

# All About Fuzzy Description Logics

Umberto Straccia

ISTI-CNR, Pisa, Italy  
straccia@isti.cnr.it

1

### About Vagueness

- On the Existence of Vague Concepts
- On the Existence of Vague Objects
- Vague Statements
- Sources of Vagueness
- Uncertainty vs Vagueness: a clarification

2

### From Fuzzy Sets to Mathematical Fuzzy Logic

- Fuzzy Sets Basics
- Mathematical Fuzzy Logics Basics

3

### Fuzzy Description Logics and OWL 2

- Crisp DLs
- Fuzzy DLs
- Representing Fuzzy OWL Ontologies in OWL
- Reasoning Problems and Algorithms

## About Vagueness

From Fuzzy Sets to Mathematical Fuzzy Logic  
Fuzzy Description Logics and OWL 2

On the Existence of Vague Concepts

On the Existence of Vague Objects

Vague Statements

Sources of Vagueness

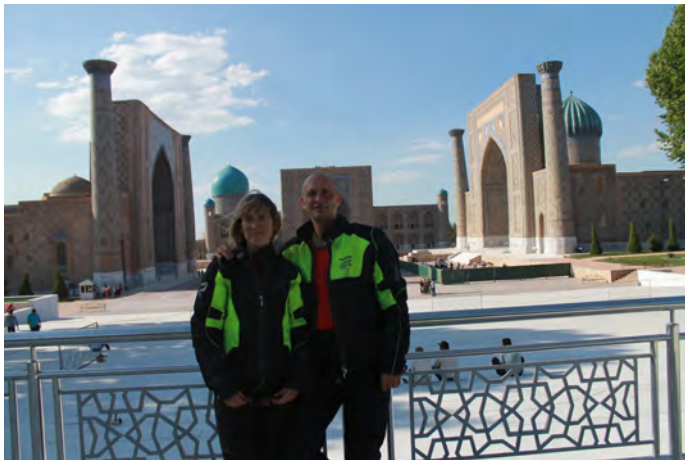
Uncertainty vs Vagueness: a clarification

# About Vagueness

# On the Existence of Vague Concepts

What are vague concepts and do they exist?

- What are the pictures about?



## About Vagueness

From Fuzzy Sets to Mathematical Fuzzy Logic  
Fuzzy Description Logics and OWL 2

## On the Existence of Vague Concepts

On the Existence of Vague Objects  
Vague Statements  
Sources of Vagueness  
Uncertainty vs Vagueness: a clarification



## About Vagueness

From Fuzzy Sets to Mathematical Fuzzy Logic  
Fuzzy Description Logics and OWL 2

## On the Existence of Vague Concepts

On the Existence of Vague Objects  
Vague Statements  
Sources of Vagueness  
Uncertainty vs Vagueness: a clarification



- A **concept is vague** whenever its extension is deemed lacking in clarity
  - **Aboutness** of a picture or piece of text
  - **Tall** person
  - **High** temperature
  - **Nice** weather
  - **Adventurous** trip
  - **Similar** proof
- Vague concepts:
  - Are abundant in everyday speech and almost inevitable
  - Their meaning is often subjective and context dependent

# On the Existence of Vague Objects

What are vague objects and do they exist?

- Are there vague objects in the pictures?



## About Vagueness

From Fuzzy Sets to Mathematical Fuzzy Logic  
Fuzzy Description Logics and OWL 2

## On the Existence of Vague Concepts

### On the Existence of Vague Objects

Vague Statements

Sources of Vagueness

Uncertainty vs Vagueness: a clarification



## About Vagueness

From Fuzzy Sets to Mathematical Fuzzy Logic  
Fuzzy Description Logics and OWL 2

On the Existence of Vague Concepts

On the Existence of Vague Objects

Vague Statements

Sources of Vagueness

Uncertainty vs Vagueness: a clarification

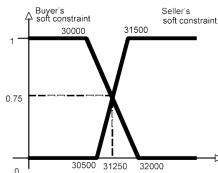


- An **object is vague** whenever its identity is lacking in clarity
  - **Dust**
  - **Cloud**
  - **Dunes**
  - **Sun**
- Vague objects:
  - Are not identical to anything, except to themselves (reflexivity)
  - Are characterised by a **vague identity** relation (e.g. a **similarity** relation)
- BTW: example of *uncertain object*: “habitable Earth-like planet in universe”

# Vague Statements

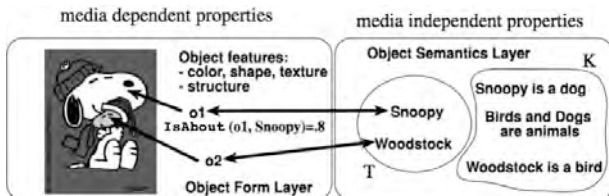
- A **statement is vague** whenever it involves vague concepts or vague objects
  - **Heavy** rain
  - **Tall** person
  - **Hot** temperature
- The **truth** of a vague statement is a matter of **degree**, as it is intrinsically difficult to establish whether the statement is entirely true or false
  - There are 33 °C. Is it **hot**?

# Sources of Vagueness: Matchmaking



- A car seller sells an Audi TT for 31500 €, as from the catalog price.
- A buyer is looking for a sports-car, but wants to pay not more than around 30000 €
- Classical DLs: the problem relies on the crisp conditions on price.
- More fine grained approach: to consider prices as vague constraints (fuzzy sets) (as usual in negotiation)
  - Seller would sell above 31500 €, but can go down to 30500 €
  - The buyer prefers to spend less than 30000 €, but can go up to 32000 €
  - Highest degree of matching is 0.75 . The car may be sold at 31250 €.

# Sources of Vagueness: Multimedia information retrieval

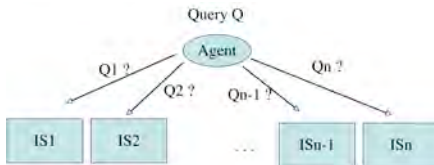


IsAbout		
ImageRegion	Object ID	degree
o1	snoopy	0.8
o2	woodstock	0.7
⋮	⋮	
⋮	⋮	

“Find top-k image regions about animals”

$Query(x) \leftarrow ImageRegion(x) \wedge isAbout(x, y) \wedge Animal(y)$

# Sources of Vagueness: Distributed Information Retrieval



Then the agent has to perform **automatically** the following steps:

- 1 The agent has to select a subset of relevant resources  $\mathcal{S}' \subseteq \mathcal{S}$ , as it is not reasonable to assume to access to and query all resources (**resource selection/resource discovery**);
- 2 For every selected source  $S_i \in \mathcal{S}'$  the agent has to reformulate its information need  $Q_A$  into the query language  $\mathcal{L}_i$  provided by the resource (**schema mapping/ontology alignment**);
- 3 The results from the selected resources have to be merged together

## Sources of Vagueness: Vague database query

<i>HotelID</i>	<i>hasLoc</i>	<i>ConferenceID</i>	<i>hasLoc</i>
<i>h1</i>	<i>hl1</i>	<i>c1</i>	<i>cl1</i>
<i>h2</i>	<i>hl2</i>	<i>c2</i>	<i>cl2</i>
⋮	⋮	⋮	⋮

<i>hasLoc</i>	<i>hasLoc</i>	<i>distance</i>	<i>hasLoc</i>	<i>hasLoc</i>	<i>close</i>	<i>cheap</i>
<i>hl1</i>	<i>cl1</i>	300	<i>hl1</i>	<i>cl1</i>	0.7	0.3
<i>hl1</i>	<i>cl2</i>	500	<i>hl1</i>	<i>cl2</i>	0.5	0.5
<i>hl2</i>	<i>cl1</i>	750	<i>hl2</i>	<i>cl1</i>	0.25	0.8
<i>hl2</i>	<i>cl2</i>	800	<i>hl2</i>	<i>cl2</i>	0.2	0.9
⋮	⋮		⋮	⋮	⋮	

“Find top-*k* cheapest hotels close to the train station”

$$q(h) \leftarrow \text{hasLocation}(h, hl) \wedge \text{hasLocation}(\text{train}, cl) \wedge \text{close}(hl, cl) \wedge \text{cheap}(h)$$

# Sources of Vagueness: Health-care: diagnosis of pneumonia



## Health Care Guideline:

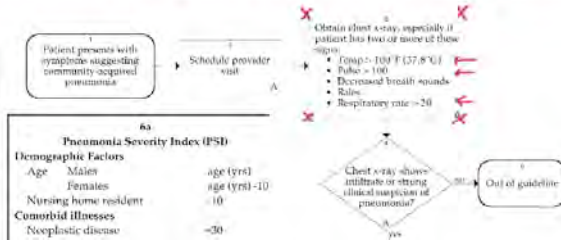
## Community-Acquired Pneumonia in Adults

INSTITUTE FOR CLINICAL  
SYSTEMS IMPROVEMENT

Seventh Edition  
May 2006

**Work Group Leader**  
John Degelau, MD  
*Internal Medicine,  
HealthPartners Medical Group*

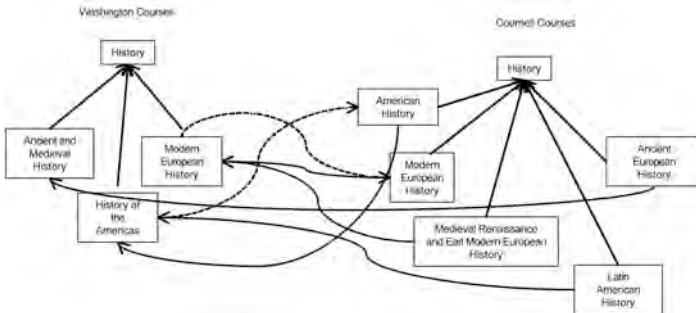
**Work Group Members**  
Family Medicine  
Garrett Tinker, MD



- E.g.,  $Temp = 37.5$ ,  $Pulse = 98$ ,  $RespiratoryRate = 18$  are in “danger zone” already
- Temperature, Pulse and Respiratory rate: these constraints are rather vague

# Sources of Vagueness: Ontology alignment (schema matching)

- To which **degree** are two concepts of two ontologies similar?



## Sources of Vagueness: Lifezone mapping

- To which **degree** do certain areas have a specific bioclimate



Holdridge life zones of USA

# Sources of Vagueness: ARPAT, Air quality in the province of Lucca

I dati di domenica 21/02/2010

Sintesi dei dati rilevati dalle ore 0 alle ore 24 del giorno domenica 21/02/2010

Stazione		Tipo stazione	SO <sub>2</sub> µg/m <sup>3</sup> (media su 24h)	NO <sub>2</sub> µg/m <sup>3</sup> (max oraria)	CO mg/m <sup>3</sup> (max oraria)	O <sub>3</sub> µg/m <sup>3</sup> (max oraria)	PM <sub>10</sub> µg/m <sup>3</sup> (media su 24h)	Giudizio di qualità dell'aria
Lucca	P.za San Michele (RETE REGIONALE **)	urbana - traffico	1	75	---	---	37	Accettabile
Lucca	V.le Carducci	urbana - traffico	1	---	2,3	---	49	Accettabile
Lucca	Carignano (RETE REGIONALE **)	rurale - fondo	---	---	---	86 (h.16*)	---	Buona
Viareggio	Largo Risorgimento	urbana - traffico	---	---	1,8	---	n.d.	Buona
Viareggio	Via Maroncelli (RETE REGIONALE **)	urbana - fondo	4	97	---	61 (h.15*)	33	Accettabile
Capannori	V. di Piaggia (RETE REGIONALE **)	urbana - fondo	---	62	1,3	---	25	Accettabile
Porcari	V. Carrara (RETE REGIONALE **)	periferica - fondo	1	51	---	84 (h.15*)	24	Accettabile

Giudizio di qualità	SO <sub>2</sub> µg/m <sup>3</sup> (media su 24h)	NO <sub>2</sub> µg/m <sup>3</sup> (max oraria)	CO mg/m <sup>3</sup> (max oraria)	O <sub>3</sub> µg/m <sup>3</sup> (max oraria)	PM <sub>10</sub> µg/m <sup>3</sup> (media su 24h)
Buona	0-50	0-50	0-2,5	0-120	0-25
Accettabile	51-125	51-200	2,6-15	121-180	26-50
Scadente	126-250	201-400	15,1-30	181-240	51-74
Pessima	>250	>400	>30	>240	>74

Sintesi dei dati rilevati dalle ore 0 alle ore 24 del giorno domenica 14/02/2010

Stazione		Tipo stazione	SO <sub>2</sub> µg/m <sup>3</sup> (media su 24h)	NO <sub>2</sub> µg/m <sup>3</sup> (max oraria)	CO mg/m <sup>3</sup> (max oraria)	O <sub>3</sub> µg/m <sup>3</sup> (max oraria)	PM <sub>10</sub> µg/m <sup>3</sup> (media su 24h)	Giudizio di qualità dell'aria
Lucca	P.za San Micheletto (RETE REGIONALE **)	urbana - traffico	1	75	---	---	56	Scadente
Lucca	V.le Carducci	urbana - traffico	2	---	2	---	75	Pessima
Lucca	Carignano (RETE REGIONALE **)	rurale - fondo	---	---	---	87 (h.18*)	---	Buona
Viareggio	Largo Risorgimento	urbana - traffico	---	---	1,7	---	n.d.	Buona
Viareggio	Via Maroncelli (RETE REGIONALE **)	urbana - fondo	1	121	---	60 (h.17*)	45	Accettabile
Capannori	V. di Piaggia (RETE REGIONALE **)	urbana - fondo	---	79	2	---	59	Scadente
Porcari	V. Carrara (RETE REGIONALE **)	periferica - fondo	2	72	---	82 (h.16*)	63	Scadente

Giudizio di qualità	SO <sub>2</sub> µg/m <sup>3</sup> (media su 24h)	NO <sub>2</sub> µg/m <sup>3</sup> (max oraria)	CO mg/m <sup>3</sup> (max oraria)	O <sub>3</sub> µg/m <sup>3</sup> (max oraria)	PM <sub>10</sub> µg/m <sup>3</sup> (media su 24h)
Buona	0-50	0-50	0-2,5	0-120	0-25
Accettabile	51-125	51-200	2,6-15	121-180	26-50
Scadente	126-250	201-400	15,1-30	181-240	51-74
Pessima	>250	>400	>30	>240	>74

# TripAdvisor: Hotel User Judgments

## 2,889 Reviews from our TripAdvisor Community



Your overall rating of this property



[Click to rate](#)

### Traveler rating



### See reviews for



### Rating summary



# Uncertainty vs Vagueness: a clarification

- Initial difficulty:
  - Understand the conceptual differences between **uncertainty** and **vagueness**
- Main problem:
  - Interpreting a **degree** as a measure of **uncertainty** rather than as a measure of **vagueness**

# Uncertain Statements

- A statement is **true** or **false** in any world/interpretation
  - We are “**uncertain**” about which world to consider
  - We may have e.g. a probability distribution over possible worlds
- E.g., “it will rain tomorrow”
  - We cannot exactly establish whether it will rain tomorrow or not, due to our **incomplete** knowledge about our world
  - We can estimate to which **degree** this is **probable**

- Consider a propositional statement (formula)  $\phi$
- Interpretation (world)  $\mathcal{I} \in \mathcal{W}$ ,

$$\mathcal{I} : \mathcal{W} \rightarrow \{0, 1\}$$

- $\mathcal{I}(\phi) = 1$  means  $\phi$  is true in  $\mathcal{I}$ , denoted  $\mathcal{I} \models \phi$
- Each interpretation  $\mathcal{I}$  depicts some concrete world
- Given  $n$  propositional letters,  $|\mathcal{W}| = 2^n$
- In uncertainty theory, we do not know which interpretation  $\mathcal{I}$  is the actual one

- One may construct a **probability distribution** over the worlds

$$Pr : \mathcal{W} \rightarrow [0, 1]$$
$$\sum_{\mathcal{I}} Pr(\mathcal{I}) = 1$$

- $Pr(\mathcal{I})$  indicates the probability that  $\mathcal{I}$  is the actual world
- **Probability**  $Pr(\phi)$  of a statement  $\phi$  in  $Pr$

$$Pr(\phi) = \sum_{\mathcal{I} \models \phi} Pr(\mathcal{I})$$

- $Pr(\phi)$  is the probability of the event: " $\phi$  is true"

# Vague Statements

- A statement is **true** to some **degree**, which is taken from a truth space (usually  $[0, 1]$ )
- The convention prescribing that a proposition is either true or false is changed towards **graded propositions**
- E.g., “heavy rain”
  - The compatibility of “heavy” in the phrase “heavy rain” is graded and the degree depends on the amount of rain is falling
    - The intensity of precipitation is expressed in terms of a precipitation rate  $R$ : volume flux of precipitation through a horizontal surface, i.e.  $m^3/m^2s = ms^{-1}$
    - It is usually expressed in  $mm/h$

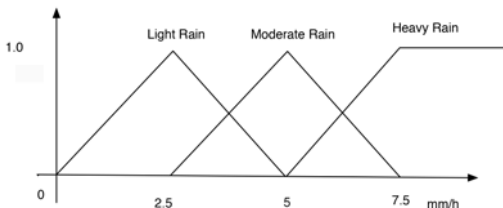
“Heavy rain” continued...E.g., in weather forecasts one may find:

- Rain intensity measured as precipitation rate  $R$ : volume flux of precipitation through a horizontal surface, i.e.  $m^3/m^2h = mh^{-1}$

Rain.	Falling drops of water larger than 0.5 mm in diameter. “Rain” usually implies that the rain will fall steadily over a period of time;
Light rain.	Rain falls at the rate of 2.6 mm or less an hour;
Moderate rain.	Rain falls at the rate of 2.7 mm to 7.6 mm an hour;
Heavy rain.	Rain falls at the rate of 7.7 mm an hour or more.

