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**: $q(\bar{x}) = \exists \bar{y} \varphi$ with $\varphi$ conjunction of atoms
- $\mathcal{K} \models q(\bar{a})$ if $q(\bar{a})$ holds in every model of $\mathcal{K}$

Query

Ontology

Data

\[ q(x) = \exists y \text{ hasDisease}(x, y) \land \text{LungCancer}(y) \]

\[ \text{hasDisease}(bob, d_1) \]

\[ \text{SmallCellCarcinoma}(d_1) \]

\[ \text{primaryTumor}(d_1, o_1) \]

\[ \text{Lung}(o_1) \]
Handling inconsistent data

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

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

\[
\text{hasDisease}(bob, d_1)
\]
\textbf{SmallCellCarcinoma($d_1$)} \quad \textbf{Adenocarcinoma($d_1$)}

\textbf{primary Tumor($d_1, o_1$)} \quad \textbf{primary Tumor($d_1, o_2$)}
\textbf{Lung($o_1$)} \quad \textbf{Breast($o_2$)}
Handling inconsistent data

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

\[
\begin{align*}
\text{Cancer} \sqcap \exists \text{primaryTumor}.\text{Lung} \sqsubseteq \text{LungCancer} \\
\text{SmallCellCarcinoma} \sqsubseteq \text{Cancer} \\
\text{Adenocarcinoma} \sqsubseteq \text{Cancer} \\
\text{Adenocarcinoma} \sqcap \text{SmallCellCarcinoma} \sqsubseteq \bot \\
(\text{functional primaryTumor}) \\
\text{Lung} \sqcap \text{Breast} \sqsubseteq \bot
\end{align*}
\]

\[
\begin{align*}
\text{hasDisease}(bob, d_1) \\
\text{SmallCellCarcinoma}(d_1) & \quad \text{Adenocarcinoma}(d_1) \\
\text{primaryTumor}(d_1, o_1) & \quad \text{primaryTumor}(d_1, o_2) \\
\text{Lung}(o_1) & \quad \text{Breast}(o_2)
\end{align*}
\]
Handling inconsistent data

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

\[
\begin{align*}
\text{Cancer} \sqcap \exists \text{primaryTumor.Lung} & \subseteq \text{LungCancer} \\
\text{SmallCellCarcinoma} & \subseteq \text{Cancer} \\
\text{Adenocarcinoma} & \subseteq \text{Cancer} \\
\text{Adenocarcinoma} \sqcap \text{SmallCellCarcinoma} & \subseteq \bot \\
\text{(functional primaryTumor)} \\
\text{Lung} \sqcap \text{Breast} & \subseteq \bot
\end{align*}
\]

\[
\begin{align*}
\text{hasDisease}(bob, d_1) \\
\text{SmallCellCarcinoma}(d_1) & \quad \text{Adenocarcinoma}(d_1) \\
\text{primaryTumor}(d_1, o_1) & \quad \text{primaryTumor}(d_1, o_2) \\
\text{Lung}(o_1) & \quad \text{Breast}(o_2)
\end{align*}
\]

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

$\Rightarrow$ Use inconsistency-tolerant semantics
Handling inconsistent data

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

(functional primaryTumor)
\text{Lung} \sqcap \text{Breast} \sqsubseteq \bot

\begin{align*}
\text{hasDisease}(bob, d_1) \\
\text{SmallCellCarcinoma}(d_1) & \quad \text{Adenocarcinoma}(d_1) \\
\text{primaryTumor}(d_1, o_1) & \quad \text{primaryTumor}(d_1, o_2) \\
\text{Lung}(o_1) & \quad \text{Breast}(o_2)
\end{align*}

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

Cancer ⊓ ∃primaryTumor.Lung ⊆ LungCancer
SmallCellCarcinoma ⊆ Cancer
Adenocarcinoma ⊆ Cancer
Adenocarcinoma ⊓ SmallCellCarcinoma ⊆ ⊥
(functional primaryTumor)
Lung ⊓ Breast ⊆ ⊥

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

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

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

Cancer ⊓ ∃primaryTumor.Lung ⊆ LungCancer
SmallCellCarcinoma ⊆ Cancer
Adenocarcinoma ⊆ Cancer
Adenocarcinoma ⊓ SmallCellCarcinoma ⊆ ⊥
(functional primaryTumor)
Lung ⊓ Breast ⊆ ⊥

hasDisease(bob, d₁)

SmallCellCarcinoma(d₁)  Adenocarcinoma(d₁)
primaryTumor(d₁, o₁)  primaryTumor(d₁, o₂)
Lung(o₁)  Breast(o₂)

(Subset) repair: inclusion-maximal \( \mathcal{R} \subseteq \mathcal{D} \) such that \((\mathcal{R}, \mathcal{T}) \not\models ⊥\)
Handling inconsistent data

Cancer \sqcap \exists \text{primaryTumor} \cdot \text{Lung} \sqsubseteq \text{LungCancer}

\text{SmallCellCarcinoma} \sqsubseteq \text{Cancer}

\text{Adenocarcinoma} \sqsubseteq \text{Cancer}

\text{Adenocarcinoma} \sqcap \text{SmallCellCarcinoma} \sqsubseteq \bot

(has functional primaryTumor)

