Does Query Evaluation Tractability Help Query Containment?

Pablo Barceló    Miguel Romero    Moshe Y. Vardi

Univ. of Chile    Univ. of Chile    Rice University

June 24, PODS 2014, Utah, USA
Containment of Datalog

Datalog Programs

- Unions of Conjunctive Queries + Recursion.
- Ontological reasoning, graph databases, distributed computing,
- Set of rules of the form: \( S(\overline{x}) \leftarrow R_1(\overline{y}_1), \ldots, R_m(\overline{y}_m) \).

Notation:

- **IDB predicates**: in the head of some rule.
- **EDB predicates**: only in the body of rules.
  (input database schema)
- **Ans predicate**: special IDB predicate.
PROBLEM: Containment of Datalog

INPUT: Datalog programs \( \Pi \) and \( \Pi' \)

QUESTION: \( \Pi \subseteq \Pi' \)?

\( \Pi \subseteq \Pi' \) if

\( \Pi(D) \subseteq \Pi'(D) \) for every DB \( D \)
Containment of Datalog

Containment of Datalog is **undecidable** (Shmueli 1993). Decidable restrictions:

1. **Monadic Datalog**: All IDB predicates have arity 1  
   (Cosmadakis, Gaifman, Kanellakis, Vardi 1988)

2. **Datalog in UCQs/UC2RPQs**  
   (Chaudhuri, Vardi 1992; Calvanese, De Giacomo, Vardi 2003)

3. **(Nested) Monadically Defined Queries**  
   (Rudolph, Krotzsch 2013)

In this work:

Datalog in UCQs/UC2RPQs
Datalog in UCQs

- Containment of Datalog in UCQs is $2\text{EXPTIME}$-complete (Chaudhuri, Vardi 1992).
- Several restrictions that leads to better complexity bounds.
  - Conditions on the Datalog program, not in the UCQ. (Chaudhuri, Vardi 1994; Benedikt, Bourhis, Senellart 2012)

Our Goal:

Obtain better complexity bounds for containment of Datalog in UCQs by
- restricting the UCQs,
- while retaining the full expressive power of Datalog programs.
- Is it possible to obtain $\text{EXPTIME}$ upper bounds?
  - the norm for several static analysis and verification tasks.
Outline

Motivation

Restrictions on UCQs

Containment of Datalog in UCQs

Containment of Datalog in UC2RPQs

Open Questions
Outline

Motivation

Restrictions on UCQs

Containment of Datalog in UCQs

Containment of Datalog in UC2RPQs

Open Questions
Restrictions of UCQs

Datalog program $\Pi$, UCQ $\Theta$.

$\Pi \subseteq \Theta$ iff $\Theta$ is true in every model of $\Pi$.

Strong connection between Containment of Datalog in UCQs and Evaluation of UCQs.

Idea:

Consider tractable restrictions for UCQs.

- Restrictions of UCQs, whose evaluation problem is in PTIME.
Several tractable restrictions for CQs:
- Acyclic (Yannakakis 1981)
- Bounded Treewidth (Chekuri, Rajaraman 2000)
- Bounded Hypertree width (Gottlob, Leone, Scarcello 2002)

We can extend these restrictions to UCQs:
A UCQ is acyclic if each disjunct of the UCQ is acyclic.
Main Goal: Datalog in UCQs

Let $\mathcal{C}$ be a class of UCQs.

Problem: CONT(Datalog, $\mathcal{C}$)

INPUT: Datalog Program $\Pi$ and UCQ $\Theta \in \mathcal{C}$.

QUESTION: Is $\Pi \subseteq \Theta$?

Main Goal:

- Study the complexity of $\text{CONT}(\text{Datalog}, \mathcal{C})$ for tractable classes $\mathcal{C}$.
- Is $\text{CONT}(\text{Datalog}, \mathcal{C})$ in EXPTIME, for some tractable class $\mathcal{C}$?
Contributions

1. Complexity analysis of $\text{CONT(Datalog, } \mathcal{C} \text{)}$, for tractable restrictions $\mathcal{C}$.
2. We identify natural classes $\mathcal{C}$, for which $\text{CONT(Datalog, } \mathcal{C} \text{)}$ is in $\text{EXPTIME}$. 
3. We also analyze UC2RPQs.
Outline

Motivation

Restrictions on UCQs

Containment of Datalog in UCQs

Containment of Datalog in UC2RPQs

Open Questions
Acyclic UCQs

A CQ is **acyclic** if it can be decomposed in a **tree** that “preserves” its structure.

Formally:
A CQ \( \theta(\bar{x}) \leftarrow R_1(\bar{y}_1), \ldots, R_m(\bar{y}_m) \) is acyclic if it has a join tree, that is, a tree \( T \) such that

1. The nodes of \( T \) are the **atoms** of \( \theta \).
2. For each variable \( x \) in \( \theta \) the set \( \{ t \in T \mid x \in t \} \) is **connected**.
Acyclic UCQs: Examples

The CQ

\[ \theta_1() \leftarrow E(x, y), R(y, w, z), E(z, t) \]

is acyclic.

