] and H, B_i are atoms with truth values in (0, 1]. If we take the operator f_1 as the product following Goguen implication and the operator f_2 as the Gödel t-norm (minimum), this is exactly of the form by van Emden [15].

In the current paper, we follow this work and use the following form to represent a fuzzy rule.

$$H(\vec{x}) \leftarrow B_1(\vec{x}_1), \cdots, B_n(\vec{x}_n) \quad / c \qquad (\mathbf{2})$$

Here $H(\vec{x}), B_i(\vec{x}_i)$ are atoms, $\vec{x}, \vec{x}_i$ are vectors of variables or constants, $n \geq 0$ and the confidence factor c (also called certainty degree) is a real number in the interval (0,1]. For the special case of fuzzy facts this becomes $H \quad / c$. These forms with a "/" symbol have the advantages of avoiding possible confusion with the equality symbol usually used for functions in logics with equality, as well as using a unified and compact format to represent fuzzy rules and fuzzy facts.

The semantics of such fuzzy LP is an extension of classical LP semantics. Let B_R stand for the Herbrand base of a fuzzy knowledge base KB_{LP}. A fuzzy Herbrand interpretation H_I for KB_{LP} is defined as a mapping $B_R \rightarrow [0,1]$. It is a fuzzy subset of B_R under Zadeh's semantics and can be specified by a function val with two arguments: a variable-free atom H (or $B_1, \cdots, B_n$) and a fuzzy Herbrand interpretation H_I. The returned result of the function val is the membership value of H (or $B_1, \cdots, B_n$) under H_I, denoted as $val(H, H_I)$ (or $val(B_i, H_I)$).

Therefore, a fuzzy knowledge base KB_{LP} is true under H_I iff every rule in KB_{LP} is true under H_I. Such a Herbrand interpretation H_I is called a Herbrand model of KB_{LP}. Furthermore, a rule is true under H_I iff each variable-free instance of this rule is true under H_I. A variable-free instance of a rule (**3**) is true under H_I iff $val(H, H_I) \geq c \times \min\{val(B_i, H_I) \mid i \in \{1, \cdots, n\}\}$ ($\min\{\} = 1\ if\ n = 0$). In other words, such an interpretation can be separated into the following two parts [26-28].

(1) The body of the rule consists of n atoms. Our confidence that all these atoms are true is interpreted under Gödel's semantics for fuzzy logic:

$val((B_1, \cdots, B_n), H_I) = \min\{val(B_i, H_I) \mid i \in \{1, \cdots, n\}\}$

(2) The implication is interpreted as the product:

$val(H, H_I) = c \times val((B_1, \cdots, B_n), H_I)$

For a fuzzy knowledge base KB_{LP}, the reasoning task is a fuzzy entailment problem written as $KB_{LP} \models H \quad / c$ ($H \in B_R, c \in (0,1]$).

Example 3.1. Consider the following fuzzy LP knowledge base:

$cheapFlight(x,y) \leftarrow affordableFlight(x,y) \quad / \, 0.9$ (1)
$affordableFlight(x,y) \quad / \, left\_shoulder0k4k1k3k(y)$ (2)

Fig. 1 shows the left__shoulder membership function $left\_shoulder(0,4000,1000,3000)$. We use the name $left\_shoulder0k4k1k3k$ for this parameterization. The function has the mathematical form

$$left\_shoulder0k4k1k3k(y) = \begin{cases} 1 & 0 \le y \le 1000 \\ -0.0005y + 1.5 & 1000 < y \le 3000 \\ 0 & 3000 < y \le 4000 \end{cases}.$$

For example, the certainty degree computed by this function for the fact $affordableFlight(flight0001,1800)$ is 0.7.

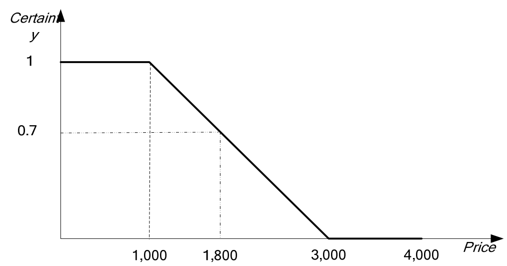

Fig. 1. A Left__shoulder Membership Function

Applying the semantics we discussed, $val(cheapFlight(flight0001,1800), H_I) = 0.9 * 0.7 = 0.63$, so we have that $KB_{LP} \models cheapFlight(flight0001,1800) \quad / \, 0.63$.