\text{Lung} \sqcap \text{Breast} \sqsubseteq \bot

\begin{align*}
\text{hasDisease}(bob, d_1) \\
\text{SmallCellCarcinoma}(d_1) & \quad \text{Adenocarcinoma}(d_1) \\
\text{primaryTumor}(d_1, o_1) & \quad \text{primaryTumor}(d_1, o_2) \\
\text{Lung}(o_1) & \quad \text{Breast}(o_2)
\end{align*}

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

Cancer ⊓ ∃primaryTumor.Lung ⊑ LungCancer
SmallCellCarcinoma ⊑ Cancer
Adenocarcinoma ⊑ Cancer
Adenocarcinoma ⊓ SmallCellCarcinoma ⊑ ⊥
(functional primaryTumor)
Lung ⊓ Breast ⊑ ⊥

(Subset) repair: inclusion-maximal \( R \subseteq D \) such that \((R, T) \nvDash \bot\)
Handling inconsistent data

Cancer ⊓ ∃primaryTumor.Lung ⊑ LungCancer
SmallCellCarcinoma ⊑ Cancer
Adenocarcinoma ⊑ Cancer
Adenocarcinoma ⊓ SmallCellCarcinoma ⊑ ⊥
(functional primaryTumor)
Lung ⊓ Breast ⊑ ⊥

(hasDisease(bob, d₁)

SmallCellCarcinoma(d₁)  Adenocarcinoma(d₁)
primaryTumor(d₁, o₁)  primaryTumor(d₁, o₂)
Lung(o₁)  Breast(o₂)

(Subset) repair: inclusion-maximal \( \mathcal{R} \subseteq \mathcal{D} \) such that \( (\mathcal{R}, \mathcal{T}) \not\models ⊥ \)
Handling inconsistent data

Cancer $\sqcap \exists$ primaryTumor.Lung $\subseteq$ LungCancer
SmallCellCarcinoma $\subseteq$ Cancer
Adenocarcinoma $\subseteq$ Cancer
Adenocarcinoma $\sqcap$ SmallCellCarcinoma $\subseteq \bot$
(functional primaryTumor)
Lung $\sqcap$ Breast $\subseteq \bot$

(Subset) repair: inclusion-maximal $\mathcal{R} \subseteq \mathcal{D}$ such that $(\mathcal{R}, \mathcal{T}) \not|= \bot
Handling inconsistent data

Cancer ⊓ ∃primaryTumor.Lung ⊑ LungCancer
SmallCellCarcinoma ⊑ Cancer
Adenocarcinoma ⊑ Cancer
Adenocarcinoma ⊓ SmallCellCarcinoma ⊑ ⊥
(functional primaryTumor)
Lung ⊓ Breast ⊑ ⊥

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

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

Cancer ⊓ ∃primaryTumor.Lung ⊆ LungCancer
SmallCellCarcinoma ⊆ Cancer
Adenocarcinoma ⊆ Cancer
Adenocarcinoma ⊓ SmallCellCarcinoma ⊆ ⊥
(functional primaryTumor)
Lung ⊓ Breast ⊆ ⊥

<table>
<thead>
<tr>
<th>hasDisease(bob, d₁)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SmallCellCarcinoma(d₁)</td>
</tr>
<tr>
<td>primaryTumor(d₁, o₁)</td>
</tr>
<tr>
<td>Lung(o₁)</td>
</tr>
</tbody>
</table>

- (Subset) repair: inclusion-maximal \( \mathcal{R} \subseteq D \) such that \( (\mathcal{R}, \mathcal{T}) \not\models \bot \)
- AR semantics: queries that hold in every repair
  \[
  \exists y \hspace{1em} \text{hasDisease}(bob, y) \land \text{Cancer}(y) \hspace{1em} \text{plausible/likely}
  \]
Handling inconsistent data

\[ \text{Cancer} \sqcap \exists \text{primaryTumor}.\text{Lung} \sqsubseteq \text{LungCancer} \]
\[ \text{SmallCellCarcinoma} \sqsubseteq \text{Cancer} \]
\[ \text{Adenocarcinoma} \sqsubseteq \text{Cancer} \]
\[ \text{Adenocarcinoma} \sqcap \text{SmallCellCarcinoma} \sqsubseteq \bot \]

(functional primaryTumor)
\[ \text{Lung} \sqcap \text{Breast} \sqsubseteq \bot \]

\[ \exists y \text{ hasDisease}(bob, y) \land \text{LungCancer}(y) \quad \text{possible} \]

- (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
Handling inconsistent data

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) \quad \text{Adenocarcinoma}(d_1)

\text{primaryTumor}(d_1, o_1) \quad \text{primaryTumor}(d_1, o_2)
\text{Lung}(o_1) \quad \text{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
  \[
  \exists y \text{ hasDisease}(bob, y) \quad \text{surest}
  \]
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

<table>
<thead>
<tr>
<th>Fact</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>primaryTumor($d_1$, $o_1$)</td>
<td>08.10.2023</td>
</tr>
<tr>
<td>primaryTumor($d_1$, $o_2$)</td>
<td>05.22.2023</td>
</tr>
</tbody>
</table>

most recent fact gives the last, revised, diagnosis
⇒ primaryTumor($d_1$, $o_1$) ≻ 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...)

