Representing Fuzzy Ontologies in OWL 2

Fernando Bobillo
fbobillo@unizar.es

Department of Computer Science and Systems Engineering
University of Zaragoza, Spain

Joint research with U. Straccia (ISTI-CNR, Pisa, Italy)

FUZZ-IEEE 2010
Barcelona (Spain)
July 2010
Outline

1. Introduction
2. The Fuzzy DL $SROIQ(D)$
3. Representing Fuzzy Ontologies using OWL 2
4. Related Work
5. Conclusions and Future Work
Outline

1. Introduction
2. The Fuzzy DL $SROIQ(D)$
3. Representing Fuzzy Ontologies using OWL 2
4. Related Work
5. Conclusions and Future Work
Motivation

- Fuzzy ontologies emerge as useful in several applications.
  - Several extension of Description Logics (DLs) can be found.
  - Some **fuzzy DL reasoners** have been implemented, such as FUZZYDL, DELOREAN, and FIRE.
  - Not surprisingly, each reasoner uses its own language for representing fuzzy ontologies and, thus, there is a need for a standard way to represent such information.

- In this work, as we do not expect a fuzzy OWL extension to become a W3C proposed standard in the near future, we identify the **syntactic differences** that a fuzzy ontology language has to cope with, and propose to use OWL 2 itself to represent them.

- More precisely, **we use OWL 2 annotation properties** to encode fuzzy $\text{SROIQ(D)}$ ontologies, making it possible:
  - To use current OWL 2 editors for fuzzy ontology representation.
  - OWL 2 reasoners discard the fuzzy part of a fuzzy ontology, producing the same results as if would not exist.
Outline

1. Introduction

2. The Fuzzy DL $\text{SROIQ}(\mathcal{D})$

3. Representing Fuzzy Ontologies using OWL 2

4. Related Work

5. Conclusions and Future Work
Fuzzy concrete domains and fuzzy modifiers

Fuzzy concrete domains. A pair $\langle \Delta_D, \Phi_D \rangle$, a concrete interpretation domain $\Delta_D$, and fuzzy concrete predicates $d \in \Phi_D$:

- $d \rightarrow \left( k_1, k_2, a, b \right)$  \hspace{1cm} (D1)
- $d \rightarrow \left( k_1, k_2, a, b \right)$  \hspace{1cm} (D2)
- $d \rightarrow \text{triangular}(k_1, k_2, a, b, c)$  \hspace{1cm} (D3)
- $d \rightarrow \text{trapezoidal}(k_1, k_2, a, b, c, d)$  \hspace{1cm} (D4)

Fuzzy modifiers A function $f_{mod} : [0, 1] \rightarrow [0, 1]$ applies to a fuzzy set to change its membership function:

- $mod \rightarrow \text{linear}(c)$  \hspace{1cm} (M1)
- $mod \rightarrow \text{triangular}(a, b, c)$  \hspace{1cm} (M2)
### Fuzzy concepts

