OWL API for $\mathcal{ALC}$-LTL Reasoning

M. Fareed Arif

June 03/04 , 2013

Outline

Semantic Web & OWL

Semantic Web:

The Semantic Web is the extension of the World Wide Web that facilitates machines to understand the meaning of information on the Web.

Web Ontology Language (OWL): ([Michael K. Smith 2003])

OWL provides a family of languages (i.e., OWL Lite, OWL DL and OWL Full) which are used to author OWL Ontologies. OWL provides a common standard for representing, exchanging and deriving logical consequences from different domains. Thus, providing machine-processable descriptions of domains including World Wide Web.

The OWL API: ([Sean Bechhofer 2003])

The OWL API is a programming interface to access and manupulate OWL Ontologies.

Pizza Examples:

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF
<!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">
<!ENTITY owl "http://www.w3.org/2002/07/owl#">
<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">
<!ENTITY family "http://www.example.org/family#">
]>
...
SubClassOf(<http://www.co-ode.org/ontologies/pizza/pizza.owl#
Soho> ObjectSomeValuesFrom(<<http://www.co-ode.org/ontologies/
pizza/pizza.owl#hasTopping> <http://www.co-ode.org/ontologies/
pizza/pizza.owl#GarlicTopping>))
...
```

$\mathtt{Soho} \sqsubseteq \exists\ \mathtt{hasTopping}\ .\ \mathtt{GarlicTopping}$

The Description Logic $\mathcal{ALC}$

$\mathcal{ALC}$ [Schmidt-Schau 1991] is a DL with conjunction ($\sqcap$), disjunction($\sqcup$), negation($\neg$), existential restriction ($\exists$) and **value restriction ($\forall$)**.

Syntax ([Franz Baader 2008])

Let N_C is a set of concept names, N_R is a set of role names and N_I is a set individual names. The set of $\mathcal{ALC}$ concept descriptions is the smallest set satisfying the following properties:

- Every concept name, $\top$ and $\bot$ are $\mathcal{ALC}$-concept descriptions,
- If C, D are $\mathcal{ALC}$ concept descriptions, $r \in N_R$, then the following are $\mathcal{ALC}$-concept descriptions:

$$\begin{array}{rl}
\neg C & \text{(complement)} \\
C \sqcap D & \text{(conjunction)} \\
C \sqcup D & \text{(disjunction)} \\
\exists r.C & \text{(existential restriction)} \\
\forall r.C & \text{(value restriction)}
\end{array}$$

The Description Logic $\mathcal{ALC}$

General Concept Inclusion (GCI):

A general concept inclusion (GCI) is of form $C \sqsubseteq D$ where C, D are $\mathcal{ALC}$-concept descriptions.

Assertion:

An assertion is of the form $a : C$ or $(a, b) : r$ where C is an $\mathcal{ALC}$-concept description, $r \in N_C$ and $a, b \in N_I$.

Example:

$$Person \sqcap \exists hasChild.Person$$

$$GermanCitizen \sqsubseteq \exists insured\_by.HealthInsurance$$

$$Germany : \exists winner.FIFA\_WORLD\_CUP$$

The Description Logic $\mathcal{ALC}$

Interpretation ([Franz Baader 2008])

An $\mathcal{ALC}$ interpretation $\mathcal{I}$ is a pair $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}}$ is a non-empty set, called the domain of $\mathcal{I}$, and a mapping $\cdot^{\mathcal{I}}$ that assigns:

- a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ to each concept name $A \in N_C$,
- $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$ and $\bot^{\mathcal{I}} = \emptyset$,
- an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ to each individual name $a \in N_I$ and
- a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ to each role name $r \in N_R$.
- $(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$,
- $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$,
- $(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$,
- $(\exists r.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} |$ there is a $y \in \Delta^{\mathcal{I}}$ with $(x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$,
- $(\forall r.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} |$ for all $y \in \Delta^{\mathcal{I}}, (x, y) \in r^{\mathcal{I}} \Rightarrow y \in C^{\mathcal{I}}\}$.

Example: Pizza OWL API in $\mathcal{ALC}$

$\texttt{Soho} \sqsubseteq \exists\ \texttt{hasTopping} \ .\ \texttt{GarlicTopping}$

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF
<!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">
<!ENTITY owl "http://www.w3.org/2002/07/owl#">
<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">
<!ENTITY family "http://www.example.org/family#">
]>
...
SubClassOf(<http://www.co-ode.org/ontologies/pizza/pizza.owl#
Soho> ObjectSomeValuesFrom(<<http://www.co-ode.org/ontologies/
pizza/pizza.owl#hasTopping> <http://www.co-ode.org/ontologies/
pizza/pizza.owl#GarlicTopping>))
...
```

The OWL API & Descripton Logic $\mathcal{ALC}$

The OWL API provides appropriate data structure to deal with the description logic $\mathcal{ALC}$.