<table>
<thead>
<tr>
<th>Fact</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adenocarcinoma($d_1$)</td>
<td>X-ray report</td>
</tr>
<tr>
<td>SmallCellCarcinoma($d_1$)</td>
<td>biopsy report</td>
</tr>
</tbody>
</table>

the second diagnostic method is more reliable
⇒ SmallCellCarcinoma($d_1$) ≻ 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

\[
\text{hasDisease}(\text{bob}, d_1), \\
\text{primaryTumor}(d_1, o_1), \text{Lung}(o_1), \\
\text{primaryTumor}(d_1, o_2), \text{Breast}(o_2), \\
\text{gotSurgery}(\text{bob}, s), \text{BronchialDebridement}(s)
\]

the dataset indicates that the patient got a surgery common in the case of lung cancer but nothing about a breast cancer treatment

\[\Rightarrow \text{primaryTumor}(d_1, o_1), \text{Lung}(o_1) \succ \text{primaryTumor}(d_1, o_2), \text{Breast}(o_2)\]
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
- ...

\[
\begin{align*}
\text{hasDisease}(bob, d_1) \\
\text{SmallCellCarcinoma}(d_1) \succ \text{Adenocarcinoma}(d_1) \\
\text{primaryTumor}(d_1, o_1) \succ \text{primaryTumor}(d_1, o_2) \\
\text{Lung}(o_1) \succ \text{Breast}(o_2)
\end{align*}
\]
Adding priorities

Formally:

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

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

Lung(o_1) \succ Breast(o_2)
```
Adding priorities

Formally:

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

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

- $\succ$ is **total** if for all $\alpha \neq \beta$ such that $\{\alpha, \beta\} \subseteq C$ for some conflict $C$, 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 $R$ be a repair of $\mathcal{K}$ ($R \in SRep(\mathcal{K})$)

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

\[\{\alpha_1, \alpha_2, \delta, \epsilon\} \in SRep(\mathcal{K})\]

\[\{\beta_1, \delta, \epsilon\} \text{ Pareto improvement} \]

\[\Rightarrow \quad \{\alpha_1, \alpha_2, \delta, \epsilon\} \notin PRep(\mathcal{K}_{\succ})\]
Optimal repairs

Import three notions of optimal repair from database setting

[Staworko, Chomicki, and Marcinkowski, 2012]

Let \( R \) be a repair of \( K \) (\( R \in SRep(K) \))

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

\[
\{\alpha_1, \alpha_2, \delta, \epsilon\} \in PRep(K_\succ)
\]

\[
\{\beta_1, \beta_2, \delta, \epsilon\} \text{ global improvement}
\]

\[
\Rightarrow \quad \{\alpha_1, \alpha_2, \delta, \epsilon\} \notin GRep(K_\succ)
\]
Optimal repairs

Import three notions of optimal repair from database setting

[Staworko, Chomicki, and Marcinkowski, 2012]

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

- $\mathcal{R}$ is completion-optimal ($\mathcal{R} \in CRep(\mathcal{K}_{\succ})$) 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

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

Pareto- and globally-optimal

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

Completion-optimal

$CRep(\mathcal{K}_{\succ}) = \{ \{\alpha\}, \{\gamma\} \}$
Import three notions of optimal repair from database setting

[Staworko, Chomicki, and Marcinkowski, 2012]

\[ CRep(\mathcal{K}_{\succ}) \subseteq GRep(\mathcal{K}_{\succ}) \subseteq PRep(\mathcal{K}_{\succ}) \subseteq SRep(\mathcal{K}) \]
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 \( CRep(\mathcal{K}_{\succ}) = GRep(\mathcal{K}_{\succ}) = PRep(\mathcal{K}_{\succ}) \)

\[
\begin{align*}
\text{hasDisease}(bob, d_1) \\
\text{SmallCellCarcinoma}(d_1) \succ \text{Adenocarcinoma}(d_1) \\
\text{primaryTumor}(d_1, o_1) \succ \text{primaryTumor}(d_1, o_2) \\
\text{Lung}(o_1) \succ \text{Breast}(o_2)
\end{align*}
\]

\[
\{ \text{hasDisease}(bob, d_1), \text{SmallCellCarcinoma}(d_1), \text{primaryTumor}(d_1, o_1), \text{Lung}(o_1), \text{primaryTumor}(d_1, o_2) \} \\
\{ \text{hasDisease}(bob, d_1), \text{SmallCellCarcinoma}(d_1), \text{primaryTumor}(d_1, o_1), \text{Lung}(o_1), \text{Breast}(o_2) \}
\]
Inconsistency-tolerant semantics

Use optimal repairs instead of subset repairs

1. **X-AR**: every $X$-optimal repair

   \[ \mathcal{K} \models_{\text{AR}} X q \iff \forall R \in X\text{Rep}(\mathcal{K} \triangleright), (R, T) \models q \]

2. **X-brave**: some $X$-optimal repair

   \[ \mathcal{K} \models_{\text{brave}} X q \iff \exists R \in X\text{Rep}(\mathcal{K} \triangleright), (R, T) \models q \]

3. **X-IAR**: intersection of all $X$-optimal repairs

   \[ \mathcal{K} \models_{\text{IAR}} X q \iff (\bigcap R, T) \models q, \quad \bigcap R = \bigcap_{R \in X\text{Rep}(\mathcal{K} \triangleright)} R \]

   \[ \mathcal{K} \models_{\text{IAR}} X q \implies \mathcal{K} \models_{\text{AR}} X q \implies \mathcal{K} \models_{\text{brave}} X q \]
Complexity of reasoning with optimal repairs

Data complexity of query entailment

<table>
<thead>
<tr>
<th></th>
<th>Globally-optimal</th>
<th>Pareto-optimal</th>
<th>Completion-optimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR</td>
<td>$\Pi^p_2$-complete</td>
<td>coNP-complete</td>
<td>coNP-complete</td>
</tr>
<tr>
<td>IAR</td>
<td>$\Pi^p_2$-complete</td>
<td>coNP-complete</td>
<td>coNP-complete</td>
</tr>
<tr>
<td>Brave</td>
<td>$\Sigma^p_2$-complete</td>
<td>NP-complete</td>
<td>NP-complete</td>
</tr>
</tbody>
</table>

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

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$_R$ 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$_R$ 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
    \[\text{[Bienvenu, Bourgaux, and Goasdoué, 2014]}\]
  - **CAvSAT**: databases + denial constraints, S-AR
    \[\text{[Dixit and Kolaitis, 2019]}\]

- **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 $\alpha$ to $\beta$ 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 $\alpha$ to $\beta$ 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 $x_\alpha$ to true means that fact $\alpha$ 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: $\text{neg}_1$ and $\text{neg}_2$, 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 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
