fuzzyDL: An Expressive Fuzzy Description Logic Reasoner

Fernando Bobillo[1], Umberto Straccia[2]

[1]University of Granada, Spain,

[2]ISTI-CNR, Pisa, Italy

Web Ontology Languages

- Wide variety of languages for “Explicit Specification”
  - Graphical notations
    - Semantic networks
    - UML
    - RDF/RDFS
  - Logic based
    - Description Logics (e.g., OIL, DAML+OIL, OWL, OWL-DL, OWL-Lite)
    - Rules (e.g., RuleML, RIF, SWRL, LP/Prolog)
    - First Order Logic (e.g., KIF)
- RDF and OWL-DL are the major players (so far ...)

OWL

- Three species of OWL
  - OWL full is union of OWL syntax and RDF (Undecidable)
  - OWL DL restricted to FOL fragment (decidable in NEXPTIME)
  - OWL Lite is “easier to implement” subset of OWL DL (decidable in EXPTIME)
- Semantic layering
  - OWL DL within Description Logic (DL) fragment
- OWL DL based on $\mathcal{SHOIN}(D_n)$ DL
- OWL Lite based on $\mathcal{SHIF}(D_n)$ DL

Description Logics (DLs)

- The logics behind OWL-DL and OWL-Lite, `http://dl.kr.org/`.
- Concept/Class: names are equivalent to unary predicates
  - In general, concepts equiv to formulae with one free variable
- Role or attribute: names are equivalent to binary predicates
  - In general, roles equiv to formulae with two free variables
- Taxonomy: Concept and role hierarchies can be expressed
- Individual: names are equivalent to constants
- Operators: restricted so that:
  - Language is decidable and, if possible, of low complexity
  - No need for explicit use of variables
    - Restricted form of $\exists$ and $\forall$
  - Features such as counting can be succinctly expressed

The DL Family

- A given DL is defined by set of concept and role forming operators
- Basic language: $\mathcal{ALC}$($\mathcal{A}$ttributive $\mathcal{L}$anguage with $\mathcal{C}$omplement)

Syntax				Semantics	Example
C, D	$\rightarrow$	$\top$	|	$\top(x)$	
		$\bot$	|	$\bot(x)$	
		A	|	$A(x)$	*Human*
		$C \sqcap D$	|	$C(x) \wedge D(x)$	$Human \sqcap Male$
		$C \sqcup D$	|	$C(x) \vee D(x)$	$Nice \sqcup Rich$
		$\neg C$	|	$\neg C(x)$	$\neg Meat$
		$\exists R.C$	|	$\exists y.R(x,y) \wedge C(y)$	$\exists has\_child.Blond$
		$\forall R.C$		$\forall y.R(x,y) \Rightarrow C(y)$	$\forall has\_child.Human$
$C \sqsubseteq D$				$\forall x.C(x) \Rightarrow D(x)$	$Happy\_Father \sqsubseteq Man \sqcap \exists has\_child.Female$
$a{:}C$				$C(a)$	$John{:}Happy\_Father$

Toy Example

$$\begin{array}{rcl} Sex & = & Male \sqcup Female \\ Male \sqcap Female & \sqsubseteq & \bot \\ Person & \sqsubseteq & Human \sqcap \exists hasSex.Sex \\ MalePerson & \sqsubseteq & Person \sqcap \exists hasSex.Male \end{array}$$

$$umberto : Person \sqcap \exists hasSex.\neg Female$$

$$KB \models umberto : MalePerson$$

Note on DL Naming

$\mathcal{AL}$: $C, D \longrightarrow \top \mid \bot \mid A \mid C \sqcap D \mid \neg A \mid \exists R.\top \mid \forall R.C$

