

# Merging Incommensurable Possibilistic *DL-Lite* Assertional Bases

S. Benferhat<sup>1</sup>   Z. Bouraoui<sup>1</sup>   S. Lagrue<sup>1</sup>   J. Rossit<sup>2</sup>

<sup>1</sup> CRIL-CNRS, Univ. d'Artois, {benferhat,bouraoui,lagrue}@cril.fr,

<sup>2</sup> LIPADE, Univ Paris Descartes, julien.rossit@parisdescartes.fr

## 3 main notions

- Merging multiple-source uncertain information
- Incommensurability of uncertainty scales
  - Assessment marks
    - ▶ marked on the 0-100 scale
    - ▶ marked on the 0-20 scale
    - ▶ Using qualitative scale : A+, A, A-, etc
- Lightweight ontologies (DL-lite)

# Why lightweight DL?

## Which language to use?

- Each knowledge base format is suitable for some applications

# Why lightweight DL?

## Which language to use?

- Each knowledge base format is suitable for some applications
- In general, the more expressive is the language the more hard is its inference relations

# Why lightweight DL?

## Which language to use?

- Each knowledge base format is suitable for some applications
- In general, the more expressive is the language the more hard is its inference relations
- Always, one needs to reach for a good compromise between expressiveness and computational issues.

# Why lightweight DL?

## Which language to use?

- Each knowledge base format is suitable for some applications
- In general, the more expressive is the language the more hard is its inference relations
- Always, one needs to reach for a good compromise between expressiveness and computational issues.

## Nice features of DL-Lite

- A reasonable expressive language

# Why lightweight DL?

## Which language to use?

- Each knowledge base format is suitable for some applications
- In general, the more expressive is the language the more hard is its inference relations
- Always, one needs to reach for a good compromise between expressiveness and computational issues.

## Nice features of DL-Lite

- A reasonable expressive language
- DL-lite logics are appropriate for applications where queries need to be efficiently handled

# Why lightweight DL?

## Which language to use?

- Each knowledge base format is suitable for some applications
- In general, the more expressive is the language the more hard is its inference relations
- Always, one needs to reach for a good compromise between expressiveness and computational issues.

## Nice features of DL-Lite

- A reasonable expressive language
- DL-lite logics are appropriate for applications where queries need to be efficiently handled
- Tractable methods for computing conflicts.



## DL-lite: vocabulary

The starting points are  $N_C$ ,  $N_R$  and  $N_I$ , three pairwise disjoint sets :

## DL-lite: vocabulary

The starting points are  $N_C$ ,  $N_R$  and  $N_I$ , three pairwise disjoint sets :

- set of atomic concepts,

## DL-lite: vocabulary

The starting points are  $N_C$ ,  $N_R$  and  $N_I$ , three pairwise disjoint sets :

- set of atomic concepts,
- set of atomic roles and

## DL-lite: vocabulary

The starting points are  $N_C$ ,  $N_R$  and  $N_I$ , three pairwise disjoint sets :

- set of atomic concepts,
- set of atomic roles and
- set of individuals.

## DL-lite: vocabulary

The starting points are  $N_C$ ,  $N_R$  and  $N_I$ , three pairwise disjoint sets :

- set of atomic concepts,
- set of atomic roles and
- set of individuals.

## ABox

Let  $a$  and  $b$  be two individuals. An ABox is a set of:

- Membership assertions on atomic concepts:

$$A(a)$$

## DL-lite: vocabulary

The starting points are  $N_C$ ,  $N_R$  and  $N_I$ , three pairwise disjoint sets :

- set of atomic concepts,
- set of atomic roles and
- set of individuals.

## ABox

Let  $a$  and  $b$  be two individuals. An ABox is a set of:

- Membership assertions on atomic concepts:

$$A(a)$$

- membership assertions on atomic roles:

$$P(a, b)$$

DL-lite: unary connectors

To define complex concepts and roles:

### DL-lite: unary connectors

To define complex concepts and roles:

- $\neg$  (negated concepts or roles),



## DL-lite: unary connectors

To define complex concepts and roles:

- $\neg$  (negated concepts or roles),
- $\exists$  (set of individuals obtained by projection on the first dimension of a role)

## DL-lite: unary connectors

To define complex concepts and roles:

- $\neg$  (negated concepts or roles),
- $\exists$  (set of individuals obtained by projection on the first dimension of a role)
- $^{-1}$  (inverse relation)