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 ($\text{neg}_1$) than the other ($\text{neg}_2$)
- **Score-structured case**: P-encodings are much better than C-encoding
## Practical SAT-based approaches

### Some experimental results

<table>
<thead>
<tr>
<th>Alg. 1</th>
<th>P₁</th>
<th>P₂</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>q₁</td>
<td>q₂</td>
<td>q₃</td>
</tr>
<tr>
<td>P₁</td>
<td>417</td>
<td>141</td>
<td>350</td>
</tr>
<tr>
<td>P₂</td>
<td>804</td>
<td>142</td>
<td>379</td>
</tr>
<tr>
<td>C</td>
<td>252,694</td>
<td>179</td>
<td>550</td>
</tr>
<tr>
<td></td>
<td>q₄</td>
<td>q₅</td>
<td>q₆</td>
</tr>
<tr>
<td></td>
<td>12,799</td>
<td>224</td>
<td>4,009</td>
</tr>
<tr>
<td></td>
<td>326,594</td>
<td>213</td>
<td>8,684</td>
</tr>
<tr>
<td></td>
<td>oom</td>
<td>284</td>
<td>t.o</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Alg. 2</th>
<th>P₁</th>
<th>P₂</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>q₁</td>
<td>q₂</td>
<td>q₃</td>
</tr>
<tr>
<td>P₁</td>
<td>268</td>
<td>166</td>
<td>326</td>
</tr>
<tr>
<td>P₂</td>
<td>502</td>
<td>163</td>
<td>333</td>
</tr>
<tr>
<td>C</td>
<td>oom</td>
<td>632</td>
<td>t.o</td>
</tr>
<tr>
<td></td>
<td>q₄</td>
<td>q₅</td>
<td>q₆</td>
</tr>
<tr>
<td></td>
<td>1,730</td>
<td>214</td>
<td>11,263</td>
</tr>
<tr>
<td></td>
<td>2,961</td>
<td>221</td>
<td>10,833</td>
</tr>
<tr>
<td></td>
<td>t.o</td>
<td>551</td>
<td>oom</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Alg. 3</th>
<th>P₁</th>
<th>P₂</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>q₁</td>
<td>q₂</td>
<td>q₃</td>
</tr>
<tr>
<td>P₁</td>
<td>272</td>
<td>154</td>
<td>313</td>
</tr>
<tr>
<td>P₂</td>
<td>466</td>
<td>146</td>
<td>281</td>
</tr>
<tr>
<td>C</td>
<td>oom</td>
<td>624</td>
<td>t.o</td>
</tr>
<tr>
<td></td>
<td>q₄</td>
<td>q₅</td>
<td>q₆</td>
</tr>
<tr>
<td></td>
<td>t.o</td>
<td>211</td>
<td>245,804</td>
</tr>
<tr>
<td></td>
<td>t.o</td>
<td>201</td>
<td>241,030</td>
</tr>
<tr>
<td></td>
<td>t.o</td>
<td>550</td>
<td>oom</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Alg. 4</th>
<th>P₁</th>
<th>P₂</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>q₁</td>
<td>q₂</td>
<td>q₃</td>
</tr>
<tr>
<td>P₁</td>
<td>362</td>
<td>166</td>
<td>997</td>
</tr>
<tr>
<td>P₂</td>
<td>566</td>
<td>193</td>
<td>972</td>
</tr>
<tr>
<td>C</td>
<td>oom</td>
<td>764</td>
<td>t.o</td>
</tr>
<tr>
<td></td>
<td>q₄</td>
<td>q₅</td>
<td>q₆</td>
</tr>
<tr>
<td></td>
<td>42,544</td>
<td>281</td>
<td>559,923</td>
</tr>
<tr>
<td></td>
<td>36,923</td>
<td>304</td>
<td>546,199</td>
</tr>
<tr>
<td></td>
<td>t.o</td>
<td>846</td>
<td>oom</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Alg. 5</th>
<th>P₁</th>
<th>P₂</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>q₁</td>
<td>q₂</td>
<td>q₃</td>
</tr>
<tr>
<td>P₁</td>
<td>383</td>
<td>135</td>
<td>335</td>
</tr>
<tr>
<td>P₂</td>
<td>565</td>
<td>157</td>
<td>309</td>
</tr>
<tr>
<td>C</td>
<td>192,429</td>
<td>164</td>
<td>544</td>
</tr>
<tr>
<td></td>
<td>q₄</td>
<td>q₅</td>
<td>q₆</td>
</tr>
<tr>
<td></td>
<td>8,192</td>
<td>211</td>
<td>3,419</td>
</tr>
<tr>
<td></td>
<td>225,170</td>
<td>207</td>
<td>5,963</td>
</tr>
<tr>
<td></td>
<td>oom</td>
<td>238</td>
<td>t.o</td>
</tr>
</tbody>
</table>

### Note

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

<table>
<thead>
<tr>
<th></th>
<th>q1</th>
<th>q2</th>
<th>q3</th>
<th>q4</th>
<th>q5</th>
<th>q6</th>
<th>q7</th>
<th>q8</th>
<th>q9</th>
<th>q10</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>∼CQAPri</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(single ans.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P1 1</td>
<td>t.o</td>
<td>232,386</td>
<td>2,622</td>
<td>t.o</td>
<td>846</td>
<td>4,772</td>
<td>2,602</td>
<td>713</td>
<td>370,340</td>
<td>1,983</td>
</tr>
<tr>
<td>P1 2</td>
<td>t.o</td>
<td>224,840</td>
<td>2,672</td>
<td>t.o</td>
<td>863</td>
<td>4,919</td>
<td>2,453</td>
<td>718</td>
<td>365,374</td>
<td>1,901</td>
</tr>
<tr>
<td>C</td>
<td>t.o</td>
<td>t.o</td>
<td>oom</td>
<td>t.o</td>
<td>t.o</td>
<td>oom</td>
<td>oom</td>
<td>oom</td>
<td>t.o</td>
<td>oom</td>
</tr>
<tr>
<td><strong>∼CAvSAT</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(multiple ans.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P1 1</td>
<td>135,263</td>
<td>84,826</td>
<td>456</td>
<td>oom</td>
<td>1,155</td>
<td>434</td>
<td>2,060</td>
<td>369</td>
<td>t.o</td>
<td>1,043</td>
</tr>
<tr>
<td>P1 2</td>
<td>208,831</td>
<td>50,469</td>
<td>379</td>
<td>oom</td>
<td>1,109</td>
<td>504</td>
<td>1,909</td>
<td>364</td>