- $\mathcal{C}$: Concept negation, $\neg C$. Thus, $\mathcal{ALC} = \mathcal{AL} + \mathcal{C}$
- $\mathcal{S}$: Used for $\mathcal{ALC}$ with transitive roles $\mathcal{R}_+$
- $\mathcal{U}$: Concept disjunction, $C_1 \sqcup C_2$
- $\mathcal{E}$: Existential quantification, $\exists R.C$
- $\mathcal{H}$: Role inclusion axioms, $R_1 \sqsubseteq R_2$, *e.g.*, $\mathit{is\_component\_of} \sqsubseteq \mathit{is\_part\_of}$
- $\mathcal{N}$: Number restrictions, $(\geqslant n\ R)$ and $(\leqslant n\ R)$, *e.g.*, $(\geqslant 3\ \mathit{has\_Child})$ (has at least 3 children)
- $\mathcal{Q}$: Qualified number restrictions, $(\geqslant n\ R.C)$ and $(\leqslant n\ R.C)$, *e.g.*, $(\leqslant 2\ \mathit{has\_Child}.\mathit{Adult})$ (has at most 2 adult children)
- $\mathcal{O}$: Nominals (singleton class), $\{a\}$, *e.g.*, $\exists \mathit{has\_child}.\{\mathit{mary}\}$.
  Note: $a{:}C$ equiv to $\{a\} \sqsubseteq C$ and $(a, b){:}R$ equiv to $\{a\} \sqsubseteq \exists R.\{b\}$
- $\mathcal{I}$: Inverse role, R^-, *e.g.*, $\mathit{isPartOf} = \mathit{hasPart}^-$
- $\mathcal{F}$: Functional role, f, *e.g.*, $\mathit{functional}(\mathit{hasAge})$
- $\mathcal{R}_+$: transitive role, *e.g.*, $\mathit{transitive}(\mathit{isPartOf})$

For instance,

$$\mathcal{SHIF} = \mathcal{S} + \mathcal{H} + \mathcal{I} + \mathcal{F} = \mathcal{ALCR}_+\mathcal{HIF} \quad \text{OWL-Lite (EXPTIME)}$$

$$\mathcal{SHOIN} = \mathcal{S} + \mathcal{H} + \mathcal{O} + \mathcal{I} + \mathcal{N} = \mathcal{ALCR}_+\mathcal{HOIN} \quad \text{OWL-DL (NEXPTIME)}$$

Concrete Domains

- Concrete domains: reals, integers, strings, . . .

  $(tim, 14) : hasAge$
  $(sf, \text{``}SoftComputing\text{''}) : hasAcronym$
  $(source1, \text{``}ComputerScience\text{''}) : isAbout$
  $(service2, \text{``}InformationRetrievalTool''\,) : Matches$
  $Minor = Person \sqcap \exists hasAge. \leqslant_{18}$

- Semantics: a clean separation between “object” classes and concrete domains
  - $D = \langle \Delta_\beta, \Phi_\beta \rangle$
  - Δ_β is an interpretation domain
  - Φ_β is the set of concrete domain predicates d with a predefined arity n and fixed interpretation $d^\beta \subseteq \Delta_\beta^n$
  - Concrete properties: $R^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta_D$
- Notation: (D). E.g., $\mathcal{ALC}(D)$ is $\mathcal{ALC}$ + concrete domains

Fuzzy DLs Basics

The semantics is an immediate consequence of applying mathematical fuzzy logic to the First-Order-Logic translation of DLs expressions

Interpretation:

$$\mathcal{I} = \Delta^{\mathcal{I}}$$
$$C^{\mathcal{I}} : \Delta^{\mathcal{I}} \rightarrow [0,1]$$
$$R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0,1]$$

$\otimes$ = t-norm
$\oplus$ = s-norm
$\ominus$ = negation
$\Rightarrow$ = implication

Concepts:

Syntax		Semantics		
$C, D \longrightarrow$	$\top$ |	$\top^{\mathcal{I}}(x)$	$=$	1
	$\bot$ |	$\bot^{\mathcal{I}}(x)$	$=$	0
	A |	$A^{\mathcal{I}}(x)$	$\in$	$[0,1]$
	$C \sqcap D$ |	$(C_1 \sqcap C_2)^{\mathcal{I}}(x)$	$=$	$C_1{}^{\mathcal{I}}(x) \otimes C_2{}^{\mathcal{I}}(x)$
	$C \sqcup D$ |	$(C_1 \sqcup C_2)^{\mathcal{I}}(x)$	$=$	$C_1{}^{\mathcal{I}}(x) \oplus C_2{}^{\mathcal{I}}(x)$
	$\neg C$ |	$(\neg C)^{\mathcal{I}}(x)$	$=$	$\ominus C^{\mathcal{I}}(x)$
	$\exists R.C$ |	$(\exists R.C)^{\mathcal{I}}(x)$	$=$	$\sup_{y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x,y) \otimes C^{\mathcal{I}}(y)$
	$\forall R.C$	$(\forall R.C)^{\mathcal{I}}(u)$	$=$	$\inf_{y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x,y) \Rightarrow C^{\mathcal{I}}(y)\}$

Assertions: $\langle a{:}C, r\rangle, \mathcal{I} \models \langle a{:}C, r\rangle$ iff $C^{\mathcal{I}}(a^{\mathcal{I}}) \geqslant r$ (similarly for roles)

- individual a is instance of concept C at least to degree r, $r \in [0,1] \cap \mathbb{Q}$

Inclusion axioms: $\langle C \sqsubseteq D, r\rangle$,

- $\mathcal{I} \models \langle C \sqsubseteq D, r\rangle$ iff $\inf_{x \in \Delta^{\mathcal{I}}} C^{\mathcal{I}}(x) \leqslant D^{\mathcal{I}}(x) \geqslant r$

fuzzyDL: Informally

- A fuzzy variant of $\mathcal{SHIF}$ with concrete data types
  - *e.g.*, $\textit{Sedan} \sqcap (\geqslant \textit{price}\ 22.000)$
- fuzzy constraints
  - numerical features may be constrained by so-called fuzzy membership functions

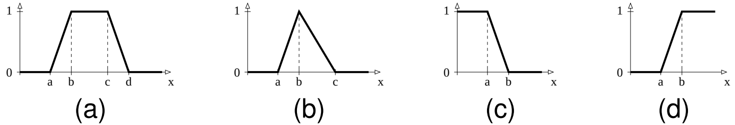

Figure: (a) Trapezoidal function $trz(a, b, c, d)$, (b) triangular function $tri(a, b, c)$, (c) left shoulder function $ls(a, b)$, and (d) right shoulder function $rs(a, b)$.

  - For instance, $(\exists price.ls(22000, 26000))$ dictates that given a price it returns the degree to which the constraint is satisfied

Definition (Concept expressions)

As for $\mathcal{SHIF}$

+

$$C \to DR \quad (\text{datatype restriction})$$
$$DR \to (\geqslant t\ val) \mid (\leqslant t\ val) \mid (= t\ val)$$

e.g. $Sedan \sqcap (\leqslant price\ 26.000)$

+

$$C \to \forall t.d \mid \exists t.d \quad (\text{fuzzy constraints})$$
$$d \to ls(a,b) \mid rs(a,b) \mid tri(a,b,c) \mid trz(a,b,c,d)$$

e.g. $Car \sqcap (\exists price.ls(22000, 26000))$

+

$$C \to TC \quad (\text{threshold concept})$$
$$TC \to C[\geqslant n] \mid C[\leqslant n]$$

e.g. $(Sedan \sqcap Cheap \sqcap (\leqslant price\ 30.000))[\geqslant 0.8]$

+

$$C \to WC \quad (\text{weighted sum concept})$$
$$WC \to (w_1 \cdot C_1 + w_2 \cdot C_2 + \ldots + w_k \cdot C_k)$$

where $\sum_{i=1}^{k} w_i = 1$. E.g., $0.2 \cdot (\leqslant price\ 30.000) + 0.8 \cdot (\exists hasColor.Red)$