Example 3.2. Consider the following fuzzy LP knowledge base:

$A(x) \leftarrow B(x), C(x) \quad / \, 0.5$ (1)
$C(x) \leftarrow D(x) \quad / \, 0.5$ (2)
$B(d) \quad / \, 0.5$ (3)
$D(d) \quad / \, 0.8$ (4)

We have that $KB_{LP} \models A(d) \quad / \, 0.2$. The reasoning steps of example 3.2 are described as follows:

$$\begin{aligned} val(A(d), H_I) &= 0.5 \times \min(val(B(d), H_I), val(C(d), H_I)) && \text{**}\textit{according to } (1) \\ &= 0.5 \times \min(val(B(d), H_I), 0.5 \times val(D(d), H_I)) && \text{**}\textit{according to } (2) \\ &= 0.5 \times \min(0.5, 0.5 \times val(D(d), H_I)) && \text{**}\textit{according to } (3) \\ &= 0.5 \times \min(0.5, 0.5 \times 0.8) && \text{**}\textit{according to } (4) \\ &= 0.5 \times 0.4 \\ &= 0.2 \end{aligned}$$

3.2. Encoding Uncertainty Using RIF Functions

One technique to encode uncertainty in logics with equality such as the current RIF-BLD (where equality in the head is "At Risk") is mapping all predicates to functions and using equality for letting them return uncertainty values [29]. We assume that H, B_i of the fuzzy rule of equation (2) from Section 3.1 contain variables in {$?x_1$, …, $?x_k$} and that the head and body predicates are applied to terms t_1 … t_r and $t_{j,1}$ … $t_{j,sj}$ ($1 \le j \le n$) respectively, which can all be variables, constants or complex terms. A fuzzy rule in the form of equation (2) from Section 3.1 can then be represented in RIF-BLD as (for simplicity, we will omit prefix declarations)

```
Document(
   Group
   (      Forall ?x1 … ?xk (
             h(t1 … tr)=?ch :- And(b1(t1,1 … t1,s1)=?c1  ⋯  bn(tn,1 … tn,sn)=?cn
                                   ?ct =External(numeric-minimum(?c1  ⋯  ?cn))
                                   ?ch=External(numeric-multiply(c ?ct)) )
   )    )
```

Each predicate in the fuzzy rule thus becomes a function with a return value between 0 and 1. The semantics of the fuzzy rules is encoded in RIF-BLD using the built-in functions numeric-multiply from RIF-DTB[30] and an aggregate function numeric-minimum proposed here as an addition to RIF-DTB (this could also be defined using rules).

A fact of the form $H \quad / c$ can be represented in RIF-BLD presentation syntax as

```
h(t1 … tr)=c
```

Example 3.3 We can rewrite example 3.1 using RIF functions as follows:

```
(* <http://example.org/fuzzy/membershipfunction > *)
Document(
  Group
  (      (* "Definition of membership function  left_shoulder(0,4000,1000,3000) "[] *)
     Forall ?y(
           left__shoulder0k4k1k3k(?y)=1 :- And(External(numeric-less-than-or-equal(0 ?y))
                                              External(numeric-less-than-or-equal(?y 1000))))
     Forall ?y(
     left__shoulder0k4k1k3k(?y)=External(numeric-add(External(numeric-multiply(-0.0005 ?y)) 1.5))
           :- And(External(numeric-less-than(1000 ?y))
                 External(numeric-less-than-or-equal(?y 3000))))
     Forall ?y(
           left__shoulder0k4k1k3k(?y)=0 :- And(External(numeric-less-than(3000 ?y))
                                              External(numeric-less-than-or-equal(?y 4000))))
      . .
  )    )
```

Note that membership function $left\_shoulder(0,4000,1000,3000)$ is encoded as three rules.

```
Document(
   Import (<http://example.org/fuzzy/membershipfunction >)
   Group
   (      Forall ?x ?y(
         cheapFlight(?x ?y)=?ch :- And(affordableFlight(?x ?y)=?c1
                                      ?ch=External(numeric-multiply(0.4 ?c1))))
      Forall ?x ?y(affordableFlight(?x ?y)=left__shoulder0k4k1k3k(?y))
   )    )
```

The Import statement loads the left__shoulder0k4k1k3k function defined at the given "<…>" IRI.

Example 3.4 We can rewrite example 3.2 in RIF functions as follows:

```
Document(
   Group
   (      Forall ?x(
             A(?x)=?ch :- And(B(?x)=?c1 C(?x)=?c2
                              ?ct =External(numeric-minimum(?c1 ?c2))
                              ?ch=External(numeric-multiply(0.5 ?ct))))
        Forall ?x(
             C(?x)= ?ch :- And(D(?x)=?c1 ?ch=External(numeric-multiply(0.5 ?c1))) )
             B(d)=0.5
             D(d)=0.8
   )    )
```