## DL-lite: unary connectors

To define complex concepts and roles:

- $\neg$  (negated concepts or roles),
- $\exists$  (set of individuals obtained by projection on the first dimension of a role)
- $\bar{\phantom{x}}$  (inverse relation)

## TBOX of DL-lite<sub>core</sub>

DL-Lite<sub>core</sub> TBox consists of a set of concept inclusion assertions:

$$B_1 \sqsubseteq B_2, \quad B_1 \sqsubseteq \neg B_2,$$

with

## DL-lite: unary connectors

To define complex concepts and roles:

- $\neg$  (negated concepts or roles),
- $\exists$  (set of individuals obtained by projection on the first dimension of a role)
- $^{-}$  (inverse relation)

## TBOX of DL-lite<sub>core</sub>

DL-Lite<sub>core</sub> TBox consists of a set of concept inclusion assertions:

$$B_1 \sqsubseteq B_2, \quad B_1 \sqsubseteq \neg B_2,$$

with

$$B_i \longrightarrow A \mid \exists P \mid \exists P^{-}$$

# Problem : merging DL-Lite<sup>π</sup>

## Contexte

- DL-Lite<sup>π</sup> :  $\mathcal{K} = \mathcal{T} \cup \mathcal{A} = \{(\phi, \alpha) : \phi \in \text{DL-Lite} \text{ and } \alpha \in ]0, 1]\}$

# Problem : merging DL-Lite $^\pi$

## Contexte

- DL-Lite $^\pi$  :  $\mathcal{K} = \mathcal{T} \cup \mathcal{A} = \{(\phi, \alpha) : \phi \in \text{DL-Lite} \text{ and } \alpha \in ]0, 1]\}$
- Input :  $E = \{\mathcal{K}_1, \dots, \mathcal{K}_n\}$  where  $\mathcal{K}_i = \mathcal{T}_i \cup \mathcal{A}_i$  is a DL-Lite $^\pi$

# Problem : merging DL-Lite $^\pi$

## Contexte

- DL-Lite $^\pi$  :  $\mathcal{K} = \mathcal{T} \cup \mathcal{A} = \{(\phi, \alpha) : \phi \in \text{DL-Lite} \text{ and } \alpha \in ]0, 1]\}$
- Input :  $E = \{\mathcal{K}_1, \dots, \mathcal{K}_n\}$  where  $\mathcal{K}_i = \mathcal{T}_i \cup \mathcal{A}_i$  is a DL-Lite $^\pi$
- Output : weighted DL-lite base  $\Delta(E) = \mathcal{T} \cup \mathcal{A}$

# Problem : merging DL-Lite $^\pi$

## Contexte

- DL-Lite $^\pi$  :  $\mathcal{K} = \mathcal{T} \cup \mathcal{A} = \{(\phi, \alpha) : \phi \in \text{DL-Lite} \text{ and } \alpha \in ]0, 1]\}$
- Input :  $E = \{\mathcal{K}_1, \dots, \mathcal{K}_n\}$  where  $\mathcal{K}_i = \mathcal{T}_i \cup \mathcal{A}_i$  is a DL-Lite $^\pi$
- Output : weighted DL-lite base  $\Delta(E) = \mathcal{T} \cup \mathcal{A}$

## Assumptions

- Sources share the same ontology :  $\mathcal{T}_1 = \dots = \mathcal{T}_n$



# Problem : merging DL-Lite <sup>$\pi$</sup>

## Contexte

- DL-Lite <sup>$\pi$</sup>  :  $\mathcal{K} = \mathcal{T} \cup \mathcal{A} = \{(\phi, \alpha) : \phi \in \text{DL-Lite} \text{ and } \alpha \in ]0, 1]\}$
- Input :  $E = \{\mathcal{K}_1, \dots, \mathcal{K}_n\}$  where  $\mathcal{K}_i = \mathcal{T}_i \cup \mathcal{A}_i$  is a DL-Lite <sup>$\pi$</sup>
- Output : weighted DL-lite base  $\Delta(E) = \mathcal{T} \cup \mathcal{A}$

## Assumptions

- Sources share the same ontology :  $\mathcal{T}_1 = \dots = \mathcal{T}_n$
- $\mathcal{T} = \mathcal{T}_i$  is viewed as a constraint (degree = 1)