Join tree:
Acyclic UCQs: Examples

The CQ

\[ \theta_2() \leftarrow E(x, y), R(y, w, z), E(z, x) \]

is not acyclic.
Acyclic UCQs: Examples

The CQ

\[ \theta_3() \leftarrow E(x,y), R(y,w,z), E(z,x), R(x,y,z). \]

is acyclic.

Join tree:

```
R(x, y, z)
  /    |
E(x, y) R(y, w, z) E(z, x)
```
Acyclic UCQs

- **AC**: class of acyclic UCQs.
- Is \( \text{CONT}(\text{Datalog}, \text{AC}) \) in \text{EXPTIME}?

**Theorem:**
\( \text{CONT}(\text{Datalog}, \text{AC}) \) is \text{2EXPTIME}-complete.
Bounded hypertree width UCQs

- $\text{HW}(k)$: class of UCQs of hypertree width at most $k$.
- $\text{AC}=\text{HW}(1)$.

**Corollary:**
For each $k \geq 1$, $\text{CONT}(\text{Datalog}, \text{HW}(k))$ is $2\text{EXPTIME}$-complete.

**EXPTIME** upper bounds?
The $AC_k$ Hierarchy

For $k \geq 1$,
$AC_k$: the class of acyclic UCQs such that in each disjunct
- the number of common variables
  between any pair of distinct atoms is at most $k$. 
The AC$_k$ Hierarchy: Examples

- The CQ

\[ \theta_1() \leftarrow E(x, y), R(y, w, z), E(z, t) \]

is in AC$_1$.

- The CQ

\[ \theta_3() \leftarrow E(x, y), R(y, w, z), E(z, x), R(x, y, z) \]

is in AC$_2$. 
The AC$_k$ Hierarchy

Theorem:
For each $k \geq 1$, CONT(Datalog, AC$_k$) is EXPTIME-complete.

- Proof based on two-way alternating tree automata.
A HW(2)\(_k\) Hierarchy?

\textbf{HW(2)}\(_k\): class of UCQs in HW(2) with at most \(k\) common variables between atoms.

\textbf{Proposition:}

\text{\textit{\textbf{CONT(Datalog, HW(2)\(_1\)}} is 2EXPTIME-complete.}
Map of results

<table>
<thead>
<tr>
<th>AC=HW(1)</th>
<th>HW(2)</th>
<th>HW(3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2EXPTIME</td>
<td>2EXPTIME</td>
<td>2EXPTIME</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AC3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXPTIME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AC2</td>
<td>HW(2)</td>
<td></td>
</tr>
<tr>
<td>EXPTIME</td>
<td>2EXPTIME</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>HW(2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2EXPTIME</td>
<td></td>
</tr>
<tr>
<td>AC1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXPTIME</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>HW(3)</td>
</tr>
<tr>
<td></td>
<td>2EXPTIME</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Bounded treewidth: map of results
1. Traditional tractable restrictions for UCQs do not help to reduce the complexity of \textsc{CONT(Datalog, UCQs)}.

2. Restricting the UCQ to be acyclic and have a bounded number of common variables between atoms reduces the complexity to \textsc{EXPTIME}.
Outline

Motivation

Restrictions on UCQs

Containment of Datalog in UCQs

Containment of Datalog in UC2RPQs

Open Questions
Graph Databases and UC2RPQs

**Graph Database:** Directed graph whose edges are labeled over a finite alphabet $\Sigma$.

**C2RPQ:**

$$\gamma(\bar{x}) \leftarrow L_1(y_1, y'_1), \ldots, L_m(y_m, y'_m)$$

Each $L_i$ is a 2RPQ

- regular expression over $\Sigma$.

**UC2RPQ:** Union of C2RPQ.
Datalog in UC2RPQs

- Containment of Datalog in UC2RPQs is 2EXPTIME-complete (Calvanese, De Giacomo, Vardi 2003).
- What is the complexity of $\text{CONT}(\text{Datalog}, C)$, for tractable classes $C$ of UC2RPQs.
Summary of results

1. Traditional tractable restrictions for UC2RPQs (Acyclicity and Bounded treewidth) do not help to reduce the complexity of CONT(Datalog, UC2RPQs).

2. We can define a hierarchy (ACR$_k$ hierarchy) inside acyclic UC2RPQs, by bounding the number of parallel atoms.
   - For each level of this hierarchy, the problem is EXPTIME.
Outline

Motivation

Restrictions on UCQs

Containment of Datalog in UCQs

Containment of Datalog in UC2RPQs

Open Questions
Open Questions

1. Are the classes $AC_k$ and $ACR_k$ useful in other contexts?
2. $\text{CONT}(\text{monadic Datalog, UCQs})$ is $2\text{EXPTIME}$-complete (Benedikt, Bourhis, Senellart 2012)
   - What is the complexity of $\text{CONT}(\text{monadic Datalog, } C)$, for tractable restrictions $C$ of UCQs?