<td>t.o</td>
<td>1,275</td>
</tr>
<tr>
<td>C</td>
<td>oom</td>
<td>75,688</td>
<td>1,050</td>
<td>oom</td>
<td>oom</td>
<td>oom</td>
<td>oom</td>
<td>oom</td>
<td>oom</td>
<td>oom</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>q11</th>
<th>q12</th>
<th>q13</th>
<th>q14</th>
<th>q15</th>
<th>q16</th>
<th>q17</th>
<th>q18</th>
<th>q19</th>
<th>q20</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>∼CQAPri</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(single ans.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P1 1</td>
<td>10,004</td>
<td>71,135</td>
<td>21,990</td>
<td>5,822</td>
<td>113,593</td>
<td>t.o</td>
<td>2,696</td>
<td>86,121</td>
<td>2,460</td>
<td>1,762</td>
</tr>
<tr>
<td>P1 2</td>
<td>10,096</td>
<td>62,227</td>
<td>21,038</td>
<td>5,830</td>
<td>115,590</td>
<td>t.o</td>
<td>2,551</td>
<td>88,288</td>
<td>2,378</td>
<td>1,702</td>
</tr>
<tr>
<td>C</td>
<td>oom</td>
<td>oom</td>
<td>oom</td>
<td>t.o</td>
<td>t.o</td>
<td>oom</td>
<td>oom</td>
<td>oom</td>
<td>oom</td>
<td>oom</td>
</tr>
<tr>
<td><strong>∼CAvSAT</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(multiple ans.)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P1 1</td>
<td>34,912</td>
<td>323,386</td>
<td>84,726</td>
<td>19,074</td>
<td>508,810</td>
<td>t.o</td>
<td>7,194</td>
<td>359,410</td>
<td>6,188</td>
<td>4,470</td>
</tr>
<tr>
<td>P1 2</td>
<td>35,362</td>
<td>284,407</td>
<td>84,111</td>
<td>20,755</td>
<td>586,030</td>
<td>t.o</td>
<td>7,555</td>
<td>391,599</td>
<td>6,410</td>
<td>4,399</td>
</tr>
<tr>
<td>C</td>
<td>t.o</td>
<td>oom</td>
<td>t.o</td>
<td>t.o</td>
<td>oom</td>
<td>t.o</td>
<td>oom</td>
<td>t.o</td>
<td>oom</td>
<td>oom</td>
</tr>
</tbody>
</table>

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’?

⇒ Study links with other formalisms

[Bienvenu and Bourgaux 2020, 2023]
Abstract argumentation: well-known framework to deal with contradictory information in AI

An (abstract) argumentation framework (AF) is a pair \((\text{Args}, \sim\nearrow)\) where

- \(\text{Args}\) is a finite set of arguments
- \(\sim\nearrow \subseteq \text{Args} \times \text{Args}\) is the attack relation: \(\alpha\) attacks \(\beta\) if \(\alpha \sim\nearrow \beta\)

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

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

\[
\begin{align*}
\alpha & \sim \sim \rightarrow \beta \sim \sim \rightarrow \gamma \\
\delta & \leftarrow \sim \sim \rightarrow \epsilon
\end{align*}
\]

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: $\beta \rightsquigarrow \alpha$ if $\beta \rightsquigarrow \alpha$ and $\alpha \ntriangleright \beta$

- **Set-based AF (SETAF)**
  - collective attacks $S \rightsquigarrow \alpha$ with $S$ finite set of arguments

Combined into **PSETAF** $(\text{Args}, \rightsquigarrow, \succ)$

- $S \rightsquigarrow \alpha$ if $S \rightsquigarrow \alpha$ and $\alpha \ntriangleright \beta$ for every $\beta \in S$
Translation of a prioritized KB $\mathcal{K}_\succ = (\mathcal{D}, \mathcal{T}, \succ)$ into a PSEPAF $F_{\mathcal{K},\succ}$

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

$$\Rightarrow C \setminus \{\alpha\} \succ\sim \alpha \text{ if } \alpha \nprec \beta \text{ for every } \beta \in C$$
Connections with argumentation

Translation

\( R \) is a Pareto-optimal repair of \( K \)

iff

\( R \) is a stable extension of \( F_{K,\succ} \)

If \( \succ \) is transitive or if \( K \) has only binary conflicts, then \( F_{K,\succ} \) is coherent:

\( R \) is a Pareto-optimal repair of \( K \)

iff

\( R \) is a preferred extension of \( F_{K,\succ} \)

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

⇒ 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:
\[
\text{Child}(x) \land \text{Adult}(x) \rightarrow \bot \\
\text{Child}(x) \land \text{Adult}(x) \rightarrow \{\neg \text{Child}(x)\}
\]

Example of universal constraint and active universal constraint:
\[
\text{Lung}(x) \land \neg \text{LeftLg}(x) \land \neg \text{RightLg}(x) \rightarrow \bot \\
\text{Lung}(x) \land \neg \text{LeftLg}(x) \land \neg \text{RightLg}(x) \rightarrow \{+\text{LeftLg}(x), +\text{RightLg}(x)\}
\]

A ground active integrity constraint (AIC) is a formula of the form
\[
\alpha_1 \land \cdots \land \alpha_n \land \neg \beta_1 \land \cdots \land \neg \beta_m \rightarrow \{A_1, \ldots, A_k\}
\]

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 \( D \circ \mathcal{U} \) is a repair, where \( D \circ \mathcal{U} = D \setminus \{ \alpha \mid -\alpha \in \mathcal{U} \} \cup \{ \alpha \mid +\alpha \in \mathcal{U} \} \)

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 \( D \circ \mathcal{U} \setminus \{ A \} \not\models r \)