Definition (Concept expressions)

As for $\mathcal{SHIF}$

+

$$\begin{array}{rcl} C & \rightarrow & DR \quad \text{(datatype restriction)} \\ DR & \rightarrow & (\geqslant t\ val) \mid (\leqslant t\ val) \mid (= t\ val) \end{array}$$

e.g. $Sedan \sqcap (\leqslant price\ 26.000)$

+

$$\begin{array}{rcl} C & \rightarrow & \forall t.d \mid \exists t.d \quad \text{(fuzzy constraints)} \\ d & \rightarrow & ls(a,b) \mid rs(a,b) \mid tri(a,b,c) \mid trz(a,b,c,d) \end{array}$$

e.g. $Car \sqcap (\exists price.ls(22000, 26000))$

+

$$\begin{array}{rcl} C & \rightarrow & TC \quad \text{(threshold concept)} \\ TC & \rightarrow & C[\geqslant n] \mid C[\leqslant n] \end{array}$$

e.g. $(Sedan \sqcap Cheap \sqcap (\leqslant price\ 30.000))[\geqslant 0.8]$

+

$$\begin{array}{rcl} C & \rightarrow & WC \quad \text{(weighted sum concept)} \\ WC & \rightarrow & (w_1 \cdot C_1 + w_2 \cdot C_2 + \ldots + w_k \cdot C_k) \end{array}$$

where $\sum_{i=1}^{k} w_i = 1$. E.g., $0.2 \cdot (\leqslant price\ 30.000) + 0.8 \cdot (\exists hasColor.Red)$

Definition (Concept expressions)

As for $\mathcal{SHIF}$

+

$$\begin{array}{lcl} C & \rightarrow & DR \quad \text{(datatype restriction)} \\ DR & \rightarrow & (\geqslant t\ val) \mid (\leqslant t\ val) \mid (= t\ val) \end{array}$$

e.g. $Sedan \sqcap (\leqslant price\ 26.000)$

+

$$\begin{array}{lcl} C & \rightarrow & \forall t.d \mid \exists t.d \quad \text{(fuzzy constraints)} \\ d & \rightarrow & ls(a,b) \mid rs(a,b) \mid tri(a,b,c) \mid trz(a,b,c,d) \end{array}$$

e.g. $Car \sqcap (\exists price.ls(22000, 26000))$

+

$$\begin{array}{lcl} C & \rightarrow & TC \quad \text{(threshold concept)} \\ TC & \rightarrow & C[\geqslant n] \mid C[\leqslant n] \end{array}$$

e.g. $(Sedan \sqcap Cheap \sqcap (\leqslant price\ 30.000))[\geqslant 0.8]$

+

$$\begin{array}{lcl} C & \rightarrow & WC \quad \text{(weighted sum concept)} \\ WC & \rightarrow & (w_1 \cdot C_1 + w_2 \cdot C_2 + \ldots + w_k \cdot C_k) \end{array}$$

where $\sum_{i=1}^{k} w_i = 1$. E.g., $0.2 \cdot (\leqslant price\ 30.000) + 0.8 \cdot (\exists \text{hasColor}.Red)$

Definition (Concept expressions)

As for $\mathcal{SHIF}$

+

$$\begin{array}{lcl} C & \rightarrow & DR \quad \text{(datatype restriction)} \\ DR & \rightarrow & (\geqslant t\ val) \mid (\leqslant t\ val) \mid (= t\ val) \end{array}$$

e.g. $Sedan \sqcap (\leqslant price\ 26.000)$

+

$$\begin{array}{lcl} C & \rightarrow & \forall t.d \mid \exists t.d \quad \text{(fuzzy constraints)} \\ d & \rightarrow & ls(a,b) \mid rs(a,b) \mid tri(a,b,c) \mid trz(a,b,c,d) \end{array}$$