# Problem : merging DL-Lite <sup>$\pi$</sup>

## Contexte

- DL-Lite <sup>$\pi$</sup>  :  $\mathcal{K} = \mathcal{T} \cup \mathcal{A} = \{(\phi, \alpha) : \phi \in \text{DL-Lite} \text{ and } \alpha \in ]0, 1]\}$
- Input :  $E = \{\mathcal{K}_1, \dots, \mathcal{K}_n\}$  where  $\mathcal{K}_i = \mathcal{T}_i \cup \mathcal{A}_i$  is a DL-Lite <sup>$\pi$</sup>
- Output : weighted DL-lite base  $\Delta(E) = \mathcal{T} \cup \mathcal{A}$

## Assumptions

- Sources share the same ontology :  $\mathcal{T}_1 = \dots = \mathcal{T}_n$
- $\mathcal{T} = \mathcal{T}_i$  is viewed as a constraint (degree = 1)
- Each  $\mathcal{T}_i \cup \mathcal{A}_i$  is consistent

# Problem : merging DL-Lite $^{\pi}$

## Contexte

- DL-Lite $^{\pi}$  :  $\mathcal{K} = \mathcal{T} \cup \mathcal{A} = \{(\phi, \alpha) : \phi \in \text{DL-Lite} \text{ and } \alpha \in ]0, 1]\}$
- Input :  $E = \{\mathcal{K}_1, \dots, \mathcal{K}_n\}$  where  $\mathcal{K}_i = \mathcal{T}_i \cup \mathcal{A}_i$  is a DL-Lite $^{\pi}$
- Output : weighted DL-lite base  $\Delta(E) = \mathcal{T} \cup \mathcal{A}$

## Assumptions

- Sources share the same ontology :  $\mathcal{T}_1 = \dots = \mathcal{T}_n$
- $\mathcal{T} = \mathcal{T}_i$  is viewed as a constraint (degree = 1)
- Each  $\mathcal{T}_i \cup \mathcal{A}_i$  is consistent
- Sources do not share the same uncertainty scale

## Principle

- If  $\mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$  is consistent with  $\mathcal{T}$ , then

$$\Delta_{\pi}^{\mathcal{T}}(E) = \mathcal{T} \cup \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$$

## Principle

- If  $\mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$  is consistent with  $\mathcal{T}$ , then

$$\Delta_{\pi}^{\mathcal{T}}(E) = \mathcal{T} \cup \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$$

- For each source  $i$ , rank-order the interpretations  $\mathcal{I}$  with respect to the highest assertion that is rejected from  $A_i$ .

## Principle

- If  $\mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$  is consistent with  $\mathcal{T}$ , then

$$\Delta_{\pi}^{\mathcal{T}}(E) = \mathcal{T} \cup \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$$

- For each source  $i$ , rank-order the interpretations  $\mathcal{I}$  with respect to the highest assertion that is rejected from  $A_i$ .
- More precisely:

$$\pi_i(\mathcal{I}) = 1 - \max\{f : f \in \mathcal{A}_i, \mathcal{I} \not\models f\}.$$

## Principle

- If  $\mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$  is consistent with  $\mathcal{T}$ , then

$$\Delta_{\pi}^{\mathcal{T}}(E) = \mathcal{T} \cup \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$$

- For each source  $i$ , rank-order the interpretations  $\mathcal{I}$  with respect to the highest assertion that is rejected from  $\mathcal{A}_i$ .
- More precisely:

$$\pi_i(\mathcal{I}) = 1 - \max\{f : f \in \mathcal{A}_i, \mathcal{I} \not\models f\}.$$

- Combine  $\pi_i$ 's (with the minimum operation) to select the result of merging.

# Possibilistic merging

## Example

- $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg C\}$
- $\mathcal{A}_1 = \{(A(a), .6) (C(b), .5)\}$
- $\mathcal{A}_2 = \{(C(a), .4) (B(b), .8), (A(b), .7)\}$ .