<table>
<thead>
<tr>
<th>Concepts</th>
<th>Syntax ((C))</th>
<th>Semantics of (C^\mathcal{I}(x))</th>
</tr>
</thead>
<tbody>
<tr>
<td>((C1))</td>
<td>(A)</td>
<td>(A^\mathcal{I}(x))</td>
</tr>
<tr>
<td>((C2))</td>
<td>(\top)</td>
<td>(1)</td>
</tr>
<tr>
<td>((C3))</td>
<td>(\bot)</td>
<td>(0)</td>
</tr>
<tr>
<td>((C4))</td>
<td>(C \cap_X D)</td>
<td>(C^\mathcal{I}(x) \otimes_X D^\mathcal{I}(x))</td>
</tr>
<tr>
<td>((C5))</td>
<td>(C \cup_X D)</td>
<td>(C^\mathcal{I}(x) \oplus_X D^\mathcal{I}(x))</td>
</tr>
<tr>
<td>((C6))</td>
<td>(\neg_X C)</td>
<td>(\ominus_X C^\mathcal{I}(x))</td>
</tr>
<tr>
<td>((C7))</td>
<td>(\forall_X R.C)</td>
<td>(\inf_{y \in \Delta^\mathcal{I}} { R^\mathcal{I}(x, y) \Rightarrow_X C^\mathcal{I}(y) })</td>
</tr>
<tr>
<td>((C8))</td>
<td>(\exists_X R.C)</td>
<td>(\sup_{y \in \Delta^\mathcal{I}} { R^\mathcal{I}(x, y) \otimes_X C^\mathcal{I}(y) })</td>
</tr>
<tr>
<td>((C9))</td>
<td>(\forall_X T.d)</td>
<td>(\inf_{v \in \Delta_D^\mathcal{I}} { T^\mathcal{I}(x, v) \Rightarrow_X d_D^\mathcal{I}(v) })</td>
</tr>
<tr>
<td>((C10))</td>
<td>(\exists_X T.d)</td>
<td>(\sup_{v \in \Delta_D^\mathcal{I}} { T^\mathcal{I}(x, v) \otimes_X d_D^\mathcal{I}(v) })</td>
</tr>
<tr>
<td>((C11))</td>
<td>{(\alpha/a)}</td>
<td>(\alpha) if (x = \sigma_i^\mathcal{I}), 0 otherwise</td>
</tr>
<tr>
<td>((C12))</td>
<td>(\geq_X m S.C)</td>
<td>(\sup_{y_1, \ldots, y_m \in \Delta^\mathcal{I}} { \min_{i=1}^{m+1} { S^\mathcal{I}(x, y_i) \otimes_X C^\mathcal{I}(y_i) } } \otimes_X (((\otimes_X)_{1 \leq j \leq k \leq m} { y_j \neq y_k })))</td>
</tr>
<tr>
<td>((C13))</td>
<td>(\leq_X n S.C)</td>
<td>(\inf_{y_1, \ldots, y_{n+1} \in \Delta^\mathcal{I}} { \min_{i=1}^{n+1} { S^\mathcal{I}(x, y_i) \otimes_X C^\mathcal{I}(y_i) } } \Rightarrow_X (((\otimes_X)_{1 \leq j \leq k \leq n+1} { y_j = y_k })))</td>
</tr>
<tr>
<td>((C14))</td>
<td>(\geq_X m T.d)</td>
<td>(\sup_{v_1, \ldots, v_m \in \Delta_D^\mathcal{I}} { \min_{i=1}^{m+1} { T^\mathcal{I}(x, v_i) \otimes_X d_D^\mathcal{I}(v_i) } } \otimes_X (((\otimes_X)_{j \leq k} { v_j \neq v_k })))</td>
</tr>
<tr>
<td>((C15))</td>
<td>(\leq_X n T.d)</td>
<td>(\inf_{v_1, \ldots, v_{n+1} \in \Delta_D^\mathcal{I}} { \min_{i=1}^{n+1} { T^\mathcal{I}(x, v_i) \otimes_X d_D^\mathcal{I}(v_i) } } \Rightarrow_X (((\otimes_X)_{j \leq k} { v_j = v_k })))</td>
</tr>
<tr>
<td>((C16))</td>
<td>(\exists S.Self)</td>
<td>(S^\mathcal{I}(x, x))</td>
</tr>
<tr>
<td>((C17))</td>
<td>(C \rightarrow_X D)</td>
<td>(C^\mathcal{I}(x) \Rightarrow_X D^\mathcal{I}(x))</td>
</tr>
<tr>
<td>((C18))</td>
<td>(\mod(C))</td>
<td>(f_{\mod}(C^\mathcal{I}(x)))</td>
</tr>
<tr>
<td>((C19))</td>
<td>([C \geq \alpha])</td>
<td>(1) if (C^\mathcal{I}(x) \geq \alpha), 0 otherwise</td>
</tr>
<tr>
<td>((C20))</td>
<td>([C \leq \alpha])</td>
<td>(1) if (C^\mathcal{I}(x) \leq \alpha), 0 otherwise</td>
</tr>
<tr>
<td>((C21))</td>
<td>(\alpha \cdot C)</td>
<td>(\alpha \cdot C^\mathcal{I}(x))</td>
</tr>
</tbody>
</table>
## Fuzzy roles and axioms

