Query Containment under Bag and Bag-Set Semantics

Foto Afrati, Matthew Damigos and Manolis Gergatsoulis

Information Processing Letters (IPL) 2010
Query Containment Problem

- \( Q_1, Q_2 \) over schema \( S \).
- \( \mathcal{D} \) is a database instance of \( S \).

For every database instance \( \mathcal{D} \):

- \( Q_2 \sqsubseteq Q_1 \)
- \( Q_1(\mathcal{D}) \)
- \( Q_2(\mathcal{D}) \)
Motivation - Previous Work

- Related problems:
  - Query rewriting using views.
  - Information integration.
  - Query optimization.
  - ...
- The query containment problem under set semantics has been extensively investigated
  - Most of the queries’ classes give decidable results.
- SQL semantics: manipulation of duplicate tuples.
- The query containment problem for conjunctive queries under both bag and bag-set semantics remains open for more than a decade.
  - Most of the super-classes give undecidable results.
Conjunctive queries

- **Conjunctive query (CQ, for short):**

  \[
  Q : \quad q(\overline{X}) :\leftarrow g_1(\overline{X}_1), \ldots, g_n(\overline{X}_n),
  \]

  **Head** \quad **Body**

  Subgoal \quad Subgoal

  - Select-Project-Join SQL queries with equality comparisons.
  - Distinguished variables: \( Vars(\overline{X}) \).
  - Safe CQ: every variable in \( Vars(\text{head}(Q)) \) appears in the body of \( Q \).

  - (True) valuation from \( Q \) to \( D \):
    1. every variable of \( Q \) is valuated by a constant appear in \( D \).
    2. If every valuated subgoal appears in database instance \( D \) then the valuated head is in the answer of \( Q (Q(D)) \).
Query Containment under Bag and Bag-Set Semantics

Foto Afrati, Matthew Damigos and Manolis Gergatsoulis

Semantics

Set-valued DB $\mathcal{D}$

<table>
<thead>
<tr>
<th>link</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>a</td>
<td>c</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>c</td>
<td>c</td>
</tr>
</tbody>
</table>

Bag-valued DB $\mathcal{D}$

<table>
<thead>
<tr>
<th>link</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>c</td>
<td>c</td>
</tr>
</tbody>
</table>

VS.

Query:

$Q : q(X) : \neg \text{link}(X,Y), \text{link}(Y,Y)$
### Semantics

**Set-valued DB** $\mathcal{D}$

<table>
<thead>
<tr>
<th>link</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>a</td>
<td>c</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>c</td>
<td>c</td>
</tr>
</tbody>
</table>

**Bag-valued DB** $\mathcal{D}$

<table>
<thead>
<tr>
<th>link</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>c</td>
<td>c</td>
</tr>
</tbody>
</table>

---

**Query: $Q : q(X) : \neg \text{link}(X, Y), \text{link}(Y, Y)$**

**$Q(\mathcal{D})$:**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td></td>
</tr>
</tbody>
</table>

---

1. **Set semantics:** Relations are sets (using DISTINCT in SQL)
## Query Containment under Bag and Bag-Set Semantics

**Foto Afrati, Matthew Damigos and Manolis Gergatsoulis**

### Semantics

<table>
<thead>
<tr>
<th>Set-valued DB $\mathcal{D}$</th>
<th>Bag-valued DB $\mathcal{D}$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>link</strong></td>
<td><strong>link</strong></td>
</tr>
<tr>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>a</td>
<td>c</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>c</td>
<td>c</td>
</tr>
</tbody>
</table>

**VS.**

**Query:**

$$Q : \text{q}(X) : \neg \text{link}(X, Y), \text{link}(Y, Y)$$

**Bag-operators:** treat duplicates as distinct tuples

1. **Set semantics:** Relations are sets (using DISTINCT in SQL)
2. **Bag semantics:** Relations are bags (SQL semantics)

---

**Bag-operators:**

1. **Set operators**
2. **Bag operators**

---

**Bag-operators:**

1. Set semantics: Relations are sets (using DISTINCT in SQL)
2. Bag semantics: Relations are bags (SQL semantics)
Query Containment under Bag and Bag-Set Semantics

Foto Afrati, Matthew Damigos and Manolis Gergatsoulis

Set-valued DB $\mathcal{D}$

<table>
<thead>
<tr>
<th>link</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
</tr>
<tr>
<td>b</td>
</tr>
<tr>
<td>a</td>
</tr>
<tr>
<td>c</td>
</tr>
<tr>
<td>b</td>
</tr>
<tr>
<td>b</td>
</tr>
<tr>
<td>c</td>
</tr>
<tr>
<td>c</td>
</tr>
</tbody>
</table>

Bag-valued DB $\mathcal{D}$

<table>
<thead>
<tr>
<th>link</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
</tr>
<tr>
<td>b</td>
</tr>
<tr>
<td>a</td>
</tr>
<tr>
<td>b</td>
</tr>
<tr>
<td>b</td>
</tr>
<tr>
<td>c</td>
</tr>
<tr>
<td>c</td>
</tr>
</tbody>
</table>

VS.

