1. Motivation

- Information exchange between rules and ontology can cause inconsistency.

- DL-program $\Pi = (\mathcal{O}, P)$ is inconsistent.

\[ \mathcal{O} = \{ \]
- (1) Child $\sqsubseteq$ HasParent
- (2) Adopted $\sqsubseteq$ Child
- (3) Male $\sqsubseteq$ John
- (4) Male $\sqsubseteq$ Pat
- (5) Female $\sqsubseteq$ Male
- (6) hasParent(john, pat)
- (7) isnchildof(john, alex)
- (8) boy(john)
- (9) hasFather(john, pat) $\rightarrow$ DL: Male $\sqsubseteq$ boy; Male $\sqsubseteq$ pat; DL: hasParent(john, pat)
- (10) $\bot$ $\not\models$ notDL; Adopted(john), pat, boy, notDL, Child $\sqcup$ boy; $\sqcup$ Male(alex)

\[ P = \{ \]
- (11) $\bot$ $\not\models$ notDL; Adopted(john), pat, boy, notDL, Child $\sqcup$ boy; $\sqcup$ Male(alex)

Aim of this work: change ontology ABox to make DL-program consistent.

- $\mathcal{A}' = \{ \text{Male}(john), \text{hasParent}(john, pat) \}$ is a possible repair of $\Pi$ that yields $\text{flp}$-repair answer set $I = \{ \text{ischild}(john, alex), \text{boy}(john) \}$.

Contributions:

- Notion of repair and repair answer set;
- Preference selection function $\sigma$ and its independence property;
- Sound and complete algorithm for repair computation;
- Tractable cases of special ontology repair problem for DL-Lite$_A$.

2. DL-programs

- DL-program: ontology + rules (loose-coupling approach);
- DL-atoms serve as query interfaces to ontology;
- Bidirectional information flow between ontology and rules.

\[ \Pi = (\mathcal{O}, P) \] is a DL-program.

\[ \Pi = \{ \mathcal{O} \subseteq (1) C \sqsubseteq (2) A \} \]

\[ P = \{ \]
- (3) $r(c) \quad \text{D}L[C \sqsubseteq r; D(c)]$
- (4) $q(c) \quad \text{D}L[C \sqsubseteq r; D(c)]$

\[ \Pi = (\mathcal{O}, P) \] is consistent.

3. DL-program Evaluation

Given:

\[ \Pi = (\mathcal{O}, P), P = \{ r(c); q(c) \quad \text{D}L[C \sqsubseteq r; D(c)] \}, \mathcal{O} = \{ C \sqsubseteq D; A(c) \}. \]

Construct:

\[ \Pi = \{ r(c); q(c) \quad \exists c \in c ; \exists c > n c \} \]

Compute:

- $\text{Answer sets of } \Pi: \text{AS}(\Pi) = \{ r(c), q(c) \quad \text{D}L[C \sqsubseteq r; D(c)] \}$

Check:

- $\text{Consistency: } \text{AS}(\Pi) = \{ r(c), q(c) \quad \text{D}L[C \sqsubseteq r; D(c)] \}$
- $\text{Minimality: } \text{AS}(\Pi) = \{ r(c), q(c) \quad \text{D}L[C \sqsubseteq r; D(c)] \}$

4. Ontology Repair Problem (ORP)

Ontology repair problem (ORP) is a triple $P = (\mathcal{O}, D_1, D_2)$, where $\mathcal{O} = (T, A)$; ontology;

\[ \Pi = \{ \mathcal{O} \subseteq (1) C \sqsubseteq (2) A \} \]

Related problems were studied in [Sakama, et al., 2003; Calvanese et al., 2012].

Repair (solution) for $P$ is any ABox $\mathcal{A}'$ s.t.

- $\mathcal{O}' = \{ T, A' \}$ is consistent;
- $\tau((T, A' \cup U_j^i) \sqsubseteq Q_j \text{ holds for } 1 \leq j \leq m_1)$
- $\tau((T, A' \cup U_j^i) \sqsubseteq Q_j \text{ holds for } 1 \leq j \leq m_2)$

Given:

\[ \Pi = (\mathcal{O}, P), s.t. P = \{ r(c); q(c) \quad \text{D}L[C \sqsubseteq r; D(c)] \}

\[ \Pi = \{ r(c); q(c) \quad \text{D}L[C \sqsubseteq r; D(c)] \}

\[ \mathcal{A}' = \{ A(c) \} \] is a possible repair for $P$ if $\mathcal{O} = \{ E \sqsubseteq D; A \sqsubseteq D; \neg C(c) \}$.

5. Selection Preferences and Tractable Cases of ORP

Selection function $\sigma$: given set of ABoxes $\mathcal{S}$ and ABox $\mathcal{A}$ selects $\sigma$-preferred $\mathcal{S}' \subseteq \mathcal{S}$.

Independent $\sigma$: ABox one can immediately decide whether $\mathcal{A}' \in \mathcal{S}$ is $\sigma$-selected.

- $\sigma$ deletion repair is independent;
- $\sigma$ set-minimal (cardinality minimal) change repair is independent.

6. Repair Answer Set Computation

- RepAns extends DL-program evaluation to DL-program repair computation;
- RepAnsSet uses RepAns to compute answer sets of repaired program.

7. References