Querying Inconsistent Prioritized Data

Camille Bourgaux

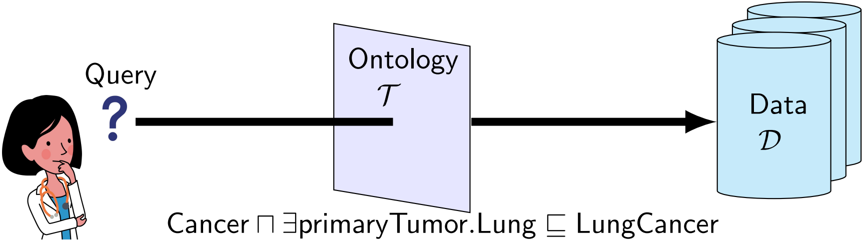
CNRS & DI ENS, Paris, France

joint work with Meghyn Bienvenu

DL workshop - June 2024

Ontology-mediated query answering

- Knowledge base: $\mathcal{K} = (\mathcal{D}, \mathcal{T})$
  - $\mathcal{D}$ dataset
  - $\mathcal{T}$ (consistent) logical theory (DL ontology, database constraints...)
- Conjunctive query: $q(\vec{x}) = \exists \vec{y} \varphi$ with φ conjunction of atoms
- $\mathcal{K} \models q(\vec{a})$ if $q(\vec{a})$ holds in every model of $\mathcal{K}$



$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor}.\mathsf{Lung} \sqsubseteq \mathsf{LungCancer}$

$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$

$q(x) =$
$\exists y \ \mathsf{hasDisease}(x, y)$
$\wedge \ \mathsf{LungCancer}(y)$

$\mathsf{hasDisease}(bob, d_1)$
$\mathsf{SmallCellCarcinoma}(d_1)$
$\mathsf{primaryTumor}(d_1, o_1)$
$\mathsf{Lung}(o_1)$

Handling inconsistent data

Problem: if $\mathcal{K}$ is inconsistent, $\mathcal{K} \models q$ for every BCQ q

$\text{Cancer} \sqcap \exists\text{primaryTumor.Lung} \sqsubseteq \text{LungCancer}$
$\text{SmallCellCarcinoma} \sqsubseteq \text{Cancer}$
$\text{Adenocarcinoma} \sqsubseteq \text{Cancer}$
$\text{Adenocarcinoma} \sqcap \text{SmallCellCarcinoma} \sqsubseteq \bot$
(functional primaryTumor)
$\text{Lung} \sqcap \text{Breast} \sqsubseteq \bot$

$\text{hasDisease}(bob, d_1)$

$\text{SmallCellCarcinoma}(d_1)$ $\text{Adenocarcinoma}(d_1)$

$\text{primaryTumor}(d_1, o_1)$ $\text{primaryTumor}(d_1, o_2)$
$\text{Lung}(o_1)$ $\text{Breast}(o_2)$

Handling inconsistent data

Problem: if $\mathcal{K}$ is inconsistent, $\mathcal{K} \models q$ for every BCQ q

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor}.\mathsf{Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
(functional primaryTumor)
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

$\mathsf{hasDisease}(bob, d_1)$

$\mathsf{SmallCellCarcinoma}(d_1)$ $\mathsf{Adenocarcinoma}(d_1)$

$\mathsf{primaryTumor}(d_1, o_1)$ $\mathsf{primaryTumor}(d_1, o_2)$
$\mathsf{Lung}(o_1)$ $\mathsf{Breast}(o_2)$

Handling inconsistent data

Problem: if $\mathcal{K}$ is inconsistent, $\mathcal{K} \models q$ for every BCQ q

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor}.\mathsf{Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
$(\mathsf{functional}\ \mathsf{primaryTumor})$
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

$\mathsf{hasDisease}(bob, d_1)$

$\mathsf{SmallCellCarcinoma}(d_1)$ $\mathsf{Adenocarcinoma}(d_1)$

$\mathsf{primaryTumor}(d_1, o_1)$ $\mathsf{primaryTumor}(d_1, o_2)$
$\mathsf{Lung}(o_1)$ $\mathsf{Breast}(o_2)$

$\mathcal{K} \models \exists y \mathsf{hasDisease}(x) \wedge \mathsf{LungCancer}(x)$ for $x \in \{bob, d_1, d_2, o_1, o_2\}$

$\Rightarrow$ Use inconsistency-tolerant semantics

Handling inconsistent data

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor}.\mathsf{Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
$(\mathsf{functional}\ \mathsf{primaryTumor})$
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

$\mathsf{hasDisease}(bob, d_1)$

$\mathsf{SmallCellCarcinoma}(d_1)$ $\mathsf{Adenocarcinoma}(d_1)$

$\mathsf{primaryTumor}(d_1, o_1)$ $\mathsf{primaryTumor}(d_1, o_2)$
$\mathsf{Lung}(o_1)$ $\mathsf{Breast}(o_2)$

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$

Handling inconsistent data

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor.Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
(functional primaryTumor)
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

hasDisease(bob, d_1)

~~SmallCellCarcinoma(d_1)~~ Adenocarcinoma(d_1)

~~primaryTumor(d_1, o_1)~~ primaryTumor(d_1, o_2)

Lung(o_1) Breast(o_2)

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$

Handling inconsistent data

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor}.\mathsf{Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
(functional primaryTumor)
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

$\mathsf{hasDisease}(bob, d_1)$

$\mathsf{SmallCellCarcinoma}(d_1)$ ~~$\mathsf{Adenocarcinoma}(d_1)$~~

~~$\mathsf{primaryTumor}(d_1, o_1)$~~ $\mathsf{primaryTumor}(d_1, o_2)$

$\mathsf{Lung}(o_1)$ $\mathsf{Breast}(o_2)$

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$

Handling inconsistent data

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor}.\mathsf{Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
(functional primaryTumor)
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

hasDisease(bob, d_1)

~~SmallCellCarcinoma(d_1)~~ Adenocarcinoma(d_1)

primaryTumor(d_1, o_1) ~~primaryTumor(d_1, o_2)~~

Lung(o_1) Breast(o_2)

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$

Handling inconsistent data

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor.Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
$(\mathsf{functional\ primaryTumor})$
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

$\mathsf{hasDisease}(bob, d_1)$

$\mathsf{SmallCellCarcinoma}(d_1)$ ~~$\mathsf{Adenocarcinoma}(d_1)$~~

$\mathsf{primaryTumor}(d_1, o_1)$ ~~$\mathsf{primaryTumor}(d_1, o_2)$~~
$\mathsf{Lung}(o_1)$ $\mathsf{Breast}(o_2)$

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$

Handling inconsistent data

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor.Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
(functional primaryTumor)
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

hasDisease(bob, d_1)

~~SmallCellCarcinoma(d_1)~~ Adenocarcinoma(d_1)

primaryTumor(d_1, o_1) primaryTumor(d_1, o_2)
~~Lung(o_1)~~ Breast(o_2)

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$

Handling inconsistent data

$\text{Cancer} \sqcap \exists \text{primaryTumor}.\text{Lung} \sqsubseteq \text{LungCancer}$
$\text{SmallCellCarcinoma} \sqsubseteq \text{Cancer}$
$\text{Adenocarcinoma} \sqsubseteq \text{Cancer}$
$\text{Adenocarcinoma} \sqcap \text{SmallCellCarcinoma} \sqsubseteq \bot$
$(\text{functional primaryTumor})$
$\text{Lung} \sqcap \text{Breast} \sqsubseteq \bot$

$\text{hasDisease}(bob, d_1)$

$\text{SmallCellCarcinoma}(d_1)$ ~~$\text{Adenocarcinoma}(d_1)$~~

$\text{primaryTumor}(d_1, o_1)$ $\text{primaryTumor}(d_1, o_2)$

~~$\text{Lung}(o_1)$~~ $\text{Breast}(o_2)$

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$

Handling inconsistent data

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor}.\mathsf{Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
$(\mathsf{functional\ primaryTumor})$
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