Query:

$Q : q(X) : \neg \text{link}(X, Y), \text{link}(Y, Y)$

(1) set-operators

$Q(\mathcal{D})$

| a |
| b |
| c |

(2) bag-operators

$Q(\mathcal{D})$

| a |
| a |
| b |
| c |

(3) bag-operators

Bag-operators: treat duplicates as distinct tuples

(1): Set semantics: Relations are sets (using DISTINCT in SQL)

(2): Bag semantics: Relations are bags (SQL semantics)

(3): Bag-Set semantics: Relations are sets (normalized DB + SQL)
Bag-Set Semantics - Projection causes duplicate tuples

- Bag-set semantics: Set-valued database + Bag-operators
  - Each tuple is unique in a relation
Bag-Set Semantics - Projection causes duplicate tuples

- Bag-set semantics: Set-valued database + Bag-operators
  - Each tuple is unique in a relation
- Queries: Select, Join, Cartesian Product, Projection
- CQ \( Q \) without projection ⇔ the answer of \( Q \) is set.
  - Afrati, Damigos, Gergatsoulis IPL 2009

\[
\begin{align*}
Q_1(D) := & \quad q(X) : \neg \text{link}(X, Y), \text{link}(Y, Y) \\
Q_2(D) := & \quad q(X, Y) : \neg \text{link}(X, Y), \text{link}(Y, Y)
\end{align*}
\]
Query Containment under Bag and Bag-Set Semantics

Foto Afrati, Matthew Damigos and Manolis Gergatsoulis

Formal Definition

Definition

$Q_2 \subseteq Q_1$, if for every database instance $\mathcal{D}$ of $S$, we have that $Q_2(\mathcal{D}) \subseteq Q_1(\mathcal{D})$.

<table>
<thead>
<tr>
<th>Semantics</th>
<th>$\sqsubseteq$</th>
<th>$\mathcal{D}$</th>
<th>$\subseteq$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set</td>
<td>$\sqsubseteq_s$</td>
<td>set-valued</td>
<td>$\subseteq_s$</td>
</tr>
<tr>
<td>Bag</td>
<td>$\sqsubseteq_b$</td>
<td>bag-valued</td>
<td>$\subseteq_b$</td>
</tr>
<tr>
<td>Bag-Set</td>
<td>$\sqsubseteq_{bs}$</td>
<td>set-valued</td>
<td>$\subseteq_{bs}$</td>
</tr>
</tbody>
</table>

- $Q_2$ **bag-contained** in $Q_1$ $\Rightarrow$ $Q_2$ **bag-set-contained** in $Q_1$
- $Q_2$ **bag-set-contained** in $Q_1$ $\Rightarrow$ $Q_2$ **set-contained** in $Q_1$
\[
\sqsubseteq_{bs} \not\supseteq \sqsubseteq_{b} \quad \text{and} \quad \sqsubseteq_{s} \not\supseteq \sqsubseteq_{bs}
\]

- Relation “path” stores paths of length 2.

**Queries**

\[
Q_1 : q(X) :- path(X, Y) \\
Q_2 : q(X) :- path(X, Y), path(Y, Z) \\
Q_3 : q(X) :- path(X, Y), path(Y, Y)
\]

**Database** \(D\)

**Answers of Queries**

- \(Q_1(D) = \{1, 2, 3, 3\}\)
- \(Q_2(D) = \{1, 1, 2, 3, 3\}\)
- \(Q_3(D) = \{1, 2, 3\}\)
\( \sqsubseteq_{bs} \not\Rightarrow \sqsubseteq_{b} \) and \( \sqsubseteq_{s} \not\Rightarrow \sqsubseteq_{bs} \)

- Relation “path” stores paths of length 2.

**Queries**

\begin{align*}
Q_1 : q(X) & :- \text{path}(X, Y) \\
Q_2 : q(X) & :- \text{path}(X, Y), \text{path}(Y, Z) \\
Q_3 : q(X) & :- \text{path}(X, Y), \text{path}(Y, Y)
\end{align*}

**Database** \( \mathcal{D} \)

**Answers of Queries**

\begin{enumerate}
  \item \( Q_1(\mathcal{D}) = \{1, 2, 3, 3, 2, 3, 4, 4\} \)
  \item \( Q_2(\mathcal{D}) = \{1, 1, 2, 3, 3, 1, 2, 2, 3, 3, 3, 3, 4, 4\} \)
  \item \( Q_3(\mathcal{D}) = \{1, 2, 3, 1, 2, 3, 3, 3, 3, 4, 4\} \)
\end{enumerate}
Query Containment under Bag and Bag-Set Semantics

Foto Afrati, Matthew Damigos and Manolis Gergatsoulis

Containment Mapping

- Containment mapping from $Q_1$ to $Q_2$: Every distinct tuple appears in $Q_2(D)$ also appears in $Q_1(D)$
- each valuation over $Q_2 \Rightarrow$ at least one valuation over $Q_1$

\[ \mu_1 : Q_1 \rightarrow Q_2 \]

\[
Q_1 : \begin{align*}
q(a) & \leftarrow \text{blue}(a, b), \ \text{red}(b, c) \ \text{red}(b, c) \\
q(X) & :- \text{blue}(X, Y), \ \text{red}(Y, Z) \ \text{red}(W, Z)
\end{align*}
\]

\[
Q_2 : \begin{align*}
q(a) & \leftarrow \text{blue}(a, b), \ \text{red}(b, c) \ \text{red}(b, c) \\
q(A) & :- \text{blue}(A, B), \ \text{red}(B, C), \ \text{red}(B, D)
\end{align*}
\]
Containment Mapping