- Quite harsh distinction:  $R = 7.7mm/h \rightarrow$  heavy rain  
 $R = 7.6mm/h \rightarrow$  moderate rain
- This is clearly unsatisfactory, as quite naturally
  - The more rain is falling, the more the sentence “heavy rain” is true
  - Vice-versa, the less rain is falling the less the sentence is true

- In other words, that the sentence “heavy rain” is no longer either true or false, but is **intrinsically graded**
  - Even if we have complete knowledge about the current world, i.e. exact specification of the precipitation rate
- More fine grained approach:
  - Define the various types of rains as



- Light rain, moderate rain and heavy rain are **vague concepts**

- Consider a propositional statement  $\phi$
- A propositional interpretation  $\mathcal{I}$  maps  $\phi$  to a truth degree in  $[0, 1]$

$$\mathcal{I}(\phi) \in [0, 1]$$

- I.e., we are unable to establish whether a statement is entirely true or false due the occurrence of vague concept
- Vague statements are truth-functional
  - Degree of truth of a statement can be calculated from the degrees of truth of its constituents
  - Note that this is not possible for uncertain statements
- Example of truth functional interpretation of vague statements:

$$\begin{aligned}\mathcal{I}(\phi \wedge \psi) &= \min(\mathcal{I}(\phi), \mathcal{I}(\psi)) \\ \mathcal{I}(\phi \vee \psi) &= \max(\mathcal{I}(\phi), \mathcal{I}(\psi)) \\ \mathcal{I}(\neg\phi) &= 1 - \mathcal{I}(\phi)\end{aligned}$$

# Uncertain Vague Statements

- Recap:

- In a **probabilistic** setting each statement is either true or false, but there is e.g. a probability distribution telling us how probable each interpretation/sentence is

$$\mathcal{I}(\phi) \in \{0, 1\}, Pr(\mathcal{I}) \in [0, 1] \text{ and } Pr(\phi) = \sum_{\mathcal{I} \models \phi} Pr(\mathcal{I}) \in [0, 1]$$

- In **vagueness** theory instead, sentences are **graded**

$$\mathcal{I}(\phi) \in [0, 1]$$

- Are there sentences combining the two orthogonal concepts of uncertainty and vagueness?
- Yes, and we use them daily !
  - E.g. “there will be heavy rain tomorrow”
- This type of sentences are called **uncertain vague sentences**
- Essentially, there is
  - **uncertainty** about the world we will have tomorrow
  - **vagueness** about the various types of rain

- Consider a propositional statement  $\phi$
- A model for uncertain vague sentences:
  - Define probability distribution over worlds  $\mathcal{I} \in \mathcal{W}$ , i.e.

$$Pr(\mathcal{I}) \in [0, 1], \sum_{\mathcal{I}} Pr(\mathcal{I}) = 1$$

- Sentences are graded: each interpretation  $\mathcal{I} \in \mathcal{W}$  is truth functional and maps sentences into  $[0, 1]$

$$\mathcal{I}(\phi) \in [0, 1]$$

- For a sentence  $\phi$ , consider the **expected truth** of  $\phi$

$$ET(\phi) = \sum_{\mathcal{I}} Pr(\mathcal{I}) \cdot \mathcal{I}(\phi) .$$

- Note: if  $\mathcal{I}$  is bivalent (that is,  $\mathcal{I}(\phi) \in \{0, 1\}$ ) then  $ET(\phi) = Pr(\phi)$

# From Fuzzy Sets to Mathematical Fuzzy Logic

# Fuzzy Sets Basics

*From Crisp Sets to Fuzzy Sets.*

- Let  $X$  be a **universal set** of objects
- The **power set**, denoted  $2^A$ , of a set  $A \subset X$ , is the set of subsets of  $A$ , i.e.,

$$2^A = \{B \mid B \subseteq A\}$$

- Often sets are defined as

$$A = \{x \mid P(x)\}$$

- $P(x)$  is a statement “ $x$  has property  $P$ ”
- $P(x)$  is either **true** or **false** for any  $x \in X$

- Examples of universe  $X$  and subsets  $A, B \in 2^X$  may be

$$X = \{x \mid x \text{ is a day}\}$$

$$A = \{x \mid x \text{ is a rainy day}\}$$

$$B = \{x \mid x \text{ is a day with precipitation rate } R \geq 7.5\text{mm/h}\}$$

- In the above case:  $B \subseteq A \subseteq X$
- The **membership function** of a set  $A \subseteq X$ :

$$\chi_A: X \rightarrow \{0, 1\}$$

where  $\chi_A(x) = 1$  iff  $x \in A$

- Note that for sets  $A, B \in 2^X$

$$A \subseteq B \text{ iff } \forall x \in X. \chi_A(x) \leq \chi_B(x)$$

- **Complement** of a set  $A$ , i.e.  $\bar{A} = X \setminus A$ :  $\forall x \in X$ :

$$\chi_{\bar{A}}(x) = 1 - \chi_A(x)$$

- **Intersection** and **union**:  $\forall x \in X$

$$\chi_{A \cap B}(x) = \min(\chi_A(x), \chi_B(x))$$

$$\chi_{A \cup B}(x) = \max(\chi_A(x), \chi_B(x))$$

- **Cartesian product** of two sets  $A, B \in 2^X$

$$A \times B = \{\langle a, b \rangle \mid a \in A, b \in B\}$$

- A relation  $R \subseteq X \times X$

- is **reflexive** if for all  $x \in X$

$$\chi_R(x, x) = 1$$

- is **symmetric** if for all  $x, y \in X$

$$\chi_R(x, y) = \chi_R(y, x)$$

- **Inverse** of  $R$ ,  $\chi_{R^{-1}}: X \times X \rightarrow \{0, 1\}$ :  $\forall x, y \in X$ :

$$\chi_{R^{-1}}(y, x) = \chi_R(x, y)$$

- **Fuzzy set**  $A$ :  $\chi_A: X \rightarrow [0, 1]$ , or simply

$$A: X \rightarrow [0, 1]$$

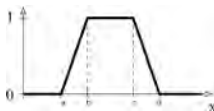
- **Fuzzy power set** over  $X$ , is denoted  $\tilde{2}^X$ , i.e. the set of all fuzzy sets over  $X$
- Example: the fuzzy set

$$C = \{x \mid x \text{ is a day with heavy precipitation rate } R\}$$

is defined via the membership function

$$\chi_C(x) = \begin{cases} 1 & \text{if } R \geq 7.5 \\ (x - 5)/2.5 & \text{if } R \in [5, 7.5) \\ 0 & \text{otherwise} \end{cases}$$

- Fuzzy membership functions may depend on the context and may be subjective
- **Shape** may be quite different
- Usually, it is sufficient to consider functions



(a)



(b)



(c)



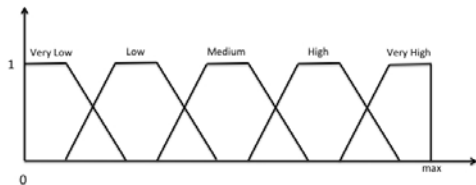
(d)

(a) Trapezoidal  $trz(a, b, c, d)$ ; (b) Triangular  $tri(a, b, c)$ ; (c) left-shoulder  $ls(a, b)$ ; (d) right-shoulder  $rs(a, b)$

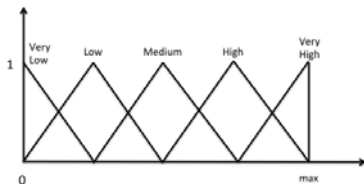
# Fuzzy Sets Construction

- The usefulness of fuzzy sets depends critically on appropriate membership functions
- Methods for fuzzy membership functions construction is largely addressed in literature

- Easy and typically satisfactory method (numerical domain)
  - uniform partitioning into 5 fuzzy sets

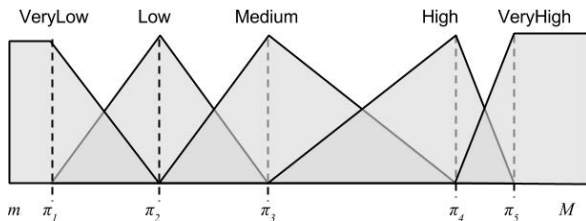


Fuzzy sets construction using trapezoidal functions



Fuzzy sets construction using triangular functions

- Another popular method is based on **clustering**
- Use **Fuzzy C-Means** to cluster data into 5 clusters
  - Fuzzy C-Means extends K-Means to accommodate graded membership
- From the clusters  $c_1, \dots, c_5$  take the centroids  $\pi_1, \dots, \pi_5$
- Build the fuzzy sets from the centroids



Fuzzy sets construction using clustering

# Norm-Based Fuzzy Set Operations

- Standard fuzzy set operations are not the only ones
- Most notable ones are **triangular norms**
  - **t-norm**  $\otimes$  for set intersection
  - **t-conorm**  $\oplus$  (also called **s-norm**) for set union
  - **negation**  $\ominus$  for set complementation
  - **implication**  $\Rightarrow$ 
    - set inclusion  $A \sqsubseteq B$  is defined as

$$\inf_{x \in X} A(x) \Rightarrow B(x)$$

- $\Rightarrow$  is often defined from  $\otimes$  as *r-implication*

$$a \Rightarrow b = \sup \{c \mid a \otimes c \leq b\} .$$

- These functions satisfy some properties that one expects to hold

## Properties for t-norms and s-norms

Axiom Name	T-norm	S-norm
Taututology/Contradiction	$a \otimes 0 = 0$	$a \oplus 1 = 1$
Identity	$a \otimes 1 = a$	$a \oplus 0 = a$
Commutativity	$a \otimes b = b \otimes a$	$a \oplus b = b \oplus a$
Associativity	$(a \otimes b) \otimes c = a \otimes (b \otimes c)$	$(a \oplus b) \oplus c = a \oplus (b \oplus c)$
Monotonicity	if $b \leq c$ , then $a \otimes b \leq a \otimes c$	if $b \leq c$ , then $a \oplus b \leq a \oplus c$

# Properties for implication and negation functions

Axiom Name	Implication Function	Negation Function
Tautology / Contradiction	$0 \Rightarrow b = 1, a \Rightarrow 1 = 1, 1 \Rightarrow 0 = 0$	$\ominus 0 = 1, \ominus 1 = 0$
Antitonicity	if $a \leq b$ , then $a \Rightarrow c \geq b \Rightarrow c$	if $a \leq b$ , then $\ominus a \geq \ominus b$
Monotonicity	if $b \leq c$ , then $a \Rightarrow b \leq a \Rightarrow c$	

- By commutativity,  $\otimes$  and  $\oplus$  are monotone also in the first argument
- $\otimes$  is **idempotent** if  $a \otimes a = a$ , for all  $a \in [0, 1]$
- Negation function  $\ominus$  is **involution** iff  $\ominus \ominus a = a$ , for all  $a \in [0, 1]$ .
- Salient negation functions are:
  - Standard or Łukasiewicz negation:  $\ominus_l a = 1 - a$ ;
  - Gödel negation:  $\ominus_g a$  is 1 if  $a = 0$ , else is 0.
- Łukasiewicz negation is involutive, Gödel negation is not.

- Salient t-norm functions are:

Gödel t-norm:  $a \otimes_g b = \min(a, b)$ ;

Bounded difference or Łukasiewicz t-norm:

$$a \otimes_l b = \max(0, a + b - 1);$$

Algebraic product or product t-norm:  $a \otimes_p b = a \cdot b$ ;

Drastic product:  $a \otimes_d b =$

$$\begin{cases} 0 & \text{when } (a, b) \in [0, 1[ \times [0, 1[ \\ \min(a, b) & \text{otherwise} \end{cases}$$

- Salient s-norm functions are:

Gödel s-norm:  $a \oplus_g b = \max(a, b)$ ;

Bounded sum or Łukasiewicz s-norm:

$$a \oplus_l b = \min(1, a + b);$$

Algebraic sum or product s-norm:  $a \oplus_p b = a + b - ab$ ;

Drastic sum:  $a \oplus_d b =$

$$\begin{cases} 1 & \text{when } (a, b) \in ]0, 1] \times ]0, 1] \\ \max(a, b) & \text{otherwise} \end{cases}$$

## Salient properties of norms:

- Ordering among t-norms ( $\otimes$  is any t-norm):

$$\otimes_d \leq \otimes \leq \otimes_g$$

$$\otimes_d \leq \otimes_l \leq \otimes_p \leq \otimes_g .$$

- The only idempotent t-norm is  $\otimes_g$ .
- The only t-norm satisfying  $a \otimes a = 0$  for all  $a \in [0, 1[$  is  $\otimes_d$ .
- Ordering among s-norms ( $\oplus$  is any s-norm):

$$\oplus_g \leq \oplus \leq \oplus_d$$

$$\oplus_g \leq \oplus_p \leq \oplus_l \leq \oplus_d .$$

- The only idempotent s-norm is  $\oplus_g$ .
- The only s-norm satisfying  $a \oplus a = 1$  for all  $a \in ]0, 1]$  is  $\oplus_d$ .
- The **dual s-norm** of  $\otimes$  is defined as

$$a \oplus b = 1 - (1 - a) \otimes (1 - b) .$$

- **Kleene-Dienes implication**:  $x \Rightarrow y = \max(1 - x, y)$  is called
- **Fuzzy modus ponens**: let  $a \geq n$  and  $a \Rightarrow b \geq m$ 
  - Under Kleene-Dienes implication, we infer that if  $n > 1 - m$  then  $b \geq m$
  - Under r-implication relative to a t-norm  $\otimes$ , we infer that  $b \geq n \otimes m$
- **composition** of two fuzzy relations  $R_1 : X \times X \rightarrow [0, 1]$  and  $R_2 : X \times X \rightarrow [0, 1]$ : for all  $x, z \in X$ 
  - $(R_1 \circ R_2)(x, z) = \sup_{y \in X} R_1(x, y) \otimes R_2(y, z)$
- A fuzzy relation  $R$  is **transitive** iff for all  $x, z \in X$   
 $R(x, z) \geq (R \circ R)(x, z)$