<table>
<thead>
<tr>
<th>Roles</th>
<th>Syntax ($R$)</th>
<th>Semantics of $R^\mathcal{I}$ ($x, y$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(R1)</td>
<td>$R_A$</td>
<td>$R_A^\mathcal{I}(x, y)$</td>
</tr>
<tr>
<td>(R2)</td>
<td>$R^-$</td>
<td>$R^\mathcal{I}(y, x)$</td>
</tr>
<tr>
<td>(R3)</td>
<td>$U$</td>
<td>1</td>
</tr>
<tr>
<td>(R4)</td>
<td>$mod(R)$</td>
<td>$f_{mod}(R^\mathcal{I}(x, y))$</td>
</tr>
<tr>
<td>(R5)</td>
<td>$[R \geq \alpha]$</td>
<td>1 if $R^\mathcal{I}(x, y) \geq \alpha$, 0 otherwise</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Axiom</th>
<th>Syntax ($\tau$)</th>
<th>Semantics ($\mathcal{I}$ satisfies $\tau$ if ...)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A1)</td>
<td>$\langle a : C \bowtie \alpha \rangle$</td>
<td>$C^\mathcal{I}(a^\mathcal{I}) \bowtie \alpha$</td>
</tr>
<tr>
<td>(A2)</td>
<td>$\langle (a, b) : R \bowtie \alpha \rangle$</td>
<td>$R^\mathcal{I}(a^\mathcal{I}, b^\mathcal{I}) \bowtie \alpha$</td>
</tr>
<tr>
<td>(A3)</td>
<td>$\langle (a, b) : \neg x R \bowtie \alpha \rangle$</td>
<td>$\otimes_X R^\mathcal{I}(a^\mathcal{I}, b^\mathcal{I}) \bowtie \alpha$</td>
</tr>
<tr>
<td>(A4)</td>
<td>$\langle (a, v) : T \bowtie \alpha \rangle$</td>
<td>$T^\mathcal{I}(a^\mathcal{I}, v_D) \bowtie \alpha$</td>
</tr>
<tr>
<td>(A5)</td>
<td>$\langle (a, v) : \neg x T \bowtie \alpha \rangle$</td>
<td>$\otimes_X T^\mathcal{I}(a^\mathcal{I}, v_D) \bowtie \alpha$</td>
</tr>
<tr>
<td>(A6)</td>
<td>$\langle a \neq b \rangle$</td>
<td>$a^\mathcal{I} \neq b^\mathcal{I}$</td>
</tr>
<tr>
<td>(A7)</td>
<td>$\langle a = b \rangle$</td>
<td>$a^\mathcal{I} = b^\mathcal{I}$</td>
</tr>
<tr>
<td>(A8)</td>
<td>$\langle C \subseteq_X D \bowtie \alpha \rangle$</td>
<td>$\inf_{x \in \Delta\mathcal{I}} { C^\mathcal{I}(x) \Rightarrow_X D^\mathcal{I}(x) } \bowtie \alpha$</td>
</tr>
<tr>
<td>(A9)</td>
<td>$\langle R_1 \ldots R_n \subseteq_X R \bowtie \alpha \rangle$</td>
<td>$\inf_{x_1, x_{n+1} \in \Delta\mathcal{I}} { \sup_{x_2 \ldots x_n \in \Delta\mathcal{I}} { (R_1^\mathcal{I}(x_1, x_2) \otimes_X \ldots \otimes_X R_n^\mathcal{I}(x_n, x_{n+1})) \Rightarrow_X } } \bowtie \alpha$</td>
</tr>
<tr>
<td>(A10)</td>
<td>$\langle T_1 \subseteq_X T_2 \bowtie \alpha \rangle$</td>
<td>$\inf_{x \in \Delta\mathcal{I}, v \in \Delta_D { T_1^\mathcal{I}(x, v) \Rightarrow_X T_2^\mathcal{I}(x, v) } \bowtie \alpha$</td>
</tr>
<tr>
<td>(A11)</td>
<td>$\text{trans}_X(R)$</td>
<td>$\forall x, y, z \in \Delta^\mathcal{I}, R^\mathcal{I}(x, z) \otimes_X R^\mathcal{I}(z, y) \leq R^\mathcal{I}(x, y)$</td>
</tr>
<tr>
<td>(A12)</td>
<td>$\text{dis}_X(S_1, S_2)$</td>
<td>$\forall x, y \in \Delta^\mathcal{I}, S_1^\mathcal{I}(x, y) \otimes_X S_2^\mathcal{I}(x, y) = 0$</td>
</tr>
<tr>
<td>(A13)</td>
<td>$\text{dis}_X(T_1, T_2)$</td>
<td>$\forall x \in \Delta^\mathcal{I}, v \in \Delta_D, T_1^\mathcal{I}(x, v) \otimes_X T_2^\mathcal{I}(x, v) = 0$</td>
</tr>
<tr>
<td>(A14)</td>
<td>$\text{ref}(R)$</td>
<td>$\forall x \in \Delta^\mathcal{I}, R^\mathcal{I}(x, x) = 1$</td>
</tr>
<tr>
<td>(A15)</td>
<td>$\text{irr}(S)$</td>
<td>$\forall x \in \Delta^\mathcal{I}, S^\mathcal{I}(x, x) = 0$</td>
</tr>
<tr>
<td>(A16)</td>
<td>$\text{sym}(R)$</td>
<td>$\forall x, y \in \Delta^\mathcal{I}, R^\mathcal{I}(x, y) = R^\mathcal{I}(y, x)$</td>
</tr>
<tr>
<td>(A17)</td>
<td>$\text{asy}(S)$</td>
<td>$\forall x, y \in \Delta^\mathcal{I}, \text{if } S^\mathcal{I}(x, y) &gt; 0 \text{ then } S^\mathcal{I}(y, x) = 0$</td>
</tr>
</tbody>
</table>
Definable elements

