Reasoning about

DL-Lite ontologies

Michael Zakharyaschev

School of Computer Science and Information Systems

Birkbeck College, London

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

Joint work with A. Artale, D. Calvanese, R. Kontchakov, F. Wolter
What is the ontology language?

- What is the natural language? Italian, Latin, English, . . . ?
- What is the logic? Classical, intuitionistic, . . . , 1-order, 2-order, . . . ?
- What is the modal logic? $S_5$, $S_4$, $K$, . . . , 1-order, . . . ?

A naïve ‘ontology’ of ontology languages

- ‘Engineering:' OWL, RDF/S, OIL, DAML, DAML+OIL, . . .
- Mathematical logic:
  - Description logic: OWL-DL ($SHOIN(D)$, $SROIQ(D)$),
    OWL-Lite ($SHIF(D)$), . . ., $EL$ family, $DL$-Lite family
  - First-order logic: Prolog, Prolog + DL, . . .
  - Fuzzy logic: . . .
  - . . .

Open problem: design an ontology of ontology languages and reasoners
DL ‘islands of tractability’: $\mathcal{EL}$ family

- A number of huge life science ontologies (such as SNOMED CT, NCI) can be represented in (mild extensions of) DL $\mathcal{EL}$ with concepts

$$C ::= \top \mid A \mid C_1 \sqcap C_2 \mid \exists R.C$$

**Theorem** (Dresden) The problem ‘$\{C_i \sqsubseteq D_i \mid i \in I\} \models C \sqsubseteq D$?’ is in P.

- This theorem holds for the extensions of $\mathcal{EL}$ with:
  - Role inclusions $R_1 \circ R_2 \circ \cdots \circ R_n \sqsubseteq R$
  - Nominals
  - Concrete domains

Atomic negations, inverse roles, (non-qualified) number restrictions, functional roles $\leadsto$ EXPTime-complete reasoning
DL ‘islands of tractability’: DL-Lite family

- Reasoning about conceptual database schemas gives rise to an ‘orthogonal’ family of DLs

Translating into DL:

- \( \exists \text{passengers}. \top \sqsubseteq \text{Flight} \)
- \( \exists \text{passengers}^- \cdot \top \sqsubseteq \text{Passenger} \)
- \( \geq 2 \text{ origin}. \top \sqsubseteq \bot \)

Flight \( \sqsubseteq \geq 2 \text{ crew-members}. \top \)
CabinCrew \( \sqsubseteq \text{Person} \)
CabinCrew \( \sqsubseteq \text{Pilot} \sqcup \text{Steward} \)
Pilot \( \sqcap \text{Steward} \sqsubseteq \bot \)
...
DL ‘islands of tractability’: DL-Lite family (cont.)

1. **DL-Lite**\textsubscript{bool} (captures full ER) \hspace{1cm} \textbf{NP-complete}

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

   \[ B ::= \bot \mid A \mid \geq qR \]

   \[ C ::= B \mid \neg C \mid C_1 \sqcap C_2 \]

   TBox axioms \hspace{0.5cm} C_1 \sqsubseteq C_2, \hspace{0.5cm} \text{ABox assertions: } a : C, \ aRb

2. **DL-Lite**\textsubscript{horn} \hspace{1cm} \textbf{P-complete}

   TBox axioms \hspace{0.5cm} B_1 \sqcap \cdots \sqcap B_n \sqsubseteq B

3. **DL-Lite**\textsubscript{krom} (ER without covering constraints, e.g., \( B \sqsubseteq B_1 \sqcup B_2 \)) \hspace{1cm} \textbf{NLogSpace-complete}

   TBox axioms \hspace{0.5cm} B_1 \sqsubseteq B_2 \hspace{1cm} B_1 \sqsubseteq \neg B_2 \hspace{1cm} \neg B_1 \sqsubseteq B_2

   (subclass) \hspace{1cm} (disjointness)

\textbf{NB:} these complexity results are closely related to

complexity of reasoning in fragments of propositional logic

Bolzano 20.11.07
**Theorem** The satisfiability problem for $DL-Lite_{bool}$ knowledge bases is $NP$-complete.

**Proof** by embedding into the 1-variable fragment of first-order logic ($= S5$)

$DL-Lite_{bool}$ can only make statements about the whole **domain** and **range** of a role.