- Containment mapping from $Q_1$ to $Q_2$: Every **distinct** tuple appears in $Q_2(D)$ also appears in $Q_1(D)$
  - each valuation over $Q_2 \Rightarrow$ at least one valuation over $Q_1$

$$Q_1 : \ X \rightarrow Y \rightarrow W \rightarrow Z \quad \text{and} \quad Q_2 : \ A \rightarrow B \rightarrow C \rightarrow D$$

$$q(a) \leftarrow blue(a, b), \ red(b, d) \quad red(b, d)$$

$$Q_1 : q(X) :- blue(X, Y), \ red(Y, Z) \quad red(W, Z)$$

$$Q_2 : q(A) :- blue(A, B), \ red(B, C), \ red(B, D)$$

$$q(a) \leftarrow blue(a, b), \ red(b, c) \quad red(b, c)$$

$$\mu_1 : Q_1 \rightarrow Q_2$$

$$\mu_2 : Q_1 \rightarrow Q_2$$
Containment Mapping

- Containment mapping from $Q_1$ to $Q_2$: Every **distinct** tuple appears in $Q_2(D)$ also appears in $Q_1(D)$
  - each valuation over $Q_2 \Rightarrow$ at least one valuation over $Q_1$
- What about multiplicity of each tuple (under bag(-set) semantics)?
  - Many valuations over $Q_2 \Rightarrow$ same valuation over $Q_1$

\[
q(a) \leftarrow \text{blue}(a, b), \quad \text{red}(b, d) \quad \text{red}(b, d)
\]

\[
Q_1 : q(X) :- \text{blue}(X, Y), \quad \text{red}(Y, Z) \quad \text{red}(W, Z)
\]

\[
Q_2 : q(A) :- \text{blue}(A, B), \quad \text{red}(B, C), \quad \text{red}(B, D)
\]

\[
q(a) \leftarrow \text{blue}(a, b), \quad \text{red}(b, c) \quad \text{red}(b, d)
\]

\[
q(a) \leftarrow \text{blue}(a, b), \quad \text{red}(b, d) \quad \text{red}(b, c)
\]
**Containment Mapping**

- Containment mapping from $Q_1$ to $Q_2$: Every **distinct** tuple appears in $Q_2(D)$ also appears in $Q_1(D)$
  - each valuation over $Q_2 \Rightarrow$ at least one valuation over $Q_1$
- What about multiplicity of each tuple (under bag(-set) semantics)?
  - Many valuations over $Q_2 \Rightarrow$ same valuation over $Q_1$
  - $\Rightarrow Q_2 \not\sqsubseteq_{bs} Q_1 \Rightarrow Q_2 \not\sqsubseteq_{b} Q_1$

$q(a) \leftarrow \text{blue}(a, b), \ \text{red}(b, d) \ \text{red}(b, d)$
$q(a) \leftarrow \text{blue}(a, b), \ \text{red}(b, c) \ \text{red}(b, c)$

$Q_1 : q(X) :- \text{blue}(X, Y), \ \text{red}(Y, Z) \ \text{red}(W, Z)$

$Q_2 : q(A) :- \text{blue}(A, B), \ \text{red}(B, C), \ \text{red}(B, D)$
$q(a) \leftarrow \text{blue}(a, b), \ \text{red}(b, c) \ \text{red}(b, d)$
$q(a) \leftarrow \text{blue}(a, b), \ \text{red}(b, d) \ \text{red}(b, c)$
$q(a) \leftarrow \text{blue}(a, b), \ \text{red}(b, c) \ \text{red}(b, c)$
$q(a) \leftarrow \text{blue}(a, b), \ \text{red}(b, d) \ \text{red}(b, d)$
Query containment and Containment mapping

$Q_1: q(X) :- edge(X, Y), edge(Y, Z), edge(Y, W)$
$Q_2: q(A) :- edge(A, B), edge(B, B), edge(B, C)$

$Q_1 : \begin{array}{c}
\xrightarrow{X} \xrightarrow{Y} \xrightarrow{Z} \xrightarrow{W}
\end{array}$

$Q_2 : \begin{array}{c}
\xrightarrow{A} \xrightarrow{B} \xrightarrow{C}
\end{array}$
Query containment and Containment mapping

\[ Q_1 : q(X) :- \text{edge}(X, Y), \text{edge}(Y, Z), \text{edge}(Y, W) \]
\[ Q_2 : q(A) :- \text{edge}(A, B), \text{edge}(B, B), \text{edge}(B, C) \]

- \( \mu_1 : Q_1 \rightarrow Q_2 \)

\( \mu_1 \): containment mapping from \( Q_1 \) to \( Q_2 \).