3.3 Encoding Uncertainty Using RIF Predicates

Another encoding technique is making all n-ary predicates into (1+n)-ary predicates, each being functional in the first argument which captures the certainty factor of predicate applications. A fuzzy rule in the form of equation (2) from Section 3.1 can then be represented in RIF-BLD as

```
Document(
   Group
   (       Forall ?x_1 … ?x_k (
              h(?c_h t_1 … t_r) :- And(b_1(?c_1 t_1,1 … t_1,s1)  ⋯  b_n(?c_n t_n,1 … t_n,sn)
                                         ?c_t =External(numeric-minimum(?c_1  ⋯  ?c_n))
                                         ?c_h=External(numeric-multiply(c ?c_t)) )
   )       )
```

Likewise, a fact of the form $H \quad / c$ can be represented in RIF-BLD as

```
h(c t_1 … t_r)
```

Example 3.5 We can rewrite example 3.1 in RIF predicates as follows,

```
Document(
   Import (<http://example.org/fuzzy/membershipfunction >)
   Group
   (
      Forall ?x ?y(
         cheapFlight(?c_h ?x ?y) :- And(affordableFlight(?c_1 ?x ?y)
                                          ?c_h=External(numeric-multiply(0.4 ?c_1)))
      )
      Forall ?x ?y(affordableFlight(?c_1 ?x ?y) :- ?c_1 =left__shoulder0k4k1k3k(?y))
   )       )
```

4. Uncertainty Extension of RIF

In this section, we adapt the definition of the set of truth values from RIF-FLD and its semantic structure. We then propose a RIF extension for directly representing uncertain knowledge.

4.1 Definition of Truth Values and Truth Valuation

In previous sections, we showed how to represent the semantics of fuzzy LP with RIF functions and predicates in RIF presentation syntax. We now propose to introduce a new dialect for RIF, RIF Uncertainty Rule Dialect (RIF-URD), so as to directly represent uncertain knowledge and extend the expressive power of RIF.

The set TV of truth values in RIF-BLD consists of just two values, t and f. This set forms a two-element Boolean algebra with $t=1$ and $f=0$. However, in order to represent uncertain knowledge, all intermediate truth values must be allowed. Therefore, the set TV of truth values is extended to a set with infinitely many truth values ranging between 0 and 1. Our uncertain knowledge representation is specifically based on Fuzzy Logic, thus a member function maps a variable to a truth value in the 0 to 1 range.

Definition 1. (Set of truth values as a specialization of the set in RIF-FLD). In RIF-FLD, $\le_t$ denotes the truth order, a binary relation on the set of truth values TV. Instantiating RIF-FLD, which just requires a partial order, the set of truth values in RIF-URD is equipped with a total order over the 0 to 1 range. In RIF-URD, we specialize $\le_t$ to $\le$, denoting the numerical truth order. Thus, we observe that the following statements hold for any element e_i, e_j or e_k in the set of truth values TV in the 0 to 1 range, justifying to write it as the interval [0,1].

(1) The set TV is a complete lattice with respect to $\le$, i.e., the least upper bound (lub) and the greatest lower bound (glb) exist for any subset of $\le$.

(2) Antisymmetry. If $e_i \le e_j$ and $e_j \le e_i$ then $e_i = e_j$.

(3) Transitivity. If $e_i \le e_j$ and $e_j \le e_k$ then $e_i \le e_k$.

(4) Totality. Any two elements should satisfy one of these two relations: $e_i \le e_j$ or $e_j \le e_i$.

(5) The set TV has an operator of negation, $\sim: TV \to TV$, such that

a). $\sim e_i = 1 - e_i$.

b). ~ is self-inverse, i.e., $\sim\sim e_i = e_i$.

Let $TVal(\varphi)$ denote the truth value of a non-document formula, φ, in RIF-BLD. $TVal(\varphi)$ is a mapping from the set of all non-document formulas to TV , I denotes an interpretation, and c is a real number in the interval (0,1].

Definition 2. **(*Truth valuation adapted from RIF-FLD*)**. Truth valuation for well-formed formulas in RIF-URD is determined as in RIF-FLD, adapting the following three cases.

(8) Conjunction (glb_t becomes min): $TVal_I(And(B_1 \cdots B_n)) = \min(TVal(B_1) \cdots TVal(B_n))$.

(9) Disjunction (lub_t becomes max): $TVal_I(Or(B_1 \cdots B_n)) = \max(TVal(B_1) \cdots TVal(B_n))$

(11) Rule implication (t becomes 1, f becomes 0, condition valuation is multiplied with c):

$TVal_I(conclusion :- condition \quad / c) = 1$ if $TVal_I(conclusion) \geq c \times TVal_I(condition)$

$TVal_I(conclusion :- condition \quad / c) = 0$ if $TVal_I(conclusion) < c \times TVal_I(condition)$

4.2 Using RIF-URD to Represent Uncertain Knowledge

A fuzzy rule in the form of equation (2) from Section 3.1 can be directly represented in RIF-URD as

```
Document(
   Group
   (
      Forall ?x1 … ?xk (
            h(t1 … tr) :- And(b1(t1,1 … t1,s1)  ⋯  bn(tn,1 … tn,sn))
      )   / c
     )
```

Likewise, a fact of the form $H \quad / c$ can be represented in RIF-URD as

```
h(t1 … tr)    / c
```

Such a RIF-URD document of course cannot be executed by an ordinary RIF-compliant reasoner. RIF-URD-compliant reasoners will need to be extended to support the above semantics and the reasoning process shown in Section 3.1.