e.g. $Car \sqcap (\exists price.ls(22000, 26000))$

+

$$\begin{array}{lcl} C & \rightarrow & TC \quad \text{(threshold concept)} \\ TC & \rightarrow & C[\geqslant n] \mid C[\leqslant n] \end{array}$$

e.g. $(Sedan \sqcap Cheap \sqcap (\leqslant price\ 30.000))[\geqslant 0.8]$

+

$$\begin{array}{lcl} C & \rightarrow & WC \quad \text{(weighted sum concept)} \\ WC & \rightarrow & (w_1 \cdot C_1 + w_2 \cdot C_2 + \ldots + w_k \cdot C_k) \end{array}$$

where $\sum_{i=1}^{k} w_i = 1$. E.g., $0.2 \cdot (\leqslant price\ 30.000) + 0.8 \cdot (\exists \text{hasColor}.Red)$

Some other features

- Variables are allowed in place of truth degrees
  - For instance, "umberto likes more a lemon than an apple"

$$\langle (umberto, lemon) : Likes, x_1 \rangle \quad , \quad \langle (umberto, apple) : Likes, x_2 \rangle \quad , \quad x_1 \geqslant x_2$$

- Variables are allowed in place of numeric values
  - For instance, Fahrenheit to Celsius conversion

$$(= hasDegreeCelsius\ v_1) = (= hasDegreeFahrenheit\ v_2) \quad , \quad v_1 = (v_2 - 32) * \frac{5}{9}$$

- Full support of Mixed Integer Linear Programming
  - linear inequations can be considered

Constraint	Semantics
(linear inequation)	$a_1 var_1 + \cdots + a_k * var_k \bowtie n$
(binary var)	$var \in \{0, 1\}$
(free var)	$var \in (-\infty, \infty)$

- Support for Łukasiewicz and Gödel t-norm, s-norm and implication.

An application: Matchmaking

- Suppose we have a buyer and a seller (agents)
  - the buyer describes what he is intended to buy and his preferences
  - the seller describes what he is intended to sell and his preferences
  - there is some background knowledge
- The objective is determine "an optimal" agreement among the two

  Pareto optimal agreement: The utility of one agent cannot increase without decreasing the utility of the other

Running Example (buying/selling a car)

- A car seller sells a sedan car
- A buyer is looking for a second hand passenger car
- Both the buyer as well as the seller have preferences (constraints)
- Our aim is to find an optimal (Pareto) agreement

Problem Encoding

- Background knowledge: terminology $\mathcal{T}$ (GCI's may be fuzzy)

$$\mathcal{T} = \begin{cases} \mathit{Sedan} \sqsubseteq \mathit{PassengerCar} \\ \mathit{ExternalColorBlack} \sqsubseteq \neg \mathit{ExternalColorGray} \\ \mathit{SatelliteAlarm} \sqsubseteq \mathit{AlarmSystem} \\ \mathit{InsurancePlus} = \mathit{DriverInsurance} \sqcap \mathit{TheftInsurance} \\ \mathit{NavigatorPack} = \mathit{SatelliteAlarm} \sqcap \mathit{GPS\_system} \end{cases}$$

Hard Contraints

- The seller and the buyer model with concept definition σ and β the minimal requirements they accept for the negotiation
  - $\mathcal{T} \cup \{\sigma, \beta\}$ has to be satisfiable
  - possible agreement: interpretation $\mathcal{I}$ between β and σ such that $\mathcal{I} \models \mathcal{T} \cup \{\sigma, \beta\}$
  - Note: if $\mathcal{T} \cup \{\sigma, \beta\}$ has no models, then the negotiation ends immediately
- E.g., for the buyer, β is

$$B = (\mathit{PassengerCar} \sqcap (\leqslant \mathit{price}\ 30000))[\geqslant 1]$$