- \( Q_2 \sqsubseteq_s Q_1 \iff \text{containment mapping from } Q_1 \text{ to } Q_2 \).
  (Chandra-Merlin, STOC 1977)
Query containment and Containment mapping

\[ Q_1 : q(X) :- \text{edge}(X, Y), \text{edge}(Y, Z), \text{edge}(Y, W) \]
\[ Q_2 : q(A) :- \text{edge}(A, B), \text{edge}(B, B), \text{edge}(B, C) \]

- \( \mu_1 \): containment mapping from \( Q_1 \) to \( Q_2 \).
  - \( Q_2 \sqsubseteq_s Q_1 \Leftrightarrow \) containment mapping from \( Q_1 \) to \( Q_2 \).
    (Chandra-Merlin, STOC 1977)
- \( \mu_2 \): variables-onto containment mapping from \( Q_1 \) to \( Q_2 \).
  - variables-onto containment mapping from \( Q_1 \) to \( Q_2 \) \( \Rightarrow \)
    \( Q_2 \sqsubseteq_{bs} Q_1 \).
    (Chaudhuri-Vardi, PODS 1993)
Query containment and Containment mapping

\[ Q_1 : q(X) :- \text{edge}(X, Y), \text{edge}(Y, Z), \text{edge}(Y, W) \]
\[ Q_2 : q(A) :- \text{edge}(A, B), \text{edge}(B, B), \text{edge}(B, C) \]

\[ \begin{align*}
Q_1 : & \quad X \rightarrow Y \rightarrow Z \rightarrow W \\
Q_2 : & \quad A \rightarrow B \rightarrow B \rightarrow C
\end{align*} \]

- \( \mu_1 \): containment mapping from \( Q_1 \) to \( Q_2 \).
  - \( Q_2 \sqsubseteq_s Q_1 \iff \text{containment mapping from } Q_1 \text{ to } Q_2 \).
    (Chandra-Merlin, STOC 1977)
- \( \mu_2 \): \textbf{variables-onto} containment mapping from \( Q_1 \) to \( Q_2 \).
  - \text{variables-onto containment mapping from } Q_1 \text{ to } Q_2 \Rightarrow \n Q_2 \sqsubseteq_{bs} Q_1 . \) (Chaudhuri-Vardi, PODS 1993)
- \( \mu_3 \): \textbf{subgoals-onto} containment mapping from \( Q_1 \) to \( Q_2 \).
  - \text{subgoals-onto containment mapping from } Q_1 \text{ to } Q_2 \Rightarrow \n Q_2 \sqsubseteq b Q_1 . \) (Chaudhuri-Vardi, PODS 1993)
Necessary Conditions for CQ Containment

- $Q_2 \subseteq_{bs} Q_1 \Rightarrow$ Every variable of $Q_2$ must be mapped using a containment mapping from $Q_1$ (Chaudhuri-Vardi, PODS 1993)
  - otherwise, there is a subgoal of $Q_2$ that is not mapped by $Q_1$
- $Q_2 \subseteq_b Q_1 \Rightarrow$ Every subgoal of $Q_2$ must be mapped using a containment mapping from $Q_1$ (Chaudhuri-Vardi, PODS 1993)

Example

$Q_1 : q(X) :- link(X, Y), link(X, Z)$

$Q_2 : q(A) :- link(A, C), link(C, D)$

$Q_2 \not\subseteq_{bs} Q_1$

$q(a) \leftarrow link(a, b), link(a, b)$

$Q_2 : q(A) :- link(A, C), link(C, D)$

$q(a) \leftarrow link(a, b), link(a, c_1)$

$\vdots$

$\text{link}(a, c_\ell)$
Necessary Conditions for CQ Containment

- $Q_2 \sqsubseteq_{bs} Q_1 \Rightarrow$ Every variable of $Q_2$ must be mapped using a containment mapping from $Q_1$ (Chaudhuri-Vardi, PODS 1993)
  - otherwise, there is a subgoal of $Q_2$ that is not mapped by $Q_1$
- $Q_2 \sqsubseteq_{b} Q_1 \Rightarrow$ Every subgoal of $Q_2$ must be mapped using a containment mapping from $Q_1$ (Chaudhuri-Vardi, PODS 1993)

Example

$Q_1 : q(X) :- \text{link}(X, Y), \text{link}(X, Z)$

$q(a) \leftarrow \text{link}(a, b), \text{link}(a, b)$

$Q_2 : q(A) :- \text{link}(A, C), \text{link}(C, C)$

$q(a) \leftarrow \text{link}(a, b), \text{link}(a, a)$

$\vdots$

$\text{link}(a, a)$
CQs without projections

- \( Q_1 \) is CQ, \( Q_2 \) is CQ without projections:
  \[ Q_2 \sqsubseteq_{bs} Q_1 \iff Q_2 \sqsubseteq_s Q_1 \]
- i.e. searching for a containment mapping from \( Q_1 \) to \( Q_2 \) (in NP)
- What about bag-containment? i.e. \( Q_2 \sqsubseteq_b Q_1 ?? Q_2 \sqsubseteq_{bs} Q_1 \)

Example

\[
\begin{align*}
Q_1 &: q(X, Y) :- \text{link}(X, Y) \\
Q_2 &: q(X, Y) :- \text{link}(X, Y), \text{link}(Y, Y)
\end{align*}
\]

\( Q_2 \sqsubseteq_{bs} Q_1 \)

\[
\begin{array}{|c|c|}
\hline
\text{link} & a \\
\hline
a & a \\
\hline
a & a \\
\hline
a & a \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|}
\hline
\text{link} & a \\
\hline
a & a \\
\hline
a & a \\
\hline
a & a \\
\hline
\end{array}
\]
CQs without projections - Cont.

• $Q_1$, $Q_2$ both CQs without projections:
  
  - $Q_2 \sqsubseteq_b Q_1 \iff$ subgoals-onto containment mapping from $Q_1$ to $Q_2$
  
  - Check whether or not the mapping of distinguished vars is also a subgoals-onto (resp. variables-onto): $O(n^2 \log(n))$
  
  - Why $Q_1$ without projection?