Definable concepts:

- **Weighted sum:** \((\alpha_1 \cdot C_1) \sqcup_L \cdots \sqcup_L (\alpha_k \cdot C_k)\).
- **Fuzzy one-of:** \(\{\alpha_1/\omega_1\} \sqcup_G \{\alpha_2/\omega_2\} \sqcup_G \cdots \sqcup_G \{\alpha_k/\omega_k\}\).

Definable axioms (given an R-implication):

- **Concept equivalence:** \(\langle C_1 \sqsubseteq_X C_2 \geq 1 \rangle\) and \(\langle C_2 \sqsubseteq_X C_1 \geq 1 \rangle\).
- **Disjoint concepts:** \(\langle C_1 \sqcap_X \cdots \sqcap_X C_n \sqsubseteq_X \bot \geq 1 \rangle\).
- **Role domain:** \(\langle \exists_X R.T \sqsubseteq_X C \geq 1 \rangle\).
- **Role range:** \(\langle T \sqsubseteq_X \forall_X R.C \geq 1 \rangle\).
- **Role functionality:** \(\langle T \sqsubseteq_X (\leq_X 1 \ R.\top) \geq 1 \rangle\).

Syntactic sugar (not assumed for similarity with OWL 2):

- \(\text{irr}(S) = \top \sqsubseteq_X \neg \exists S.\text{Self}\)
- \(\text{trans}(R) = RR \sqsubseteq_X R\)
- \(\text{sym}(R) = R \sqsubseteq_X R^\sim\)
Outline

1. Introduction
2. The Fuzzy DL $SROIQ(D)$
3. Representing Fuzzy Ontologies using OWL 2
4. Related Work
5. Conclusions and Future Work
The idea of our representation is to use an OWL 2 ontology, extending their elements with annotation properties of type `rdfs:comment`, representing the features of the fuzzy ontology that OWL 2 cannot directly encode.