$\mathcal{I}$	$\mathcal{I}$	$\pi_{\mathcal{A}_1}$	$\pi_{\mathcal{A}_2}$	$\Delta_{\mathcal{T}}^{\min}(\mathcal{A})$
$\mathcal{I}_1$	$A = \{a\}, B = \{a\}, C = \{b\}$	1	.2	.2
$\mathcal{I}_2$	$A = \{\}, B = \{\}, C = \{a, b\}$	.4	.2	.4
$\mathcal{I}_3$	$A = \{a, b\}, B = \{a, b\}, C = \{\}$	.5	.6	.5
$\mathcal{I}_4$	$A = \{b\}, B = \{b\}, C = \{a\}$	.4	1	.4



# Possibilistic merging

## Example

- $\mathcal{T} = \{A \sqsubseteq B, B \sqsubseteq \neg C\}$
- $\mathcal{A}_1 = \{(A(a), .6) (C(b), .5)\}$
- $\mathcal{A}_2 = \{(C(a), .4) (B(b), .8), (A(b), .7)\}$ .

$\mathcal{I}$	$\mathcal{I}$	$\pi_{\mathcal{A}_1}$	$\pi_{\mathcal{A}_2}$	$\Delta_{\mathcal{T}}^{min}(\mathcal{A})$
$\mathcal{I}_1$	$A=\{a\}, B=\{a\}, C=\{b\}$	1	.2	.2
$\mathcal{I}_2$	$A=\{\}, B=\{\}, C=\{a,b\}$	.4	.2	.4
$\mathcal{I}_3$	$A=\{a,b\}, B=\{a,b\}, C=\{\}$	.5	.6	.5
$\mathcal{I}_4$	$A=\{b\}, B=\{b\}, C=\{a\}$	.4	1	.4

- $[\Delta_{\mathcal{T}}^{min}(\mathcal{A})] = \mathcal{I}_3$

# At the syntactic level

## Method

1 Define :  $\mathcal{A}_\oplus = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$

# At the syntactic level

## Method

- 1 Define :  $\mathcal{A}_\oplus = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$
- 2 Compute  $x = \text{Inc}(\mathcal{T} \cup \mathcal{A}_\oplus)$

# At the syntactic level

## Method

- 1 Define :  $\mathcal{A}_\oplus = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$
- 2 Compute  $x = \text{Inc}(\mathcal{T} \cup \mathcal{A}_\oplus)$
- 3  $\Delta_\pi^{\mathcal{T}}(E) = \mathcal{T} \cup \{(\phi, \alpha) : (\phi, \alpha) \in \mathcal{A}_\oplus \text{ and } \alpha > x\}$

# At the syntactic level

## Method

- 1 Define :  $\mathcal{A}_\oplus = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$
- 2 Compute  $x = \text{Inc}(\mathcal{T} \cup \mathcal{A}_\oplus)$
- 3  $\Delta_\pi^{\mathcal{T}}(E) = \mathcal{T} \cup \{(\phi, \alpha) : (\phi, \alpha) \in \mathcal{A}_\oplus \text{ and } \alpha > x\}$

## Remarks

- Computing  $\Delta_\pi^{\mathcal{T}}(E)$  is done in a polynomial time.

# At the syntactic level

## Method

- 1 Define :  $\mathcal{A}_\oplus = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$
- 2 Compute  $x = \text{Inc}(\mathcal{T} \cup \mathcal{A}_\oplus)$
- 3  $\Delta_\pi^{\mathcal{T}}(E) = \mathcal{T} \cup \{(\phi, \alpha) : (\phi, \alpha) \in \mathcal{A}_\oplus \text{ and } \alpha > x\}$

## Remarks

- Computing  $\Delta_\pi^{\mathcal{T}}(E)$  is done in a polynomial time.
- Question:

# At the syntactic level

## Method

- 1 Define :  $\mathcal{A}_\oplus = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \dots \cup \mathcal{A}_n$
- 2 Compute  $x = \text{Inc}(\mathcal{T} \cup \mathcal{A}_\oplus)$
- 3  $\Delta_\pi^{\mathcal{T}}(E) = \mathcal{T} \cup \{(\phi, \alpha) : (\phi, \alpha) \in \mathcal{A}_\oplus \text{ and } \alpha > x\}$

## Remarks

- Computing  $\Delta_\pi^{\mathcal{T}}(E)$  is done in a polynomial time.
- Question:  
How to extend the possibilistic merging when the uncertainty scales are incommensurable?