Example

$Q_1 : q(X, Y) :- e(X, Y), e(W, U), e(W, V)$

$Q_2 : q(X, Y) :- e(X, Y), e(X, X), e(Y, Y)$
CQs without projections - Cont.

- $Q_1, Q_2$ both CQs without projections:
  - $Q_2 \sqsubseteq_b Q_1 \iff$ subgoals-onto containment mapping from $Q_1$ to $Q_2$
  - $Q_2 \sqsubseteq_{bs} Q_1 \iff$ containment mapping from $Q_1$ to $Q_2$
  - Check whether or not the mapping of distinguished vars is also a subgoals-onto (resp. variables-onto): $O(n^2 \log(n))$
  - Why $Q_1$ without projection?

Example

\[
\begin{align*}
Q_1: & \text{ } q(X, Y) \leftarrow e(X, Y), e(W, U), e(W, V) \\
Q_2: & \text{ } q(X, Y) \leftarrow e(X, Y), e(X, X), e(Y, Y) \\
& \text{ } q(a, b) \leftarrow e(a, b), e(a, a), e(b, b)
\end{align*}
\]
CQs without self-joins

- $Q_1$ is a CQ, $Q_2$ is CQ without self-joins:
  - i.e. every relation-name appears at most once
  - every subgoal of $Q_1$ can map at most one subgoal of $Q_2$
- $Q_2 \sqsubseteq_b Q_1$ (resp. $Q_2 \sqsubseteq_{bs} Q_1$) $\iff$ subgoals-onto (resp. variables-onto) containment mapping from $Q_1$ to $Q_2$
- Complexity: $O(n \log(n))$
  1. Sort w.r.t. relation-names
  2. Check whether or not each subgoal of $Q_1$ maps the unique subgoal of $Q_2$, with the same relation name
  3. Check whether or not there is a subgoals-onto (resp. variables-onto) containment mapping from $Q_1$ to $Q_2$

Example

$Q_1 : q(X) :- blue(X,Y), \ blue(X,Z)$

$Q_2 : q(X) :- blue(X,Y), \ green(Y,Y), \ red(Y,Z)$,
CQs without self-joins

- $Q_1$ is a CQ, $Q_2$ is CQ without self-joins:
  - i.e. every relation-name appears at most once
  - every subgoal of $Q_1$ can map at most one subgoal of $Q_2$

\[ Q_2 \sqsubseteq_b Q_1 \text{ (resp. } Q_2 \sqsubseteq_{bs} Q_1) \iff \text{ subgoals-onto (resp. variables-onto) containment mapping from } Q_1 \text{ to } Q_2 \]

- Complexity: $O(n \log(n))$
  1. Sort w.r.t. relation-names
  2. Check whether or not each subgoal of $Q_1$ maps the unique subgoal of $Q_2$, with the same relation name
  3. Check whether or not there is a subgoals-onto (resp. variables-onto) containment mapping from $Q_1$ to $Q_2$

Example

\[ Q_1 : q(X) :- \text{blue}(X, Y), \text{blue}(X, Z) \]
\[ Q_2 : q(X) :- \text{blue}(X, Y), \text{green}(Y, Y), \text{red}(Y, Z), \text{q}(a) \leftarrow \text{blue}(a, b), \text{green}(a, a), \text{red}(a, c), \text{red}(a, d) \]
\[ Q_2 \not\sqsubseteq_{bs} Q_1 \]
Generalized-Star Queries

- **Labeled path**: $r_1(W_0, W_1), r_2(W_1, W_2), \ldots, r_k(W_{k-1}, W_k), k \geq 1$
  - $r_1, r_2, \ldots, r_k$ are not necessarily distinct relation names, and
  - $W_0, W_1, \ldots, W_k$ are distinct variables.

- **Star** $S(X)$: collection of labeled paths starting from the same variable $X$ (root).

- **Generalized-star query of arity** $n$:
  \[ Q : q(X_1, \ldots, X_n) : -S_1(X_1), \ldots, S_n(X_n), N_1(Y_1), \ldots, N_m(Y_m) \]
  - $m = 0 \Rightarrow Q$ is a star query
  - **Simple** generalized-star query: the length of each labeled path is 1 (i.e. it is of the form: $r(W_0, W_1)$)
Star Queries

• $Q_1$, $Q_2$ are star queries of arity $n$:
  • $Q_2 \subseteq b Q_1$ (resp. $Q_2 \subseteq bs Q_1$) $\iff$ subgoals-onto (resp. variables-onto) containment mapping from $Q_1$ to $Q_2$
  
  - Variables-onto (resp. Subgoals-onto) containment mapping $\Rightarrow Q_2 \subseteq bs Q_1$ (resp. $Q_2 \subseteq b Q_1$)
  - Existence of a containment mapping from $Q_1$ to $Q_2$
  - Every variable (resp. subgoal) of $Q_2$ is mapped by $Q_1$
Query Containment under Bag and Bag-Set Semantics

Foto Afrati, Matthew Damigos and Manolis Gergatsoulis

Star Queries

- $Q_1, Q_2$ are star queries of arity $n$:
  - $Q_2 \sqsubseteq_b Q_1$ (resp. $Q_2 \sqsubseteq_{bs} Q_1$) $\iff$ subgoals-onto (resp. variables-onto) containment mapping from $Q_1$ to $Q_2$