Example

Consider the fuzzy concept assertion `<paul: Tall ≥ 0.5>`. To represent it in OWL 2, we consider the crisp assertion `paul: Tall` as represented in OWL 2, i.e., `ClassAssertion(paul Tall)`. Next, we add an annotation property stating `≥ 0.5` to it.

- It is worth to note that OWL 2 only provides for annotations on ontologies, axioms, and entities.
- OWL DL is less expressive and only provides for annotations on ontologies and entities.
- For the sake of clarity, we will combine OWL 2 abstract syntax (for OWL 2), and an XML syntax (for annotation properties).
Syntactic Requirements of Fuzzy Ontologies

We will summarize the syntactic differences between the fuzzy and non-fuzzy ontologies. There are 8 cases which are non-exclusive (cases 3–5 can occur simultaneously, as well as cases 7–8).

1. **Fuzzy datatypes** do not have an equivalence in OWL 2: (D1–D4).
2. **Fuzzy modifiers** do not have an equivalence in OWL 2: (M1–M2).
3. Some **fuzzy concepts** require a **fuzzy logic**: (C4–C10), (C12–C15), (C17).
4. Some **fuzzy concepts** require a **degree** of truth: (C11), (C21).
5. Some **fuzzy concepts** do not have an equivalence in OWL 2: (C17)–(C21).
6. Some **fuzzy roles** do not have an equivalence in OWL 2: (R4–R5).
7. Some **axioms** require an inequality sign and a **degree** of truth: (A1–(A5), (A8)–(A10).
8. Some **axioms** require a **fuzzy logic**: (A3), (A5), (A8–A13).
1. Representing fuzzy datatypes

Example

Represent the age of a young person as \( \text{left}(0, 200, 10, 30) \). We use a datatype definition of base type \( \text{xsd:nonNegativeInteger} \) with range in \([0, 200]\):

\[
\text{DatatypeDefinition( YoungAge DatatypeRestriction(}
\text{xsd:nonNegativeInteger}
\text{xsd:minInclusive "0"^^xsd:integer}
\text{xsd:maxInclusive "200"^^xsd:integer})
\)
\]

Then we add the following annotation property to it:

\[
<\text{fuzzyOwl2 fuzzyType="datatype">}
<\text{Datatype type="leftshoulder" a="10" b="30" />}
<\text{fuzzyOwl2>}
\]
2. Representing fuzzy modifiers

- Our fuzzy modifiers have parameters $a, b, c$.
- They can be represented as in the previous case, without representing `xsd:minInclusive` and `xsd:maxInclusive`.
- The value of `fuzzyType` will be `modifier`, and there will be a tag `Modifier` with an attribute `type` (possible values `linear`, and `triangular`), and attributes $a, b, c$, depending on the type.

**Example**

We define the datatype `very`

```
DatatypeDefinition( very xsd:nonNegativeInteger )
```

Then, we add this annotation property to it:

```
<fuzzyOwl2 fuzzyType="modifier">
  <Modifier type="linear" c="0.8" />
</fuzzyOwl2>
```
An annotation in a named concept can specify the fuzzy logic.

- Anonymous concept expressions must be explicitly named.

The value of fuzzyType is concept.

There is an optional tag Logic (possible values goedel, lukasiewicz, product, and zadeh). Default value: goedel.

Example (Concept \{1/germany\} \sqcap_G \{0.67/switzerland\})

```
Class ( C Annotation ( rdfs:comment
    <fuzzyOwl2 fuzzyType="concept">
        <Logic>goedel</Logic>
    </fuzzyOwl2>
)
EquivalentClasses( C ObjectUnionOf( Nom1 Nom2 ) )
```
An annotation in a named concept can specify the degree.

Anonymous concept expressions must be explicitly named.

The value of fuzzyType is concept.

There is an optional tag Degree (with attribute value).

