The Combined Approach to Query Answering in DL-Lite

Roman Kontchakov

Department of Computer Science and Inf. Systems, Birkbeck College, London

http://www.dcs.bbk.ac.uk/~roman

joint work with

Carsten Lutz, David Toman, Frank Wolter and Michael Zakharyaschev
'DL-Lite family includes the first DLs specifically tailored for effective query answering over large amounts of instances.'

D. Calvanese et al., 2007

effective = in $AC^0$ for data complexity

---

**Pure Query Rewriting**

---

---
Pure Query Rewriting: an Example of PerfectRef

\[ q(x) \leftarrow \text{TeachesTo}(x, y), \; \text{HasTutor}(y, z) \]
\[ \text{Student} \sqsubseteq \exists \text{HasTutor} \]
\[ \text{HasTutor}(x_1, y_1) \leftarrow \text{Student}(x_1) \]
\[ q(x) \leftarrow \text{TeachesTo}(x, y), \; \text{Student}(y) \]
\[ \exists \text{TeachesTo} \sqsubseteq \text{Student} \]
\[ \text{Student}(x_2) \leftarrow \text{TeachesTo}(y_2, x_2) \]
\[ q(x) \leftarrow \text{TeachesTo}(x, y), \; \text{TeachesTo}(x_2, y) \]
\[ \text{unification} \]
\[ q(x) \leftarrow \text{TeachesTo}(x, y) \]
\[ \text{Professor} \sqsubseteq \exists \text{TeachesTo} \]
\[ \text{TeachesTo}(x_3, y_3) \leftarrow \text{Professor}(x_3) \]
\[ q(x) \leftarrow \text{Professor}(x) \]
\[ \exists \text{HasTutor} \sqsubseteq \text{Professor} \]
\[ \text{Professor}(x_4) \leftarrow \text{HasTutor}(y_4, x_4) \]
\[ q(x) \leftarrow \text{HasTutor}(y_4, x) \]

**Intuitive!**

**NB.** what if Student has many subclasses? \( \exists \text{TeachesTo}? \)

\[ \mathcal{O}( (|\mathcal{T}| \cdot |q|)^{|q|} ) \text{ subqueries} \]
Combined Approach in $\mathcal{EL}$

(Lutz, Toman & Wolter, 2008)

query answering in $\mathcal{EL}$ is \textsc{PTIME}-complete for data complexity

$\mathcal{A}'$ is computed in polytime in $\mathcal{A}$
and only when $\mathcal{A}$ is updated
Variants of DL-Lite

\[ R := P \mid P^- \]

\[ C := \bot \mid A \mid \geq kR \]

TBox concept inclusions

\[ DL-Lite^N_{\text{horn}}: \quad C_1 \land \cdots \land C_n \sqsubseteq C \]

\[ DL-Lite^N_{\text{core}}: \quad C_1 \sqsubseteq C_2, \quad C_1 \sqsubseteq \neg C_2 \]

ABox assertions: \( C(a), \ R(a, b) \)

\[ DL-Lite^F_\alpha = DL-Lite^N_\alpha \text{ with } \exists R \text{ and } \geq 2 R \sqsubseteq \bot \text{ only} \]

\[ DL-Lite^{(HN)}_\alpha = DL-Lite^N_\alpha \text{ with (restricted) role inclusions, role disjointness, etc.} \]

\[ DL-Lite^{(HF)}_\alpha = DL-Lite^F_\alpha \text{ with (restricted) role inclusions, role disjointness, etc.} \]

In all these languages, answering positive existential queries (under UNA) is in \( \text{AC}^0 \) for data complexity.

Positive existential formulas are built from \( A(x) \) and \( R(x, y) \) using \( \exists, \land \) and \( \lor \).
ABox Expansion in DL-Lite

**canonical interpretation** \( \mathcal{I}_K \):

\[
\Delta^\mathcal{I} = \text{Ind}(\mathcal{A}) \cup \{c_R \mid R \text{ is generating in } K\}
\]

\begin{align*}
\begin{array}{ll}
\mathcal{A}^\mathcal{I}_K &= \{a \mid K \models A(a)\} \cup \{c_R \mid \mathcal{T} \models \exists R^- \sqsubseteq A\} \\
\mathcal{P}^\mathcal{I}_K &= \{(a, b) \mid P(a, b) \in \mathcal{A}\} \cup \{(d, c_P) \mid d \leadsto c_P\} \cup \{(c_{P^-}, d) \mid d \leadsto c_{P^-}\}
\end{array}
\end{align*}