Example 3.6 We can directly represent example 3.1 in RIF-URD as follows:

```
Document(
   Import (<http://example.org/fuzzy/membershipfunction >)
   Group
   (
      Forall ?x ?y(
            cheapFlight(?x ?y) :- affordableFlight(?x ?y)
      )   / 0.4
      Forall ?x ?y(affordableFlight(?x ?y))    / left__shoulder0k4k1k3k(?y)
   )      )
```

5. Fuzzy Description Logic Programs and Their Representation in RIF

In this section, we extend Description Logic Programs (DLP) [2] to support mappings between fuzzy DL and fuzzy LP; we also show how to represent such mappings in RIF-BLD and RIF-URD based on the three uncertainty treatment methods addressed in previous sections.

Based on Fuzzy Sets and Fuzzy Logic [23], the semantics for fuzzy DL [12] and fuzzy LP [15], as well as the previous work cited in Section 1 and 3, we extend the work on Description Logic Programs (DLP) [2] to fuzzy Description Logic Programs (Fuzzy DLP).

Since DL is a subset of FOL, it can also be seen in terms of that subset of FOL, where individuals are equivalent to FOL constants, concepts and concept descriptions are equivalent to FOL formulas with one free variable, and roles and role descriptions are equivalent to FOL formulas with two free variables.

A concept inclusion axiom of the form $C \sqsubseteq D$ is equivalent to an FOL sentence of the form $\forall x.C(x) \rightarrow D(x)$, i.e. an FOL implication. In uncertainty representation and reasoning, it is important to represent and compute the degree of subsumption between two fuzzy concepts, i.e., the degree of overlap, in addition to crisp subsumption. Therefore, we consider fuzzy axioms of the form $C \sqsubseteq D \quad = c$ generalizing the crisp $C \sqsubseteq D$. The above equivalence leads to a straightforward mapping from a fuzzy concept inclusion axiom of the form $C \sqsubseteq D = c$ ($c \in (0,1]$) to an LP rule as follows: $D(x) \leftarrow C(x) \quad / c$.

The intersection of two fuzzy concepts in fuzzy DL is defined as $(C_1 \sqcap C_2)^I(x) = \min(C_1{}^I(x), C_2{}^I(x))$; therefore, a fuzzy concept inclusion axiom of the form $C_1 \sqcap C_2 \sqsubseteq D \quad = c$ including the intersection of C_1 and C_2 can be transformed to an LP rule $D(x) \leftarrow C_1(x), C_2(x) \quad / c$. Here the certainty degree of (variable-free) instantiations of the atom $D(x)$ is defined by the valuation $val(D, H_I) = c \times \min\{val(C_i, H_I) \mid i \in \{1,2\}\}$. It is easy to see that such a fuzzy concept inclusion axiom can be extended to include the intersection of n concepts ($n > 2$).

Similarly, a role inclusion axiom of the form $R \sqsubseteq P$ is equivalent to an FOL sentence consisting of an implication between two roles. Thus we map a fuzzy role inclusion axiom of the form $R \sqsubseteq P = c$ ($c \in (0,1]$) to a fuzzy LP rule as $P(x, y) \leftarrow R(x, y) \quad / c$. Moreover, $\bigcap_{i=1}^{n} R_i \sqsubseteq P = c$ can be transformed to $P(x, y) \leftarrow R_1(x, y), \cdots, R_n(x, y) \quad / c$.

A concept equivalence axiom of the form $C \equiv D$ can be represented as a symmetrical pair of FOL implications: $\forall x.C(x) \rightarrow D(x)$ and $\forall x.D(x) \rightarrow C(x)$. Therefore, we map the 'fuzzified' equivalence axiom $C \equiv D \quad = c$ into $C(x) \leftarrow D(x) \quad / c$ and $D(x) \leftarrow C(x) \quad / c$ ($c \in (0,1]$). As later examples show, such mappings in hybrid knowledge bases are directed from rules to DL expressions, hence if we have two rules of the forms $C(x) \leftarrow D(x) \quad / c_1$ and $D(x) \leftarrow C(x) \quad / c_2$ ($c_1, c_2 \in (0,1]$), they are mapped to a DL expression as $C \equiv D \quad = c$ with $c = \min(c_1, c_2)$. Similarly, we map two rules $R(x, y) \leftarrow P(x, y) \quad / c_1$ and $P(x, y) \leftarrow R(x, y) \quad / c_2$ into a role equivalence axiom of the form $R \equiv P \quad = \min(c_1, c_2)$, as well as two rules $R(x, y) \leftarrow P(y, x) \quad / c_1$ and $P(y, x) \leftarrow R(x, y) \quad / c_2$ into an inverse role equivalence axiom of the form $P \equiv R^- \quad = \min(c_1, c_2)$.