![Diagram of star queries]

- Variables-onto (resp. Subgoals-onto) containment mapping $\Rightarrow Q_2 \sqsubseteq_{bs} Q_1$ (resp. $Q_2 \sqsubseteq_b Q_1$)

- Existence of a containment mapping from $Q_1$ to $Q_2$

- Every variable (resp. subgoal) of $Q_2$ is mapped by $Q_1$
Star Queries

• \( Q_1, Q_2 \) are star queries of arity \( n \):
  
  • \( Q_2 \sqsubseteq_b Q_1 \) (resp. \( Q_2 \sqsubseteq_{bs} Q_1 \)) \iff \text{subgoals-onto (resp. variables-onto) containment mapping from} \ Q_1 \ \text{to} \ Q_2

• Variables-onto (resp. Subgoals-onto) containment mapping \( \Rightarrow Q_2 \sqsubseteq_{bs} Q_1 \) (resp. \( Q_2 \sqsubseteq_b Q_1 \))

• Existence of a containment mapping from \( Q_1 \) to \( Q_2 \)

• Every variable (resp. subgoal) of \( Q_2 \) is mapped by \( Q_1 \)

\[
\begin{align*}
Q_1 & : \\
X_1 & \quad X_2 \quad \ldots \quad X_N \\
& | \quad | \quad | \\
P_{11} & \quad P_{21} & \quad P_{N1} & \quad P_{NK_N} \\
W_{11} & \quad W_{21} & \quad W_{N1} & \quad W_{NK_N} \\
\end{align*}
\]

\[
\begin{align*}
Q_2 & : \\
Y_1 & \quad Y_2 \quad \ldots \quad Y_N \\
& | \quad | \quad | \\
P'_{11} & \quad P'_{21} & \quad P'_{N1} & \quad P'_{NK_N} \\
Z_{11} & \quad Z_{21} & \quad Z_{N1} & \quad Z_{NK_N} \\
\end{align*}
\]
Query Containment under Bag and Bag-Set Semantics

Star Queries

- $Q_1$, $Q_2$ are star queries of arity $n$:
  - $Q_2 \sqsubseteq_b Q_1$ (resp. $Q_2 \sqsubseteq_{bs} Q_1$) $\iff$ subgoals-onto (resp. variables-onto) containment mapping from $Q_1$ to $Q_2$

- Variables-onto (resp. Subgoals-onto) containment mapping $\Rightarrow Q_2 \sqsubseteq_{bs} Q_1$ (resp. $Q_2 \sqsubseteq_b Q_1$)

- Existence of a containment mapping from $Q_1$ to $Q_2$

- Every variable (resp. subgoal) of $Q_2$ is mapped by $Q_1$
Star Queries

- $Q_1, Q_2$ are star queries of arity $n$:
  - $Q_2 \subseteq_b Q_1$ (resp. $Q_2 \subseteq_{bs} Q_1$) $\iff$ subgoals-onto (resp. variables-onto) containment mapping from $Q_1$ to $Q_2$

- Variables-onto (resp. Subgoals-onto) containment mapping $\Rightarrow Q_2 \subseteq_{bs} Q_1$ (resp. $Q_2 \subseteq_b Q_1$)

- Existence of a containment mapping from $Q_1$ to $Q_2$

- Every variable (resp. subgoal) of $Q_2$ is mapped by $Q_1$

- Check whether or not for each distinct labeled path $P$:
  - number of $P(Y_i)$ in $Q_2 \leq$ number of $P(X_i)$ in $Q_1$
  - $O(n^2 \log n)$
No existence of variables-onto containment mapping

Suppose the following two queries:

\[ Q_1 : q(X, Y) :- r(X, X'), r(Z, U), r(Z, W), r(Y, Y') \]
\[ Q_2 : q(X, Y) :- r(X, X'), r(X, U), r(Y, W), r(Y, Y') \]

- Neither subgoal-onto nor variables-onto containment mapping from \( Q_1 \) to \( Q_2 \).
- Each variable and each subgoal of \( Q_2 \) are mapped by \( Q_1 \).
- For each distinct set of tuples:
  \( \text{number of valuations over } Q_2 \leq \text{number of valuations over } Q_1 \)
No existence of variables-onto containment mapping

Suppose the following two queries:

\[ Q_1 : q(X, Y) := r(X, X'), r(Z, U), r(Z, W), r(Y, Y') \]
\[ Q_2 : q(X, Y) := r(X, X'), r(X, U), r(Y, W), r(Y, Y') \]

- Neither subgoal-onto nor variables-onto containment mapping from \( Q_1 \) to \( Q_2 \).
- Each variable and each subgoal of \( Q_2 \) are mapped by \( Q_1 \).
- For each distinct set of tuples:
  number of valuations over \( Q_2 \) \( \leq \) number of valuations over \( Q_1 \).
No existence of variables-onto containment mapping

Suppose the following two queries:

\[ Q_1 : q(X, Y) :\neg r(X, X') , r(Z, U) , r(Z, W) , r(Y, Y') \]
\[ Q_2 : q(X, Y) :\neg r(X, X') , r(X, U) , r(Y, W) , r(Y, Y') \]