Example (Concepts \{1/germany\}, \{0.67/switzerland\})

```xml
Class ( Nom1 Annotation ( rdfs:comment
  <fuzzyOwl2 fuzzyType="concept">
    <Degree value="1" />
  </fuzzyOwl2>
) )
Class ( Nom2 Annotation ( rdfs:comment
  <fuzzyOwl2 fuzzyType="concept">
    <Degree value="0.67" />
  </fuzzyOwl2>
) )
EquivalentClasses ( Nom1 ObjectOneOf ( germany ) )
EquivalentClasses ( Nom2 ObjectOneOf ( switzerland ) )
```
5–6. Representing fuzzy concepts and roles without an equivalence in OWL

- We have **to create a new entity** (concept or role) denoting the elements, and to add an annotation property to it, describing the type of the constructor and the value of their parameters.

Example (very(R))

```xml
ObjectProperty ( veryR Annotation ( rdfs:comment
   <fuzzyOwl2 fuzzyType="role">
    <Role type="modified" modifier="very" base="R" />
   </fuzzyOwl2>
 ) )
```
7–8. Representing fuzzy axioms

- Some axioms may require a fuzzy logic, an inequality sign, or a degree of truth.
- Similarly to cases 3–4, there are two optional tags:
  - **Degree**, with attributes `value` and `sign` (possible values `geq`, `gre`, `leq`, and `les`),
  - **Logic** (possible values `goedel`, `lukasiewicz`, `product`, or `zadeh`).

**Example**

Consider again the fuzzy concept assertion `<paul: Tall ≥ 0.5>`. We extend the OWL 2 axiom `ClassAssertion(paul Tall)` with the following annotation property:

```xml
<fuzzyOwl2 fuzzyType="axiom">
  <Degree sign="geq" value="0.5" />
  <Logic>lukasiewicz</Logic>
</fuzzyOwl2>
```
Outline

1 Introduction

2 The Fuzzy DL $\mathcal{SROIQ}(\mathcal{D})$

3 Representing Fuzzy Ontologies using OWL 2

4 Related Work

5 Conclusions and Future Work
Related work

- **An OWL ontology** for fuzzy ontology representation using individuals to represent concepts, roles and axioms.
  - (Meta) logical problems, completely different and user-unfriendly way of modelling, and inefficient representation (it grows exponentially with the size of the ontology).

- **Fuzzy OWL and Fuzzy OWL 2.**
  - Obviously not complaint with OWL 2 and current ontology editors.

- Similar work covers just some of the cases:
  - A **pattern for uncertainty representation** in ontologies restricted to a subset of our case 7, axioms (A1).
  - **Probabilistic constraints** restricted to a subset of our case 7, axioms (A1) and (A8).

- **Crisp representations for fuzzy ontologies.**
  - Ok to reuse current DL reasoners, but not for modelling.
Outline

1 Introduction

2 The Fuzzy DL $\text{SROIQ}(\text{D})$

3 Representing Fuzzy Ontologies using OWL 2

4 Related Work

5 Conclusions and Future Work
Conclusions and future work

- Our objective is **not to provide a standard** language for fuzzy ontology representation. This should involve the whole community.
- We identified the **syntactical differences** of a fuzzy ontology language and provided a **representation using OWL 2**.
  - A similar approach **cannot be represented in OWL DL** as it does not support rich enough annotation capabilities.
- Our logic is very expressive, but it is **extensible** and can easily be augmented to support more fuzzy logics, fuzzy predicates . . .
- **Methodology for fuzzy ontology development.**
  - First, we can build the core part of the ontology as usual.
  - Then, we add the fuzzy part with annotation properties.
  - Non-fuzzy reasoners **discard the fuzzy part**.
- Parsers translating this representation into the syntax of some popular fuzzy DL reasoners (**FUZZYDL** and **DELOREAN**).
- In **future work**, we will develop a **graphical interface** (e.g. a Protégé plug-in) to make annotation properties transparent.
Comments?

Thank you very much for your attention