A DL assertion $C(a)$ (respectively, $R(a,b)$) is equivalent to an FOL atom of the form $C(a)$ (respectively, $R(a,b)$), where a and b are individuals. Therefore, a fuzzy DL concept-individual assertion of the form $C(a) = c$ corresponds to a ground fuzzy atom $C(a) \quad / c$ in fuzzy LP, while a fuzzy DL role-individual assertion of the form $R(a,b) = c$ corresponds to a ground fuzzy fact $R(a,b) \quad / c$.

Table 2 summarizes the mappings in Fuzzy DLP. For simplicity, in Fuzzy DLP as defined in this paper we do not use fuzzy forms for all of DLP, excluding value restrictions and transitive role axiom, and assuming $c = 1$ whenever $/ c$ is omitted.

Table 2. Representing Fuzzy DLP in RIF

LP syntax	$D(x) \leftarrow C_1(x), \cdots, C_n(x) \quad / c$
DL syntax	$\bigcap_{i=1}^{n} C_i \sqsubseteq D \quad = c$
RIF function	Forall ?x(D(?x)=?c_h :- And(C_1(?x)=?c_1 ⋯ C_n(?x)=?c_n ?c_t =External(numeric-minimum(?c_1 ⋯ ?c_n)) ?c_h=External(numeric-multiply(c ?c_t)))
RIF predicate	Forall ?x(D(?c_h ?x) :- And(C_1(?c_1 ?x) ⋯ C_n(?c_n ?x) ?c_t =External(numeric-minimum(?c_1 ⋯ ?c_n)) ?c_h=External(numeric-multiply(c ?c_t)))

RIF-URD	Forall ?x(D(?x) :- And(C_1(?x) $\cdots$ C_n(?x))) / c
LP syntax	$P(x,y) \leftarrow R_1(x,y),\cdots,R_n(x,y) \quad / c$
DL syntax	$\bigcap_{i=1}^{n} R_i \sqsubseteq P \quad = c$
RIF function	Forall ?x ?y(P(?x ?y)=?c_h :- And(R_1(?x ?y)=?c_1 $\cdots$ R_n(?x ?y)=?c_n ?c_t =External(numeric-minimum(?c_1 $\cdots$?c_n)) ?c_h=External(numeric-multiply(c ?c_t)))
RIF predicate	Forall ?x ?y(P(?c_h ?x ?y) :- And(R_1(?c_1 ?x ?y) $\cdots$ R_n(?c_n ?x ?y) ?c_t =External(numeric-minimum(?c_1 $\cdots$?c_n)) ?c_h=External(numeric-multiply(c ?c_t)))
RIF-URD	Forall ?x ?y(P(?x ?y) :- And(R_1(?x ?y) $\cdots$ R_n(?x ?y))) / c
LP syntax	$C(x) \leftarrow D(x) \quad / c, \quad D(x) \leftarrow C(x) \quad / c'$
DL syntax	$C \equiv D \;\; = \min(c, c')$
RIF function	Forall ?x(C(?x)=?c_h :- And(D(?x)=c_1 ?c_h=External(numeric-multiply(c c_1))) Forall ?x(D(?x)=?c_h :- And(C(?x)=c_1 ?c_h=External(numeric-multiply(c' c_1)))
RIF predicate	Forall ?x(C(?c_h ?x) :- And(D(?c_1 ?x) ?c_h=External(numeric-multiply(c c_1))) Forall ?x(D(?c_h ?x) :- And(C(?c_1 ?x) ?c_h=External(numeric-multiply(c' c_1)))
RIF-URD	Forall ?x(C(?x) :- D(?x)) / c, Forall ?x(D(?x) :- C(?x)) / c
LP syntax	$R(x,y) \leftarrow P(x,y) \quad / c, \quad P(x,y) \leftarrow R(x,y) \quad / c'$
DL syntax	$R \equiv P \;\; = \min(c, c')$
RIF function	Forall ?x ?y(R(?x ?y)=?c_h :- And(P(?x ?y)=c_1 ?c_h=External(numeric-multiply(c c_1))) Forall ?x ?y(P(?x ?y)=?c_h :- And(R(?x ?y)=c_1 ?c_h=External(numeric-multiply(c' c_1)))
RIF predicate	Forall ?x ?y(R(?c_h ?x ?y) :- And(P(?c_1 ?x ?y) ?c_h=External(numeric-multiply(c c_1))) Forall ?x ?y(P(?c_h ?x ?y) :- And(R(?c_1 ?x ?y) ?c_h=External(numeric-multiply(c' c_1)))
RIF-URD	Forall ?x ?y(R(?x ?y) :- P(?x ?y)) / c, Forall ?x ?y(P(?x ?y) :- R(?x ?y)) / c
LP syntax	$R(x,y) \leftarrow P(y,x) \quad / c, \quad P(y,x) \leftarrow R(x,y) \quad / c'$
DL syntax	$P \equiv R^{-} \;\; = \min(c, c')$
RIF function	Forall ?x ?y(R(?x ?y)=?c_h :- And(P(?y ?x)=c_1 ?c_h=External(numeric-multiply(c c_1))) Forall ?x ?y(P(?y ?x)=?c_h :- And(R(?x ?y)=c_1 ?c_h=External(numeric-multiply(c' c_1)))
RIF predicate	Forall ?x ?y(R(?c_h ?x ?y) :- And(P(?c_1 ?y ?x) ?c_h=External(numeric-multiply(c c_1))) Forall ?x ?y(P(?c_h ?y ?x) :- And(R(?c_1 ?x ?y) ?c_h=External(numeric-multiply(c' c_1)))
RIF-URD	Forall ?x ?y(R(?x ?y) :- P(?y ?x)) / c, Forall ?x ?y(P(?y ?x) :- R(?x ?y)) / c