No fmp, but only **linear** number of (domain and range) witnesses needed!
**DL-Lite with role hierarchies**

\( DL-Lite^R = DL-Lite + \text{role inclusions (} R_1 \sqsubseteq R_2) \)

<table>
<thead>
<tr>
<th>Theorem</th>
<th>Satisfiability of ( DL-Lite_{horn}^R ) KBs is ExpTime-complete</th>
</tr>
</thead>
</table>

**Proof** by simulating alternating Turing machines with polynomial tapes:

- \( C(q, i, a, j, b) \rightarrow \ 'M \text{ is in state } q, \text{ scans cell } i \text{ with symbol } a, \text{ and cell } j \text{ contains } b' \)
- **Initialisation:** ABox \( \mathcal{A} = \{x : C(q_1, 1, a_1, j, a_j) \mid j \leq p(n)\}, \quad \vec{a} = (a_1, \ldots, a_n) \)
- **Transition** \( (q, a) \sim M (q', a', \rightarrow) \):
  \( C(q, i, a, i, a) \sqsubseteq \exists H_k(q', i + 1) \)
  \( C(q, i, a, i, a) \sqsubseteq \exists L_k(i, a') \), \( C(q, i, a, j, b) \sqsubseteq \exists L_k(j, b) \), \( \text{for all } j \neq i, \quad k = 0, 1 \)
- **Synchronisation:** \( \geq 2 T_k \sqsubseteq \perp \), \( L_k(j, b) \sqsubseteq T_k \), \( H_k(q, i) \sqsubseteq T_k \), \( \text{for } k = 0, 1 \)
  \( \exists L_k^-(i, a) \cap \exists L_k^-(j, b) \cap \exists H_k^-(q, i) \sqsubseteq C(q, i, a, j, b) \), \( \text{for all } (q, i, a, j, b) \)
- **Acceptance:** \( C(h, i, a, j, b) \sqsubseteq \exists S_\forall \), \( h \text{ the halting (existential) state} \)
  \( \geq 2 S_\forall \sqsubseteq \exists S_\exists \), \( \exists S_\exists \sqsubseteq \exists S_\forall \), \( S_\exists \sqsubseteq T \), \( S_\forall \sqsubseteq T \)
  \( T_k \sqsubseteq T \), \( \geq 2 T^- \sqsubseteq \perp \)

\( M \text{ accepts } \vec{a} \iff (T, \mathcal{A} \cup \{x : \neg \exists S_\exists\}) \text{ is not satisfiable} \)
DL-Lite with role hierarchies (cont.)

**Theorem**  
Satisfiability of $DL$-Lite$^R_{krom}$ \(\cap\) $DL$-Lite$^R_{horn}$ KBs is \textbf{ExpTime}-complete

**Proof** role inclusions + number restrictions can simulate $A \cap B \cap C \subseteq D$:

- $A \subseteq \exists R_A$, $B \subseteq \exists R_B$, $C \subseteq \exists R_C$
- $\exists R_A^- \subseteq \neg \exists R_B^-$, $\exists R_A^- \subseteq \neg \exists R_C^-$, $\exists R_B^- \subseteq \neg \exists R_C^-$
- $R_A \subseteq S$, $R_B \subseteq S$, $R_C \subseteq S$
- $\geq 3 S \subseteq D$

**Open problem:** What is the complexity of $DL$-Lite$^R_{krom}$ with functionality constraints?  
($DL$-Lite$^R_{horn}$ with functionality constraints is \textbf{ExpTime}-complete)
Ontology-based data access

Aim: to achieve **logical transparency** in accessing data
- hide from the user where and how data are stored
- present to the user a **conceptual view** of the data
- query the data sources through the conceptual model

**NB:** In fact, this is a form of Data Integration with a rich conceptual description as the global view.
Modularity problem

- If a set of new concepts names, roles, and axioms is added to $\mathcal{T}_1$, does it affect the meaning of a set $\Sigma$ of concepts and roles names defined in $\mathcal{T}_1$?
- When importing an ontology, do we change the meaning of its vocabulary?
- Do $\mathcal{T}_1$ and $\mathcal{T}_2$ say the same about $\Sigma$?
  
  Can I always import $\mathcal{T}_2$ instead of $\mathcal{T}_1$ into my ontology?

Changing the meaning of $\Sigma$ could mean:

- changing the classification of $\Sigma$-concepts
- changing the set of derivable implications between complex concepts over $\Sigma$
- changing the set of answers to (conjunctive) queries over $\Sigma$
- ...
Deductive conservative extensions

\( T_1 \cup T_2 \) is a **deductive conservative extension** of \( T_1 \) w.r.t. \( \Sigma \) if, for every general concept inclusion \( C_1 \sqsubseteq C_2 \) with \( \text{sig}(C_1 \sqsubseteq C_2) \subseteq \Sigma \),

\[
T_1 \models C_1 \sqsubseteq C_2 \quad \text{iff} \quad T_1 \cup T_2 \models C_1 \sqsubseteq C_2
\]

**Reasoning service:** is \( T_1 \cup T_2 \) a deductive conservative extension of \( T_1 \) w.r.t. \( \Sigma \)? If not, return ‘interesting’ counterexamples.