# Łukasiewicz, Gödel, Product logic and Standard Fuzzy logic

- One distinguishes three different sets of fuzzy set operations (called **fuzzy logics**)
  - Łukasiewicz, Gödel, and Product logic
  - Standard Fuzzy Logic (SFL) is a sublogic of Łukasiewicz
    - $\min(a, b) = a \otimes_I (a \Rightarrow_I b)$ ,  $\max(a, b) = 1 - \min(1 - a, 1 - b)$

	Łukasiewicz Logic	Gödel Logic	Product Logic	SFL
$a \otimes b$	$\max(a + b - 1, 0)$	$\min(a, b)$	$a \cdot b$	$\min(a, b)$
$a \oplus b$	$\min(a + b, 1)$	$\max(a, b)$	$a + b - a \cdot b$	$\max(a, b)$
$a \Rightarrow b$	$\min(1 - a + b, 1)$	$\begin{cases} 1 & \text{if } a \leq b \\ b & \text{otherwise} \end{cases}$	$\min(1, b/a)$	$\max(1 - a, b)$
$\ominus a$	$1 - a$	$\begin{cases} 1 & \text{if } a = 0 \\ 0 & \text{otherwise} \end{cases}$	$\begin{cases} 1 & \text{if } a = 0 \\ 0 & \text{otherwise} \end{cases}$	$1 - a$

- Mostert–Shields theorem: any continuous t-norm can be obtained as an ordinal sum of these three

## Some additional properties

Property	Łukasiewicz Logic	Gödel Logic	Product Logic	SFL
$x \otimes \ominus x = 0$	•			
$x \oplus \ominus x = 1$	•			
$x \otimes x = x$		•		•
$x \oplus x = x$		•		•
$\ominus \ominus x = x$	•			•
$x \Rightarrow y = \ominus x \oplus y$	•			•
$\ominus(x \Rightarrow y) = x \otimes \ominus y$	•			•
$\ominus(x \otimes y) = \ominus x \oplus \ominus y$	•	•	•	•
$\ominus(x \oplus y) = \ominus x \otimes \ominus y$	•	•	•	•

- **Note:** If all conditions in the upper part of a column have to be satisfied then we collapse to classical two-valued logic

# Fuzzy Modifiers

- Fuzzy modifiers: interesting feature of fuzzy set theory
- A fuzzy modifier apply to fuzzy sets to change their membership function
  - Examples: **very**, **more\_or\_less**, and **slightly**
- A **fuzzy modifier**  $m$  represents a function

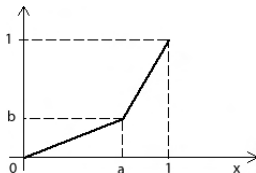
$$f_m: [0, 1] \rightarrow [0, 1]$$

Example:  $f_{\text{very}}(x) = x^2$ ,  $f_{\text{more\_or\_less}}(x) = \text{tri}(0, x, 1)$ ,  $f_{\text{slightly}}(x) = \sqrt{x}$

- Modelling the fuzzy set of **very heavy rain**:

$$\begin{aligned}\chi_{\text{very heavy rain}}(x) &= f_{\text{very}}(\chi_{\text{heavyrain}}(x)) \\ &= (\chi_{\text{heavyrain}}(x))^2 \\ &= (rs(5, 7.5)(x))^2\end{aligned}$$

- A typical shape of modifiers: **linear modifiers**  $lm(a, b)$



- Note: linear modifiers require one parameter  $c$  only

$$lm(a, b) = lm(c)$$

where  $a = c/(c + 1)$ ,  $b = 1/(c + 1)$

# Mathematical Fuzzy Logics Basics

- OWL 2 is grounded on Mathematical Logic
- Fuzzy OWL 2 is grounded on **Mathematical Fuzzy Logic**
- A statement is no longer either true or false, but is graded
- **Truth space**: set of truth values  $L$  with some structure
- Given a statement  $\phi$ 
  - **Fuzzy Interpretation**: a function  $\mathcal{I}$  mapping  $\phi$  into  $L$ , i.e.

$$\mathcal{I}(\phi) \in L$$

- Usually

$$L = [0, 1]$$
$$L_n = \left\{0, \frac{1}{n}, \dots, \frac{n-2}{n-1}, \dots, 1\right\} \quad (n \geq 1)$$

- **Fuzzy statement:** for  $r \in [0, 1]$

$$\langle \phi, r \rangle$$

*The degree of truth of  $\phi$  is equal or greater than  $r$*

- **Examples:**
  - Fuzzy FOL:  $\langle \text{RainyDay}(d), 0.75 \rangle$
  - Fuzzy LPs:  $\langle \text{RainyDay}(d) \leftarrow, 0.75 \rangle$
  - Fuzzy RDFS:  $\langle \langle d, \text{type}, \text{RainyDay} \rangle, 0.75 \rangle$
  - Fuzzy DLs:  $\langle d:\text{RainyDay}, 0.75 \rangle$

- **Fuzzy interpretation  $\mathcal{I}$ :**

- Maps each basic statement  $p_i$  into  $[0, 1]$
- Extended inductively to all statements

$$\mathcal{I}(\phi \wedge \psi) = \mathcal{I}(\phi) \otimes \mathcal{I}(\psi)$$

$$\mathcal{I}(\phi \vee \psi) = \mathcal{I}(\phi) \oplus \mathcal{I}(\psi)$$

$$\mathcal{I}(\phi \rightarrow \psi) = \mathcal{I}(\phi) \Rightarrow \mathcal{I}(\psi)$$

$$\mathcal{I}(\phi \leftrightarrow \psi) = \mathcal{I}(\phi \rightarrow \psi) \otimes \mathcal{I}(\psi \rightarrow \phi)$$

$$\mathcal{I}(\neg\phi) = \ominus \mathcal{I}(\phi)$$

$$\mathcal{I}(\exists x.\phi) = \sup_{a \in \Delta^{\mathcal{I}}} \mathcal{I}_x^a(\phi)$$

$$\mathcal{I}(\forall x.\phi) = \inf_{a \in \Delta^{\mathcal{I}}} \mathcal{I}_x^a(\phi),$$

where

- $\Delta^{\mathcal{I}}$  is the domain of  $\mathcal{I}$
- $\otimes$ ,  $\oplus$ ,  $\Rightarrow$ , and  $\ominus$  are the t-norms, t-conorms, implication functions, a negation functions
- The function  $\mathcal{I}_x^a$  is as  $\mathcal{I}$  except that  $x$  is interpreted as  $a$

## Example

- In Lukasiewicz logic:

$$\varphi = \text{Cold} \wedge \text{Cloudy}$$

$\mathcal{I}$	<i>Cold</i>	<i>Cloudy</i>	$\mathcal{I}(\varphi)$
$\mathcal{I}_1$	0	0.1	$\max(0, 0 + 0.1 - 1) = 0.0$
$\mathcal{I}_2$	0.3	0.4	$\max(0, 0.3 + 0.4 - 1) = 0.0$
$\mathcal{I}_3$	0.7	0.8	$\max(0, 0.7 + 0.9 - 1) = 0.6$
$\mathcal{I}_4$	1	1	$\max(0, 1 + 1 - 1) = 1.0$
$\vdots$	$\vdots$	$\vdots$	$\vdots$

- Note:** given  $m$  propositional letters
  - Fuzzy interpretations over  $L = [0, 1]$  are **not recursively enumerable**
  - There are  $n^m$  fuzzy interpretations over  $L_n$

- One may also consider the following abbreviations:

$$\phi \wedge_g \psi \stackrel{\text{def}}{=} \phi \wedge (\phi \rightarrow \psi)$$

$$\phi \vee_g \psi \stackrel{\text{def}}{=} (\phi \rightarrow \psi) \rightarrow \phi \wedge_g (\psi \rightarrow \phi) \rightarrow \psi$$

$$\neg_{\otimes} \phi \stackrel{\text{def}}{=} \phi \rightarrow 0$$

$$\langle \phi \leq r \rangle \stackrel{\text{def}}{=} \langle \neg_I \phi, 1 - r \rangle$$

- In case  $\Rightarrow$  is the r-implication based on  $\otimes$ , then
  - $\wedge_g$  is Gödel t-norm
  - $\vee_g$  is Gödel s-norm
  - $\neg_{\otimes}$  is interpreted as the negation function related to  $\otimes$

- $\mathcal{I}$  **satisfies**  $\langle \phi, r \rangle$ , or  $\mathcal{I}$  is a **model** of  $\langle \phi, r \rangle$

$$\mathcal{I} \models \langle \phi, r \rangle \text{ iff } \mathcal{I}(\phi) \geq r$$

- $\mathcal{I}$  is a **model** of  $\phi$  if  $\mathcal{I}(\phi) = 1$
- **Fuzzy knowledge base**  $\mathcal{K}$ : finite set of fuzzy statements
- $\mathcal{I}$  **satisfies** (is a **model** of)  $\mathcal{K}$ :  $\mathcal{I} \models \mathcal{K}$  iff it satisfies each element in it
- **Best entailment degree** of  $\phi$  w.r.t.  $\mathcal{K}$ :

$$\text{bed}(\mathcal{K}, \phi) = \sup \{r \mid \mathcal{K} \models \langle \phi, r \rangle\}$$

- **Best satisfiability degree** of  $\phi$  w.r.t.  $\mathcal{K}$ :

$$\text{bsd}(\mathcal{K}, \phi) = \sup_{\mathcal{I}} \{\mathcal{I}(\phi) \mid \mathcal{I} \models \mathcal{K}\}$$

### Proposition (Fuzzy Modus Ponens)

For  $r$ -implication  $\rightarrow$ , for  $r, s \in [0, 1]$ :

$$\langle \phi, r \rangle, \langle \phi \rightarrow \psi, s \rangle \models \langle \psi, r \otimes s \rangle$$

### Proposition

Salient equivalences:

$$\begin{aligned}\neg\neg\phi &\equiv \phi \quad (\perp, SFL) \\ \phi \wedge \phi &\equiv \phi \quad (G, SFL) \\ \neg(\phi \wedge \neg\phi) &\equiv 1 \quad (\perp, G, \Pi) \\ \phi \vee \neg\phi &\equiv 1 \quad (\perp) \\ \forall x.\phi &\equiv \neg\exists x.\neg\phi \quad (\perp, SFL)\end{aligned}$$

## Proposition

*Salient equivalences:*

$$\mathcal{L} + G \equiv \textit{Boolean Logic}$$

$$\mathcal{L} + \Pi \equiv \textit{Boolean Logic}$$

$$G + \Pi \equiv \textit{Boolean Logic}$$

### Proposition (BED)

$bed(\mathcal{K}, \phi) = \min x$ . *such that  $\mathcal{K} \cup \{\langle \phi \leq x \rangle\}$  satisfiable.*

### Proposition (BSD)

$bsd(\mathcal{K}, \phi) = \max x$ . *such that  $\mathcal{K} \cup \{\langle \phi, x \rangle\}$  satisfiable.*

## On Witnessed Models

- **Witnessed interpretation**  $\mathcal{I}$ :

$$\mathcal{I}(\exists x.\phi) = \mathcal{I}_x^a(\phi), \text{ for some } a \in \Delta^{\mathcal{I}} \quad (1)$$

$$\mathcal{I}(\forall x.\phi) = \mathcal{I}_x^a(\phi), \text{ for some } a \in \Delta^{\mathcal{I}} \quad (2)$$

- The supremum (resp. infimum) are attained at some point
- Classical interpretations are witnessed
- Fuzzy interpretations **may not be witnessed**
- E.g.,  $\mathcal{I}$  is not witnessed as Eq. (1) not satisfied:

$$\begin{aligned} \Delta^{\mathcal{I}} &= \mathbb{N} \\ \mathcal{I}_x^n(A(x)) &= 1 - 1/n < 1, \text{ for all } n \\ \mathcal{I}(\exists x.A(x)) &= \sup_n \mathcal{I}_x^n(A(x)) \\ &= \sup_n 1 - 1/n = 1 \end{aligned}$$

## Proposition (Witnessed model property)

*In Łukasiewicz logic and SFL over  $L = [0, 1]$ , or for all cases in which the truth space  $L$  is finite, a fuzzy KB has a witnessed fuzzy model iff it has a fuzzy model.*

- Not true for Gödel and product logic over  $L = [0, 1]$ 
  - $\neg\forall x p(x) \wedge \neg\exists x \neg p(x)$  has no classical model
  - In Gödel logic it has no finite model, but has an **infinite** model: for integer  $n \geq 1$ , let  $\mathcal{I}$  such that  $\mathcal{I}(p(n)) = 1/n$

$$\begin{aligned}\mathcal{I}(\forall x p(x)) &= \inf_n 1/n = 0 \\ \mathcal{I}(\exists x \neg p(x)) &= \sup_n \neg 1/n = \sup_n 0 = 0\end{aligned}$$

- IMHO: non-witnessed models make little sense in KR
- We will always assume that interpretations are witnessed

# Fuzzy Propositional Logic: Reasoning

- We need to distinguish if truth space is  $L = [0, 1]$  or  $L_n = \{0, \frac{1}{n}, \dots, \frac{n-2}{n-1}, \dots, 1\}$
- Case  $L_n$  easier: given  $m$  propositional letters, there are  $m^n$  possible interpretations
- We may use
  - Operational Research
  - Analytic Tableaux, Non-Deterministic Analytic Tableaux
  - Reduction into Classical Propositional Logic

# Operational Research: Case Łukasiewicz Logic & SFL

- Basic idea: translate formulae into equational constraints about truth degrees
- For a formula  $\phi$  consider a variable  $x_\phi$ 
  - Intuition:  $x_\phi$  will hold the degree of truth of statement  $\phi$
  - Example: constraints under Łukasiewicz for  $\langle \neg\phi, 0.6 \rangle$

$$x_{\neg\phi} \in [0, 1]$$

$$x_\phi \in [0, 1]$$

$$x_{\neg\phi} = 1 - x_\phi$$

- We may use **Mixed Integer Linear Programming** for the encodings of constraints

For Łukasiewicz:

- $x_1 \otimes_l x_2 = z$   
 $\mapsto \{x_1 + x_2 - 1 \leq z, x_1 + x_2 - 1 \geq z - y, z \leq 1 - y, y \in \{0, 1\}\}$ ,  
where  $y$  is a new variable.
- $x_1 \oplus_l x_2 = z \mapsto \{x_1 + x_2 \leq z + y, y \leq z, x_1 + x_2 \geq z, y \in \{0, 1\}\}$ ,  
where  $y$  is a new variable.
- $x_1 \Rightarrow_l x_2 = z \mapsto \{(1 - x_1) \oplus_l x_2 = z\}$ .

For SFL:

- $x_1 \otimes_g x_2 = z$   
 $\mapsto \{z \leq x_1, z \leq x_2, x_1 \leq z + y, x_2 \leq z + (1 - y), y \in \{0, 1\}\}$ ,  
where  $y$  is a new variable.
- $x_1 \oplus_g x_2 = z$   
 $\mapsto \{z \geq x_1, z \geq x_2, x_1 + y \geq z, x_2 + (1 - y) \geq z, y \in \{0, 1\}\}$ ,  
where  $y$  is a new variable.
- $x_1 \Rightarrow_{kd} x_2 = z \mapsto (1 - x_1) \oplus_g x_2 = z$ .