LP syntax	$C(a) \quad / c$	$R(a,b) \quad / c$
DL syntax	$C(a) \quad = c$	$R(a,b) \quad = c$
RIF function	C(a)=c	R(a b)=c
RIF predicate	C(c a)	R(c a b)
RIF-URD	C(a) /c	R(a b) /c

In summary, Fuzzy DLP is an extension of Description Logic Programs supporting the following concept and role inclusion axioms, range and domain axioms, concept and role assertion axioms to build a knowledge base: $\bigcap_{i=1}^{n} C_i \sqsubseteq D \;\; = c$, $C \equiv D \;\; = c$, $\top \sqsubseteq \forall R.C$, $\top \sqsubseteq \forall R^{-}.C$,

$\bigcap_{i=1}^{n} R_i \sqsubseteq P \quad = c$, $P \equiv R \quad = c$, $P \equiv R^{-} \quad = c$, $R^{+} \sqsubseteq R$, $C(a) \quad = c$, and $R(a,b) \quad = c$, where $C, D, C_1, \cdots C_n$ are atomic concepts, P, R are atomic roles, a, b are individuals, $c \in (0,1]$ and $n \geq 1$. Notice that the crisp DLP axioms in DLP are special cases of their counterparts in Fuzzy DLP. For example, $C \sqsubseteq D$ is equal to its fuzzy version $\bigcap_{i=1}^{n} C_i \sqsubseteq D \quad = c$ for $n = 1$ and $c = 1$.

In previous sections, we presented two techniques of encoding uncertainty in RIF and proposed a method based on an extension of RIF for uncertainty representation. Subsequently, we also showed how to represent Fuzzy DLP in RIF-BLD and RIF-URD in Table 2.

Layered on Fuzzy DLP, we can build fuzzy hybrid knowledge bases in order to build fuzzy rules on top of ontologies for the Semantic Web and reason on such KBs.

Definition 3. A fuzzy hybrid knowledge base KB_{hf} is a pair $< K_{DL}, K_{LP} >$, where K_{DL} is the finite set of (fuzzy) concept inclusion axioms, role inclusion axioms, and concept and role assertions of a decidable DL defining an ontology. K_{LP} consists of a finite set of (fuzzy) hybrid rules and (fuzzy) facts.

A hybrid rule r in K_{LP} is of the following generalized form (we use the BNF choice bar, |):

$$\left(H(\vec{y}) \,\middle|\, \& H(\vec{z})\right) \leftarrow B_1(\vec{y}_1), \cdots, B_l(\vec{y}_l), \& Q_1(\vec{z}_1), \cdots, \& Q_n(\vec{z}_n) \quad / c \qquad \textbf{(4)}$$

Here, $H(\vec{y}), H(\vec{z}), B_i(\vec{y}_i), Q_j(\vec{z}_j)$ are atoms, $\&$ precedes a DL atom, $\vec{y}, \vec{z}, \vec{y}_i, \vec{z}_j$ are vectors of variables or constants, where $\vec{y}$ and each $\vec{y}_i$ have arbitrary lengths, $\vec{z}$ and each $\vec{z}_j$ have length 1 or 2, and $c \in (0,1]$. Also, $\&$ atoms and $/c$ degrees are optional (if all $\&$ atoms and $/c$ degrees are missing from a rule, it becomes a classical rule of Horn Logic).

Such a fuzzy hybrid rule must satisfy the following constraints:

(1) H is either a DL predicate or a rule predicate ($H \in \sum_T \bigcup \sum_R$). H is a DL predicate with the form $\&H$, while it is a rule predicate without the $\&$ operator.