**Complexity of deciding deductive conservative extensions**

<table>
<thead>
<tr>
<th>Logic</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horn-fragment of propositional logic</td>
<td>coNP-complete</td>
</tr>
<tr>
<td>Propositional logic</td>
<td>( \Pi_2 )-complete (( \forall \bar{p} T_1 \rightarrow \exists \bar{q} T_2 ))</td>
</tr>
<tr>
<td>DL-Lite\text{horn}</td>
<td>coNP-complete</td>
</tr>
<tr>
<td>DL-Lite\text{bool}</td>
<td>( \Pi_2 )-complete (see next slide)</td>
</tr>
<tr>
<td>EL</td>
<td>EXP\text{TIME}-complete</td>
</tr>
<tr>
<td>ALC, ALCQI</td>
<td>2EXP\text{TIME}-complete</td>
</tr>
<tr>
<td>ALCQIO</td>
<td>undecidable</td>
</tr>
</tbody>
</table>
Semantic criterion of deductive conservativity for $\text{DL-Lite}_{\text{bool}}$

$\mathcal{T}_1 \cup \mathcal{T}_2$ is a deductive conservative extension of $\mathcal{T}_1$ w.r.t. $\Sigma$ iff every $\mathcal{T}_1$-consistent $\Sigma\mathcal{Q}$-type is also $\mathcal{T}_1 \cup \mathcal{T}_2$-consistent, where $\mathcal{Q}$ is the set of all numerical parameters from $\mathcal{T}_1 \cup \mathcal{T}_2$

Can be expressed as a QBF of the form $\forall \vec{p} \exists \vec{q} \varphi(\vec{p}, \vec{q})$ (and so is $\Pi^p_2$-complete):

$$\forall \vec{t}_\Sigma \left[ \text{Real}_{\mathcal{T}_1}(\vec{t}_\Sigma) \rightarrow \text{Real}_{\mathcal{T}_1 \cup \mathcal{T}_2}(\vec{t}_\Sigma) \right]$$

$$\text{Real}_T(\vec{t}_\Sigma) = \exists \vec{e}_T \left[ \text{Witns}_T(\vec{t}_\Sigma, \vec{e}_T) \land \bigwedge_{P_i \in \text{role}(T)} \left( \neg e_i \rightarrow \text{Witns}_{T, \geq 1 P_i}(\vec{t}_\Sigma, \vec{e}_T) \land \text{Witns}_{T, \geq 1 P_i^-}(\vec{t}_\Sigma, \vec{e}_T) \right) \right]$$

$$\text{Witns}_{T,C}(\vec{t}_\Sigma, \vec{e}_T) = \exists \vec{t}_T \left[ \bigwedge_{P_i \in \text{role}(T)} \left( e_i \rightarrow \neg(\geq 1 P_i) \land \neg(\geq 1 P_i^-) \right) \land C \land \bigwedge_{R_i \in \text{role}^\pm(T)} \left( (\geq q' R_i) \rightarrow (\geq q R_i) \right) \land \bigwedge_{C_1 \sqsubseteq C_2 \in \mathcal{T}} (C_1 \rightarrow C_2) \right]$$
**Query conservative extensions**

\( \mathcal{T}_1 \cup \mathcal{T}_2 \) is a **query conservative extension** of \( \mathcal{T}_1 \) w.r.t. \( \Sigma \) if,

- for every ABox \( \mathcal{A} \) with \( \text{sig}(\mathcal{A}) \subseteq \Sigma \),
- for every positive existential query \( q \) with \( \text{sig}(q) \subseteq \Sigma \),
- and every tuple \( \vec{a} \) of object names from \( \mathcal{A} \), we have:

\[
(\mathcal{T}_1, \mathcal{A}) \models q(\vec{a}) \iff (\mathcal{T}_1 \cup \mathcal{T}_2, \mathcal{A}) \models q(\vec{a}).
\]

Every query conservative extension is a deductive conservative extension, but not the other way round:

Let \( \mathcal{T}_1 = \emptyset, \mathcal{T}_2 = \{ A \sqsubseteq \exists P, \exists P^- \sqsubseteq B \}, \Sigma = \{ A, B \} \).