$\mathsf{hasDisease}(bob, d_1)$

~~$\mathsf{SmallCellCarcinoma}(d_1)$~~ $\mathsf{Adenocarcinoma}(d_1)$

$\mathsf{primaryTumor}(d_1, o_1)$ $\mathsf{primaryTumor}(d_1, o_2)$
$\mathsf{Lung}(o_1)$ ~~$\mathsf{Breast}(o_2)$~~

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$

Handling inconsistent data

$\text{Cancer} \sqcap \exists \text{primaryTumor.Lung} \sqsubseteq \text{LungCancer}$
$\text{SmallCellCarcinoma} \sqsubseteq \text{Cancer}$
$\text{Adenocarcinoma} \sqsubseteq \text{Cancer}$
$\text{Adenocarcinoma} \sqcap \text{SmallCellCarcinoma} \sqsubseteq \bot$
(functional primaryTumor)
$\text{Lung} \sqcap \text{Breast} \sqsubseteq \bot$

hasDisease(bob, d_1)

SmallCellCarcinoma(d_1) ~~Adenocarcinoma(d_1)~~

primaryTumor(d_1, o_1) primaryTumor(d_1, o_2)
Lung(o_1) ~~Breast(o_2)~~

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$

Handling inconsistent data

$$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor}.\mathsf{Lung} \sqsubseteq \mathsf{LungCancer}$$
$$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$$
$$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$$
$$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$$
$$(\mathsf{functional\ primaryTumor})$$
$$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$$

$$\mathsf{hasDisease}(bob, d_1)$$
$$\mathsf{SmallCellCarcinoma}(d_1) \qquad \mathsf{Adenocarcinoma}(d_1)$$
$$\mathsf{primaryTumor}(d_1, o_1) \qquad \mathsf{primaryTumor}(d_1, o_2)$$
$$\mathsf{Lung}(o_1) \qquad \mathsf{Breast}(o_2)$$

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$
- AR semantics: queries that hold in every repair

$$\exists y\, \mathsf{hasDisease}(bob, y) \wedge \mathsf{Cancer}(y)$$ plausible/likely

Handling inconsistent data

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor.Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
$(\mathsf{functional\ primaryTumor})$
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

$\mathsf{hasDisease}(bob, d_1)$

$\mathsf{SmallCellCarcinoma}(d_1)$ $\mathsf{Adenocarcinoma}(d_1)$

$\mathsf{primaryTumor}(d_1, o_1)$ $\mathsf{primaryTumor}(d_1, o_2)$
$\mathsf{Lung}(o_1)$ $\mathsf{Breast}(o_2)$

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$
- Brave semantics: queries that hold in some repair

$\exists y\, \mathsf{hasDisease}(bob, y) \wedge \mathsf{LungCancer}(y)$ possible

Handling inconsistent data

$\mathsf{Cancer} \sqcap \exists \mathsf{primaryTumor}.\mathsf{Lung} \sqsubseteq \mathsf{LungCancer}$
$\mathsf{SmallCellCarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqsubseteq \mathsf{Cancer}$
$\mathsf{Adenocarcinoma} \sqcap \mathsf{SmallCellCarcinoma} \sqsubseteq \bot$
$(\mathsf{functional\ primaryTumor})$
$\mathsf{Lung} \sqcap \mathsf{Breast} \sqsubseteq \bot$

$\mathsf{hasDisease}(bob, d_1)$

$\mathsf{SmallCellCarcinoma}(d_1)$ $\mathsf{Adenocarcinoma}(d_1)$

$\mathsf{primaryTumor}(d_1, o_1)$ $\mathsf{primaryTumor}(d_1, o_2)$
$\mathsf{Lung}(o_1)$ $\mathsf{Breast}(o_2)$

- (Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not\models \bot$
- IAR semantics: queries that hold in the intersection of all repairs

$\exists y\, \mathsf{hasDisease}(bob, y)$ surest

Adding priorities

When information about relative reliability of facts is available, define priorities between conflicting facts

Examples of possible preferences

- prefer more recent (updated) or older (curated) facts

Fact	Date
primaryTumor(d_1, o_1)	08.10.2023
primaryTumor(d_1, o_2)	05.22.2023

most recent fact gives the last, revised, diagnosis

$\Rightarrow$ primaryTumor(d_1, o_1) $\succ$ primaryTumor(d_1, o_2)

Adding priorities

When information about relative reliability of facts is available, define priorities between conflicting facts

Examples of possible preferences

- prefer more recent (updated) or older (curated) facts
- prefer facts that come from some source (process, user...)

Fact	Source
Adenocarcinoma(d_1)	X-ray report
SmallCellCarcinoma(d_1)	biopsy report

the second diagnostic method is more reliable

$\Rightarrow$ SmallCellCarcinoma(d_1) $\succ$ Adenocarcinoma(d_1)

Adding priorities

When information about relative reliability of facts is available, define priorities between conflicting facts

Examples of possible preferences

- prefer more recent (updated) or older (curated) facts
- prefer facts that come from some source (process, user...)
- take into account presence or absence of other facts in the dataset

  hasDisease(bob, d_1),
  primaryTumor(d_1, o_1), Lung(o_1),
  primaryTumor(d_1, o_2), Breast(o_2),
  gotSurgery(bob, s), BronchialDebridement(s)

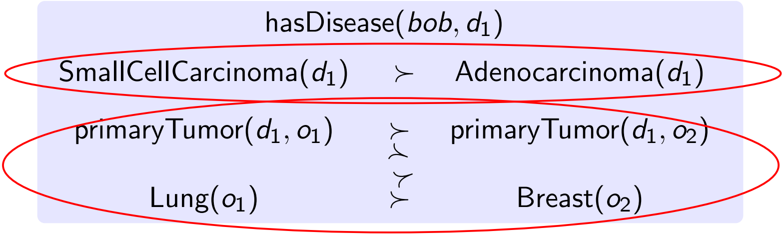
  the dataset indicates that the patient got a surgery common in the case of lung cancer but nothing about a breast cancer treatment
  $\Rightarrow$ primaryTumor(d_1, o_1), Lung(o_1) $\succ$ primaryTumor(d_1, o_2), Breast(o_2)

Adding priorities

When information about relative reliability of facts is available, define priorities between conflicting facts

Examples of possible preferences

- prefer more recent (updated) or older (curated) facts
- prefer facts that come from some source (process, user...)
- take into account presence or absence of other facts in the dataset
- ...



Adding priorities

Formally:

- Conflict: inclusion-minimal $\mathcal{C} \subseteq \mathcal{D}$ such that $(\mathcal{C}, \mathcal{T}) \models \bot$
- Priority relation $\succ$: acyclic binary relation over $\mathcal{D}$ such that $\alpha \succ \beta$ implies $\{\alpha, \beta\} \subseteq \mathcal{C}$ for some conflict $\mathcal{C}$

A prioritized KB $\mathcal{K}_{\succ}$ is a KB $\mathcal{K} = (\mathcal{D}, \mathcal{T})$ with a priority relation $\succ$ for $\mathcal{K}$

hasDisease(bob, d_1)

SmallCellCarcinoma(d_1) $\succ$ Adenocarcinoma(d_1)

primaryTumor(d_1, o_1) $\succ$ primaryTumor(d_1, o_2)

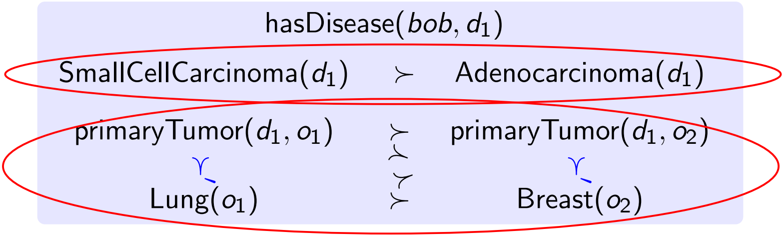
$\succ$ $\succ$

Lung(o_1) $\succ$ Breast(o_2)