# Compatible-based merging

## Principle

Incommensurable merging

=

Family of compatible and commensurable merging

## Example

$$\begin{aligned}\mathcal{T} &= \{A \sqsubseteq B, B \sqsubseteq \neg C\} \\ \mathcal{A}_1 &= \{(A(a), .6) (C(b), .5)\} \\ \mathcal{A}_2 &= \{(C(a), .4) (B(b), .8), (A(b), .7)\}.\end{aligned}$$



# Compatible-based merging

## Principle

Incommensurable merging  
=  
Family of compatible and commensurable merging

## Example

$$\begin{aligned}\mathcal{T} &= \{A \sqsubseteq B, B \sqsubseteq \neg C\} \\ \mathcal{A}_1 &= \{(A(a), .6) (C(b), .5)\} \\ \mathcal{A}_2 &= \{(C(a), .4) (B(b), .8), (A(b), .7)\}.\end{aligned}$$

	$\mathcal{R}_1(A_i, f_{ij})$
$A_1$	.6 .5
$A_2$	.4 .8 .7

# Compatible-based merging

## Principle

Incommensurable merging  
=  
Family of compatible and commensurable merging

## Example

$$\begin{aligned}\mathcal{T} &= \{A \sqsubseteq B, B \sqsubseteq \neg C\} \\ \mathcal{A}_1 &= \{(A(a), .6) (C(b), .5)\} \\ \mathcal{A}_2 &= \{(C(a), .4) (B(b), .8), (A(b), .7)\}.\end{aligned}$$

	$\mathcal{R}_1(A_i, f_{ij})$
$A_1$	.6 .5
$A_2$	.4 .8 .7

	$\mathcal{R}_2(A_i, f_{ij})$
$A_1$	.5 .2
$A_2$	.3 .7 .4

# Compatible-based merging

## Principle

Incommensurable merging

=

Family of compatible and commensurable merging

## Example

$$\begin{aligned}\mathcal{T} &= \{A \sqsubseteq B, B \sqsubseteq \neg C\} \\ \mathcal{A}_1 &= \{(A(a), .6) (C(b), .5)\} \\ \mathcal{A}_2 &= \{(C(a), .4) (B(b), .8), (A(b), .7)\}.\end{aligned}$$

	$\mathcal{R}_1(A_i, f_{ij})$
$A_1$	.6 .5
$A_2$	.4 .8 .7

	$\mathcal{R}_2(A_i, f_{ij})$
$A_1$	.5 .2
$A_2$	.3 .7 .4

	$\mathcal{R}_3(A_i, f_{ij})$
$A_1$	.4 .7
$A_2$	.3 .6 .2

- Define a partial pre-order over interprétations

$$\mathcal{I} <_{\forall}^{\mathcal{A}} \mathcal{I}' \iff \forall \mathcal{R} \in \mathcal{R}(\mathcal{A}), \mathcal{I} \triangleleft_{\text{Min}}^{\mathcal{A}^{\mathcal{R}}} \mathcal{I}'$$

- Select the best ones to define the result of merging

$$\text{Mod}(\Delta_{\mathcal{T}}^{\forall}(\mathcal{A})) = \{\mathcal{I} \in \text{Mod}(\mathcal{T}) : \nexists \mathcal{I}' \in \text{Mod}(\mathcal{T}), \mathcal{I}' <_{\forall}^{\mathcal{A}} \mathcal{I}\}$$

- Define a partial pre-order over interprétations

$$\mathcal{I} <_{\forall}^{\mathcal{A}} \mathcal{I}' \iff \forall \mathcal{R} \in \mathcal{R}(\mathcal{A}), \mathcal{I} \triangleleft_{\text{Min}}^{\mathcal{A}^{\mathcal{R}}} \mathcal{I}'$$

- Select the best ones to define the result of merging

$$\text{Mod}(\Delta_{\mathcal{T}}^{\forall}(\mathcal{A})) = \{\mathcal{I} \in \text{Mod}(\mathcal{T}) : \nexists \mathcal{I}' \in \text{Mod}(\mathcal{T}), \mathcal{I}' <_{\forall}^{\mathcal{A}} \mathcal{I}\}$$

# Exemple

## Example (continued)

- $\mathcal{A}_1 = \{(A(a), .6), (C(b), .5)\}$
- $\mathcal{A}_2 = \{(C(a), .4), (B(b), .8), (A(b), .7)\}$