\( \mathcal{T}_2 \) is deductive but not query conservative extension of \( \mathcal{T}_1 \) w.r.t. \( \Sigma \) because

for \( \mathcal{A} = \{ A(a) \}, \quad q = \exists y B(y), \quad (\mathcal{T}_1, \mathcal{A}) \not\models q, \quad \text{but} \quad (\mathcal{T}_2, \mathcal{A}) \models q \).
Deciding query conservativity

- Deciding query conservativity is \textbf{coNP-complete} for $DL-Lite_{horn}$ and $\Pi^p_2$-complete for $DL-Lite_{bool}$

$\mathcal{T}_1 \cup \mathcal{T}_2$ is a query conservative extension of $\mathcal{T}_1$ w.r.t. $\Sigma$ iff every precisely $\mathcal{T}_1$-realisable set of $\Sigma Q$-types is also precisely $\mathcal{T}_1 \cup \mathcal{T}_2$-realisable.

Can also be expressed as a \textbf{QBF} of the form $\forall \vec{p} \exists \vec{q} \varphi(\vec{p}, \vec{q})$

(\text{where } N \text{ is the number of roles in } \mathcal{T}_1 \cup \mathcal{T}_2)

\[
\forall \vec{t}_\Sigma^0, \ldots, \vec{t}_\Sigma^N \left[ P-\text{Real}^N_{\mathcal{T}_1}(\vec{t}_\Sigma^0, \ldots, \vec{t}_\Sigma^N) \rightarrow P-\text{Real}^N_{\mathcal{T}_1 \cup \mathcal{T}_2}(\vec{t}_\Sigma^0, \ldots, \vec{t}_\Sigma^N) \right]
\]

$P-\text{Real}^N_{\mathcal{T}}(\vec{t}_\Sigma^0, \ldots, \vec{t}_\Sigma^N) = \exists \vec{e}_\mathcal{T} \left[ \bigwedge_{j=0}^{N} \text{Witness}_{\mathcal{T}}(\vec{t}_\Sigma^j, \vec{e}_\mathcal{T}) \right. \wedge \\
\left. \bigwedge_{P_i \in \text{role}(\mathcal{T})} \left( \neg e_i \rightarrow \bigvee_{j=0}^{N} \text{Witness}_{\mathcal{T}, \geq 1} P_i(\vec{t}_\Sigma^j, \vec{e}_\mathcal{T}) \wedge \bigvee_{j=0}^{N} \text{Witness}_{\mathcal{T}, \geq 1} P_i^{-}(\vec{t}_\Sigma^j, \vec{e}_\mathcal{T}) \right) \right]\]
Strong deductive/query conservative extensions

Do $\mathcal{T}_1$ and $\mathcal{T}_1 \cup \mathcal{T}_2$ say the same about $\Sigma$ whenever they are imported into an ontology?

$\mathcal{T}_1 \cup \mathcal{T}_2$ is a strong deductive/query conservative extension of $\mathcal{T}_1$ w.r.t. $\Sigma$ if,

for every TBox $\mathcal{T}$ with $\text{sig}(\mathcal{T}) \cap \text{sig}(\mathcal{T}_1 \cup \mathcal{T}_2) \subseteq \Sigma$,

$\mathcal{T}_1 \cup \mathcal{T}_2 \cup \mathcal{T}$ is a deductive/query conservative extension of $\mathcal{T}_1 \cup \mathcal{T}$ w.r.t. $\Sigma$.

Not every deductive conservative extension is a strong one:

Let $\mathcal{T}_1 = \emptyset, \mathcal{T}_2 = \{A \sqsubseteq \exists R, A \sqcap \exists R^\bot \sqsubseteq \bot\}, \Sigma = \{A\}$

Then $\mathcal{T}_1 \cup \mathcal{T}_2$ is a deductive conservative extension of $\mathcal{T}_1$ w.r.t. $\Sigma$.

But for $\mathcal{T} = \{\top \sqsubseteq A\}$,

$\mathcal{T}_1 \cup \mathcal{T}_2 \cup \mathcal{T} \models \top \sqsubseteq \bot, \quad \mathcal{T}_1 \cup \mathcal{T} \not\models \top \sqsubseteq \bot$
Final results

\[ DL-Lite_{\text{horn}}: \text{deductive} \preceq \text{query} \preceq \text{strong deductive} \equiv \text{strong query} \]
\[ DL-Lite_{\text{bool}}: \text{deductive} \preceq \text{query} \equiv \text{strong deductive} \equiv \text{strong query} \]

- All the decision problems for \( DL-Lite_{\text{horn}} \) are \text{coNP-complete}
- All the decision problems for \( DL-Lite_{\text{bool}} \) are \( \Pi_2^p \)-complete
Further work

- Now experimenting with QBF + SAT solvers
- Experiments required for ‘interesting counterexamples’ to conservativity
- Complexity of model conservativity still open
- Applications to data integration
- Add nominals, concrete domains to DL-Lite, transitive roles?
- What about finite model reasoning?