Formally:

- Conflict: inclusion-minimal $\mathcal{C} \subseteq \mathcal{D}$ such that $(\mathcal{C}, \mathcal{T}) \models \bot$
- Priority relation $\succ$: acyclic binary relation over $\mathcal{D}$ such that $\alpha \succ \beta$ implies $\{\alpha, \beta\} \subseteq \mathcal{C}$ for some conflict $\mathcal{C}$

A prioritized KB $\mathcal{K}_\succ$ is a KB $\mathcal{K} = (\mathcal{D}, \mathcal{T})$ with a priority relation $\succ$ for $\mathcal{K}$



- $\succ$ is total if for all $\alpha \neq \beta$ such that $\{\alpha, \beta\} \subseteq \mathcal{C}$ for some conflict $\mathcal{C}$, either $\alpha \succ \beta$ or $\beta \succ \alpha$
- Completion of $\succ$: total priority relation $\succ' \supseteq \succ$
  - example: complete $\succ$ with primaryTumor$(d_1, o_1) \succ'$ Lung(o_1) and primaryTumor$(d_1, o_2) \succ'$ Breast(o_2)

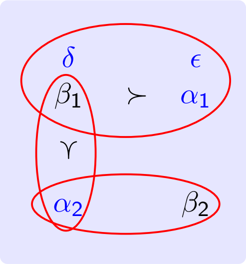
Optimal repairs

Import three notions of optimal repair from database setting

[Staworko, Chomicki, and Marcinkowski, 2012]

Let $\mathcal{R}$ be a repair of $\mathcal{K}$ ($\mathcal{R} \in \mathit{SRep}(\mathcal{K})$)

- A Pareto improvement of $\mathcal{R}$ is a $\mathcal{T}$-consistent $\mathcal{B} \subseteq \mathcal{D}$ such that there is $\beta \in \mathcal{B} \setminus \mathcal{R}$ with $\beta \succ \alpha$ for every $\alpha \in \mathcal{R} \setminus \mathcal{B}$
- $\mathcal{R}$ is Pareto-optimal ($\mathcal{R} \in \mathit{PRep}(\mathcal{K}_\succ)$) if there is no Pareto improvement of $\mathcal{R}$



$\{\alpha_1, \alpha_2, \delta, \epsilon\} \in \mathit{SRep}(\mathcal{K})$

$\{\beta_1, \delta, \epsilon\}$ Pareto improvement

$\Rightarrow \quad \{\alpha_1, \alpha_2, \delta, \epsilon\} \notin \mathit{PRep}(\mathcal{K}_\succ)$

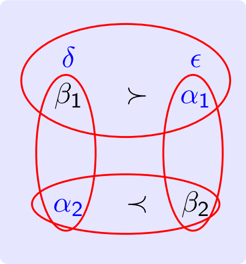
Optimal repairs

Import three notions of optimal repair from database setting

[Staworko, Chomicki, and Marcinkowski, 2012]

Let $\mathcal{R}$ be a repair of $\mathcal{K}$ ($\mathcal{R} \in \mathit{SRep}(\mathcal{K})$)

- A global improvement of $\mathcal{R}$ is a $\mathcal{T}$-consistent $\mathcal{B} \subseteq \mathcal{D}$ such that $\mathcal{B} \neq \mathcal{R}$ and for every $\alpha \in \mathcal{R} \setminus \mathcal{B}$, there is $\beta \in \mathcal{B} \setminus \mathcal{R}$ such that $\beta \succ \alpha$
- $\mathcal{R}$ is globally-optimal ($\mathcal{R} \in \mathit{GRep}(\mathcal{K}_\succ)$) if there is no global improvement of $\mathcal{R}$



$\{\alpha_1, \alpha_2, \delta, \epsilon\} \in \mathit{PRep}(\mathcal{K}_\succ)$

$\{\beta_1, \beta_2, \delta, \epsilon\}$ global improvement

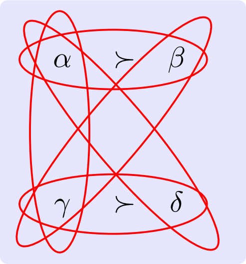
$\Rightarrow \quad \{\alpha_1, \alpha_2, \delta, \epsilon\} \notin \mathit{GRep}(\mathcal{K}_\succ)$

Import three notions of optimal repair from database setting

[Staworko, Chomicki, and Marcinkowski, 2012]

Let $\mathcal{R}$ be a repair of $\mathcal{K}$ ($\mathcal{R} \in \mathit{SRep}(\mathcal{K})$)

- $\mathcal{R}$ is completion-optimal ($\mathcal{R} \in \mathit{CRep}(\mathcal{K}_\succ)$) if $\mathcal{R}$ is globally-optimal w.r.t. some completion $\succ'$ of $\succ$
- Equivalently: obtained by greedily selecting some fact maximal w.r.t. $\succ$ among those not yet considered, and keeping it if still consistent



Subset repairs

$$\mathit{SRep}(\mathcal{K}) = \{ \{\alpha\}, \{\gamma\}, \{\beta, \delta\} \}$$

Pareto- and globally-optimal

$$\mathit{PRep}(\mathcal{K}_\succ) = \mathit{GRep}(\mathcal{K}_\succ) = \{ \{\alpha\}, \{\gamma\}, \{\beta, \delta\} \}$$

Completion-optimal

$$\mathit{CRep}(\mathcal{K}_\succ) = \{ \{\alpha\}, \{\gamma\} \}$$

Optimal repairs

Import three notions of optimal repair from database setting

[Staworko, Chomicki, and Marcinkowski, 2012]

$$\mathit{CRep}(\mathcal{K}_{\succ}) \subseteq \mathit{GRep}(\mathcal{K}_{\succ}) \subseteq \mathit{PRep}(\mathcal{K}_{\succ}) \subseteq \mathit{SRep}(\mathcal{K})$$

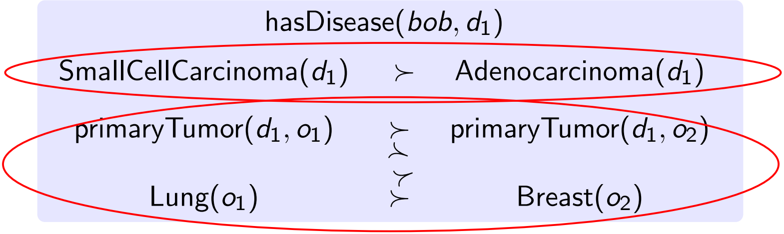
Optimal repairs

Import three notions of optimal repair from database setting

[Staworko, Chomicki, and Marcinkowski, 2012]

$$CRep(\mathcal{K}_\succ) \subseteq GRep(\mathcal{K}_\succ) \subseteq PRep(\mathcal{K}_\succ) \subseteq SRep(\mathcal{K})$$

If $\succ$ is score-structured (i.e., can be induced by assigning scores to facts), then $CRep(\mathcal{K}_\succ) = GRep(\mathcal{K}_\succ) = PRep(\mathcal{K}_\succ)$



$\{$hasDisease(bob, d_1), SmallCellCarcinoma(d_1), primaryTumor(d_1, o_1), Lung(o_1), primaryTumor(d_1, o_2)$\}$

$\{$hasDisease(bob, d_1), SmallCellCarcinoma(d_1), primaryTumor(d_1, o_1), Lung(o_1), Breast(o_2)$\}$