- **Negation Normal Form,  $nnf(\phi)$**

$$\neg \perp = \top$$

$$\neg \top = \perp$$

$$\neg \neg \phi \mapsto \phi$$

$$\neg(\phi \wedge \psi) \mapsto \neg\phi \vee \neg\psi$$

$$\neg(\phi \vee \psi) \mapsto \neg\phi \wedge \neg\psi$$

$$\neg(\phi \rightarrow \psi) \mapsto \phi \wedge \neg\psi .$$

- 1 Transform  $\mathcal{K}$  into NNF
- 2 Initialize the fuzzy theory  $\mathcal{T}_{\mathcal{K}}$  and the initial set of constraints  $\mathcal{C}_{\mathcal{K}}$  by

$$\begin{aligned}\mathcal{T}_{\mathcal{K}} &= \{\phi \mid \langle \phi, n \rangle \in \mathcal{K}\} \\ \mathcal{C}_{\mathcal{K}} &= \{x_{\psi} \geq n \mid \langle \phi, n \rangle \in \mathcal{K}\}\end{aligned}$$

- 3 Apply the following inference rules until no more rules can be applied

- (var). For variable  $x_{\phi}$  occurring in  $\mathcal{C}_{\mathcal{K}}$  add  $x_{\phi} \in [0, 1]$  to  $\mathcal{C}_{\mathcal{K}}$
- ( $\bar{\text{var}}$ ). For variable  $x_{\neg\phi}$  occurring in  $\mathcal{C}_{\mathcal{K}}$  add  $x_{\phi} = 1 - x_{\neg\phi}$  to  $\mathcal{C}_{\mathcal{K}}$
- ( $\perp$ ). If  $\perp \in \mathcal{T}_{\mathcal{K}}$  then  $\mathcal{C}_{\mathcal{K}} := \mathcal{C}_{\mathcal{K}} \cup \{x_{\perp} = 0\}$
- ( $\top$ ). If  $\top \in \mathcal{T}_{\mathcal{K}}$  then  $\mathcal{C}_{\mathcal{K}} := \mathcal{C}_{\mathcal{K}} \cup \{x_{\top} = 1\}$
- ( $\wedge$ ). If  $\phi \wedge \psi \in \mathcal{T}_{\mathcal{K}}$ , then
  - 1 add  $\phi$  and  $\psi$  to  $\mathcal{T}_{\mathcal{K}}$
  - 2  $\mathcal{C}_{\mathcal{K}} := \mathcal{C}_{\mathcal{K}} \cup \{x_{\phi} \otimes x_{\psi} = x_{\phi \wedge \psi}\}$
- ( $\vee$ ). If  $\phi \vee \psi \in \mathcal{T}_{\mathcal{K}}$ , then
  - 1 add  $\phi$  and  $\psi$  to  $\mathcal{T}_{\mathcal{K}}$
  - 2  $\mathcal{C}_{\mathcal{K}} := \mathcal{C}_{\mathcal{K}} \cup \{x_{\phi} \oplus x_{\psi} = x_{\phi \vee \psi}\}$
- ( $\rightarrow$ ). If  $\phi \rightarrow \psi \in \mathcal{T}_{\mathcal{K}}$ , then
  - 1 add  $\text{nnf}(\neg\phi)$  and  $\psi$  to  $\mathcal{T}_{\mathcal{K}}$
  - 2  $\mathcal{C}_{\mathcal{K}} := \mathcal{C}_{\mathcal{K}} \cup \{(1 - x_{\text{nnf}(\neg\phi)}) \Rightarrow x_{\psi} = x_{\phi \rightarrow \psi}\}$

***sat*( $\mathcal{K}$ ):**  $\mathcal{K}$  is satisfiable iff the final set of constraints  $\mathcal{C}_{\mathcal{K}}$  has a solution

- bed*( $\mathcal{K}, \phi$ ):**
- Add  $\neg\phi$  to  $\mathcal{T}_{\mathcal{K}}$
  - Add  $x_{\neg\phi} \geq 1 - x, x \in [0, 1]$  to  $\mathcal{C}_{\mathcal{K}}$ ,  $x$  new
  - Compute final set of constraints  $\mathcal{C}_{\mathcal{K}}$
  - Then, solve the optimisation problem

$bed(\mathcal{K}, \phi) = \min x.$  such that  $\mathcal{C}_{\mathcal{K}}$  has a solution

- bsd*( $\mathcal{K}, \phi$ ):**
- Add  $\phi$  to  $\mathcal{T}_{\mathcal{K}}$
  - Add  $x_{\phi} \geq x, x \in [0, 1]$  to  $\mathcal{C}_{\mathcal{K}}$ ,  $x$  new
  - Compute final set of constraints  $\mathcal{C}_{\mathcal{K}}$
  - Then, solve the optimisation problem

$bsd(\mathcal{K}, \phi) = \max x.$  such that  $\mathcal{C}_{\mathcal{K}}$  has a solution

## Analytical Fuzzy Tableau: Case SFL

- Main property the method is based on:
  - if  $\mathcal{I}$  is model of  $\langle \phi \wedge \psi, n \rangle$  then  $\mathcal{I}$  is a model of both  $\langle \phi, n \rangle$  and  $\langle \psi, n \rangle$ ;
  - if  $\mathcal{I}$  is model of  $\langle \phi \vee \psi, n \rangle$  then  $\mathcal{I}$  is a model of either  $\langle \phi, n \rangle$  or  $\langle \psi, n \rangle$ .
  - $\mathcal{I}$  cannot be a model of both  $\langle p, n \rangle$  and  $\langle \neg p, m \rangle$  if  $n > 1 - m$ .
- A **clash** is either
  - a fuzzy statement  $\langle \perp, n \rangle$  with  $n > 0$ ; or
  - a pair of fuzzy statements  $\langle p, n \rangle$  and  $\langle \neg p, m \rangle$  with  $n > 1 - m$
- **Clash-free**: does not contain a clash

- 1 Transform  $\mathcal{K}$  into NNF
- 2 Initialize the completion  $\mathcal{S}_{\mathcal{K}} = \mathcal{K}$
- 3 Apply the following inference rules to  $\mathcal{S}_{\mathcal{K}}$  until no more rules can be applied
- 4 We call a set of fuzzy statements  $\mathcal{S}_{\mathcal{K}}$  **complete** iff none of the rules below can be applied to  $\mathcal{S}_{\mathcal{K}}$
- 5 Note that rule ( $\vee$ ) is non-deterministic
  - ( $\wedge$ ). If  $\langle \phi \wedge \psi, n \rangle \in \mathcal{S}_{\mathcal{K}}$  and  $\{\langle \phi, n \rangle, \langle \psi, n \rangle\} \not\subseteq \mathcal{S}_{\mathcal{K}}$ , then add both  $\langle \phi, n \rangle$  and  $\langle \psi, n \rangle$  to  $\mathcal{S}_{\mathcal{K}}$
  - ( $\vee$ ). If  $\langle \phi \vee \psi, n \rangle \in \mathcal{S}_{\mathcal{K}}$  and  $\{\langle \phi, n \rangle, \langle \psi, n \rangle\} \cap \mathcal{S}_{\mathcal{K}} = \emptyset$ , then add either  $\langle \phi, n \rangle$  or  $\langle \psi, n \rangle$  to  $\mathcal{S}_{\mathcal{K}}$
  - ( $\rightarrow$ ). If  $\langle \phi \rightarrow \psi, n \rangle \in \mathcal{S}_{\mathcal{K}}$  and  $\langle nnf(\neg\phi) \vee \psi, n \rangle \notin \mathcal{S}_{\mathcal{K}}$ , then add  $\langle nnf(\neg\phi) \vee \psi, n \rangle$  to  $\mathcal{S}_{\mathcal{K}}$

**sat( $\mathcal{K}$ )**:  $\mathcal{K}$  is satisfiable iff we find a complete and clash-free completion  $\mathcal{S}_{\mathcal{K}}$  of  $\mathcal{K}$

- For BED and BSD we need some more work
- Given  $\mathcal{K}$ , define

$$\begin{aligned} N^{\mathcal{K}} &= \{0, 0.5, 1\} \cup \{n \mid \langle \phi, n \rangle \in \mathcal{K}\} \\ \bar{N}^{\mathcal{K}} &= N^{\mathcal{K}} \cup \{1 - n \mid n \in N^{\mathcal{K}}\} \\ \epsilon &= \min\{d/2 \mid n, m \in \bar{N}^{\mathcal{K}}, n \neq m, d = |n - m|\} \end{aligned}$$

## Proposition

*Under SFL, given  $\mathcal{K}$ , then for  $n > 0$*

*$\mathcal{K} \models \langle \phi, n \rangle$  iff  $\mathcal{K} \cup \{\langle \neg\phi, 1 - n + \epsilon \rangle\}$  is not satisfiable .*

*Moreover,  $\mathcal{K}$  is satisfiable iff it has a model over  $\bar{N}^{\mathcal{K}}$ .*

$bed(\mathcal{K}, \phi)$ : Find greatest  $n \in \bar{N}^{\mathcal{K}}$  such that  $\mathcal{K} \models \langle \phi, n \rangle$   
 $bsd(\mathcal{K}, \phi)$ : Find greatest  $n \in \bar{N}^{\mathcal{K}}$  such that  $\mathcal{K} \cup \{\langle \phi, n \rangle\}$   
satisfiable

# Non Deterministic Analytic Fuzzy Tableau

- Works for finitely-valued fuzzy propositional logic over  $L_n$
- Works also for SFL (as in place of  $[0, 1]$ , we may use  $\bar{N}^K$ )
- Basic idea is as for fuzzy tableau, but now we **guess** the truth degrees
  - ( $\wedge$ ). If  $\langle \phi \wedge \psi, n \rangle \in \mathcal{S}_{\mathcal{K}}$ ,  $n_1, n_2 \in L_n$  such that  $n_1 \otimes n_2 = n$  and  $\{\langle \phi, n_1 \rangle, \langle \psi, n_2 \rangle\} \not\subseteq \mathcal{S}_{\mathcal{K}}$ , then add both  $\langle \phi, n_1 \rangle$  and  $\langle \psi, n_2 \rangle$  to  $\mathcal{S}_{\mathcal{K}}$
  - ( $\vee$ ). If  $\langle \phi \vee \psi, n \rangle \in \mathcal{S}_{\mathcal{K}}$ ,  $n_1, n_2 \in L_n$  such that  $n_1 \oplus n_2 = n$  and  $\{\langle \phi, n_1 \rangle, \langle \psi, n_2 \rangle\} \not\subseteq \mathcal{S}_{\mathcal{K}}$ , then add both  $\langle \phi, n_1 \rangle$  and  $\langle \psi, n_2 \rangle$  to  $\mathcal{S}_{\mathcal{K}}$
  - ( $\rightarrow$ ). If  $\langle \phi \rightarrow \psi, n \rangle \in \mathcal{S}_{\mathcal{K}}$ ,  $n_1, n_2 \in L_n$  such that  $n_1 \Rightarrow n_2 = n$  and  $\{\langle \phi, n_1 \rangle, \langle \psi, n_2 \rangle\} \not\subseteq \mathcal{S}_{\mathcal{K}}$ , then add both  $\langle \phi, n_1 \rangle$  and  $\langle \psi, n_2 \rangle$  to  $\mathcal{S}_{\mathcal{K}}$
- A **clash** is either
  - a fuzzy statement  $\langle \perp, n \rangle$  with  $n > 0$ ; or
  - a pair of fuzzy statements  $\langle p, n \rangle$  and  $\langle \neg p, m \rangle$  such that

$$x_p \geq n, \quad \ominus x_p \geq m, \quad x_p \in L_n$$

has no solution

# Reduction to Classical Propositional Logic: Case SFL over $[0, 1]$

- Given  $\mathcal{K}$ , we know that we can use

$$L_n = \bar{N}^{\mathcal{K}} = \{\gamma_1, \dots, \gamma_n\}$$

with  $\gamma_i < \gamma_{i+1}$ ,  $1 \leq i \leq n-1$

- Basic idea: use atom  $A_{\geq r}$  to represent

*The truth degree of  $A$  has to be equal or greater than  $r$*

- Similarly for  $A_{>r}$ ,  $A_{\leq r}$  and  $A_{<r}$

- To start with, build  $Crisp_{L_n}$ 
  - For all atoms  $A$ , for all  $1 \leq i \leq n-1, 2 \leq j \leq n-1$

$$A_{\geq \gamma_{i+1}} \rightarrow A_{> \gamma_i}$$

$$A_{> \gamma_j} \rightarrow A_{\geq \gamma_j}$$

- Build  $Crisp_{\mathcal{K}}$ :

$$Crisp_{\mathcal{K}} = \{\rho(\phi, n) \mid \langle \phi, n \rangle \in \mathcal{K}\} \cup Crisp_{L_n},$$

$x$	$y$	$\rho(x, y)$
$\top$	$c$	$\top$
$\perp$	$0$	$\top$
$\perp$	$c$	$\perp$ if $c > 0$
$A$	$c$	$A_{> c}$
$\neg A$	$c$	$\neg A_{> 1-c}$
$\phi \wedge \psi$	$c$	$\rho(\phi, c) \wedge \rho(\psi, c)$
$\phi \vee \psi$	$c$	$\rho(\phi, c) \vee \rho(\psi, c)$

## Proposition

Given  $\mathcal{K}$  under SFL over  $L_n$ , then  $\mathcal{K} \models \langle \phi, c \rangle$  iff  $\mathcal{K} \cup \{ \langle \neg \phi, 1 - c^- \rangle \}$  is not satisfiable, where  $c^-$  is the next smaller value than  $c$  in  $L_n$

**sat**( $\mathcal{K}$ ):  $\mathcal{K}$  is satisfiable iff  $\text{Crisp}_{\mathcal{K}}$  satisfiable

**bed**( $\mathcal{K}, \phi$ ): Find greatest  $c \in L_n$  such that  $\mathcal{K} \models \langle \phi, c \rangle$

**bsd**( $\mathcal{K}, \phi$ ): Find greatest  $c \in L_n$  such that  $\mathcal{K} \cup \{ \langle \phi, c \rangle \}$  satisfiable

## Fuzzy Description Logics and OWL 2

# The Semantic Web Family of Languages

- Wide variety of languages
  - **RDFS**: *Triple language*, -Resource Description Framework
    - The logical counterpart is  $\rho df$
  - **RIF**: *Rule language*, -Rule Interchange Format,
    - Relate to the *Logic Programming* (LP) paradigm
  - **OWL 2**: *Conceptual language*, -Ontology Web Language
    - Relate to **Description Logics** (DLs)



# OWL 2 Profiles

- OWL 2 EL
  - Useful for large size of properties and/or classes
  - Basic reasoning problems solved in polynomial time
  - The EL acronym refers to the  $\mathcal{EL}$  family of DLs
- OWL 2 QL
  - Useful for very large volumes of instance data
  - Conjunctive query answering via query rewriting and SQL
  - OWL 2 QL relates to the DL family *DL-Lite*
- OWL 2 RL
  - Useful for scalable reasoning without sacrificing too much expressive power
  - OWL 2 RL maps to Datalog
  - Computational complexity: same as for Datalog, polynomial in size of the data, EXPTIME w.r.t. size of knowledge base

# Description Logics (DLs)

- **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

- **Basic ingredients:** descriptions of classes, properties, and their instances, such as
  - $a:C$ , meaning that individual  $a$  is an instance of concept/class  $C$

$a:\text{Person} \sqcap \forall \text{hasChild.Femal}$

- $(a, b):R$ , meaning that the pair of individuals  $\langle a, b \rangle$  is an instance of the property/role  $R$

$(\text{tom}, \text{mary}):\text{hasChild}$

- $C \sqsubseteq D$ , meaning that the class  $C$  is a subclass of class  $D$

$\text{Person} \sqsubseteq \forall \text{hasChild.Person}$

# The DL Family

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

Syntax	Semantics	Example
$C, D \rightarrow$	$\top(x)$	
	$\perp(x)$	
	$A(x)$	<i>Human</i>
	$C \sqcap D$	$Human \sqcap Male$
	$C \sqcup D$	$Nice \sqcup Rich$
	$\neg C$	$\neg Meat$
	$\exists R.C$	$\exists has\_child.Blond$
	$\forall R.C$	$\forall has\_child.Human$
$C \sqsubseteq D$	$\forall x.C(x) \Rightarrow D(x)$	$Happy\_Father \sqsubseteq Man \sqcap \exists has\_child.Female$
$a:C$	$C(a)$	$John:Happy\_Father$