• Neither subgoal-onto nor variables-onto containment mapping from \( Q_1 \) to \( Q_2 \).
• Each variable and each subgoal of \( Q_2 \) are mapped by \( Q_1 \).
• For each distinct set of tuples:
  number of valuations over \( Q_2 \) \( \leq \) number of valuations over \( Q_1 \).
No existence of variables-onto containment mapping

Suppose the following two queries:

\[ Q_1 : q(X, Y) \leftarrow r(X, X'), r(Z, U), r(Z, W), r(Y, Y') \]
\[ Q_2 : q(X, Y) \leftarrow r(X, X'), r(X, U), r(Y, W), r(Y, Y') \]

- Neither subgoal-onto nor variables-onto containment mapping from \( Q_1 \) to \( Q_2 \).
- Each variable and each subgoal of \( Q_2 \) are mapped by \( Q_1 \).
- For each distinct set of tuples:
  \[
  \text{number of valuations over } Q_2 \leq \text{number of valuations over } Q_1
  \]
Query Containment under Bag and Bag-Set Semantics

No existence of variables-onto containment mapping

Suppose the following two queries:

\[ Q_1 : q(X, Y) :- r(X, X'), r(Z, U), r(Z, W), r(Y, Y') \]
\[ Q_2 : q(X, Y) :- r(X, X'), r(X, U), r(Y, W), r(Y, Y') \]

- Neither subgoal-onto nor variables-onto containment mapping from \( Q_1 \) to \( Q_2 \).
- Each variable and each subgoal of \( Q_2 \) are mapped by \( Q_1 \).
- For each distinct set of tuples:
  number of valuations over \( Q_2 \) ≤ number of valuations over \( Q_1 \).
Simple Generalized-Star Queries

- $Q_1$ is simple generalized-star query of arity $n$
- $Q_2$ is a star query of arity $n$
- Schema contains a single binary relation

\[
\begin{align*}
Q_1 : & \quad X_1 \quad N_j \quad X_2 \quad \ldots \\
& \quad W_{11} \quad \ldots \quad W_{1K_1} \quad W_{1} \quad \ldots \quad W_{K} \quad W_{21} \quad \ldots \quad W_{2K_2} \\
Q_2 : & \quad Y_1 \quad \quad Y_2 \quad \quad \ldots \\
& \quad Z_{11} \quad \ldots \quad Z_{1K_1} \quad Z_{21} \quad \ldots \quad Z_{2K_2}
\end{align*}
\]
Simple Generalized-Star Queries

- $Q_1$ is a simple generalized-star query of arity $n$
- $Q_2$ is a star query of arity $n$
- Schema contains a single binary relation

For each d-star $S$ of $Q_1$ and the corresponding d-star $S'$ of $Q_2$, calculate:

$$|\text{subgoals}(S')| - |\text{subgoals}(S)|$$

Calculate the sum $s$ of all negative differences.

Calculate the number $s_N$ of the subgoals of n-stars of $Q_1$.

Check whether or not $s + s_N \geq 0$.

Linear time.
Simple Generalized-Star Queries

- $Q_1$ is simple generalized-star query of arity $n$
- $Q_2$ is a star query of arity $n$
- Schema contains a single binary relation
Simple Generalized-Star Queries

- $Q_1$ is a simple generalized-star query of arity $n$
- $Q_2$ is a star query of arity $n$
- Schema contains a single binary relation

\[
Q_1: X_1 \rightarrow W_{11} \cdots W_{1K_1} \rightarrow Y_1 \rightarrow Z_{11} \cdots Z_{1K_1}
\]

\[
Q_2: X_2 \rightarrow W_1 \cdots W_K \rightarrow Y_2 \rightarrow Z_1 \cdots Z_{K_2}
\]
Simple Generalized-Star Queries

- $Q_1$ is a simple generalized-star query of arity $n$
- $Q_2$ is a star query of arity $n$
- Schema contains a single binary relation

$Q_2 \subseteq_b Q_1$ (resp. $Q_2 \subseteq_{bs} Q_1$) $\iff$ for every subgset of $d$-stars $S$ of $Q_1$ and the set of corresponding $d$-stars $S'$ of $Q_2$:

$$\sum_{S' \in S} |\text{subgoals}(S')| \leq \sum_{S \in S} |\text{subgoals}(S)| + \sum_{j=1}^{m} |\text{subgoals}(N_j)|$$

Diagram:

$Q_1$:
- $X_1$ connected by $W_{11}$ to $Z_{11}$, and so on...
- $N_j$ connected by $W_1$ to $X_1$, and so on...

$Q_2$:
- $X_2$ connected by $W_{21}$ to $Z_{21}$, and so on...
- $Y_2$ connected by $W_1$ to $X_2$, and so on...