Inconsistency-tolerant semantics

Use optimal repairs instead of subset repairs

- X-AR: every X-optimal repair

$$\mathcal{K}_{\succ} \models^{X}_{\text{AR}} q \quad \Leftrightarrow \quad \forall \mathcal{R} \in XRep(\mathcal{K}_{\succ}),\ (\mathcal{R}, \mathcal{T}) \models q$$

- X-brave: some X-optimal repair

$$\mathcal{K}_{\succ} \models^{X}_{\text{brave}} q \quad \Leftrightarrow \quad \exists \mathcal{R} \in XRep(\mathcal{K}_{\succ}),\ (\mathcal{R}, \mathcal{T}) \models q$$

- X-IAR: intersection of all X-optimal repairs

$$\mathcal{K}_{\succ} \models^{X}_{\text{IAR}} q \quad \Leftrightarrow \quad (\mathcal{R}^{\cap}, \mathcal{T}) \models q,\ \mathcal{R}^{\cap} = \bigcap_{\mathcal{R} \in XRep(\mathcal{K}_{\succ})} \mathcal{R}$$

$$\mathcal{K}_{\succ} \models^{X}_{\text{IAR}} q \quad \Rightarrow \quad \mathcal{K}_{\succ} \models^{X}_{\text{AR}} q \quad \Rightarrow \quad \mathcal{K}_{\succ} \models^{X}_{\text{brave}} q$$

Complexity of reasoning with optimal repairs

Data complexity of query entailment

	Globally-optimal	Pareto-optimal	Completion-optimal
AR	Π_2^p-complete	coNP-complete	coNP-complete
IAR	Π_2^p-complete	coNP-complete	coNP-complete
Brave	Σ_2^p-complete	NP-complete	NP-complete

- Upper bounds hold for conjunctive queries and FOL fragments with PTime consistency checking/PTime query entailment
- Lower bounds hold for atomic queries and any fragment that extends functional dependencies, DL-Lite$_{\text{core}}$, or $\mathcal{EL}_\perp$

[Staworko, Chomicki, and Marcinkowski, 2012, Bienvenu and Bourgaux, 2020, 2022]

Practical SAT-based approaches

Pareto- and completion-optimal repairs: NP/coNP data complexity

⇒ Reduction to propositional satisfiability: use SAT encodings to decide whether a candidate answer holds under a given semantics

[Bienvenu and Bourgaux, 2022]

Practical SAT-based approaches

- Existing SAT-based systems
  - CQAPri: DL-Lite$_{\mathcal{R}}$ ontologies, X-AR/X-IAR/X-brave with subset and optimal repairs based on priority levels

    [Bienvenu, Bourgaux, and Goasdoué, 2014]
  - CAvSAT: databases + denial constraints, S-AR

    [Dixit and Kolaitis, 2019]

Practical SAT-based approaches

- Existing SAT-based systems
  - CQAPri: DL-Lite$_{\mathcal{R}}$ ontologies, X-AR/X-IAR/X-brave with subset and optimal repairs based on priority levels

    [Bienvenu, Bourgaux, and Goasdoué, 2014]
  - CAvSAT: databases + denial constraints, S-AR

    [Dixit and Kolaitis, 2019]
- CQAPri and CAvSAT employ SAT solvers in different ways
  - CQAPri makes a single SAT call for each candidate query answer
  - CAvSAT treats all candidate answers at the same time via calls to a weighted MaxSAT solver
  - + slight difference in the way of encoding the fact that a repair does not contain any cause for a query answer

Practical SAT-based approaches

- Existing SAT-based systems
  - CQAPri: DL-Lite$_{\mathcal{R}}$ ontologies, X-AR/X-IAR/X-brave with subset and optimal repairs based on priority levels
    [Bienvenu, Bourgaux, and Goasdoué, 2014]
  - CAvSAT: databases + denial constraints, S-AR
    [Dixit and Kolaitis, 2019]
- CQAPri and CAvSAT employ SAT solvers in different ways
  - CQAPri makes a single SAT call for each candidate query answer
  - CAvSAT treats all candidate answers at the same time via calls to a weighted MaxSAT solver
  - + slight difference in the way of encoding the fact that a repair does not contain any cause for a query answer

⇒ Compare SAT-based approaches (different algorithms, different encodings) for inconsistency-tolerant query answering

Practical SAT-based approaches

Implementation: ORBITS system

Case where conflicts contain at most two facts: conflicts and priority relation can be represented as a directed graph such that there is an edge from α to β if $\{\alpha, \beta\}$ is a conflict and $\alpha \not\succ \beta$

Input: directed conflict graph + potential answers and their causes (minimal sets of facts that support the answer)

Output: answers that hold under the required semantics

Practical SAT-based approaches

Implementation: ORBITS system

Case where conflicts contain at most two facts: conflicts and priority relation can be represented as a directed graph such that there is an edge from α to β if $\{\alpha, \beta\}$ is a conflict and $\alpha \not\succ \beta$

Input: directed conflict graph + potential answers and their causes (minimal sets of facts that support the answer)

Output: answers that hold under the required semantics

High-level algorithm:

- Filter answers that are trivially X-IAR in polynomial time: those which have causes without any fact with outgoing edge in the directed conflict graph
- Check remaining potential answers using a SAT solver
  - possibility to choose among several algorithms and encoding variants

High-level ideas underlying SAT encodings

- Try to build a subset of $\mathcal{D}$ that fulfills some conditions: assigning variable x_α to true means that fact α belongs to the subset
- Consider only relevant facts
- X-AR: build a set of facts that can be extended to an X-optimal repair that does not contain any cause for the query
- X-brave: build a set of facts that contains a cause of the query and can be extended to an X-optimal repair
- X-IAR: for each cause, find a set of facts that does not contain it and can be extended to an X-optimal repair

Modular encodings with basic building blocks:

- Absence of a cause
  - two variants: neg_1 and neg_2, following encodings used by CQAPri and CAvSAT respectively
- Presence of a cause
- Consistency of selected facts
- Extension to X-optimal repair
  - two variants for Pareto-optimal repairs: P_1 and P_2

- Four generic algorithms, applicable to X-AR, X-brave and X-IAR
  - one makes a single SAT call for each candidate answer
  - the others treat all candidate answers together (global encoding with soft clauses representing answers) with different reasoning modes
- Another algorithm for X-brave and X-IAR
  - check cause by cause
- Two algorithms for X-IAR keeping track of X-IAR / non X-IAR facts
  - cause by cause and fact by fact
  - all relevant facts together

Comparing semantics w.r.t computation time

- X-AR vs X-brave vs X-IAR: depends
- priority vs no priority (for AR): depends
- completion-optimal repairs: challenging (often timeout or oom)
- finer priority relation (compare more facts): lower running times

Comparing semantics w.r.t computation time

- X-AR vs X-brave vs X-IAR: depends
- priority vs no priority (for AR): depends
- completion-optimal repairs: challenging (often timeout or oom)
- finer priority relation (compare more facts): lower running times

Comparing algorithms/encoding variants for a given semantics

- Impact on running time can be huge
- No clear winner: depends on dataset, query, semantics...
- X-IAR: one algorithm ('fact by fact') is generally better
- Pareto-optimal repairs:
  - P_1- generally better than P_2-encoding (with noteworthy exceptions)
  - P_2-encoding works better with one way of encoding absence of a cause (neg_1) than the other (neg_2)
- Score-structured case: P-encodings are much better than C-encoding

Practical SAT-based approaches

Some experimental results

		q1	q2	q3	q4	q5	q6
Alg. 1	P_1	417	141	350	12,799	224	4,009
	P_2	804	142	379	326,594	213	8,684
	C	252,694	179	550	oom	284	t.o
Alg. 2	P_1	**268**	166	326	**1,730**	214	11,263
	P_2	502	163	333	2,961	221	10,833
	C	oom	632	t.o	t.o	551	oom
Alg. 3	P_1	272	154	313	t.o	211	245,804
	P_2	466	146	**281**	t.o	**201**	241,030
	C	oom	624	t.o	t.o	550	oom
Alg. 4	P_1	362	166	997	42,544	281	559,923
	P_2	566	193	972	36,923	304	546,199
	C	oom	764	t.o	t.o	846	oom
Alg. 5	P_1	383	**135**	335	8,192	211	**3,419**
	P_2	565	157	309	225,170	207	5,963
	C	192,429	164	544	oom	238	t.o

Query answer filtering time in ms under X-brave semantics on Physicians dataset (8M facts, 2% facts in conflicts) with score-structured priority (2 levels). Best time in bold red and ‘close to best’ (i.e., not exceeding best by more than 50ms or 10%) on grey.