# DL Semantics

- Semantics is given in terms of an **interpretation**  $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ , where
  - $\Delta^{\mathcal{I}}$  is the **domain** (a non-empty set)
  - $\cdot^{\mathcal{I}}$  is an **interpretation function** that maps:
    - **Concept** (class) name  $A$  into a subset  $A^{\mathcal{I}}$  of  $\Delta^{\mathcal{I}}$
    - **Role** (property) name  $R$  into a subset  $R^{\mathcal{I}}$  of  $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
    - **Individual** name  $a$  into an element of  $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$  s.t.  $a^{\mathcal{I}} \neq b^{\mathcal{I}}$  if  $a \neq b$  (UNA)
  - Interpretation function  $\cdot^{\mathcal{I}}$  is extended to concept expressions:

$$\begin{aligned}
 \top^{\mathcal{I}} &= \Delta^{\mathcal{I}} \\
 \perp^{\mathcal{I}} &= \emptyset \\
 (C_1 \sqcap C_2)^{\mathcal{I}} &= C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}} \\
 (C_1 \sqcup C_2)^{\mathcal{I}} &= C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}} \\
 (\neg C)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\
 (\exists R.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \exists y. \langle x, y \rangle \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\} \\
 (\forall R.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \forall y. \langle x, y \rangle \in R^{\mathcal{I}} \Rightarrow y \in C^{\mathcal{I}}\}
 \end{aligned}$$

- Finally, we say that
  - $\mathcal{I}$  is a model of  $C \sqsubseteq D$ , written  $\mathcal{I} \models C \sqsubseteq D$ , iff  $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
  - $\mathcal{I}$  is a model of  $a:C$ , written  $\mathcal{I} \models a:C$ , iff  $a^{\mathcal{I}} \in C^{\mathcal{I}}$
  - $\mathcal{I}$  is a model of  $(a, b):R$ , written  $\mathcal{I} \models (a, b):R$ , iff  $\langle a^{\mathcal{I}}, b^{\mathcal{I}} \rangle \in R^{\mathcal{I}}$

# Note on DL Naming

- $\mathcal{AL}$ :  $C, D \rightarrow \top \mid \perp \mid A \mid C \sqcap D \mid \neg A \mid \exists R.C \mid \forall R.C$
- $\mathcal{C}$ : 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. *is\_component\_of*  $\sqsubseteq$  *is\_part\_of*
  - $\mathcal{N}$ : Number restrictions,  $(\geq n R)$  and  $(\leq n R)$ , e.g.  $(\geq 3 \textit{ has\_Child})$  (has at least 3 children)
  - $\mathcal{Q}$ : Qualified number restrictions,  $(\geq n R.C)$  and  $(\leq n R.C)$ , e.g.  $(\leq 2 \textit{ has\_Child.Adult})$  (has at most 2 adult children)
  - $\mathcal{O}$ : Nominals (singleton class),  $\{a\}$ , e.g.  $\exists \textit{ has\_child}.\{mary\}$ .  
**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. *isPartOf* = *hasPart*<sup>-</sup>
  - $\mathcal{F}$ : Functional role,  $f$ , e.g. *functional(hasAge)*
  - $\mathcal{R}_+$ : transitive role, e.g. *transitive(isPartOf)*

For instance,

$$\begin{aligned}
 SHIF &= S + H + I + F = \mathcal{ALCR}_+HIF && \text{OWL-Lite} \\
 SHOIN &= S + H + O + I + N = \mathcal{ALCR}_+HOIN && \text{OWL-DL} \\
 SROIQ &= S + R + O + I + Q = \mathcal{ALCR}_+ROIN && \text{OWL 2}
 \end{aligned}$$

## Semantics of Additional Constructs

- $\mathcal{H}$ : Role inclusion axioms,  $\mathcal{I} \models R_1 \sqsubseteq R_2$  iff  $R_1^{\mathcal{I}} \subseteq R_2^{\mathcal{I}}$
- $\mathcal{N}$ : Number restrictions,  
 $(\geq n R)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : |\{y \mid \langle x, y \rangle \in R^{\mathcal{I}}\}| \geq n\}$ ,  
 $(\leq n R)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : |\{y \mid \langle x, y \rangle \in R^{\mathcal{I}}\}| \leq n\}$
- $\mathcal{Q}$ : Qualified number restrictions,  
 $(\geq n R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : |\{y \mid \langle x, y \rangle \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}| \geq n\}$ ,  
 $(\leq n R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : |\{y \mid \langle x, y \rangle \in R^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}| \leq n\}$
- $\mathcal{O}$ : Nominals (singleton class),  $\{a\}^{\mathcal{I}} = \{a^{\mathcal{I}}\}$
- $\mathcal{I}$ : Inverse role,  $(R^-)^{\mathcal{I}} = \{\langle x, y \rangle \mid \langle y, x \rangle \in R^{\mathcal{I}}\}$
- $\mathcal{F}$ : Functional role,  $\mathcal{I} \models \text{fun}(f)$  iff  $\forall x \forall y \forall z$  if  $\langle x, y \rangle \in f^{\mathcal{I}}$  and  $\langle x, z \rangle \in f^{\mathcal{I}}$  then  $y = z$
- $\mathcal{R}_+$ : transitive role,  
 $(R_+)^{\mathcal{I}} = \{\langle x, y \rangle \mid \exists z \text{ such that } \langle x, z \rangle \in R^{\mathcal{I}} \wedge \langle z, y \rangle \in R^{\mathcal{I}}\}$

## Basics on Concrete Domains

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

*(tim, 14):hasAge*

*(sf, "SoftComputing"):hasAcronym*

*(source1, "ComputerScience"):isAbout*

*(service2, "InformationRetrievalTool"):Matches*

*Minor = Person  $\sqcap$   $\exists$ hasAge.  $\leq 18$*

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

# DL Knowledge Base

- A DL **Knowledge Base** is a pair  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ , where
  - $\mathcal{T}$  is a **TBox**
    - containing general inclusion axioms of the form  $C \sqsubseteq D$ ,
    - concept definitions of the form  $A = C$
    - primitive concept definitions of the form  $A \sqsubseteq C$
    - role inclusions of the form  $R \sqsubseteq P$
    - role equivalence of the form  $R = P$
  - $\mathcal{A}$  is a **ABox**
    - containing assertions of the form  $a:C$
    - containing assertions of the form  $(a, b):R$
- An interpretation  $\mathcal{I}$  is a model of  $\mathcal{K}$ , written  $\mathcal{I} \models \mathcal{K}$  iff  $\mathcal{I} \models \mathcal{T}$  and  $\mathcal{I} \models \mathcal{A}$ , where
  - $\mathcal{I} \models \mathcal{T}$  ( $\mathcal{I}$  is a model of  $\mathcal{T}$ ) iff  $\mathcal{I}$  is a model of each element in  $\mathcal{T}$
  - $\mathcal{I} \models \mathcal{A}$  ( $\mathcal{I}$  is a model of  $\mathcal{A}$ ) iff  $\mathcal{I}$  is a model of each element in  $\mathcal{A}$

## Basic Inference Problems (Formally)

**Consistency:** Check if knowledge is meaningful

- Is  $\mathcal{K}$  satisfiability?  $\mapsto$  Is there some model  $\mathcal{I}$  of  $\mathcal{K}$  ?
- Is  $C$  satisfiability?  $\mapsto C^{\mathcal{I}} \neq \emptyset$  for some some model  $\mathcal{I}$  of  $\mathcal{K}$  ?

**Subsumption:** structure knowledge, compute taxonomy

- $\mathcal{K} \models C \sqsubseteq D$  ?  $\mapsto$  Is it true that  $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$  for all models  $\mathcal{I}$  of  $\mathcal{K}$  ?

**Equivalence:** check if two classes denote same set of instances

- $\mathcal{K} \models C = D$  ?  $\mapsto$  Is it true that  $C^{\mathcal{I}} = D^{\mathcal{I}}$  for all models  $\mathcal{I}$  of  $\mathcal{K}$  ?

**Instantiation:** check if individual  $a$  instance of class  $C$

- $\mathcal{K} \models a:C$  ?  $\mapsto$  Is it true that  $a^{\mathcal{I}} \in C^{\mathcal{I}}$  for all models  $\mathcal{I}$  of  $\mathcal{K}$  ?

**Retrieval:** retrieve set of individuals that instantiate  $C$

- Compute the set  $\{a \mid \mathcal{K} \models a:C\}$

# Reduction to Satisfiability

Problems are all **reducible** to KB satisfiability

**Subsumption:**  $\mathcal{K} \models C \sqsubseteq D$  iff  $\langle \mathcal{T}, \mathcal{A} \cup \{a:C \sqcap \neg D\} \rangle$  not satisfiable, where  $a$  is a new individual

**Equivalence:**  $\mathcal{K} \models C = D$  iff  $\mathcal{K} \models C \sqsubseteq D$  and  $\mathcal{K} \models D \sqsubseteq C$

**Instantiation:**  $\mathcal{K} \models a:C$  iff  $\langle \mathcal{T}, \mathcal{A} \cup \{a:\neg C\} \rangle$  not satisfiable

**Retrieval:** The computation of the set  $\{a \mid \mathcal{K} \models a:C\}$  is reducible to the instance checking problem

# Reasoning in DLs: Basics

- OWL 2: **tableaux** based algorithms
- OWL 2 EL: **structural** based algorithms
- OWL 2 QL: **query rewriting** based algorithms
- OWL 2 RL: **logic programming** based algorithms

## Tableaux: Basics

- Tableaux algorithm deciding satisfiability
- Try to build a **tree-like model**  $\mathcal{I}$  of the KB
- Decompose concepts  $C$  syntactically
  - Apply tableau **expansion rules**
  - Infer constraints on elements of model
- Tableau rules correspond to constructors in logic ( $\sqcap, \sqcup, \dots$ )
  - Some rules are **nondeterministic** (e.g.,  $\sqcup, \leq$ )
  - In practice, this means **search**
- Stop when no more rules applicable or **clash** occurs
  - Clash is an obvious contradiction, e.g.,  $A(x), \neg A(x)$
- Cycle check (**blocking**) may be needed for termination

## Negation Normal Form (NNF)

- We have to transform concepts into **Negation Normal Form**: push negation inside using de Morgan' laws

$$\begin{aligned}\neg \top &\mapsto \perp \\ \neg \perp &\mapsto \top \\ \neg \neg C &\mapsto C \\ \neg(C_1 \sqcap C_2) &\mapsto \neg C_1 \sqcup \neg C_2 \\ \neg(C_1 \sqcup C_2) &\mapsto \neg C_1 \sqcap \neg C_2\end{aligned}$$

and

$$\begin{aligned}\neg(\exists R.C) &\mapsto \forall R.\neg C \\ \neg(\forall R.C) &\mapsto \exists R.\neg C\end{aligned}$$

# Completion-Forest

- This is a forest of trees, where
  - each node  $x$  is labelled with a set  $\mathcal{L}(x)$  of concepts
  - each edge  $\langle x, y \rangle$  is labelled with  $\mathcal{L}(\langle x, y \rangle) = \{R\}$  for some role  $R$  (edges correspond to relationships between pairs of individuals)
- The forest is initialized with
  - a root node  $a$ , labelled  $\mathcal{L}(a) = \emptyset$  for each individual  $a$  occurring in the KB
  - an edge  $\langle a, b \rangle$  labelled  $\mathcal{L}(\langle a, b \rangle) = \{R\}$  for each  $(a, b):R$  occurring in the KB
- Then, for each  $a:C$  occurring in the KB, set  $\mathcal{L}(a) \rightarrow \mathcal{L}(a) \cup \{C\}$
- The algorithm expands the tree either by extending  $\mathcal{L}(x)$  for some node  $x$  or by adding new leaf nodes.
- Edges are added when expanding  $\exists R.C$
- A completion-forest contains a **clash** if, for a node  $x$ ,  $\{C, \neg C\} \subseteq \mathcal{L}(x)$
- If nodes  $x$  and  $y$  are connected by an edge  $\langle x, y \rangle$ , then  $y$  is called a successor of  $x$  and  $x$  is called a predecessor of  $y$ . Ancestor is the transitive closure of predecessor.
- A node  $y$  is called an  $R$ -successor of a node  $x$  if  $y$  is a successor of  $x$  and  $\mathcal{L}(\langle x, y \rangle) = \{R\}$ .
- The algorithm returns "satisfiable" if rules can be applied s.t. they yield a clash-free, complete (no more rules can be applied) completion forest

## $\mathcal{ALC}$ Tableau rules without GCI's

Rule	Description
( $\sqcap$ )	if 1. $C_1 \sqcap C_2 \in \mathcal{L}(x)$ and 2. $\{C_1, C_2\} \not\subseteq \mathcal{L}(x)$ then $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{C_1, C_2\}$
( $\sqcup$ )	if 1. $C_1 \sqcup C_2 \in \mathcal{L}(x)$ and 2. $\{C_1, C_2\} \cap \mathcal{L}(x) = \emptyset$ then $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{C\}$ for some $C \in \{C_1, C_2\}$
( $\exists$ )	if 1. $\exists R.C \in \mathcal{L}(x)$ and 2. $x$ has no $R$ -successor $y$ with $C \in \mathcal{L}(y)$ then create a new node $y$ with $\mathcal{L}(\langle x, y \rangle) = \{R\}$ and $\mathcal{L}(y) = \{C\}$
( $\forall$ )	if 1. $\forall R.C \in \mathcal{L}(x)$ and 2. $x$ has an $R$ -successor $y$ with $C \notin \mathcal{L}(y)$ then $\mathcal{L}(y) \rightarrow \mathcal{L}(y) \cup \{C\}$

## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

$$\mathcal{L}(x) = \{\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D\}$$

x

## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

$$\mathcal{L}(x) = \{\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D\}$$

x

## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

$$\mathcal{L}(x) = \{\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D\}$$

$x$

## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

$x$

## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

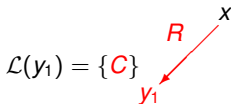
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

$x$

## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

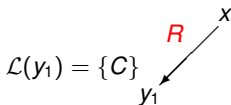
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$



## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

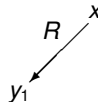
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$



## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

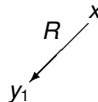
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

$$\mathcal{L}(y_1) = \{C, \neg C \sqcup \neg D\}$$


## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

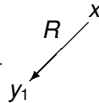
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

$$\mathcal{L}(y_1) = \{C, \neg C \sqcup \neg D\}$$


## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

$$\mathcal{L}(y_1) = \{C, \neg C \sqcup \neg D, \neg C\}$$


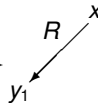
## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

$$\mathcal{L}(y_1) = \{C, \neg C \sqcup \neg D, \neg C\}$$

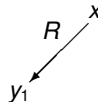
Clash



## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

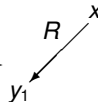
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

$$\mathcal{L}(y_1) = \{C, \neg C \sqcup \neg D\}$$


## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

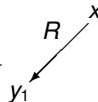
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

$$\mathcal{L}(y_1) = \{C, \neg C \sqcup \neg D, \neg D\}$$


## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

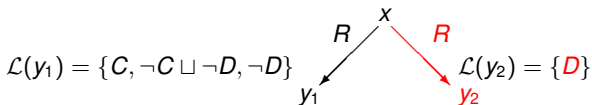
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

$$\mathcal{L}(y_1) = \{C, \neg C \sqcup \neg D, \neg D\}$$


## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

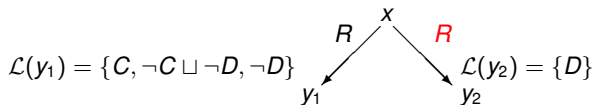
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$



## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

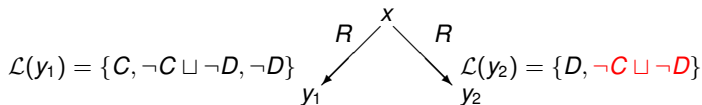
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$



## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

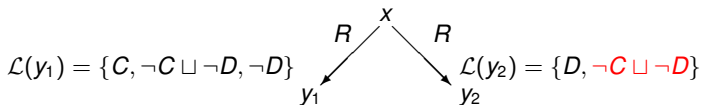
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$



## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes. w

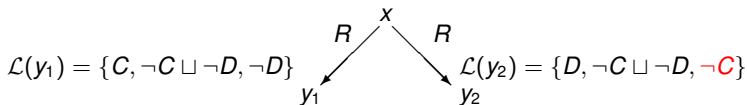
$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$



## Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

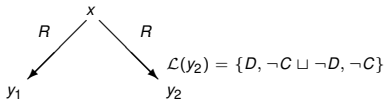


# Example

Is  $\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D$  satisfiable? Yes.

$$\mathcal{L}(x) = \{\exists R.C, \forall R.(\neg C \sqcup \neg D), \exists R.D\}$$

$$\mathcal{L}(y_1) = \{C, \neg C \sqcup \neg D, \neg D\}$$



- Finished. No more rules applicable and the tableau is complete and clash-free
- Hence, the concept is **satisfiable**
- The tree corresponds to a **model**  $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ 
  - The nodes are the elements of the domain:  $\Delta^{\mathcal{I}} = \{x, y_1, y_2\}$
  - For each atomic concept  $A$ , set  $A^{\mathcal{I}} = \{z \mid A \in \mathcal{L}(z)\}$ 
    - $C^{\mathcal{I}} = \{y_1\}, D^{\mathcal{I}} = \{y_2\}$
  - For each role  $R$ , set  $R^{\mathcal{I}} = \{\langle x, y \rangle \mid \text{there is an edge labeled } R \text{ from } x \text{ to } y\}$ 
    - $R^{\mathcal{I}} = \{\langle x, y_1 \rangle, \langle x, y_2 \rangle\}$
- It can be shown that  $x \in (\exists R.C \sqcap \forall R.(\neg C \sqcup \neg D) \sqcap \exists R.D)^{\mathcal{I}} \neq \emptyset$

# Soundness and Completeness

## Theorem

*Let  $\mathcal{A}$  be an  $\mathcal{ALC}$  ABox and  $F$  a completion-forest obtained by applying the tableau rules to  $\mathcal{A}$ . Then*

- 1 *The rule application terminates;*
- 2 *If  $F$  is clash-free and complete, then  $F$  defines a (canonical) (tree) model for  $\mathcal{A}$ ; and*
- 3 *If  $\mathcal{A}$  has a model  $\mathcal{I}$ , then the rules can be applied such that they yield a clash-free and complete completion-forest.*

## KBs with GCIs

- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable with  $\mathcal{T} \neq \emptyset$ ?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human\}$

*umberto*

## KBs with GCIs

- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable with  $\mathcal{T} \neq \emptyset$ ?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human\}$$

*umberto*

# KB Satisfiability

- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother. Human\} \\ \mathcal{A} &= \{umberto: Human\}\end{aligned}$$

$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother. Human\}$$

*umberto*

- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human, \neg Human\} \quad umberto$$

- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human, \neg Human\}$  umberto  
**Clash**

- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human\}$$

*umberto*

- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\} \quad umberto$$

- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

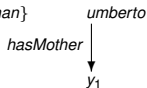
$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\} \quad umberto$$

- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$$

$$\mathcal{L}(y_1) = \{Human, \neg Human \sqcup \exists hasMother.Human\}$$

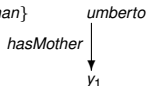


- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$$

$$\mathcal{L}(y_1) = \{Human, \neg Human \sqcup \exists hasMother.Human\}$$

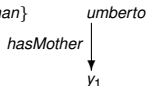


- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$$

$$\mathcal{L}(y_1) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$$



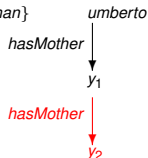
- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$$

$$\mathcal{L}(y_1) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$$

$$\mathcal{L}(y_2) = \{Human, \neg Human \sqcup \exists hasMother.Human\}$$



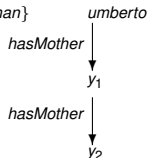
- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$

$$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$$

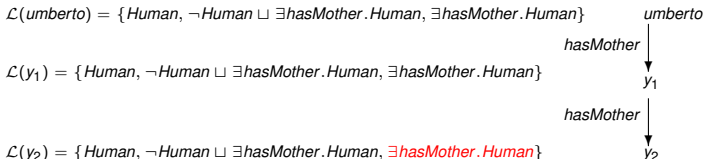
$$\mathcal{L}(y_1) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$$

$$\mathcal{L}(y_2) = \{Human, \neg Human \sqcup \exists hasMother.Human\}$$



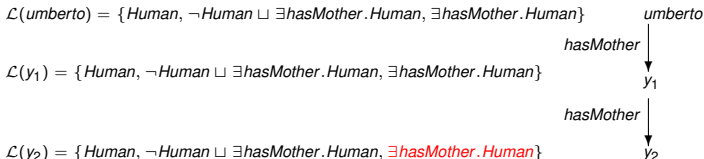
- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$



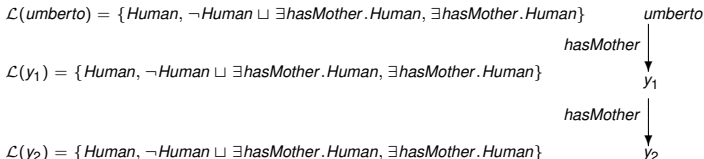
- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned} \mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\} \end{aligned}$$



- We have seen how to test the satisfiability of an ABox  $\mathcal{A}$
- But, how can we check if a KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  is satisfiable?
- Basic idea: since  $t(C \sqsubseteq D) \equiv \forall x. \neg t(C, x) \vee t(D, x)$ 
  - we use the rule: for each  $C \sqsubseteq D \in \mathcal{T}$ , add  $\neg C \sqcup D$  to **every** node
- But, **termination is not guaranteed**
  - E.g., consider  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$

$$\begin{aligned}\mathcal{T} &= \{Human \sqsubseteq \exists hasMother.Human\} \\ \mathcal{A} &= \{umberto:Human\}\end{aligned}$$



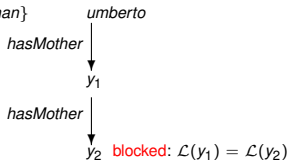
## Node Blocking in $\mathcal{ALC}$

- When creating new node, check ancestors for equal label set
- If such a node is found, new node is **blocked**
- No rule is applied to blocked nodes

$\mathcal{L}(\text{umberto}) = \{Human, \neg Human \sqcup \exists \text{hasMother}.Human, \exists \text{hasMother}.Human\}$

$\mathcal{L}(y_1) = \{Human, \neg Human \sqcup \exists \text{hasMother}.Human, \exists \text{hasMother}.Human\}$

$\mathcal{L}(y_2) = \{Human, \neg Human \sqcup \exists \text{hasMother}.Human, \exists \text{hasMother}.Human\}$



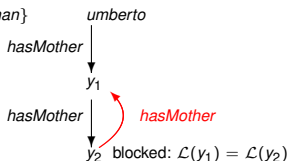
## Node Blocking in $\mathcal{ALC}$

- When creating new node, check ancestors for equal label set
- If such a node is found, new node is **blocked**
- No rule is applied to blocked nodes

$\mathcal{L}(umberto) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$

$\mathcal{L}(y_1) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$

$\mathcal{L}(y_2) = \{Human, \neg Human \sqcup \exists hasMother.Human, \exists hasMother.Human\}$



- Block represents **cyclical** model
  - $\Delta^{\mathcal{I}} = \{umberto, y_1, y_2\}$
  - $Human^{\mathcal{I}} = \{umberto, y_1, y_2\}$
  - $hasMother^{\mathcal{I}} = \{\langle umberto, y_1 \rangle, \langle y_1, y_2 \rangle, \langle y_2, y_1 \rangle\}$

## Blocking in $\mathcal{ALC}$

- A non-root node  $x$  is blocked if for some ancestor  $y$ ,  $y$  is blocked or  $\mathcal{L}(x) = \mathcal{L}(y)$ , where  $y$  is not a root node
- A blocked node  $x$  is indirectly blocked if its predecessor is blocked, otherwise it is directly blocked
- If  $x$  is directly blocked, it has a unique ancestor  $y$  such that  $\mathcal{L}(x) = \mathcal{L}(y)$
- if there existed another ancestor  $z$  such that  $\mathcal{L}(x) = \mathcal{L}(z)$  then either  $y$  or  $z$  must be blocked
- If  $x$  is directly blocked and  $y$  is the unique ancestor such that  $\mathcal{L}(x) = \mathcal{L}(y)$ , we will say that  $y$  blocks  $x$

## $\mathcal{ALC}$ Tableau rules with GCI's

Rule	Description
$(\sqcap)$	if 1. $C_1 \sqcap C_2 \in \mathcal{L}(x)$ , $x$ is not indirectly blocked and 2. $\{C_1, C_2\} \not\subseteq \mathcal{L}(x)$ then $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{C_1, C_2\}$
$(\sqcup)$	if 1. $C_1 \sqcup C_2 \in \mathcal{L}(x)$ , $x$ is not indirectly blocked and 2. $\{C_1, C_2\} \cap \mathcal{L}(x) = \emptyset$ then $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{C\}$ for some $C \in \{C_1, C_2\}$
$(\exists)$	if 1. $\exists R.C \in \mathcal{L}(x)$ , $x$ is not blocked and 2. $x$ has no $R$ -successor $y$ with $C \in \mathcal{L}(y)$ then create a new node $y$ with $\mathcal{L}(\langle x, y \rangle) = \{R\}$ and $\mathcal{L}(y) = \{C\}$
$(\forall)$	if 1. $\forall R.C \in \mathcal{L}(x)$ , $x$ is not indirectly blocked and 2. $x$ has an $R$ -successor $y$ with $C \notin \mathcal{L}(y)$ then $\mathcal{L}(y) \rightarrow \mathcal{L}(y) \cup \{C\}$
$(\sqsubseteq)$	if 1. $C \sqsubseteq D \in \mathcal{T}$ , $x$ is not indirectly blocked and 2. $\{nfn(\neg C), D\} \cap \mathcal{L}(x) = \emptyset$ then $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{E\}$ for some $E \in \{nfn(\neg C), D\}$ ( $nfn(\neg C)$ is NNF of $\neg C$ )

# Soundness and Completeness

## Theorem

*Let  $\mathcal{K}$  be an  $\mathcal{ALC}$  KB and  $F$  a completion-forest obtained by applying the tableau rules to  $\mathcal{K}$ . Then*

- 1 The rule application terminates;*
- 2 If  $F$  is clash-free and complete, then  $F$  defines a (canonical) (tree) model for  $\mathcal{K}$ ; and*
- 3 If  $\mathcal{K}$  has a model  $\mathcal{I}$ , then the rules can be applied such that they yield a clash-free and complete completion-forest.*

# Fuzzy DLs Basics

- We have seen how to “fuzzify” classical sets and FOL
  - Fuzzy statements are of the form  $\langle \phi, n \rangle$ , where  $\phi$  is a statement and  $n \in [0, 1]$
- The natural extension to fuzzy DLs consists then in replacing  $\phi$  with a DL expression
- Several **fuzzy** variants of DLs have been proposed: they can be classified according to
  - The DL resp. ontology language that they generalize
  - The allowed fuzzy constructs
  - The underlying fuzzy logic
  - Their reasoning algorithms and computational complexity results

- In classical DLs, a concept  $C$  is interpreted by an interpretation  $\mathcal{I}$  as a set of individuals
- In fuzzy DLs, a concept  $C$  is interpreted by  $\mathcal{I}$  as a fuzzy set of individuals
- Each individual is instance of a concept to a degree in  $[0, 1]$
- Each pair of individuals is instance of a role to a degree in  $[0, 1]$

- $\langle a:C, n \rangle$  states that  $a$  is an instance of concept/class  $C$  with degree at least  $n$
- $\langle (a, b):R, n \rangle$  states that  $\langle a, b \rangle$  is an instance of relation  $R$  with degree at least  $n$
- $\langle C_1 \sqsubseteq C_2, n \rangle$  states a vague subsumption relationship
  - The FOL statement  $\forall x. C_1(x) \rightarrow C_2(x)$  is true to degree at least  $n$
- **Note:** one may find also fuzzy DL expressions  $\langle \alpha \geq n \rangle$ ,  $\langle \alpha \leq n \rangle$ ,  $\langle \alpha > n \rangle$ ,  $\langle \alpha < n \rangle$ , and  $\langle \alpha = n \rangle$
- We use the form  $\langle \alpha, n \rangle$ , i.e.  $\langle \alpha \geq n \rangle$  only
  - Remind that graded axioms are intended to be produced semi- or automatically
  - Hardly they may have the form  $\langle \alpha \leq n \rangle$ ,  $\langle \alpha > n \rangle$  or  $\langle \alpha < n \rangle$

The semantics is an immediate consequence of the First-Order-Logic translation of DLs expressions

Interpretation:	$\mathcal{I}$	=	$\Delta^{\mathcal{I}}$	$\otimes$	=	t-norm
	$C^{\mathcal{I}}$	:	$\Delta^{\mathcal{I}} \rightarrow [0, 1]$	$\oplus$	=	s-norm
	$R^{\mathcal{I}}$	:	$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0, 1]$	$\neg$	=	negation
				$\Rightarrow$	=	implication

Syntax		Semantics	
$C, D \rightarrow$	$\top$	$\top^{\mathcal{I}}(x)$	= 1
	$\perp$	$\perp^{\mathcal{I}}(x)$	= 0
	$A$	$A^{\mathcal{I}}(x)$	$\in [0, 1]$
Concepts:	$C \sqcap D$	$(C_1 \sqcap C_2)^{\mathcal{I}}(x)$	= $C_1^{\mathcal{I}}(x) \otimes C_2^{\mathcal{I}}(x)$
	$C \sqcup D$	$(C_1 \sqcup C_2)^{\mathcal{I}}(x)$	= $C_1^{\mathcal{I}}(x) \oplus C_2^{\mathcal{I}}(x)$
	$C \rightarrow D$	$(C \rightarrow D)^{\mathcal{I}}(x)$	= $C^{\mathcal{I}}(x) \Rightarrow D^{\mathcal{I}}(x)$
	$\neg C$	$(\neg C)^{\mathcal{I}}(x)$	= $\neg C^{\mathcal{I}}(x)$
	$\exists R.C$	$(\exists R.C)^{\mathcal{I}}(x)$	= $\sup_{y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x, y) \otimes C^{\mathcal{I}}(y)$
	$\forall R.C$	$(\forall R.C)^{\mathcal{I}}(x)$	= $\inf_{y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x, y) \Rightarrow C^{\mathcal{I}}(y)$
	$\{a\}$	$\{a\}^{\mathcal{I}}(x)$	= 1 if $a^{\mathcal{I}} = x$ , else 0

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

General Inclusion Axioms:  $\langle C \sqsubseteq D, r \rangle,$

●  $\mathcal{I} \models \langle C \sqsubseteq D, r \rangle$  iff  $\inf_{x \in \Delta^{\mathcal{I}}} C^{\mathcal{I}}(x) \Rightarrow D^{\mathcal{I}}(x) \geq r$

## Some Remarks

- Like for fuzzy FOL,  $\forall$  and  $\exists$  are not complementary in general: i.e.  $\forall R.C \not\equiv \neg \exists R. \neg C$
- $\forall R.C \equiv \neg \exists R. \neg C$  under Łukasiewicz logic and SFL
- $\langle C \sqsubseteq D, n \rangle$  may be rewritten as  $\langle \top \sqsubseteq C \rightarrow D, n \rangle$
- In early works, a fuzzy GCI is of the form  $C \sqsubseteq D$  with semantics:
  - $\mathcal{I}$  is a model of  $C \sqsubseteq D$  iff for every  $x \in \Delta^{\mathcal{I}}$  we have that  $C^{\mathcal{I}}(x) \leq D^{\mathcal{I}}(x)$
  - This is the same of fuzzy axiom  $\langle \top \sqsubseteq C \rightarrow_x D, 1 \rangle$ , where  $\rightarrow_x$  is an  $r$ -implication
- **Disjointness**: use  $\langle C \sqcap D \sqsubseteq \perp, 1 \rangle$  rather than  $\langle C \sqsubseteq \neg D, 1 \rangle$ 
  - they are not the same, e.g.  $\langle A \sqsubseteq \neg A, 1 \rangle$  says that  $A^{\mathcal{I}}(x) \leq 0.5$ , for all  $\mathcal{I}$  and for all  $x \in \Delta^{\mathcal{I}}$