- **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 \( D \circ \{ A_1, \ldots, A_{i-1} \} \not\models r_i \)

- **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 \( D \circ \mathcal{V} \not\models r \)

- **Justified**...
Connections with active integrity constraints
Prioritized databases with universal constraints

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
  \[ \forall \vec{x} (\alpha_1 \land \cdots \land \alpha_n \land \neg \beta_1 \land \cdots \land \neg \beta_m \rightarrow \bot) \]
- 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

\[ S(x, y) \land S(x, z) \land y \neq z \rightarrow \bot \quad S(x, y) \land \neg A(x) \rightarrow \bot \]

\[ R(x, y) \land R(x, z) \land y \neq z \rightarrow \bot \quad S(x, y) \land \neg B(x) \rightarrow \bot \]

\[ R(y, x) \land S(z, x) \rightarrow \bot \]

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

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

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

\[ \text{PRep}(\mathcal{K}_{\succ}) = \text{GRep}(\mathcal{K}_{\succ}) \cup \{\{R(d, c)\}, \{R(d, c), S(a, b), A(a), B(a)\}\} \]
Connections with active integrity constraints

Translation

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

- $\eta^\mathcal{T}_\succ = \{r_C \mid C \in \text{Conf}(\mathcal{D}, \mathcal{T})\}$ where $\text{fix}(\alpha) = -\alpha$, $\text{fix}(\neg \alpha) = +\alpha$ and $r_C := \bigwedge_{\lambda \in C} \lambda \rightarrow \{\text{fix}(\lambda) \mid \lambda \in C, \forall \mu \in C, \lambda \not\succ \mu\}$

- Conflicts fixed by modifying least preferred literals according to $\succ$

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

- $S(a, b) \land S(a, c) \rightarrow \{-S(a, b), -S(a, c)\}$
- $R(d, b) \land R(d, c) \rightarrow \{-R(d, b), -R(d, c)\}$
- $R(d, b) \land S(a, b) \rightarrow \{-S(a, b)\}$
- $R(d, c) \land S(a, c) \rightarrow \{-R(d, c)\}$
- $S(a, b) \land \neg A(a) \rightarrow \{+A(a)\}$
- $S(a, c) \land \neg A(a) \rightarrow \{-S(a, c), +A(a)\}$
- $S(a, b) \land \neg B(a) \rightarrow \{-S(a, b), +B(a)\}$
- $S(a, c) \land \neg B(a) \rightarrow \{+B(a)\}$
Pareto $\equiv$ Founded $\equiv$ Grounded $\equiv$ Justified $\Rightarrow$ Well-Founded

$R = D \circ U$ is a Pareto-optimal repair of $K_\succ$
iff
$U$ is a **founded** repair update of $D$ w.r.t. $\eta^T_\succ$
iff
$U$ is a **grounded** repair update of $D$ w.r.t. $\eta^T_\succ$
iff
$U$ is a **justified** repair update of $D$ w.r.t. $\eta^T_\succ$
Connections with active integrity constraints

Translation

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

⇒ 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 $S_1, \ldots, S_n$
  - defines a score-structured priority relation $\succ$:
    $\alpha \succ \beta$ iff $\{\alpha, \beta\} \in C$ for some conflict $C$, $\alpha \in S_i$ and $\beta \in S_j$ with $i < j$

- Preordered KBs: $\triangleright$ reflexive and transitive binary relation over $\mathcal{D}$
  - defines a transitive priority relation $\succ$:
    $\alpha \succ \beta$ iff $\{\alpha, \beta\} \in C$ for some conflict $C$, $\alpha \triangleright \beta$ and $\beta \not\triangleright \alpha$
Comparison with other approaches for prioritized KBs

Two main settings that both reduce to prioritized KBs

- \( D \) partitioned into priority levels \( S_1, \ldots, S_n \)
  - defines a score-structured priority relation \( \succ \):
    \[ \alpha \succ \beta \text{ iff } \{\alpha, \beta\} \in C \text{ for some conflict } C, \alpha \in S_i \text{ and } \beta \in S_j \text{ with } i < j \]
- Preordered KBs: \( \triangleright \) reflexive and transitive binary relation over \( D \)
  - defines a transitive priority relation \( \succ \):
    \[ \alpha \succ \beta \text{ iff } \{\alpha, \beta\} \in C \text{ for some conflict } C, \alpha \triangleright \beta \text{ and } \beta \not\succ \alpha \]

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 \( \mathcal{R} \cap S_i \subset \mathcal{R}' \cap S_i \) and \( \mathcal{R} \cap S_j = \mathcal{R}' \cap S_j \) for \( 1 \leq j < i \)
  - [Bienvenu, Bourgaux, and Goasdoué, 2014]
    - coincide with optimal repairs \( (CRep(K_\succ) = GRep(K_\succ) = PRep(K_\succ) \) 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

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

$\alpha \succ \beta \succ \gamma$

$\text{Elect}(\mathcal{K}_{\succeq}) = \{\alpha\}$
$\text{grounded}(\mathcal{K}_{\succeq}) = \{\alpha, \gamma\}$
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 $\text{Elect}(\mathcal{K} \succ)$: $\alpha$ is elected iff for every conflict $\mathcal{C}$, $\alpha \in \mathcal{C}$ implies $\alpha \succ \beta$ for some $\beta \in \mathcal{C}$
  - $\text{Elect}(\mathcal{K} \succ) \subseteq \text{grounded}(\mathcal{K} \succ)$