Practical SAT-based approaches

Some experimental results

			q1	q2	q3	q4	q5	q6	q7	q8	q9	q10
∼CQAPri (single ans.)	P_1	neg_1	t.o	232,386	2,622	t.o	**846**	4,772	2,602	713	370,340	1,983
		neg_2	t.o	224,840	2,672	t.o	863	4,919	2,453	718	**365,374**	1,901
	P_2	neg_1	t.o	949,967	7,942	t.o	1,819	15,963	5,375	1,862	t.o	4,932
		neg_2	t.o	1,045,755	9,208	t.o	1,915	16,437	6,940	1,834	t.o	5,638
	C	neg_1/neg_2	t.o	t.o	oom	t.o	t.o	t.o	oom	oom	t.o	t.o
∼CAvSAT (multiple ans.)	P_1	neg_1	135,263	84,826	456	oom	1,155	**434**	2,060	369	t.o	**1,043**
		neg_2	208,831	**50,469**	**379**	oom	1,109	504	**1,909**	**364**	t.o	1,275
	P_2	neg_1	**119,041**	65,627	966	oom	2,025	942	3,484	828	t.o	2,180
		neg_2	oom	75,688	1,050	oom	2,229	1,000	3,700	816	t.o	2,397
	C	neg_1/neg_2	oom	oom	oom	oom	oom	oom	oom	oom	oom	oom

			q11	q12	q13	q14	q15	q16	q17	q18	q19	q20
∼CQAPri (single ans.)	P_1	neg_1	**10,004**	71,135	21,990	**5,822**	113,593	t.o	2,696	86,121	2,460	1,762
		neg_2	10,069	**62,227**	**21,038**	5,830	115,590	t.o	2,551	88,288	**2,378**	1,702
	P_2	neg_1	34,912	323,386	84,726	19,074	508,810	t.o	7,194	359,410	6,188	4,470
		neg_2	35,362	284,407	84,111	20,755	586,030	t.o	7,555	391,599	6,410	4,399
	C	neg_1/neg_2	t.o	oom	t.o	t.o	oom	t.o	oom	t.o	oom	oom
∼CAvSAT (multiple ans.)	P_1	neg_1	t.o	t.o	t.o	11,765	20,432	**180,360**	**1,942**	**23,016**	2,806	332
		neg_2	t.o	t.o	t.o	27,760	**18,523**	214,750	2,047	46,767	2,626	**287**
	P_2	neg_1	t.o	t.o	t.o	12,102	37,935	225,307	3,796	30,600	5,439	763
		neg_2	t.o	t.o	t.o	30,375	65,737	oom	4,007	123,690	6,334	744
	C	neg_1/neg_2	oom	oom	oom	oom	oom	oom	oom	oom	oom	oom

Query answer filtering time in ms under X-AR semantics on u20c50 (2M facts, 46% facts in conflicts) with score-structured priority (5 levels). Best time in bold red, 'close to best' (not exceeding best by more than 50ms or 10%) on grey.

Relationships with other frameworks

Pareto-optimal, globally-optimal or completion-optimal: how to choose?

- $CRep(\mathcal{K}_\succ) \subseteq GRep(\mathcal{K}_\succ) \subseteq PRep(\mathcal{K}_\succ) \subseteq SRep(\mathcal{K})$
  - completion: more X-IAR and X-AR answers, less X-brave answers
  - Pareto: less X-IAR and X-AR answers, more X-brave answers
- Complexity of reasoning: higher for globally-optimal
- Experimental comparison: completion-optimal challenging (but maybe we just need to find the right method...)

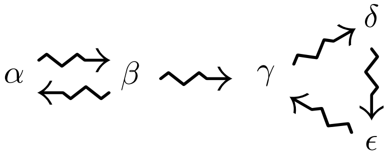
But which notion is the 'most natural'?

$\Rightarrow$ Study links with other formalisms

[Bienvenu and Bourgaux 2020, 2023]

Connections with argumentation

Abstract argumentation: well-known framework to deal with contradictory information in AI



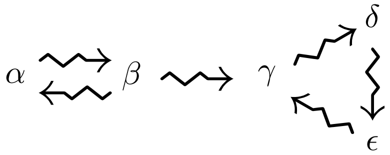
An (abstract) argumentation framework (AF) is a pair $(Args, \rightsquigarrow)$ where

- *Args* is a finite set of arguments
- $\rightsquigarrow \subseteq Args \times Args$ is the attack relation: α attacks β if $\alpha \rightsquigarrow \beta$

Semantics based on extensions (sets of arguments that represent coherent points of view) + inference mechanism (skeptical or credulous)

Several different notions of extension, in particular:

- Preferred extension: $\subseteq$-maximal conflict-free self-defending set (i.e., attacks all arguments that attack some of its arguments)
- Stable extension: conflict-free set attacking all excluded arguments



Preferred: $\{\alpha\}$, $\{\beta, \delta\}$

Stable: $\{\beta, \delta\}$

Stable extensions are also preferred extensions

Coherent AF: stable and preferred extensions coincide

Connections with argumentation

Many variants of AF have been studied, in particular:

- Preference-based AF (PAF)
  - preference relation $\succ$ between arguments
  - refines the attack relation: $\beta \rightsquigarrow_{\succ} \alpha$ if $\beta \rightsquigarrow \alpha$ and $\alpha \not\succ \beta$
- Set-based AF (SETAF)
  - collective attacks $S \rightsquigarrow \alpha$ with S finite set of arguments

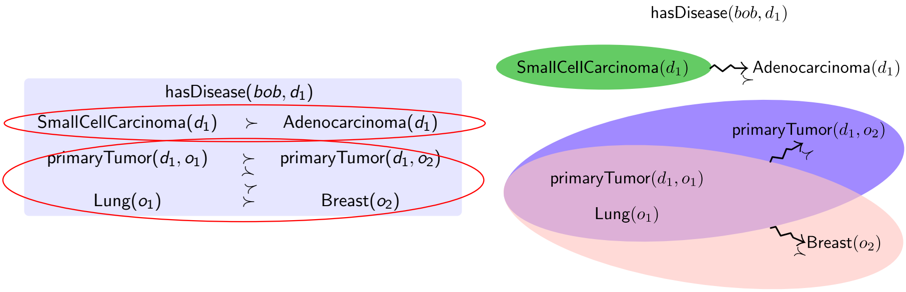
Combined into PSETAF $(Args, \rightsquigarrow, \succ)$

- $S \rightsquigarrow_{\succ} \alpha$ if $S \rightsquigarrow \alpha$ and $\alpha \not\succ \beta$ for every $\beta \in S$

Translation of a prioritized KB $\mathcal{K}_\succ = (\mathcal{D}, \mathcal{T}, \succ)$ into a PSETAF $F_{\mathcal{K},\succ}$

- Use $\mathcal{D}$ as the arguments
- Use $\succ$ as the preference relation
- Define attacks by $\mathcal{C} \setminus \{\alpha\} \rightsquigarrow \alpha$ for every conflict $\mathcal{C}$ and $\alpha \in \mathcal{C}$