\( \mathcal{I}_K \) is not a model

\( \mathcal{T} = \{A \sqsubseteq \exists P, \geq 2 P^- \sqsubseteq \bot\}, \quad \mathcal{A} = \{A(a), A(b)\} \)

\( \mathcal{I}_K \) does not give the right answers

\( q = \exists v P(v, v), \quad \mathcal{T} = \{A \sqsubseteq \exists P, \exists P^- \sqsubseteq \exists P\}, \quad \mathcal{A} = \{A(a)\} \)

\( q = \exists v_2 (P(v_1, v_2) \land P(v_3, v_2)), \quad \mathcal{T} = \{A \sqsubseteq \exists P\}, \quad \mathcal{A} = \{A(a), A(b)\} \)

The unravelling \( \mathcal{U}_K \) is almost a (canonical) model of \( \mathcal{I}_K \)

and does give the right answers
Query Rewriting for $DL$-$Lite_N^{horn}$ (1)

we rewrite a given CQ $q$ into an FO query $q^\dagger$ such that

- answers to $q$ in $\mathcal{U}_\mathcal{K}$ = answers to $q^\dagger$ in $\mathcal{I}_\mathcal{K}$
- $|q^\dagger| = O(|q| \cdot |\mathcal{T}|)$

$q^\dagger = \exists \vec{u} \left( \varphi \land \varphi_1 \land \varphi_2 \land \varphi_3 \right)$

$\varphi_1 = \bigwedge_{v \notin \vec{u}} \left( R \text{ is a role in } \mathcal{T} \right)$

\textbf{NB.} if $\varphi_1$ is replaced with $\varphi'_1 = \bigwedge_{v \notin \vec{u}} \neg \text{aux}(v)$, where aux is a new relation containing all $c_R$, then $|q^\dagger| = O(|q|)$
Query Rewriting for DL-Lite$_{\text{horn}}^N$ (2)

- answers to $q$ in $\mathcal{U}_K = \text{answers to } q^\dagger \text{ in } \mathcal{I}_K$

$\mathcal{U}_K$ is a ‘forest’ model, so if $t$ is matched to a non-ABox element then a part of $q$ containing $t$ must be homomorphically embeddable into a tree

A tree witness $f_{R,t}: \text{term}(q) \rightarrow (N^-_R)^* \text{ (finite words over roles)}$

1. $f_{R,t}(t) = \varepsilon$
2. If $f_{R,t}(s) = \varepsilon$ and $R(s, s') \in q$ then $f_{R,t}(s') = R$
3. If $f_{R,t}(s) = w \cdot S$ and $S'(s, s') \in q$ with $S' \neq S^-$ then $f_{R,t}(s') = w \cdot S \cdot S'$
4. If $f_{R,t}(s) = w \cdot S$ and $S^-(s, s') \in q$ then $f_{R,t}(s') = w$

$q = \exists v \ P(v, v): f_{P,v} \text{ does not exist}$

$q = \exists v_2 \ (P(v_1, v_2) \land P(v_3, v_2)): P_{P,v_1}(v_3) = \varepsilon$

$q = \exists t_1 t_2 t_3 t_4 \ (R(t_1, t_2) \land S(t_2, t_3) \land S(t_4, t_3)):$

- $f_{R,t_1}(t_2) = R,$
- $f_{R,t_1}(t_1) = \varepsilon,$
- $f_{R,t_1}(t_3) = R \cdot S,$
- $f_{R,t_1}(t_4) = R,$
- $f_{S,t_4}(t_3) = S,$
- $f_{S,t_4}(t_4) = \varepsilon,$
- $f_{S,t_4}(t_2) = \varepsilon,$
- $f_{S,t_4}(t_1)$ is not defined
Query Rewriting for $DL$-$Lite^N_{horn}$ (3)

$$\varphi_2 = \bigwedge_{R(t,t') \in q} (t' \neq c_R)$$

if no tree witness exists then $t$ cannot be mapped to a non-ABox element

$$\varphi_3 = \bigwedge_{R(t,t') \in q} \left( \bigvee_{R(s,s') \in q} (s' = c_R) \rightarrow \bigwedge_{f_{R,t}(s) = \varepsilon} (s = t) \right)$$

if both $s$ and $t$ are labelled with $\varepsilon$ for role $R$ and $s'$ is mapped onto $c_R$, for $R(s,s') \in q$, then $s = t$