- E.g., for the seller, σ is

$$S = (\mathit{Sedan} \sqcap (\geqslant \mathit{price}\ 22000))[\geqslant 1]$$

Soft Contraints

- The seller and the buyer model preferences with concept definitions β_i and σ_i (B_i and S_i are the defined concept names)
- The buyer's negotiation preference $\mathcal{B}$ is a concept definition of the form
$$PrefB = n_1 \cdot B_1 + \ldots + n_k \cdot B_k$$
- The seller's negotiation preference $\mathcal{S}$ is a concept definition of the form
$$PrefS = m_1 \cdot S_1 + \ldots + m_h \cdot S_h$$
- The weights n_i, m_j determine the importance of the preferences

Example: Buyer's Soft Contraints

$$\beta_1: \; B_1 = \textit{DriverInsurance} \sqcap (\textit{TheftInsurance} \sqcup \textit{FireInsurance})$$

$$\beta_2: \; B_2 = \textit{AirConditioning} \sqcap (\textit{ExternalColorBlack} \sqcup \textit{ExternalColorGray})$$

$$\beta_3: \; B_3 = \exists \textit{price}.\textit{ls}(25000, 30000)$$

$$\beta_4: \; B_4 = \exists \textit{km\_warranty}.\textit{rs}(120000, 180000)$$

$$\mathcal{B}: \; \textit{PrefB} = (0.05 \cdot B_1 + 0.05 \cdot B_2 + 0.8 \cdot B_3 + 0.1 \cdot B_4)$$

Example: Seller's Soft Contraints

σ_1: $S_1 = \mathit{InsurancePlus}$

σ_2: $S_2 = \exists \mathit{km\_warranty}.\mathit{ls}(140000, 175000)$

σ_3: $S_3 = \exists \mathit{price}.\mathit{rs}(22000, 26000)$

σ_4: $S_4 = \mathit{ExternalColorBlack} \sqcap \mathit{AirConditioning}$

$\mathcal{S}$: $\mathit{PrefS} = (0.05 \cdot S_1 + 0.3 \cdot S_2 + 0.6 \cdot S_3 + 0.05 \cdot S_4)$

Pareto Agreements

- Given background theory $\mathcal{T}$
  - Buyer's hard constraint β and Seller's hard constraint σ
  - Buyer's negotiation preference $\mathcal{B}$: $PrefB = n_1 \cdot B_1 + \ldots + n_k \cdot B_k$
  - Seller's negotiation preference $\mathcal{S}$: $PrefS = m_1 \cdot S_1 + \ldots + m_h \cdot S_h$
- Consider $\bar{\mathcal{K}}$, where

$$\begin{array}{rcl} \bar{\mathcal{T}} & := & \mathcal{T} \cup \{\beta, \sigma\} \cup \bigcup_i \{\beta_i\} \cup \bigcup_j \{\sigma_j\} \\ & & \cup \{Buy = B \sqcap PrefB[\geqslant t_b]\} \\ & & \cup \{Sell = S \sqcap PrefS[\geqslant t_s\} \\ & & \cup \{Match = Buy \sqcap Sell\} \end{array}$$

- Pareto agreement value v_P: $v_P = bsb(\bar{\mathcal{K}}, Match)$
  - where *best satisfiability bound* of a concept C is:

$$bsb(\mathcal{K}, C) = \sup_{\mathcal{I} \models \mathcal{K}} \sup_{x \in \Delta^{\mathcal{I}}} C^{\mathcal{I}}(x) \ .$$

- Pareto agreement: model $\mathcal{I}$ of $\bar{K}$ such that

$$v_P = \sup_{x \in \Delta^{\mathcal{I}}} Match^{\mathcal{I}}(x) > 0$$

  that is the *Pareto agreement value* is attained at $\mathcal{I}$ and has to be positive.
- Note that:
  - Pareto agreement value v_P is unique and optimal
  - Pareto agreement $\mathcal{I}$ is not unique

Running Example

- Pareto Agreement value: $v_P := 0.783333$
- Pareto Agreement: $\mathcal{I}$

$$km\_warranty = 140000,$$
$$price = 25000$$