# Exemple

## Example (continued)

- $\mathcal{A}_1 = \{(A(a), .6), (C(b), .5)\}$
- $\mathcal{A}_2 = \{(C(a), .4), (B(b), .8), (A(b), .7)\}$
- $\mathcal{A}_1^{\mathcal{R}_1} = \{(A(a), .8), (C(b), .4)\}$
- $\mathcal{A}_2^{\mathcal{R}_1} = \{(C(a), .2), (B(b), .9), (A(b), .6)\}$

# Exemple

## Example (continued)

- $\mathcal{A}_1 = \{(A(a), .6), (C(b), .5)\}$
- $\mathcal{A}_2 = \{(C(a), .4), (B(b), .8), (A(b), .7)\}$
- $\mathcal{A}_1^{\mathcal{R}_1} = \{(A(a), .8), (C(b), .4)\}$
- $\mathcal{A}_2^{\mathcal{R}_1} = \{(C(a), .2), (B(b), .9), (A(b), .6)\}$
- $\mathcal{A}_1^{\mathcal{R}_2} = \{(A(a), .4), (C(b), .2)\}$
- $\mathcal{A}_2^{\mathcal{R}_2} = \{(C(a), .3), (B(b), .6), (A(b), .5)\}$



# Exemple

## Example (continued)

- $\mathcal{A}_1 = \{(A(a), .6), (C(b), .5)\}$
- $\mathcal{A}_2 = \{(C(a), .4), (B(b), .8), (A(b), .7)\}$
- $\mathcal{A}_1^{\mathcal{R}_1} = \{(A(a), .8), (C(b), .4)\}$
- $\mathcal{A}_2^{\mathcal{R}_1} = \{(C(a), .2), (B(b), .9), (A(b), .6)\}$
- $\mathcal{A}_1^{\mathcal{R}_2} = \{(A(a), .4), (C(b), .2)\}$
- $\mathcal{A}_2^{\mathcal{R}_2} = \{(C(a), .3), (B(b), .6), (A(b), .5)\}$

$\mathcal{I}$	$\nu_{\mathcal{A}^{\mathcal{R}_1}}(\mathcal{I})$	Min	$\nu_{\mathcal{A}^{\mathcal{R}_2}}(\mathcal{I})$	Min
$\mathcal{I}_1$	$\langle 1, .1 \rangle$	.1	$\langle 1, .4 \rangle$	.4
$\mathcal{I}_2$	$\langle .2, .1 \rangle$	.1	$\langle .6, .4 \rangle$	.4
$\mathcal{I}_3$	$\langle .6, .8 \rangle$	<b>.6</b>	$\langle .8, .7 \rangle$	<b>.7</b>
$\mathcal{I}_4$	$\langle .2, 1 \rangle$	.2	$\langle .6, 1 \rangle$	.6

# Exemple

## Example (continued)

- $\mathcal{A}_1 = \{(A(a), .6), (C(b), .5)\}$
- $\mathcal{A}_2 = \{(C(a), .4), (B(b), .8), (A(b), .7)\}$
- $\mathcal{A}_1^{\mathcal{R}_1} = \{(A(a), .8), (C(b), .4)\}$
- $\mathcal{A}_2^{\mathcal{R}_1} = \{(C(a), .2), (B(b), .9), (A(b), .6)\}$
- $\mathcal{A}_1^{\mathcal{R}_2} = \{(A(a), .4), (C(b), .2)\}$
- $\mathcal{A}_2^{\mathcal{R}_2} = \{(C(a), .3), (B(b), .6), (A(b), .5)\}$

$\mathcal{I}$	$\nu_{\mathcal{A}^{\mathcal{R}_1}}(\mathcal{I})$	Min	$\nu_{\mathcal{A}^{\mathcal{R}_2}}(\mathcal{I})$	Min
$\mathcal{I}_1$	$\langle 1, .1 \rangle$	.1	$\langle 1, .4 \rangle$	.4
$\mathcal{I}_2$	$\langle .2, .1 \rangle$	.1	$\langle .6, .4 \rangle$	.4
$\mathcal{I}_3$	$\langle .6, .8 \rangle$	<b>.6</b>	$\langle .8, .7 \rangle$	<b>.7</b>
$\mathcal{I}_4$	$\langle .2, 1 \rangle$	.2	$\langle .6, 1 \rangle$	.6

## Important computational result

$Mod(\Delta_{\mathcal{T}}^{\forall}(\mathcal{A}))$  can be directly computed from  $A_i$ 's in a polynomial time.