(2) Each B_i $(1 < i \leq l)$ is a rule predicate ($B_i \in \sum_R$), and $B_i(y_i)$ is an LP atom.

(3) Each Q_j $(1 < j \leq n)$ is a DL predicate ($Q_j \in \sum_T$), and $Q_j(z_j)$ is a DL atom.

(4, pure DL rule) If a hybrid rule has head $\&H$, then each atom in the body must be of the form $\&Q_j$ $(1 < j \leq n)$; in other words, there is no B_i ($l = 0$). A head $\&H$ without a body ($l = 0$, $n = 0$) constitutes the special case of a pure DL fact.

Example 5.1. The rule $\&\mathit{CheapFlight}(x,y) \leftarrow \mathit{AffordableFlight}(x,y) \quad / c$ is not a pure DL rule according to (4), hence not allowed in our hybrid knowledge base, while $\mathit{CheapFlight}(x,y) \leftarrow \&\mathit{AffordableFlight}(x,y) \quad / c$ is allowed.

A hybrid rule of the form $\&\mathit{CheapFlight}(x,y) \leftarrow \&\mathit{AffordableFlight}(x,y) \quad / c$ can be mapped to a fuzzy DL role subsumption axiom $\mathit{AffordableFlight} \sqsubseteq \mathit{CheapFlight} \quad = c$.

Our approach thus allows DL atoms in the head of hybrid rules which satisfy the constraint (4, pure DL rule), supporting the mapping of DL subsumption axioms to rules. We also deal with fuzzy subsumption of fuzzy concepts of the form $C \sqsubseteq D \quad = c$ as shown in Example 5.1.

An arbitrary hybrid knowledge base cannot be fully embedded into the knowledge representation formalism of RIF with uncertainty extensions. However, in the proposed Fuzzy DLP subset, DL components (DL axioms in LP syntax) can be mapped to LP rules and facts in RIF. A RIF-compliant reasoning engine can be extended to do reasoning on a hybrid knowledge base on top of Fuzzy DLP by adding a module that first maps atoms in rules to DL atoms, and then derives the reasoning answers with a DL reasoner, e.g. Racer or Pellet, or with a fuzzy DL reasoner, e.g. fuzzyDL [31]. The specification of such a reasoning algorithm for a fuzzy hybrid knowledge base KB_{hf} based on Fuzzy DLP and a query q is treated in a companion paper[32].

6. Conclusion

In this paper, we propose two different principles of representing uncertain knowledge, encodings in RIF-BLD and an extension leading to RIF-URD. We also present a fuzzy extension to Description Logic Programs, namely Fuzzy DLP. We address the mappings between fuzzy DL and fuzzy LP within Fuzzy DLP, and give Fuzzy DLP representations in RIF. Since handling uncertain information, such as with fuzzy logic, was listed as a RIF extension in the RIF Working Group Charter [18] and RIF-URD is a manageable extension to RIF-BLD, we propose here a version of URD as a RIF dialect, realizing a fuzzy rule sublanguage for the RIF standard.

Our fuzzy extension directly relates to Lotfi Zadeh's semantics of fuzzy sets and fuzzy logic. We do not yet cover here other researchers' semantics, for example, Jan Lukasiewicz's. Nor do we cover other uncertainty formalisms, based on probability theory, possibilities, or rough sets. Future work will include generalizing our fuzzy extension of hybrid knowledge bases to some of these different kinds of uncertainty, and parameterizing RIF-URD to support different theories of uncertainty in a unified manner.

Complementing the RIF-URD presentation syntax, XML elements and attributes like <degree>, @mapkind, and @kind, following those of Fuzzy RuleML, can be introduced for the RIF-URD XML syntax. Another direction of future work would be the extension of uncertain knowledge to various combination strategies of DL and LP without being limited to DL queries.