$\mathcal{ALC}$	OWL API
$a \in N_I$	OWLIndividual
$A \in N_C$	OWLClass
$r \in N_R$	OWLObjectProperty
$\sqcap$	OWLObjectIntersectionOf
$\sqcup$	OWLObjectUnionOf
$\neg$	OWLObjectComplementOf
$\exists$	OWLObjectSomeValuesFrom
$\forall$	OWLObjectAllValuesFrom

Motivation

1. The OWL API is insufficient to deal with $\mathcal{ALC}$-LTL.
2. Extending the OWL API with appropirate data structures that are required to represent $\mathcal{ALC}$-LTL formulae.

Temporalized DL $\mathcal{ALC}$-LTL

$\mathcal{ALC}$-LTL ([Franz Baader 2008]) is a temporalized extension of a logic-based knowledge representation formalism $\mathcal{ALC}$.

- If α is an $\mathcal{ALC}$-axiom (i.e., Both GCIs and Assertions are called $\mathcal{ALC}$-axioms), then α is an $\mathcal{ALC}$-LTL formula;
- If ϕ, ψ are $\mathcal{ALC}$-LTL formulas, then $\neg\phi$, $\phi \wedge \psi$, $\phi \vee \psi$, $\phi \mathsf{U} \psi$ and $\mathsf{X}\psi$ are $\mathcal{ALC}$-LTL formulas.

Example:

$$\Diamond\Box(\mathit{USCitizen} \sqsubseteq \exists \mathit{insured\_by}.\mathit{HealthInsurance})$$

Abbreviations: $\mathit{true} \equiv \phi \vee \neg\phi$, $\Diamond\phi \equiv \mathit{true}\mathsf{U}\phi$ and $\Box\phi \equiv \neg(\mathit{true}\mathsf{U}\phi)$.

Temporalized DL $\mathcal{ALC}$-LTL

The semantics of $\mathcal{ALC}$-LTL are described in [Franz Baader 2008] by using an $\mathcal{ALC}$-LTL structure.

$\mathcal{ALC}$-LTL Structure:

An $\mathcal{ALC}$-LTL structure is a sequence $\mathfrak{I} = (\mathcal{I}_i)_{i=0,1,\ldots}$ of $\mathcal{ALC}$ interpretations $\mathcal{I}_i = (\Delta, .^{\mathcal{I}_i})$.

$$\mathcal{I}_0 \Rightarrow \mathcal{I}_1 \Rightarrow \mathcal{I}_2 \Rightarrow \ldots$$

Example:

$$\Diamond\Box(USCitizen \sqsubseteq \exists insured\_by.HealthInsurance)$$

$$\mathcal{I}_0 \Rightarrow \mathcal{I}_1 \Rightarrow \mathcal{I}_2 \Rightarrow \mathcal{I}_3 \Rightarrow \ldots$$

Temporalized DL $\mathcal{ALC}$-LTL

The semantics of $\mathcal{ALC}$-LTL are described in [Franz Baader 2008] by using an $\mathcal{ALC}$-LTL structure. An $\mathcal{ALC}$-LTL structure is a sequence of $\mathcal{ALC}$ interpretations over a non-empty domain with unique name assumption.

$\mathcal{ALC}$-LTL Structure ([Franz Baader 2008])

An $\mathcal{ALC}$-LTL structure is a sequence $\mathfrak{I} = (\mathcal{I}_i)_{i=0,1,\ldots}$ of $\mathcal{ALC}$ interpretations $\mathcal{I}_i = (\Delta, .^{\mathcal{I}_i})$ such that $a^{\mathcal{I}_i} = a^{\mathcal{I}_j}$ for all individual names a and for all $i, j \in \{0, 1, 2, \ldots, \}$. Given an $\mathcal{ALC}$-LTL formula ϕ, an $\mathcal{ALC}$-LTL structure $\mathfrak{I} = (\mathcal{I}_i)_{i=0,1,\ldots,}$ and a time point $i \in \{0, 1, 2, \ldots\}$, validity of ϕ in $\mathfrak{I}$ at time i (written $\mathfrak{I}, i \models \phi$) is defined inductively:

Temporalized DL $\mathcal{ALC}$-LTL

$\mathcal{ALC}$-LTL Structure ([Franz Baader 2008])