$\Rightarrow \quad \mathcal{C} \setminus \{\alpha\} \rightsquigarrow_\succ \alpha$ if $\alpha \not\succ \beta$ for every $\beta \in \mathcal{C}$



$\mathcal{R}$ is a Pareto-optimal repair of $\mathcal{K}_{\succ}$
iff
$\mathcal{R}$ is a stable extension of $F_{\mathcal{K},\succ}$

If $\succ$ is transitive or if $\mathcal{K}$ has only binary conflicts, then $F_{\mathcal{K},\succ}$ is coherent:
$\mathcal{R}$ is a Pareto-optimal repair of $\mathcal{K}_{\succ}$
iff
$\mathcal{R}$ is a preferred extension of $F_{\mathcal{K},\succ}$

No notion of extension corresponds to globally- or completion-optimal

The grounded extension of a (PSET)AF is the minimal conflict-free set of arguments that contains all arguments that it defends

- Add all arguments with no incoming attacks
- Iteratively add arguments defended by the selected arguments

$\Rightarrow$ Grounded semantics for prioritized KB: query grounded extension of $F_{\mathcal{K},\succ}$

- PTime-complete data complexity for DL-Lite KBs
- Under-approximation of P-IAR

Connections with active integrity constraints

In the database setting, active integrity constraints state how to resolve constraint violations: high-level similarities with prioritized databases

Example of denial constraint and active denial constraint:

$$\mathrm{Child}(x) \wedge \mathrm{Adult}(x) \rightarrow \bot$$

$$\mathrm{Child}(x) \wedge \mathrm{Adult}(x) \rightarrow \{-\mathrm{Child}(x)\}$$

Example of universal constraint and active universal constraint:

$$\mathrm{Lung}(x) \wedge \neg\mathrm{LeftLg}(x) \wedge \neg\mathrm{RightLg}(x) \rightarrow \bot$$

$$\mathrm{Lung}(x) \wedge \neg\mathrm{LeftLg}(x) \wedge \neg\mathrm{RightLg}(x) \rightarrow \{+\mathrm{LeftLg}(x), +\mathrm{RightLg}(x)\}$$

A ground active integrity constraint (AIC) is a formula of the form

$$\alpha_1 \wedge \cdots \wedge \alpha_n \wedge \neg\beta_1 \wedge \cdots \wedge \neg\beta_m \rightarrow \{A_1, \ldots, A_k\}$$

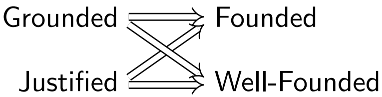
with update actions A_i of the form $-\alpha_j$ or $+\beta_j$

Connections with active integrity constraints

Semantics based on repair updates: $\mathcal{U}$ set of update actions such that $\mathcal{D} \circ \mathcal{U}$ is a repair, where $\mathcal{D} \circ \mathcal{U} = \mathcal{D} \setminus \{\alpha \mid -\alpha \in \mathcal{U}\} \cup \{\alpha \mid +\alpha \in \mathcal{U}\}$

Several different notions of repair update, in particular:

- Founded: for every $A \in \mathcal{U}$, there is an AIC r with update action A and $\mathcal{D} \circ \mathcal{U} \setminus \{A\} \not\models r$
- Well-founded: there exists a sequence of actions $A_1, \ldots, A_n$ such that $\mathcal{U} = \{A_1, \ldots, A_n\}$, and for every $1 \leq i \leq n$, there is r_i with update action A_i and $\mathcal{D} \circ \{A_1, \ldots, A_{i-1}\} \not\models r_i$
- Grounded (for normalized AICs: single update action): for every $\mathcal{V} \subsetneq \mathcal{U}$, there is r whose update action is in $\mathcal{U} \setminus \mathcal{V}$ and $\mathcal{D} \circ \mathcal{V} \not\models r$
- Justified...



Focus on prioritized databases + extend setting to universal constraints

Kind of constraints also relevant for DL with closed predicates

- $\mathcal{T}$ is a set of constraints of the form
  $$\forall \vec{x}(\alpha_1 \land \cdots \land \alpha_n \land \neg\beta_1 \land \cdots \land \neg\beta_m \to \bot)$$
- May add facts to repair the database
  - symmetric difference repairs: $\mathcal{R}$ such that $\mathcal{R} \models \mathcal{T}$ and there is no $\mathcal{R}' \models \mathcal{T}$ such that $\mathcal{R}'\Delta\mathcal{D} \subsetneq \mathcal{R}\Delta\mathcal{D}$
  - conflicts may contain absent facts of the form $\neg\alpha$: minimal sets of literals such that $\mathcal{I} \models \mathcal{C}$ implies $\mathcal{I} \not\models \mathcal{T}$
- Priority relation $\succ$ over literals in conflicts
- Pareto-, globally- and completion-optimal repair definitions extended by viewing databases as sets of literals

Connections with active integrity constraints

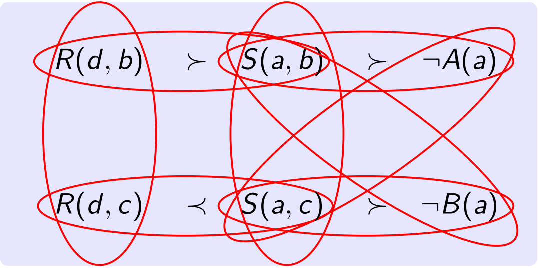
Prioritized databases with universal constraints

$$S(x,y) \wedge S(x,z) \wedge y \neq z \rightarrow \bot \qquad S(x,y) \wedge \neg A(x) \rightarrow \bot$$

$$R(x,y) \wedge R(x,z) \wedge y \neq z \rightarrow \bot \qquad S(x,y) \wedge \neg B(x) \rightarrow \bot$$

$$R(y,x) \wedge S(z,x) \rightarrow \bot$$

$$\mathcal{D} = \{S(a,b), S(a,c), R(d,b), R(d,c)\}$$



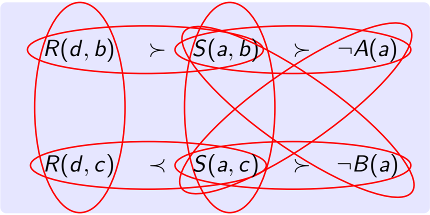
$$CRep(\mathcal{K}_\succ) = \{\ \{S(a,c), R(d,b), A(a), B(a)\}\ \}$$

$$GRep(\mathcal{K}_\succ) = CRep(\mathcal{K}_\succ) \cup \{\ \{R(d,b)\}\ \}$$

$$PRep(\mathcal{K}_\succ) = GRep(\mathcal{K}_\succ) \cup \{\ \{R(d,c)\},\ \{R(d,c), S(a,b), A(a), B(a)\}\ \}$$

Translation of a prioritized database $\mathcal{K}_\succ = (\mathcal{D}, \mathcal{T}, \succ)$ into ground AICs

- $\eta_\succ^{\mathcal{T}} = \{r_{\mathcal{C}} \mid \mathcal{C} \in Conf(\mathcal{D}, \mathcal{T})\}$ where $fix(\alpha) = -\alpha$, $fix(\neg\alpha) = +\alpha$ and
  $$r_{\mathcal{C}} := \bigwedge_{\lambda \in \mathcal{C}} \lambda \rightarrow \{fix(\lambda) \mid \lambda \in \mathcal{C}, \forall \mu \in \mathcal{C}, \lambda \not\succ \mu\}$$
- Conflicts fixed by modifying least preferred literals according to $\succ$



$$S(a, b) \wedge S(a, c) \rightarrow \{-S(a, b), -S(a, c)\}$$
$$R(d, b) \wedge R(d, c) \rightarrow \{-R(d, b), -R(d, c)\}$$
$$R(d, b) \wedge S(a, b) \rightarrow \{-S(a, b)\}$$
$$R(d, c) \wedge S(a, c) \rightarrow \{-R(d, c)\}$$
$$S(a, b) \wedge \neg A(a) \rightarrow \{+A(a)\}$$
$$S(a, c) \wedge \neg A(a) \rightarrow \{-S(a, c), +A(a)\}$$
$$S(a, b) \wedge \neg B(a) \rightarrow \{-S(a, b), +B(a)\}$$
$$S(a, c) \wedge \neg B(a) \rightarrow \{+B(a)\}$$