Test:
- For each $d$-star $S$ of $Q_1$ and the corresponding $d$-star $S'$ of $Q_2$, calculate:
  $$|\text{subgoals}(S')| - |\text{subgoals}(S)|$$
- Calculate the sum $s$ of all negative differences.
- Calculate the number $s$ of the subgoals of $n$-stars of $Q_1$.
- Check whether or not $s + s_N \geq 0$.
- Linear time.
Simple Generalized-Star Queries

- $Q_1$ is a simple generalized-star query of arity $n$
- $Q_2$ is a star query of arity $n$
- Schema contains a single binary relation

- $Q_2 \subseteq_b Q_1$ (resp. $Q_2 \subseteq_{bs} Q_1$) if for every subgset of $d$-stars $S$ of $Q_1$ and the set of corresponding $d$-stars $S'$ of $Q_2$:

$$\sum_{S' \in S} |\text{subgoals}(S')| \leq \sum_{S \in S} |\text{subgoals}(S)| + \sum_{j=1}^{m} |\text{subgoals}(N_j)|$$

Test:

- For each $d$-star $S$ of $Q_1$ and the corresponding $d$-star $S'$ of $Q_2$, calculate: $|\text{subgoals}(S')| - |\text{subgoals}(S)|$.
- Calculate the sum $s$ of all negative differences.
- Calculate the number $s_N$ of the subgoals of $n$-stars of $Q_1$.
- Check whether or not $s + s_N \geq 0$.
- Linear time.
Variables Property and CQ-Enhanced

- $Q_1, Q_2$ are CQs
- $Q_2 \sqsubseteq_{bs} Q_1 \Rightarrow$ for each finite set of tuples:
  number of valuations over $Q_2 \leq$ number of valuations over $Q_1$
Variables Property and CQ-Enhanced

- $Q_1$, $Q_2$ are CQs
- $Q_2 \sqsubseteq_{bs} Q_1 \Rightarrow$ for each finite set of tuples:
  number of valuations over $Q_2 \leq$ number of valuations over $Q_1$

- Each relation of arity $n$ is the $n$-th Cartesian Product of the set of constants
Variables Property and CQ-Enhanced

• $Q_1$, $Q_2$ are CQs

• $Q_2 \sqsubseteq_{bs} Q_1 \Rightarrow$ for each finite set of tuples:
  number of valuations over $Q_2 \leq$ number of valuations over $Q_1$

• Each relation of arity $n$ is the $n$-th Cartesian Product of the set of constants

• Each variable of each query can be valued by any constant
Variables Property and CQ-Enhanced

- $Q_1, Q_2$ are CQs
- $Q_2 \sqsubseteq_{bs} Q_1 \Rightarrow$ for each finite set of tuples:
  number of valuations over $Q_2 \leq$ number of valuations over $Q_1$

Each relation of arity $n$ is the $n$-th Cartesian Product of the set of constants
- Each variable of each query can be valuated by any constant
- If $|\text{Variables}(Q_2)| \leq |\text{Variables}(Q_1)|$ then $Q_2 \not\sqsubseteq_{bs} Q_1$
Variables Property and CQ-Enhanced

- $Q_1$, $Q_2$ are CQs
- $Q_2 \sqsubseteq_{bs} Q_1 \Rightarrow$ for each finite set of tuples: number of valuations over $Q_2 \leq$ number of valuations over $Q_1$
- $Q_2 \sqsubseteq_{bs} Q_1 \Rightarrow |\text{Variables}(Q_2)| \leq |\text{Variables}(Q_1)|$. 
Variables Property and CQ-Enhanced

- $Q_1, Q_2$ are CQs
- $Q_2 \sqsubseteq_{bs} Q_1 \implies$ for each finite set of tuples:
  number of valuations over $Q_2 \leq$ number of valuations over $Q_1$
- $Q_2 \sqsubseteq_{bs} Q_1 \implies |\text{Variables}(Q_2)| \leq |\text{Variables}(Q_1)|$.
- Suppose $Q_2$ is $Q_1$-enhanced: obtained by adding a sequence of subgoals to $Q_1$:
  - $Q_2 \sqsubseteq_{bs} Q_1 \iff$ variables-onto containment mapping from $Q_1$ to $Q_2$
  - Test: check whether any variable appearing in the additional subgoals also appears in $Q_1$’s body.
  - Linear time
# Complexity results for the Bag and Bag-Set CQ containment problem

<table>
<thead>
<tr>
<th>Containing Query ( (Q_1) )</th>
<th>Contained Query ( (Q_2) )</th>
<th>Complexity Bag-Set Semantics</th>
<th>Complexity Bag Semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>CQ</td>
<td>CQ</td>
<td>( \Pi_2^P - \text{hard}: \text{ChaudVardiPODS93} )</td>
<td>open</td>
</tr>
<tr>
<td>CQ</td>
<td>CQ without projections</td>
<td>NP: \text{AfrDamGerIPL09}</td>
<td>open</td>
</tr>
<tr>
<td>CQ without projections</td>
<td>CQ without projections</td>
<td>( O(n^2 \log(n)) ): \text{AfrDamGerIPL09}</td>
<td></td>
</tr>
<tr>
<td>CQ</td>
<td>CQ without self-joins</td>
<td>( O(n \log(n)) ): \text{AfrDamGerIPL09}</td>
<td>( O(n \log(n)) ): \text{IoanRamTODS95, AfrDamGerIPL09}</td>
</tr>
<tr>
<td>Star Query</td>
<td>Star Query</td>
<td>( O(n^2 \log(n)) ): \text{AfrDamGerIPL09}</td>
<td></td>
</tr>
<tr>
<td>Simple Gener.- Star Query</td>
<td>Simple Star Query</td>
<td>Linear: \text{AfrDamGerIPL09}</td>
<td></td>
</tr>
<tr>
<td>CQ</td>
<td>Enhanced (Q_1)</td>
<td>Linear: \text{AfrDamGerIPL09}</td>
<td>open</td>
</tr>
</tbody>
</table>
Thank You