References

1. D. Koller, A. Levy and A. Pfeffer, "P-CLASSIC: A tractable probabilistic description logic," in *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97),* 1997, pp. 390-397.
2. B. N. Grosof, I. Horrocks, R. Volz and S. Decker, "Description logic programs: Combining logic programs with description logic," in *Proceedings of the 12th International Conference on World Wide Web,* 2003, pp. 48-57.
3. F. M. Donini, M. Lenzerini, D. Nardi and A. Schaerf, "AL-log: Integrating Datalog and Description Logics," *Journal of Intelligent Information Systems,* vol. 10, pp. 227-252, 1998.
4. I. Horrocks, P. Patel-schneider, S. Bechhofer and D. Tsarkov, "OWL Rules: A Proposal and Prototype Implementation," *Journal of Web Semantics,* vol. 3, pp. 23-40, 2005.
5. J. Mei, Z. Q. Lin, H. Boley, J. Li and V. C. Bhavsar, "The Datalog^{DL} Combination of Deduction Rules and Descriprition Logics," *Computational Intelligence,* vol. 23, pp. 356-372, 2007.
6. B. Motik, U. Sattler and R. Studer, "Query Answering for OWL-DL with rules," *Web Semantics: Science, Services and Agents on the World Wide Web,* vol. 3, pp. 41-60, 7. 2005.
7. C. V. Damasio, J. Pan, G. Stoilos and U. Straccia, "Representing Uncertainty in RuleML," *Fundamenta Informaticae,* vol. 82, pp. 1-24, 2008.
8. K. Laskey, K. Laskey, P. Costa, M. Kokar, T. Martin and T. Lukasiewicz, "W3C incubator group report," W3C, Tech. Rep. http://www.w3.org/2005/Incubator/urw3/wiki/DraftFinalReport, 05 March, 2008.
9. M. Jaeger, "Probabilistic reasoning in terminological logics," in *Proc. of the 4th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'94),* 1994, pp. 305-316.
10. U. Straccia, "A fuzzy description logic," in *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI'98),* 1998, pp. 594-599.
11. U. Straccia, "Reasoning within Fuzzy Description Logics," *Journal of Artificial Intelligence Research,* vol. 14, pp. 137-166, 2001.
12. U. Straccia, "Towards a fuzzy description logic for the semantic web (preliminary report)," in *2nd European Semantic Web Conference (ESWC-05),* 2005, pp. 167-181.

13. G. Stoilos, G. Stamou, J. Pan, V. Tzouvaras and I. Horrocks, "Reasoning with Very Expressive Fuzzy Description Logics," *Journal of Artificial Intelligence Research,* vol. 30, pp. 273-320, 2007.

14. T. Lukasiewicz, "Expressive probabilistic description logics," *Artificial Intelligence,* vol. 172, pp. 852-883, 2008.

15. van Emden, M. H., "Quantitative Deduction and its Fixpoint Theory," *Journal of Logic Programming,* vol. 30, pp. 37-53, 1986.

16. P. Vojtás and L. Paulík, "Soundness and completeness of non-classical SLD-resolution," in *Extensions of Logic Programming,* 1996, pp. 289-301.

17. C. V. Damasio and L. M. Pereira, "Monotonic and residuated logic programs," in *Proceedings of the 6th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty,* 2001, pp. 748-759.

18. RIF Working Group. 2007, Rule interchange format (RIF).

19. H. Boley and M. Kifer, "RIF Basic Logic Dialect," W3C Working Draft (Last Call), Tech. Rep. http://www.w3.org/TR/rif-bld/, 30 July, 2008.

20. H. Boley and M. Kifer, "RIF Framework for Logic Dialects," W3C Working Draft, Tech. Rep. http://www.w3.org/TR/rif-fld/, 30 July, 2008.

21. U. Straccia, "Fuzzy description logic programs," in *Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, (IPMU-06),* 2006, pp. 1818-1825.

22. T. Venetis, G. Stoilos, G. Stamou and S. Kollias, "f-DLPs: Extending description logic programs with fuzzy sets and fuzzy logic," in *Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007. IEEE International,* 2007, pp. 1-6.

23. L. A. Zadeh, "Fuzzy sets," *Information and Control,* vol. 8, pp. 338-353, 1965.

24. V. Novák, *Mathematical Principles of Fuzzy Logic.* Dodrecht: Kluwer Academic, 1999.

25. P. Vojtás, "Fuzzy Logic Programming," *Fuzzy Sets and Systems,* vol. 124, pp. 361-370, 2004.

26. P. Hájek, "Fuzzy logic from the logical point of view," in *SOFSEM '95: Proceedings of the 22nd Seminar on Current Trends in Theory and Practice of Informatics,* 1995, pp. 31-49.

27. P. Hájek, *Metamathematics of Fuzzy Logic.* Kluwer, 1998.

28. P. Hájek, "Fuzzy Logic and Arithmetical Hierarchy III," *Studia Logica,* vol. 68, pp. 129-142, 2001.

29. V. Hall, "Uncertainty-valued Horn Clauses," Tech. Rep. http://www.dfki.uni-kl.de/~vega/relfun+/fuzzy/fuzzy.ps, 1994.

30. A. Polleres, H. Boley and M. Kifer, "RIF datatypes and built-ins 1.0," W3C Working Draft (Last Call), Tech. Rep. http://www.w3.org/2005/rules/wiki/DTB, 30 July, 2008.

31. F. Bobillo and U. Straccia, "fuzzyDL: An expressive fuzzy description logic reasoner," in *Proceedings of the 2008 International Conference on Fuzzy Systems (FUZZ-08),* 2008.

32. J. Zhao and H. Boley, "Combining Fuzzy Description Logics and Fuzzy Logic Programs," in *Proceedings of IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology (WI-IAT 2008) Workshops, To Appear,* 2008.