For denial constraints: data-independent reduction to non-ground AICs (assuming the priority relation is stored in the database)

Pareto $\equiv$ Founded $\equiv$ Grounded $\equiv$ Justified $\Rightarrow$ Well-Founded

$\mathcal{R} = \mathcal{D} \circ \mathcal{U}$ is a Pareto-optimal repair of $\mathcal{K}_{\succ}$

iff

$\mathcal{U}$ is a founded repair update of $\mathcal{D}$ w.r.t. $\eta_{\succ}^{\mathcal{T}}$

iff

$\mathcal{U}$ is a grounded repair update of $\mathcal{D}$ w.r.t. $\eta_{\succ}^{\mathcal{T}}$

iff

$\mathcal{U}$ is a justified repair update of $\mathcal{D}$ w.r.t. $\eta_{\succ}^{\mathcal{T}}$

In the other direction: from AICs to prioritized database

- Translation for a restricted class of ‘well-behaved’ AICs
  - such that founded, grounded and justified repair updates coincide
- Binary conflicts: Founded $\equiv$ Grounded $\equiv$ Justified $\equiv$ Pareto
- Non-binary conflicts: Founded $\equiv$ Grounded $\equiv$ Justified $\Rightarrow$ Pareto

Comparison with other approaches for prioritized KBs

Many proposals to handle inconsistent KBs with some sort of preference between facts

$\Rightarrow$ How do they compare with prioritized KBs and optimal repair-based semantics ?

Comparison with other approaches for prioritized KBs

Two main settings that both reduce to prioritized KBs

- $\mathcal{D}$ partitioned into priority levels $\mathcal{S}_1, \ldots, \mathcal{S}_n$
  - defines a score-structured priority relation $\succ$:
    $\alpha \succ \beta$ iff $\{\alpha, \beta\} \in \mathcal{C}$ for some conflict $\mathcal{C}$, $\alpha \in \mathcal{S}_i$ and $\beta \in \mathcal{S}_j$ with $i < j$
- Preordered KBs: $\trianglerighteq$ reflexive and transitive binary relation over $\mathcal{D}$
  - defines a transitive priority relation $\succ$:
    $\alpha \succ \beta$ iff $\{\alpha, \beta\} \in \mathcal{C}$ for some conflict $\mathcal{C}$, $\alpha \trianglerighteq \beta$ and $\beta \not\trianglerighteq \alpha$

Comparison with other approaches for prioritized KBs

Two main settings that both reduce to prioritized KBs

- $\mathcal{D}$ partitioned into priority levels $\mathcal{S}_1, \ldots, \mathcal{S}_n$
  - defines a score-structured priority relation $\succ$:
    $\alpha \succ \beta$ iff $\{\alpha, \beta\} \in \mathcal{C}$ for some conflict $\mathcal{C}$, $\alpha \in \mathcal{S}_i$ and $\beta \in \mathcal{S}_j$ with $i < j$
- Preordered KBs: $\trianglerighteq$ reflexive and transitive binary relation over $\mathcal{D}$
  - defines a transitive priority relation $\succ$:
    $\alpha \succ \beta$ iff $\{\alpha, \beta\} \in \mathcal{C}$ for some conflict $\mathcal{C}$, $\alpha \trianglerighteq \beta$ and $\beta \not\trianglerighteq \alpha$

Two main approaches

- Preferred repairs based on priority levels: $\mathcal{R}$ is a $\subseteq_P$-repair if $\mathcal{R} \models \mathcal{T}$ and there is no $\mathcal{R}' \models \mathcal{T}$ such that there is $1 \leq i \leq n$ such that $\mathcal{R} \cap \mathcal{S}_i \subsetneq \mathcal{R}' \cap \mathcal{S}_i$ and $\mathcal{R} \cap \mathcal{S}_j = \mathcal{R}' \cap \mathcal{S}_j$ for $1 \leq j < i$
  [Bienvenu, Bourgaux, and Goasdoué, 2014]
  - coincide with optimal repairs ($CRep(\mathcal{K}_\succ) = GRep(\mathcal{K}_\succ) = PRep(\mathcal{K}_\succ)$ for score-structured priority)
- Select a single consistent set of facts to query
  [Benferhat, Bouraoui, and Tabia, 2015, Belabbes, Benferhat, and Chomicki, 2021]

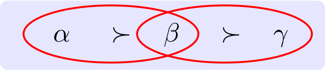
Comparison with other approaches for prioritized KBs

Selection of a single consistent set of facts to query

Preordered KBs

[Belabbes, Benferhat, and Chomicki, 2021]

- Elected facts Elect($\mathcal{K}_{\succ}$): α is elected iff for every conflict $\mathcal{C}$, $\alpha \in \mathcal{C}$ implies $\alpha \succ \beta$ for some $\beta \in \mathcal{C}$
  - Elect($\mathcal{K}_{\succ}$) $\subseteq$ grounded($\mathcal{K}_{\succ}$)
  - inclusion can be strict



Elect($\mathcal{K}_{\succ}$) $= \{\alpha\}$
grounded($\mathcal{K}_{\succ}$) $= \{\alpha, \gamma\}$

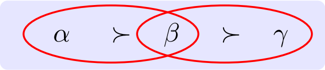
Comparison with other approaches for prioritized KBs

Selection of a single consistent set of facts to query

Preordered KBs

[Belabbes, Benferhat, and Chomicki, 2021]

- Elected facts $\mathsf{Elect}(\mathcal{K}_{\succ})$: α is elected iff for every conflict $\mathcal{C}$, $\alpha \in \mathcal{C}$ implies $\alpha \succ \beta$ for some $\beta \in \mathcal{C}$
  - $\mathsf{Elect}(\mathcal{K}_{\succ}) \subseteq \mathsf{grounded}(\mathcal{K}_{\succ})$
  - inclusion can be strict



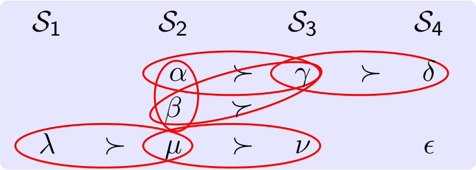
$$\mathsf{Elect}(\mathcal{K}_{\succ}) = \{\alpha\}$$
$$\mathsf{grounded}(\mathcal{K}_{\succ}) = \{\alpha, \gamma\}$$

- Preferred repair $\mathsf{Partial}_{\mathsf{PR}}(\mathcal{K}_{\succ})$: union of $\bigcap_{\mathcal{R} \in XRep(\mathcal{K}_{\succ_{\geq}})} \mathcal{R}$ for all total preorders $\geq$ extending $\trianglerighteq$ (with $\succ_{\geq}$ priority relation that corresponds to $\geq$)
  - $\mathsf{Partial}_{\mathsf{PR}}(\mathcal{K}_{\succ}) = \bigcap_{\mathcal{R} \in CRep(\mathcal{K}_{\succ})} \mathcal{R}$

$$\mathsf{Elect}(\mathcal{K}_{\succ}) \subseteq \mathsf{grounded}(\mathcal{K}_{\succ}) \subseteq \bigcap_{\mathcal{R} \in PRep(\mathcal{K}_{\succ})} \mathcal{R} \subseteq \bigcap_{\mathcal{R} \in GRep(\mathcal{K}_{\succ})} \mathcal{R} \subseteq \bigcap_{\mathcal{R} \in CRep(\mathcal{K}_{\succ})} \mathcal{R} = \mathsf{Partial}_{\mathsf{PR}}(\mathcal{K}_{\succ})$$