$Mod(\Delta_{\mathcal{T}}^{\forall}(\mathcal{A}))$  can be directly computed from  $A_i$ 's in a polynomial time.  
Thanks to the facts:

- A conflict necessarily implies:
  - One NI axiom.
  - One or two membership assertions.
- A polynomial time algorithm to compute conflicts

# Selecting one compatible scale

## Normalisation

- $\alpha_{\mathcal{A}_i}$  : set of degrees in  $\mathcal{A}_i$
- $\min_{\mathcal{A}_i}$  (resp.  $\max_{\mathcal{A}_i}$ ) is the minimum (maximum) degree used in  $\alpha_{\mathcal{A}_i}$

$$N(\alpha_j) = \frac{\alpha_j - (\min_{\mathcal{A}_i} - \epsilon)}{\max_{\mathcal{A}_i} - (\min_{\mathcal{A}_i} - \epsilon)}$$

- $\alpha_j \in \alpha_{\mathcal{A}_i}$  and  $0 < \epsilon < \min_{\mathcal{A}_i}$

## Example

- $\mathcal{A}_1 = \{(A(a), .6), (C(b), .5)\}$
- $\mathcal{A}_2 = \{(C(a), .4), (B(b), .8), (A(b), .7)\}$

## Example

- $\mathcal{A}_1 = \{(A(a), .6), (C(b), .5)\}$
- $\mathcal{A}_2 = \{(C(a), .4), (B(b), .8), (A(b), .7)\}$

$$\min_{\mathcal{A}_1} = .5, \min_{\mathcal{A}_2} = .4, \max_{\mathcal{A}_1} = .6, \max_{\mathcal{A}_2} = .8 \text{ et } \epsilon = .01$$

## Example

- $\mathcal{A}_1 = \{(A(a), .6), (C(b), .5)\}$
- $\mathcal{A}_2 = \{(C(a), .4), (B(b), .8), (A(b), .7)\}$

$min_{\mathcal{A}_1} = .5, min_{\mathcal{A}_2} = .4, max_{\mathcal{A}_1} = .6, max_{\mathcal{A}_2} = .8$  et  $\epsilon = .01$

- $\mathcal{A}_1 = \{(A(a), 1), (C(b), .09)\}$
- $\mathcal{A}_2 = \{(C(a), 0, 02), (B(b), 1), (A(b), .75)\}$



# Conclusions

- Safe possibilistic DL-Lite KB Merging without commensurability assumption using compatible scales

# Conclusions

- Safe possibilistic DL-Lite KB Merging without commensurability assumption using compatible scales
- Merging in DL-lite <sup>$\pi$</sup>  setting is tractable while it is a hard in a (weighted) propositional setting
- Rational postulates for merging in DL-lite <sup>$\pi$</sup>  setting

# Conclusions

- Safe possibilistic DL-Lite KB Merging without commensurability assumption using compatible scales
- Merging in DL-lite<sup>π</sup> setting is tractable while it is a hard in a (weighted) propositional setting
- Rational postulates for merging in DL-lite<sup>π</sup> setting  
**(CSS)**  $\forall B_i \in E$ , if  $B_i^* \models \mu$ , then  $B_i^* \wedge \Delta^\mu(E)$  inconsistent

# Conclusions

- Safe possibilistic DL-Lite KB Merging without commensurability assumption using compatible scales
  - Merging in DL-lite<sup>π</sup> setting is tractable while it is a hard in a (weighted) propositional setting
  - Rational postulates for merging in DL-lite<sup>π</sup> setting
- (CSS)**  $\forall B_i \in E$ , if  $B_i^* \models \mu$ , then  $B_i^* \wedge \Delta^\mu(E)$  inconsistent
- In the propositional setting, no way to satisfy CCS when only selecting one compatible scale.

# Conclusions

- Safe possibilistic DL-Lite KB Merging without commensurability assumption using compatible scales
  - Merging in DL-lite<sup>π</sup> setting is tractable while it is a hard in a (weighted) propositional setting
  - Rational postulates for merging in DL-lite<sup>π</sup> setting
- (CSS)**  $\forall B_i \in E$ , if  $B_i^* \models \mu$ , then  $B_i^* \wedge \Delta^\mu(E)$  inconsistent
- In the propositional setting, no way to satisfy CCS when only selecting one compatible scale.
  - Is-it the case for DL-lite<sup>π</sup> setting.