  - Inclusion can be strict

  \[
  \alpha \succ \beta \succ \gamma
  \]

  \[
  \text{Elect}(\mathcal{K} \succ) = \{\alpha\}
  \]

  \[
  \text{grounded}(\mathcal{K} \succ) = \{\alpha, \gamma\}
  \]

- Preferred repair $\text{Partial}_{PR}(\mathcal{K} \succ)$: union of $\bigcap_{\mathcal{R} \in \mathcal{XR}ep(\mathcal{K} \succ \geq)} \mathcal{R}$ for all total preorders $\geq$ extending $\succ$ (with $\succ \geq$ priority relation that corresponds to $\geq$)
  - $\text{Partial}_{PR}(\mathcal{K} \succ) = \bigcap_{\mathcal{R} \in \mathcal{C}Rep(\mathcal{K} \succ)} \mathcal{R}$

$\text{Elect}(\mathcal{K} \succ) \subseteq \text{grounded}(\mathcal{K} \succ) \subseteq \bigcap_{\mathcal{R} \in \mathcal{P}Rep(\mathcal{K} \succ)} \mathcal{R} \subseteq \bigcap_{\mathcal{R} \in \mathcal{G}Rep(\mathcal{K} \succ)} \mathcal{R} \subseteq \bigcap_{\mathcal{R} \in \mathcal{C}Rep(\mathcal{K} \succ)} \mathcal{R} = \text{Partial}_{PR}(\mathcal{K} \succ)$
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 $\pi(\mathcal{K}_{\succ})$: $S_1 \cup \cdots \cup S_{inc(\mathcal{K}_{\succ})-1}$ 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 $\pi(K_{\succ})$: $S_1 \cup \cdots \cup S_{inc(K_{\succ})-1}$ where $inc(K_{\succ})$ is the inconsistency degree of $K_{\succ}$
- Non-defeated repair $nd(K_{\succ})$: union of the intersections of the (subset) repairs of $S_1, S_1 \cup S_2, \ldots, S_1 \cup \cdots \cup S_n$
  - has been shown to coincide with $Elect(K_{\succ})$

\[
\begin{align*}
\pi(K_{\succ}) & = \{ \lambda \} \\
nd(K_{\succ}) & = \{ \lambda, \epsilon \} \\
grounded(K_{\succ}) & = \{ \lambda, \nu, \epsilon \}
\end{align*}
\]
Comparison with other approaches for prioritized KBs
Selection of a single consistent set of facts to query

**Priority levels (score-structured)**

- **Possibilistic repair** $\pi(K_\succ)$: $S_1 \cup \cdots \cup S_{inc(K_\succ)} - 1$ where $inc(K_\succ)$ is the inconsistency degree of $K_\succ$
- **Non-defeated repair** $nd(K_\succ)$: union of the intersections of the (subset) repairs of $S_1$, $S_1 \cup S_2$, ..., $S_1 \cup \cdots \cup S_n$
  - has been shown to coincide with $Elec(K_\succ)$
- **Prioritized inclusion-based non-defeated repair** $pind(K_\succ)$: as non-defeated but using optimal repairs (intractable!)
  - has been shown to coincide with $\bigcap_{\mathcal{R} \in \mathcal{XRep}(K_\succ)} \mathcal{R}$

$\begin{array}{cccc}
S_1 & S_2 & S_3 & S_4 \\
\alpha & \gamma & \delta \\
\beta & \gamma \\
\lambda & \mu & \nu & \epsilon
\end{array}$

$\pi(K_\succ) = \{\lambda\}$
$nd(K_\succ) = \{\lambda, \epsilon\}$
$grounded(K_\succ) = \{\lambda, \nu, \epsilon\}$
$pind(K_\succ) = \{\lambda, \nu, \delta, \epsilon\}$
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 $\pi(K_\succ)$: $S_1 \cup \cdots \cup S_{inc(K_\succ)} - 1$ where $inc(K_\succ)$ is the inconsistency degree of $K_\succ$
- Non-defeated repair $nd(K_\succ)$: union of the intersections of the (subset) repairs of $S_1, S_1 \cup S_2, \ldots, S_1 \cup \cdots \cup S_n$
  - has been shown to coincide with $Elec(K_\succ)$
- Prioritized inclusion-based non-defeated repair $pind(K_\succ)$: as non-defeated but using optimal repairs (intractable!)
  - has been shown to coincide with $\bigcap_{R \in XRep(K_\succ)} R$

\[
\begin{align*}
\pi(K_\succ) &= \{\lambda\} \\
nd(K_\succ) &= \{\lambda, \epsilon\} \\
grounded(K_\succ) &= \{\lambda, \nu, \epsilon\} \\
pind(K_\succ) &= \{\lambda, \nu, \delta, \epsilon\}
\end{align*}
\]

\[
\pi(K_\succ) \subseteq nd(K_\succ) \subseteq grounded(K_\succ) \subseteq pind(K_\succ) = \bigcap_{R \in XRep(K_\succ)} R
\]
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?
Meghyn Bienvenu and Camille Bourgaux.
Querying and repairing inconsistent prioritized knowledge bases: Complexity analysis and links with abstract argumentation.

Meghyn Bienvenu and Camille Bourgaux.
Querying inconsistent prioritized data with ORBITS: algorithms, implementation, and experiments.

Meghyn Bienvenu and Camille Bourgaux.
Inconsistency handling in prioritized databases with universal constraints: Complexity analysis and links with active integrity constraints.

Slawek Staworko, Jan Chomicki, and Jerzy Marcinkowski.
Prioritized repairing and consistent query answering in relational databases.

Meghyn Bienvenu, Camille Bourgaux, and François Goasdoué.
Querying inconsistent description logic knowledge bases under preferred repair semantics.

Akhil A. Dixit and Phokion G. Kolaitis.
A SAT-based system for consistent query answering.

Sihem Belabbes, Salem Benferhat, and Jan Chomicki.
Handling inconsistency in partially preordered ontologies: the Elect method.

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