# Witnessed Interpretation

- **Witnessed Interpretation:**
  - Infima and suprema are attained at some point

$$\begin{aligned}(\exists R.C)^{\mathcal{I}}(x) &= R^{\mathcal{I}}(x, y) \otimes C^{\mathcal{I}}(y) \text{ for some } y \in \Delta^{\mathcal{I}} \\ (\forall R.C)^{\mathcal{I}}(x) &= R^{\mathcal{I}}(x, y) \Rightarrow C^{\mathcal{I}}(y) \text{ for some } y \in \Delta^{\mathcal{I}} \\ (C \sqsubseteq D)^{\mathcal{I}} &= C^{\mathcal{I}}(x) \Rightarrow D^{\mathcal{I}}(x) \text{ for some } x \in \Delta^{\mathcal{I}}\end{aligned}$$

- It is customary to stick to witnessed interpretations only

- **Fuzzy knowledge base:**  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ 
  - $\mathcal{T}$  is a **fuzzy TBox**, that is a finite set of fuzzy GCI
  - $\mathcal{A}$  is a **fuzzy ABox**, that is a finite set of fuzzy assertions
- **Acyclic fuzzy ontologies:** TBox with axioms of the form

$A \sqsubseteq_n C$  (**primitive** GCI)

$A \sqsubseteq^{\sim} C$  (**primitive** GCI)

$A \simeq C$  (**definitional** GCI)

- $A$  concept name
- $A \sqsubseteq_n C$  shorthand for  $\langle \top \sqsubseteq A \rightarrow C, n \rangle$
- No nominal  $\{a\}$  occurs in the TBox

- We say that
  - concept name  $A$  **directly uses** concept name  $B$  w.r.t.  $\mathcal{T}$ , denoted  $A \rightarrow_{\mathcal{T}} B$ , if  $A$  is the head of some axiom  $\tau \in \mathcal{T}$  such that  $B$  occurs in the body of  $\tau$
  - concept name  $A$  **uses** concept name  $B$  w.r.t.  $\mathcal{T}$ , denoted  $A \rightsquigarrow_{\mathcal{T}} B$ , if there exist concept names  $A_1, \dots, A_n$ , such that  $A_1 = A$ ,  $A_n = B$  and, for every  $1 \leq i < n$ , it holds that  $A_i \rightarrow_{\mathcal{T}} A_{i+1}$
- TBox  $\mathcal{T}$  is **cyclic (acyclic)** if there is (no)  $A$  such that  $A \rightsquigarrow_{\mathcal{T}} A$
- TBox  $\mathcal{T}$  is **unfoldable** if
  - $\mathcal{T}$  is acyclic
  - If  $A \stackrel{\mathcal{T}}{\equiv} C \in \mathcal{T}$  then  $A$  does not occur in the head of any other axiom

- $\mathcal{I}$  **satisfies (is a model of)**  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  iff it satisfies each element in  $\mathcal{A}$  and  $\mathcal{T}$
- A fuzzy KB  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  **entails** an axiom  $E$ , denoted  $\mathcal{K} \models E$ , iff every model of  $\mathcal{K}$  satisfies  $E$
- We say that two concepts  $C$  and  $D$  are **equivalent**, denoted  $C \equiv_{\mathcal{K}} D$  iff in every model  $\mathcal{I}$  of  $\mathcal{K}$  and for all  $x \in \Delta^{\mathcal{I}}$ ,  $C^{\mathcal{I}}(x) = D^{\mathcal{I}}(x)$
- **Best entailment degree**: for assertion of GCI  $\phi$

$$bed(\mathcal{K}, \phi) = \sup \{r \mid \mathcal{K} \models \langle \phi, r \rangle\}$$

- **Best satisfiability degree**: for concept  $C$

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

# Some Salient Fuzzy Concept Equivalences

Property	Łukasiewicz	Gödel	Product	SFL
$C \sqcap \neg C \equiv \perp$	•	•	•	
$C \sqcup \neg C \equiv \top$	•			
$C \sqcap C \equiv C$		•		•
$C \sqcup C \equiv C$		•		•
$\neg \neg C \equiv C$	•			•
$C \rightarrow D \equiv \neg C \sqcup D$	•			•
$C \rightarrow D \equiv \neg D \rightarrow \neg C$	•			•
$\neg(C \rightarrow D) \equiv C \sqcap \neg D$	•			•
$\neg(C \sqcap D) \equiv \neg C \sqcup \neg D$	•	•	•	•
$\neg(C \sqcup D) \equiv \neg C \sqcap \neg D$	•	•	•	•
$C \sqcap (D \sqcup E) \equiv (C \sqcap D) \sqcup (C \sqcap E)$		•		•
$C \sqcup (D \sqcap E) \equiv (C \sqcup D) \sqcap (C \sqcup E)$		•		•
$\exists R.C \equiv \neg \forall R.\neg C$	•			•

## Towards Fuzzy OWL 2 and its Profiles

- Recall that OWL 2 relates to  $SROIQ(D)$
- We need to extend the semantics to fuzzy  $SROIQ(D)$
- Additionally, we add
  - **modifiers** (e.g., *very*)
  - **concrete fuzzy concepts** (e.g., *Young*)
  - both additions have **explicit** membership functions
  - other extensions:
    - aggregation functions: weighted sum, OWA, fuzzy integrals
    - fuzzy rough sets, fuzzy spatial, fuzzy numbers

# Number Restrictions, Inverse, Transitive roles, ...

- The semantics of the concept  $(\geq n R.C)$  is:  $\wedge$  interpreted as Gödel t-norm

$$\exists y_1, \dots, y_n. \bigwedge_{i=1}^n R(x, y_i) \wedge C(y_i) \wedge \bigwedge_{1 \leq i < j \leq n} y_i \neq y_j$$

- The semantics of the concept  $(\leq n R.C)$  is:  $\wedge$  interpreted as Gödel t-norm

$$(\leq n R)^{\mathcal{I}}(x) = \forall y_1, \dots, y_{n+1}. \bigwedge_{i=1}^{n+1} (R(x, y_i) \wedge C(y_i)) \Rightarrow \bigvee_{1 \leq i < j \leq n+1} y_i = y_j$$

- Note:  $(\geq 1 R) \equiv \exists R.T$
- For transitive roles  $R$  we impose: for all  $x, y \in \Delta^{\mathcal{I}}$

$$R^{\mathcal{I}}(x, y) \geq \sup_{z \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x, z) \otimes R^{\mathcal{I}}(z, y)$$

- For inverse roles we have for all  $x, y \in \Delta^{\mathcal{I}}$

$$R^{\mathcal{I}}(x, y) = R^{\mathcal{I}}(y, x)$$

- The semantics of functional roles  $fun(R)$  is

$$\forall x \forall y \forall z. R(x, y) \wedge R(x, z) \Rightarrow y = z$$

- Similar for other  $\mathcal{SROIQ}$  constructs

## Fuzzy Concrete Domains

- E.g., *Small*, *Young*, *High*, etc. with **explicit** membership function
- Use the idea of concrete domains:
  - $D = \langle \Delta_D, \Phi_D \rangle$
  - $\Delta_D$  is an interpretation domain
  - $\Phi_D$  is the set of concrete unary fuzzy domain predicates  $d$  and **fixed** interpretation  $d^D: \Delta_D \rightarrow [0, 1]$
- Specifically,

$$\mathbf{d} \quad \rightarrow \quad |s(a, b) | rs(a, b) | tri(a, b, c) | trz(a, b, c, d) \\ | \geq_v | \leq_v | =_v$$

$$C, D \quad \rightarrow \quad \forall T. \mathbf{d} \mid \exists T. \mathbf{d}$$

- Representation of **Young Person**:



$$\begin{aligned} \text{Minor} &= \text{Person} \sqcap \exists \text{hasAge}.\leq 18 \\ \text{YoungPerson} &= \text{Person} \sqcap \exists \text{hasAge}.ls(10, 30) \end{aligned}$$

- Representation of **Heavy Rain**:

$$\text{HeavyRain} = \text{Rain} \sqcap \exists \text{hasPrecipitationRate}.rs(5, 7.5)$$

## Modifiers

- *Very, moreOrLess, slightly, etc.*
- **Fuzzy modifier**  $m$  with function  $f_m: [0, 1] \rightarrow [0, 1]$

$$C \rightarrow m(C) \mid \forall T.m(\mathbf{d}) \mid \exists T.m(\mathbf{d})$$

where  $m$  is a linear modifier

- Representation of **Sport Car**



$$\text{SportsCar} = \text{Car} \sqcap \exists \text{speed}.\text{very}(\text{rs}(80, 250))$$

- Representation of **Very Heavy Rain**

$$\text{VeryHeavyRain} = \text{Rain} \sqcap \exists \text{hasPrecipitationRate}.\text{very}(\text{rs}(5, 7.5)) .$$

# Aggregation Operators

- **Aggregation operators**: aggregate concepts, using functions such as the mean, median, weighted sum operators
- Given an  $n$ -ary aggregation operator  $@ : [0, 1]^n \rightarrow [0, 1]$ 
  - We fuzzy concepts by allowing to apply  $@$  to  $n$  concepts  $C_1, \dots, C_n$ , i.e.

$$C \rightarrow @(C_1, \dots, C_n)$$

- Semantics:

$$@(C_1, \dots, C_n)^{\mathcal{I}}(x) = @(C_1^{\mathcal{I}}(x), \dots, C_n^{\mathcal{I}}(x)).$$

- Allows to express the concept

$$GoodHotel = 0.3 \cdot ExpensiveHotel + 0.7 \cdot LuxuriousHotel$$

- The membership function of good hotels is the weighted sum of being an expensive and luxurious hotel

## Some Applications

- Information retrieval
- Recommendation systems
- Image interpretation
- Ambient intelligence
- Ontology merging
- Matchmaking
- decision making
- Summarization
- Robotics perception
- Software design
- Machine learning

## Example (Graded Entailment)



<i>Car</i>	<i>speed</i>
<i>audi_tt</i>	243
<i>mg</i>	$\leq 170$
<i>ferrari_enzo</i>	$\geq 350$

*SportsCar* = *Car*  $\sqcap$   $\exists$ hasSpeed.very(High)

$\mathcal{K} \models \langle \textit{ferrari\_enzo}:\textit{SportsCar}, 1 \rangle$   
 $\mathcal{K} \models \langle \textit{audi\_tt}:\textit{SportsCar}, 0.92 \rangle$   
 $\mathcal{K} \models \langle \textit{mg}:\neg\textit{SportsCar}, 0.72 \rangle$

## Example (Graded Subsumption)

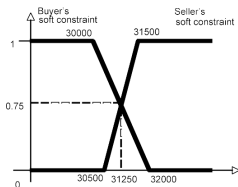


$$\begin{aligned} \text{Minor} &= \text{Person} \sqcap \exists \text{hasAge} . \leq_{18} \\ \text{YoungPerson} &= \text{Person} \sqcap \exists \text{hasAge} . \text{Young} \\ &\quad \text{fun}(\text{hasAge}) \end{aligned}$$

$$\mathcal{K} \models \langle \text{Minor} \sqsubseteq \text{YoungPerson}, 0.6 \rangle$$

Note: without an explicit membership function of *Young*, **this inference cannot be drawn**

## Example (Simplified Matchmaking)



- A car seller sells an Audi TT for 31500€, as from the catalog price.
- A buyer is looking for a sports-car, but wants to pay not more than around 30000€
- Classical sets: the problem relies on the crisp conditions on price
- More fine grained approach: to consider prices as fuzzy sets (as usual in negotiation)
  - Seller may consider optimal to sell above 31500€, but can go down to 30500€
  - The buyer prefers to spend less than 30000€, but can go up to 32000€  
 $AudiTT = SportsCar \sqcap \exists hasPrice.rs(30500, 31500)$   
 $Query = SportsCar \sqcap \exists hasPrice.ls(30000, 32000)$
  - Highest degree to which the concept  
 $C = AudiTT \sqcap Query$   
is satisfiable is 0.75 (the degree to which the Audi TT and the query **matches** is 0.75)
  - The car may be sold at 31250€

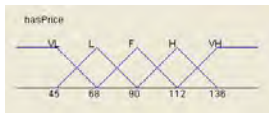
## Example: Learning fuzzy GCIs from data

- Learning of fuzzy GCIs from crisp data
- Use Case: What are **Good hotels**, using TripAdvisor data?
  - Given
    - OWL 2 Ontology about meaningful city entities and their descriptions
    - TripAdvisor data about hotels and user judgments
  - We may learn that in e.g., Pisa, Italy

$\langle \exists \text{hasAmenity.Babysitting} \sqcap \exists \text{hasPrice.fair} \sqsubseteq \text{Good\_Hotel}, 0.282 \rangle$

*“A hotel having babysitting as amenity and a fair price is a good hotel (to degree 0.282)”*

- Real valued price attribute *hasPrice* has been automatically fuzzyfied



## Example: Multi-Criteria Decision Making

- We have to select among two sites,  $A_1, A_2$
- There are two criteria ( $C_1$  -Transportation Issues, and  $C_2$  -Public Nuisance) for judgement
- There are two experts ( $E_1, E_2$ ) that make judgments
- The decision matrix of the experts is shown below:

$E_1$		Criteria	
		0.48	0.52
Alter.		$C_1$	$C_2$
$x_1$	$A_1$	$tri(0.6, 0.7, 0.8)$	$tri(0.9, 0.95, 1.0)$
$x_2$	$A_2$	$tri(0.6, 0.7, 0.8)$	$tri(0.4, 0.5, 0.6)$

$E_2$		Criteria	
		0.52	0.48
Alter.		$C_1$	$C_2$
$x_1$	$A_1$	$tri(0.55, 0.6, 0.7)$	$tri(0.4, 0.45, 0.5)$
$x_2$	$A_2$	$tri(0.35, 0.4, 0.45)$	$tri(0.5, 0.55, 0.6)$

- For each expert  $k = 1, 2$ , for each alternative  $i = 1, 2$  and for each criteria  $j = 1, 2$ , we define the concept

$$P_{ij}^k = \exists \text{hasScore}.a_{ij}^k$$

- Now, for each expert  $k$  and alternative  $i$ , we define the weighted concept

$$A_i^k = w_1^k \cdot P_{i1}^k + w_2^k \cdot P_{i2}^k$$

- Finally, we combine the two experts outcome, by defining the weighted concept

$$A_i = 0.5 \cdot A_i^1 + 0.5 \cdot A_i^2$$

- It can be verified that  $rv(\mathcal{K}, A_1) = bsd(\mathcal{K}, A_1) = 0.26$  and  $rv(\mathcal{K}, A_2) = bsd(\mathcal{K}, A_2) = 0.37$

# Representing Fuzzy OWL Ontologies in OWL

- OWL 2 is W3C standard, with classical logic semantics
  - Hence, cannot support natively Fuzzy Logic
- However, **Fuzzy OWL 2**, has been defined using OWL 2
  - Uses the axiom annotation feature of OWL 2
- Any Fuzzy OWL 2 ontology is a legal OWL 2 ontology

- A java parser for Fuzzy OWL 2 exists
- Protégé plug-in exists to encode Fuzzy OWL ontologies

The screenshot shows the Protégé ontology editor interface. The main window is titled 'FuzzyTest' and displays the configuration for a new datatype named 'rightshoulder'. The 'Type' is set to 'rightshoulder'. The 'A' parameter is 200.0, 'B' is 300.0, 'K1' is 0.0, and 'K2' is 1000.0. A graph to the right of the parameters shows a trapezoidal membership function. The 'Annotations' panel on the right shows the generated OWL code for the datatype.

```
<Datatype type="rightshoulder" >
  <B="300" />
  </FuzzyDwl2="rightshoulder" />
```

# Reasoning Problems and Algorithms

Consistency problem:

- Is  $\mathcal{K}$  satisfiable?
- Is  $\mathcal{C}$  coherent, i.e. is  $C^{\mathcal{I}}(x) > 0$  for some  $\mathcal{I} \models \mathcal{K}$  and  $x \in \Delta^{\mathcal{I}}$ ?