- $\mathfrak{I}, i \models C \sqsubseteq D$ iff $C^{\mathcal{I}_i} \subseteq D^{\mathcal{I}_i}$,
- $\mathfrak{I}, i \models a : C$ iff $a^{\mathcal{I}_i} \in C^{\mathcal{I}_i}$,
- $\mathfrak{I}, i \models (a, b) : r$ iff $(a^{\mathcal{I}_i}, b^{\mathcal{I}_i}) \in r^{\mathcal{I}_i}$,
- $\mathfrak{I}, i \models \phi \wedge \psi$ iff $\mathfrak{I}, i \models \phi$ and $\mathfrak{I}, i \models \psi$,
- $\mathfrak{I}, i \models \phi \vee \psi$ iff $\mathfrak{I}, i \models \phi$ or $\mathfrak{I}, i \models \psi$,
- $\mathfrak{I}, i \models \neg\phi$ iff $\mathfrak{I}, i \not\models \phi$,
- $\mathfrak{I}, i \models \mathrm{X}\phi$ iff $\mathfrak{I}, i+1 \models \phi$,
- $\mathfrak{I}, i \models \phi \mathrm{U} \psi$ iff there is $k \geq i$ such that $\mathfrak{I}, k \models \psi$ and $\mathfrak{I}, j \models \phi$ for all j, $i \leq j < k$.

Structure of $\mathcal{ALC}$-LTL formulae XML File

$\mathcal{ALC}$-axioms are stored in an OWL ontology file. We store $\mathcal{ALC}$-LTL formulae in an XML file using the following mapping.

$\mathcal{ALC}$-LTL XML mapping:

Operators	XML Mapping
$\neg$	$\langle$NegationOf$\rangle \ldots \langle$/NegationOf$\rangle$
$\sqcap$	$\langle$ConjunctionOf$\rangle \ldots \langle$/ConjunctionOf$\rangle$
$\sqcup$	$\langle$DisjunctionOf$\rangle \ldots \langle$/DisjunctionOf$\rangle$
X	$\langle$NextOf$\rangle \ldots \langle$/NextOf$\rangle$
U	$\langle$UntilOf$\rangle$ $\langle$LeftOf$\rangle \ldots \langle$/LeftOf$\rangle$
	$\langle$RightOf$\rangle \ldots \langle$/RightOf$\rangle$ $\langle$/UntilOf$\rangle$

$\mathcal{ALC}$-LTL Examples:

$(\mathtt{X}(\mathtt{X}((\forall \mathtt{r3.}\ (\mathtt{c3} \sqcup \mathtt{c4}) \sqsubseteq \forall \mathtt{r2.}\ \mathtt{c5}\ \mathsf{U}\ \neg\exists \mathtt{r4.}\ \mathtt{c1} \sqsubseteq (\mathtt{c4} \sqcup \mathtt{c5}))))$

```
... <Formula> <NextOf> <NextOf> <UntilOf> <LeftOf>
<Axiom> <Literal>1</Literal> </Axiom> </LeftOf> <RightOf> <Axi
<Literal>2</Literal> </Axiom> </RightOf>
</UntilOf> </NextOf> </NextOf> </Formula> ...
```

$\mathcal{ALC}$-LTL Examples:

$\forall r3.\ (c3 \sqcup c4) \sqsubseteq \forall r2.\ c5$

```
...
<SubClassOf> <Annotation> <Literal">1</Literal> </Annotation>
<ObjectAllValuesFrom> <ObjectProperty IRI="http://www.semantic
<ObjectUnionOf> <Class IRI="http://www.semanticweb.org/owlapi:
<Class IRI="http://www.semanticweb.org/owlapi:ontology944#c4"/
</ObjectUnionOf> </ObjectAllValuesFrom> <ObjectAllValuesFrom>
<ObjectProperty IRI="http://www.semanticweb.org/owlapi:ontolog
<Class IRI="http://www.semanticweb.org/owlapi:ontology944#c5"/
</ObjectAllValuesFrom> </SubClassOf>
...
```

Program Demo

For Further Reading

- Manfred Schmidt-Schauß and Gert Smolka. 1991
  Attributive Concept Descriptions with Complements.
- Franz Baader, Silvio Ghilardi and Carsten Lutz. 2008
  LTL over Description Logic Axioms.
- Franz Baader, Ian Horrocks and Ulrike Sattler. 2008
  Handbook of Knowledge Representation, "Description Logics".
- Michael K. Smith, Chris Welty and Deborah L. McGuinness. 2003
  OWL Web Ontology Language Guide
- Sean Bechhofer, Rapheal Volz and Phillip Lord. 2003
  Cooking the Semantic Web with the OWL API

Q & A