Query Answering with Transitive & Linear-Ordered Data

Antoine Amarilli¹, Michael Benedikt², Pierre Bourhis³, Michael Vanden Boom²

Query Answering (or query entailment)

- Initial set of facts \( F \) (or A-box)
- Constraints \( \Sigma \): logical rules (or T-box)
- Boolean query \( Q \): CQ or UCQ

**QA problem:**

Does \( F \land \Sigma \) entail \( Q \)?

Equivalently:
- Is \( Q \) certain given \( F \) and \( \Sigma \)?
- Is \( F \land \Sigma \land \neg Q \) unsatisfiable?

**Our goal:** identify and study constraint languages for which QA is decidable, even when some relations are restricted to be transitive or to be linear orders.

Our approach

Fix relational signature \( \sigma := \sigma_\Delta \sqcup \sigma_\bullet \) where
- \( \sigma_\bullet \): distinguished binary relations
- \( \sigma_\Delta \): base relations

We consider query answering with three different special interpretations for the distinguished relations:

- QAtr: each \( R \in \sigma_\Delta \) is transitively closed
- QAtc: each \( R^+ \in \sigma_\Delta \) is the transitive closure of \( R \in \sigma_\Delta \)
- QAlin: each \( R \in \sigma_\Delta \) is a linear order

We introduce base-frontier-guarded and base-covered constraint languages that disallow the use of distinguished relations as guards.

Main results

- QAtr & QAtc are decidable for BaseGF
  - (undecidable for FGTGDs).
- QAlin is decidable for BaseCovGF
  - (undecidable for BaseFGTGDs).

Complexity

<table>
<thead>
<tr>
<th></th>
<th>QAtr combined</th>
<th>QAtr combined</th>
<th>QAtr combined</th>
</tr>
</thead>
<tbody>
<tr>
<td>BaseCovGF</td>
<td>( \text{coNP-c} )</td>
<td>( 2\text{EXP-c} )</td>
<td>( 2\text{EXP-c} )</td>
</tr>
<tr>
<td>BaseFGTGDs</td>
<td>in ( \text{coNP} )</td>
<td>( 2\text{EXP-c} )</td>
<td>( 2\text{EXP-c} )</td>
</tr>
</tbody>
</table>

Proof ideas

**Key property:**

For \( \varphi \) in GNF, if \( \varphi \) is a tree-like witness: a set of facts satisfying \( \varphi \) has a tree decomposition of bounded tree-width.

For QAtr & QAtc with base-frontier-guarded constraints \( \Sigma \): reduce to tree automaton emptiness test.

For BaseCovGF, there are tree-like witnesses even when each distinguished relation is required to be the transitive closure of some base relation.

Hence it suffices to construct a tree automaton \( A \) that runs on encodings of tree-like sets of facts and checks \( F \land \Sigma \land \neg Q \).

For QAtr & QAlin with base-covered constraints \( \Sigma \): reduce to traditional QA with GNF \( \Sigma \).

Cannot axiomatize transitivity or totality using GNF, but can approximate using \( \Sigma \) in GNF. Key technical result shows that a tree-like approximate witness can be extended to an actual witness respecting special interpretations for \( \sigma_\Delta \) relations.

Frontier-guarded TGDs (FGTGDs)

\[ \forall x \, y \, \varphi(x, y) \land G(x) \rightarrow \exists z \, \psi(x, z) \]

guarded negation fragment (GNF)

\[ \varphi(x, y) \land G(x) \rightarrow \exists z \, \psi(x, z) \]

tuple-generating dependencies (TGDs)

\[ \forall x \, y \, \varphi(x, y) \rightarrow \exists z \, \psi(x, z) \]

Atom using variables \( x \) is called a guard for \( x \)

BaseCovGF

Generalization of BaseCovGF

BaseGF

Frontier-one TGDs

BaseCovGF

BaseGF

Frontier-guarded TGDs (FGTGDs)

\[ \forall x \, y \, \varphi(x, y) \land G(x) \rightarrow \exists z \, \psi(x, z) \]

Guarded Negation Fragment (GNF)

\[ \varphi(x, y) \land G(x) \rightarrow \exists z \, \psi(x, z) \]

Tuple-generating dependencies (TGDs)

\[ \forall x \, y \, \varphi(x, y) \rightarrow \exists z \, \psi(x, z) \]

Guarded negation fragment (GNF)

\[ \varphi(x, y) \land G(x) \rightarrow \exists z \, \psi(x, z) \]

tuple-generating dependencies (TGDs)

\[ \forall x \, y \, \varphi(x, y) \rightarrow \exists z \, \psi(x, z) \]