**NB.** in fact, $f_{R,t}(s) = \varepsilon$ induces an equivalence relation $\equiv^R_q$, and so, $|\varphi_3| = \mathcal{O}(|q|)$
Canonical Interpretation by FO Queries

regard the ABox as a relational instance and then
define (domain-independent) FO-queries $q^T_A(x)$ and $q^T_P(x, y)$ constructing $\mathcal{I}_K$

1. for each concept $C$, define queries $\text{exp}^T_C(x)$: e.g.,

$$\text{exp}^T_A(x) = A(x)$$
$$\text{exp}^T_C(x) = \text{exp}^T_C(x) \lor \bigvee_{C_1 \cap \ldots \cap C_n \subseteq C} \bigwedge_{1 \leq i \leq n} \text{exp}^T_{C_i}(x)$$

no more than $|\mathcal{T}|$ steps required

2. $q^T_P(x, y) = P(x, y) \lor (\text{gen}^T_P(x) \land (y = c_P)) \lor (\text{gen}^T_{P^{-}}(y) \land (x = c_{P^{-}}))$

3. $q^T_A(x) = \text{exp}^T_A(x) \land D(x)$, where $D(x) = \bigwedge_{c_R \in \mathcal{N}^T} ((x = c_R) \rightarrow \exists z \text{gen}^T_R(z))$

such queries can be implemented as materialised views (updates!)

Example: $h(x, y) = h(x, y) \lor$

$((\exists y' h(x, y') \lor S(x) \lor (x = c_t) \lor \exists y' t(y', x)) \land \neg \exists y' h(x, y') \land (y = c_h)) \lor$

$\exists z ((\exists y' t(z, y') \lor P(z) \lor (z = c_h) \lor \exists y' h(y', z)) \land \neg \exists y' t(z, y') \land (x = c_t) \land (y = c_h))$
Combining the two Rewriting Steps

- **polynomial** pure query rewriting for $DL-Lite^F_{core}$
- and even for $DL-Lite^N_{core}$ (if the aggregation function $\text{COUNT}$ is available)
- otherwise $|\text{exp}_{\geq kR}(x)| = \mathcal{O}(k^2)$, which is exponential in $\mathcal{T}$ if binary coding of $k$ is used

**Example:**

$$q(x) = (x \neq c_h) \land (x \neq c_t) \land$$

$$\left(t(x, y) \lor ((P(x) \lor \exists y' h(y', x)) \land \neg \exists y' t(x, y') \land (y = c_t)) \loright.$$  

$$\exists w \left( (S(w) \lor \exists y' t(y', w)) \land \neg \exists y' h(w, y') \land (x = c_h) \land (y = c_t) \right) \land$$

$$\left( h(y, z) \lor ((S(y) \lor \exists z' t(z', y)) \land \neg \exists z' h(y, z') \land (z = c_h)) \loright.$$  

$$\exists w' \left( (P(w') \lor \exists z' h(z', w')) \land \neg \exists z' t(w', z') \land (y = c_t) \land (z = c_h) \right)$$

which is equivalent to

$$q(x) = t(x, y) \lor P(x) \lor \exists y' h(y', x)$$
Other Applications of the Technique

- only exponential blowup for positive existential query answering in $DL-Lite^{(\mathcal{HN})}_{horn}$
- without the UNA, the technique is applicable to query answering in $DL-Lite^{(\mathcal{HF})}_{horn}$ (and this is P-complete for data complexity)
- experiments show that the approach is competitive with executing the original query over the data (the formulas $\varphi_1 - \varphi_3$ introduce additional selection conditions on top of the original query)

Open Questions

- is the exponential blowup unavoidable for role inclusions?
- is the exponential blowup unavoidable for positive existential queries?
- are there other fragments with pure polynomial rewriting?

more at http://www.dcs.bbk.ac.uk/~roman/