Instance checking problem:

- Does  $\mathcal{K} \models \langle a:C, n \rangle$  hold?

Subsumption problem:

- Does  $\mathcal{K} \models \langle C \sqsubseteq D, n \rangle$  hold?

Best entailment degree problem:

- What is  $bed(\mathcal{K}, \phi)$ ?

Best satisfiability degree problem:

- What is  $bsd(\mathcal{K}, \phi)$ ?

Instance retrieval problem:

- Compute the set  $\{\langle a, n \rangle \mid n = bed(\mathcal{K}, a:C)\}$

Top-k retrieval problem:

- Compute the top-k ranked elements of  $\{\langle a, n \rangle \mid n = bed(\mathcal{K}, a:C)\}$

## Some Reductions

- $\mathcal{K}$  is satisfiable iff  $bsd(\mathcal{K}, a:\perp) > 0$ , where  $a$  is a new individual.
- $C$  is coherent w.r.t.  $\mathcal{K}$  if one of the following holds:
  - $\mathcal{K} \cup \{ \langle a:C > 0 \rangle \}$  is satisfiable, where  $a$  is a new individual
  - $\mathcal{K} \not\models \langle C \sqsubseteq \perp, 1 \rangle$
  - $bsd(\mathcal{K}, C) > 0$
- $\mathcal{K} \models \langle a:C, n \rangle$  if one of the following holds:
  - $\mathcal{K} \cup \{ \langle a:C < n \rangle \}$  is not satisfiable
  - $bed(\mathcal{K}, a:C) \geq n$
- $\mathcal{K} \models \langle C \sqsubseteq D, n \rangle$  if one of the following holds:
  - $\mathcal{K} \cup \{ \langle a:C \rightarrow D < n \rangle \}$  is not satisfiable, where  $a$  is a new individual
  - $bed(\mathcal{K}, C \sqsubseteq D) \geq n$
- We have that

$$bed(\mathcal{K}, \phi) = \min x. \text{ such that } \mathcal{K} \cup \{ \langle \phi \leq x \rangle \} \text{ satisfiable}$$

$$bsd(\mathcal{K}, \phi) = \max x. \text{ such that } \mathcal{K} \cup \{ \langle \phi \geq x \rangle \} \text{ satisfiable}$$

## Reasoning in Fuzzy DLs: Basics

- Algorithms for fuzzy DLs: are a mixture of classical DLs reasoning algorithms and algorithms for Mathematical Fuzzy Logic
- Fuzzy OWL 2:
  - **Fuzzy tableaux** based algorithms
    - Tableaux and non deterministic tableaux
    - Operational Research
  - **Reduction** into classical DLs
- Fuzzy OWL 2 EL: **fuzzy structural** based algorithms
- Fuzzy OWL 2 QL: **fuzzy query rewriting** based algorithms
- fuzzy OWL 2 RL: **fuzzy logic programming** based algorithms

## OR Fuzzy Tableaux: $\mathcal{ALC}$ under SFL over $[0, 1]$

- Works as for classical  $\mathcal{ALC}$  on completion forests
  - Blocking is as for classical  $\mathcal{ALC}$
  - The completion forest is expanded by repeatedly applying inference rules
  - The completion-forest is complete when none of the rules are applicable
- Additionally, at each inference step we add equational constraints that have to hold
- Eventually, the initial KB is satisfiable if the final set of equational constraints has a solution
  - For the latter case, we may use a MILP solver

Rule	Description
(var)	For variable $x_{v:C}$ add $x_{v:C} \in [0, 1]$ to $\mathcal{C}_{\mathcal{F}}$ . For variable $x_{(v,w):R}$ , add $x_{(v,w):R} \in [0, 1]$ to $\mathcal{C}_{\mathcal{F}}$
( $\bar{A}$ )	if $\neg A \in \mathcal{L}(v)$ then add $x_{v:A} = 1 - x_{v:\neg A}$ to $\mathcal{C}_{\mathcal{F}}$
( $\perp$ )	If $\perp \in \mathcal{L}(v)$ then add $x_{v:\perp} = 0$ to $\mathcal{C}_{\mathcal{F}}$
( $\top$ )	If $\top \in \mathcal{L}(v)$ then add $x_{v:\top} = 1$ to $\mathcal{C}_{\mathcal{F}}$
( $\sqcap$ )	if $C_1 \sqcap C_2 \in \mathcal{L}(v)$ , $v$ is not indirectly blocked then $\mathcal{L}(v) \rightarrow \mathcal{L}(v) \cup \{C_1, C_2\}$ , and add $x_{v:C_1} \otimes x_{v:C_2} \geq x_{v:C_1 \sqcap C_2}$ to $\mathcal{C}_{\mathcal{F}}$
( $\sqcup$ )	if $C_1 \sqcup C_2 \in \mathcal{L}(v)$ , $v$ is not indirectly blocked then $\mathcal{L}(v) \rightarrow \mathcal{L}(v) \cup \{C_1, C_2\}$ , and add $x_{v:C_1} \oplus x_{v:C_2} \geq x_{v:C_1 \sqcup C_2}$ to $\mathcal{C}_{\mathcal{F}}$
( $\forall$ )	if $\forall R.C \in \mathcal{L}(v)$ , $v$ is not indirectly blocked then $\mathcal{L}(w) \rightarrow \mathcal{L}(w) \cup \{C\}$ , and add $x_{w:C} \geq x_{v:\forall R.C} \otimes x_{(v,w):R}$ to $\mathcal{C}_{\mathcal{F}}$
( $\exists$ )	if $\exists R.C \in \mathcal{L}(v)$ , $v$ is not blocked then create new node $w$ with $\mathcal{L}(\langle v, w \rangle) = \{R\}$ and $\mathcal{L}(w) = \{C\}$ , and add $x_{w:C} \otimes x_{(v,w):R} \geq x_{v:\exists R.C}$ to $\mathcal{C}_{\mathcal{F}}$
( $\sqsubseteq$ )	if $\langle C \sqsubseteq D, n \rangle \in \mathcal{T}$ , $v$ is not indirectly blocked then $\mathcal{L}(v) \rightarrow \mathcal{L}(v) \cup \{C, D\}$ , and add $x_{v:D} \geq x_{v:C} \otimes n$ to $\mathcal{C}_{\mathcal{F}}$

# Analytical Fuzzy Tableaux: $\mathcal{ALC}$ under SFL over $[0, 1]$

- Works as for classical  $\mathcal{ALC}$  on completion forests
  - Node labels  $\mathcal{L}(v)$  contain, rather than DL concept expressions, expressions of the form  $\langle C, n \rangle$ 

*“The truth degree of being  $v$  instance of  $C$  is  $\geq n$ ”*
  - Blocking is as for classical  $\mathcal{ALC}$
  - The completion forest is expanded by repeatedly applying inference rules
  - The completion-forest is complete when none of the rules are applicable
- Additionally, we adapt the notion of **clash**: a clash is either
  - $\langle \perp, n \rangle$  with  $n > 0$ ; or
  - a pair  $\langle C, n \rangle$  and  $\langle \neg C, m \rangle$  with  $n > 1 - m$
- Eventually, the initial KB is satisfiable if there is a clash-free complete completion forest

- ( $\sqcap$ ). If (i)  $\langle C_1 \sqcap C_2, n \rangle \in \mathcal{L}(v)$ , (ii)  $\{\langle C_1, n \rangle, \langle C_2, n \rangle\} \not\subseteq \mathcal{L}(v)$ , and (iii) node  $v$  is not indirectly blocked, then add  $\langle C_1, n \rangle$  and  $\langle C_2, n \rangle$  to  $\mathcal{L}(v)$ .
- ( $\sqcup$ ). If (i)  $\langle C_1 \sqcup C_2, n \rangle \in \mathcal{L}(v)$ , (ii)  $\{\langle C_1, n \rangle, \langle C_2, n \rangle\} \cap \mathcal{L}(v) = \emptyset$ , and (iii) node  $v$  is not indirectly blocked, then add some  $\langle C, n \rangle \in \{\langle C_1, n \rangle, \langle C_2, n \rangle\}$  to  $\mathcal{L}(v)$ .
- ( $\forall$ ). If (i)  $\langle \forall R.C, n \rangle \in \mathcal{L}(v)$ , (ii)  $\langle R, m \rangle \in \mathcal{L}(\langle v, w \rangle)$  with  $m > 1 - n$ , (iii)  $\langle C, n \rangle \notin \mathcal{L}(w)$ , and (iv) node  $v$  is not indirectly blocked, then add  $\langle C, n \rangle$  to  $\mathcal{L}(w)$ .
- ( $\exists$ ). If (i)  $\langle \exists R.C, n \rangle \in \mathcal{L}(v)$ , (ii) there is no  $\langle R, n_1 \rangle \in \mathcal{L}(\langle v, w \rangle)$  with  $\langle C, n_2 \rangle \in \mathcal{L}(w)$  such that  $\min(n_1, n_2) \geq n$ , and (iii) node  $v$  is not blocked, then create a new node  $w$ , add  $\langle R, n \rangle$  to  $\mathcal{L}(\langle v, w \rangle)$  and add  $\langle C, n \rangle$  to  $\mathcal{L}(w)$ .
- ( $\sqsubseteq$ ). If (i)  $\langle \top \sqsubseteq D, n \rangle \in \mathcal{T}$ , (ii)  $\langle D, n \rangle \notin \mathcal{L}(v)$ , and (iii) node  $v$  is not indirectly blocked, then add  $\langle D, n \rangle$  to  $\mathcal{L}(v)$ .

# Non-Deterministic Analytic Fuzzy Tableaux

- It's a combination of the analogous method for fuzzy propositional logic and analytical fuzzy tableau
- Rule example:
  - ( $\sqcap$ ). If (i)  $\langle C_1 \sqcap C_2, m \rangle \in \mathcal{L}(v)$ , (ii) there are  $m_1, m_2 \in L_n$  such that  $m_1 \otimes m_2 = m$  with  $\{\langle C_1, m_1 \rangle, \langle C_2, m_2 \rangle\} \not\subseteq \mathcal{L}(v)$ , and (iii) node  $v$  is not indirectly blocked, then add  $\langle C_1, m_1 \rangle$  and  $\langle C_2, m_2 \rangle$  to  $\mathcal{L}(v)$

## Reduction to Classical DLs

- Same principle as for the reduction for propositional fuzzy logic
- Needs adaption to the DL constructs: e.g.  $\exists$ ,  $\forall$  and  $\sqsubseteq$
- Examples of reduction rules for SFL:

$$\begin{aligned}\rho(A, \geq \gamma) &= A_{\geq \gamma} \\ \rho(C \sqcap D, \geq \gamma) &= \rho(C, \geq \gamma) \sqcap \rho(D, \geq \gamma) \\ \rho(C \sqcap D, \leq \gamma) &= \rho(C, \leq \gamma) \sqcup \rho(D, \leq \gamma) \\ \rho(\forall R.C, \geq \gamma) &= \forall \rho(R, > 1 - \gamma). \rho(C, \geq \gamma) \\ \rho(\forall R.C, \leq \gamma) &= \exists \rho(R, \geq 1 - \gamma). \rho(C, \leq \gamma) \\ \rho(\exists R.C, \geq \gamma) &= \exists \rho(R, \geq \gamma). \rho(C, \geq \gamma) \\ \rho(\exists R.C, \leq \gamma) &= \forall \rho(R, > \gamma). \rho(C, \leq \gamma) \\ \rho(R, \geq \gamma) &= R_{\geq \gamma} \\ \rho(\langle a:C, \gamma \rangle) &= \{a:\rho(C, \geq \gamma)\} \\ \rho(\langle C \sqsubseteq D, n \rangle) &= \bigcup_{\alpha \in \bar{N}_+^K, \alpha \leq n} \{ \rho(C, \geq \alpha) \sqsubseteq \rho(D, \geq \alpha) \}\end{aligned}$$

# Computational Complexity

The bad news...**undecidability!**

## Proposition

Assume that fuzzy GCIs are restricted to be classical, i.e. of the form  $\langle \alpha, 1 \rangle$  only. Then for the following fuzzy DLs, the KB satisfiability problem is undecidable over  $[0, 1]$ :

- 1  $\mathcal{ELC}$  with classical axioms only under Łukasiewicz logic and product logic;
- 2  $\mathcal{ELC}$  under any non Gödel-t-norm  $\otimes$ ;
- 3  $\mathcal{ELC}$  with concept assertions of the form  $\langle \alpha = n \rangle$  only under any non Gödel-t-norm  $\otimes$ ;
- 4  $\mathcal{AL}$  with concept implication operator  $\rightarrow$  and concept assertions of the form  $\langle \alpha = n \rangle$  only under any non Gödel-t-norm  $\otimes$ .
- 5  $\mathcal{ELC}$  under SFL with weighted sum constructor.

Some decidability results..

### Proposition

*The KB satisfiability problem is decidable for*

- *$SROIQ$  under SFL over  $[0, 1]$  and Gödel logic over  $L_n$*
- *$SROI\mathcal{N}$  under Łukasiewicz logic over  $L_n$*
- *$S\mathcal{H}I$  under any continuous  $t$ -norm over  $L_n$  without  $TBox$*
- *$\mathcal{ALC}$  with concept implication operator  $\rightarrow$ , for any continuous  $t$ -norm over  $[0, 1]$  with acyclic $TBox$*
- *$S\mathcal{H}IF$  with concept implication operator  $\rightarrow$ , for Łukasiewicz logic over  $[0, 1]$  with acyclic $TBox$*
- *$S\mathcal{I}$  under any continuous  $t$ -norm over  $[0, 1]$  without  $TBox$*

# Reasoners

Languages supported by fuzzy ontology reasoners:

Reasoner	Fuzzy DL	Logic	Degrees	Other constructors	GUI
fuzzyDL	$SHIF(\mathbf{D})$	Z, $\perp$	General	Modifiers, rough, aggregation	•
Fire	$SHIN$	Z	Numbers		•
FPLGERDS	$ACC$	$\perp$	Numbers	Role negatio/top/bottom	
YADLR	$ALCOQ$	Z, $\perp$	General	Local reflexivity	
DeLorean	$SROIQ(\mathbf{D})$	Z, G	General	Modifiers, rough DL	•
GURDL	$ACC$	General	Numbers		No
FRESG	$ACC(\mathbf{D})$	Z	Numbers	Fuzzy datatype expressions	•
LiFR	$DLP$ fragment	Z	Numbers	Weighted concepts	
SMT-based solver	$AL\mathcal{E}$	$\Pi$	No	No	
DLMedia	$DLR-Lite$	Z, G	Numbers	Image similarity	•
SoftFacts	$DLR-Lite$	Z, G	Numbers	Fuzzy datatypes	•
ONTOSEARCH2	$DL - Lite_R$	General	Numbers		•

## Reasoning services offered by fuzzy ontology reasoners

Reasoner	CON	ENT	CSAT	SUB	IR	BDB	Other tasks	OPT
fuzzyDL	•	•	•	•	•	•	Defuzzification	•
Fire	•	•	•	•		•	Classification	•
FPLGERDS		•						
YADLR		Partial			•	Partial	Realisation	
DeLorean	•	•	•	•		•		•
GURDL	•	•		•				•
FRESG	•	•	•		•		Realisation	
LiFR		Partial	•	•		•		
SMT-based solver			•					
DLMedia							Top-k Image Retrieval	•
SoftFacts							Top-k CQA	•
ONTOSEARCH2							Retrieval	

“CON”, “ENT”, “CSAT”, “SUB”, “IR”, “BED”, and “OPT” represent consistency, entailment, concept satisfiability, subsumption, instance retrieval, BED, and optimisations, respectively

That's it !