Priority levels (score-structured) [Benferhat, Bouraoui, and Tabia, 2015]

- Possibilistic repair $\pi(\mathcal{K}_\succ)$: $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_{inc(\mathcal{K}_\succ)-1}$ where $inc(\mathcal{K}_\succ)$ is the inconsistency degree of $\mathcal{K}_\succ$



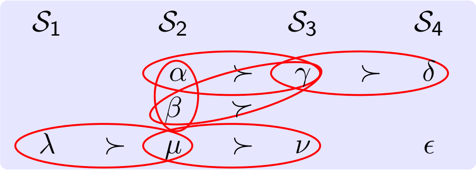
$\pi(\mathcal{K}_\succ) = \{\lambda\}$

Comparison with other approaches for prioritized KBs

Selection of a single consistent set of facts to query

Priority levels (score-structured) [Benferhat, Bouraoui, and Tabia, 2015]

- Possibilistic repair $\pi(\mathcal{K}_\succ)$: $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_{inc(\mathcal{K}_\succ)-1}$ where $inc(\mathcal{K}_\succ)$ is the inconsistency degree of $\mathcal{K}_\succ$
- Non-defeated repair nd$(\mathcal{K}_\succ)$: union of the intersections of the (subset) repairs of $\mathcal{S}_1$, $\mathcal{S}_1 \cup \mathcal{S}_2$,..., $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_n$
  - has been shown to coincide with Elect$(\mathcal{K}_\succ)$



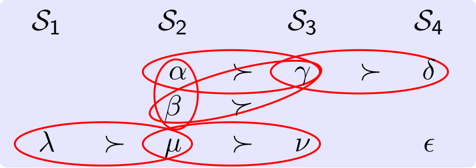
$$\pi(\mathcal{K}_\succ) = \{\lambda\}$$
$$\text{nd}(\mathcal{K}_\succ) = \{\lambda, \epsilon\}$$
$$\text{grounded}(\mathcal{K}_\succ) = \{\lambda, \nu, \epsilon\}$$

Comparison with other approaches for prioritized KBs

Selection of a single consistent set of facts to query

Priority levels (score-structured) [Benferhat, Bouraoui, and Tabia, 2015]

- Possibilistic repair $\pi(\mathcal{K}_\succ)$: $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_{inc(\mathcal{K}_\succ)-1}$ where $inc(\mathcal{K}_\succ)$ is the inconsistency degree of $\mathcal{K}_\succ$
- Non-defeated repair $\text{nd}(\mathcal{K}_\succ)$: union of the intersections of the (subset) repairs of $\mathcal{S}_1$, $\mathcal{S}_1 \cup \mathcal{S}_2$,..., $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_n$
  - has been shown to coincide with $\text{Elect}(\mathcal{K}_\succ)$
- Prioritized inclusion-based non-defeated repair $\text{pind}(\mathcal{K}_\succ)$: as non-defeated but using optimal repairs (intractable!)
  - has been shown to coincide with $\bigcap_{\mathcal{R} \in XRep(\mathcal{K}_\succ)} \mathcal{R}$



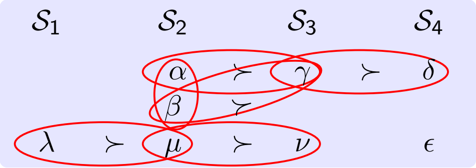
$$\pi(\mathcal{K}_\succ) = \{\lambda\}$$
$$\text{nd}(\mathcal{K}_\succ) = \{\lambda, \epsilon\}$$
$$\text{grounded}(\mathcal{K}_\succ) = \{\lambda, \nu, \epsilon\}$$
$$\text{pind}(\mathcal{K}_\succ) = \{\lambda, \nu, \delta, \epsilon\}$$

Comparison with other approaches for prioritized KBs

Selection of a single consistent set of facts to query

Priority levels (score-structured) [Benferhat, Bouraoui, and Tabia, 2015]

- Possibilistic repair $\pi(\mathcal{K}_\succ)$: $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_{inc(\mathcal{K}_\succ)-1}$ where $inc(\mathcal{K}_\succ)$ is the inconsistency degree of $\mathcal{K}_\succ$
- Non-defeated repair nd($\mathcal{K}_\succ$): union of the intersections of the (subset) repairs of $\mathcal{S}_1$, $\mathcal{S}_1 \cup \mathcal{S}_2$,..., $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_n$
  - has been shown to coincide with Elect($\mathcal{K}_\succ$)
- Prioritized inclusion-based non-defeated repair pind($\mathcal{K}_\succ$): as non-defeated but using optimal repairs (intractable!)
  - has been shown to coincide with $\bigcap_{\mathcal{R} \in XRep(\mathcal{K}_\succ)} \mathcal{R}$



$\pi(\mathcal{K}_\succ) = \{\lambda\}$

nd$(\mathcal{K}_\succ) = \{\lambda, \epsilon\}$

grounded$(\mathcal{K}_\succ) = \{\lambda, \nu, \epsilon\}$

pind$(\mathcal{K}_\succ) = \{\lambda, \nu, \delta, \epsilon\}$

$$\pi(\mathcal{K}_\succ) \subseteq \text{nd}(\mathcal{K}_\succ) \subseteq \text{grounded}(\mathcal{K}_\succ) \subseteq \text{pind}(\mathcal{K}_\succ) = \bigcap_{\mathcal{R} \in XRep(\mathcal{K}_\succ)} \mathcal{R}$$

Conclusion

- Take into account preference between facts to refine repairs
  - three kinds of optimal repair, coincide for score-structured priority
- SAT-based approaches promising for Pareto-optimal repairs
  - question of the choice of the algorithm and SAT encoding
- Translations to argumentation framework or active integrity constraints
  - get Pareto-optimal repairs analogous
  - grounded semantics inspired by argumentation

Some of the next steps

- Help users to define priorities
- Implement algorithms for non-binary conflicts
- Universal constraints for DL KBs

Thank you for your attention !
Questions ?

References

Meghyn Bienvenu and Camille Bourgaux.
Querying and repairing inconsistent prioritized knowledge bases: Complexity analysis and links with abstract argumentation.
In *Proceedings of KR*, 2020.

Meghyn Bienvenu and Camille Bourgaux.
Querying inconsistent prioritized data with ORBITS: algorithms, implementation, and experiments.
In *Proceedings of KR*, 2022.

Meghyn Bienvenu and Camille Bourgaux.
Inconsistency handling in prioritized databases with universal constraints: Complexity analysis and links with active integrity constraints.
In *Proceedings of KR*, 2023.

Slawek Staworko, Jan Chomicki, and Jerzy Marcinkowski.
Prioritized repairing and consistent query answering in relational databases.
AMAI, 64(2-3):209–246, 2012.

Meghyn Bienvenu, Camille Bourgaux, and François Goasdoué.
Querying inconsistent description logic knowledge bases under preferred repair semantics.
In *Proceedings of AAAI*, 2014.

Akhil A. Dixit and Phokion G. Kolaitis.
A SAT-based system for consistent query answering.
In *Proceedings of SAT*, 2019.

Sihem Belabbes, Salem Benferhat, and Jan Chomicki.
Handling inconsistency in partially preordered ontologies: the Elect method.
J. Log. Comput., 31(5):1356–1388, 2021.

Salem Benferhat, Zied Bouraoui, and Karim Tabia.
How to select one preferred assertional-based repair from inconsistent and prioritized DL-Lite knowledge bases?
In *Proceedings of IJCAI*, 2015.