The Query Containment Problem: Set Semantics vs. Bag Semantics

Phokion G. Kolaitis
University of California Santa Cruz
&
IBM Research - Almaden
An Old Problem in Database Theory

- Research in the connections between databases and logic has been going on for nearly five decades and since the time that E.F. Codd introduced the relational data model.

- There have been numerous successes, yet some problems have resisted solution.

- This talk is about the *conjunctive query containment problem under bag semantics*, which was introduced in 1993 ago by Surajit Chaudhuri and Moshe Y. Vardi.
Roadmap

- Background and motivation
- Query containment under set semantics
- Query containment under bag semantics
  - Problem description
  - Progress to date
- Concluding remarks and outlook.
Informally, database queries are questions that are posed against a database, and answers are retrieved.

Formally, a k-ary query, k ≥ 0, on a relational schema S is a function Q such that on every database D over S, the value Q(D) is a k-ary relation on the active domain of D.

Examples: Suppose S contains a binary relation E
- PATH2(D) = \{ (a,b): there is a path of length 2 from a to b \}
- PATH(D) = \{ (a,b): there is a path from a to b \}

Boolean query: a 0-ary query; it returns value 1 or 0.
- DIAM2(D) = 1 if and only if E has diameter at most 2.
- CONN(D) = 1 if and only if E is a connected graph.
Database Query Languages

• A query language is a formalism for expressing queries.

• Codd introduced two different query languages:
  – **Relational Algebra**: A query is an expression involving the operations $\pi$, $\sigma$, $\times$, $\cup$, $\setminus$
  
  – **Relational Calculus**: A query is a formula of first-order logic with quantifiers $\forall$ and $\exists$ ranging over the active domain.

• Codd showed that these two query languages have the same expressive power.

• **SQL**: The standard commercial database query language is based on relational algebra and relational calculus.
The Query Containment Problem

Let $Q_1$ and $Q_2$ be two database queries.

- $Q_1 \subseteq Q_2$ means that for every database $D$, we have that $Q_1(D) \subseteq Q_2(D)$, where $Q_i(D)$ is the set of all tuples returned by evaluating $Q_i$ on $D$.

- The Query Containment Problem asks: given two queries $Q_1$ and $Q_2$, is $Q_1 \subseteq Q_2$?

- For boolean queries ("true" or "false"), query containment amounts to logical implication $Q_1 \models Q_2$ in the finite.
The Query Containment Problem

- Encountered in several different areas, including
  - Query processing
    query equivalence reduces to query containment:

  \[ Q_1 \equiv Q_2 \text{ if and only if } Q_1 \subseteq Q_2 \text{ and } Q_2 \subseteq Q_1. \]

- Decision-support
  - \( Q_1 \) may be much easier to evaluate than \( Q_2 \).
  - If \( Q_1 \subseteq Q_2 \), then
    \( Q_1 \) provides a sound approximation to \( Q_2 \).

- Tight connections with constraint satisfaction (but this is another talk).
Complexity of Query Containment

The Query Containment Problem:
Given queries $Q_1$, $Q_2$, is $Q_1 \subseteq Q_2$?
In other words:
Is $Q_1(D)$ contained in $Q_2(D)$, for all databases $D$?

Note: Can’t just try every database $D$ – infinitely many!

Trakhtenbrot’s Theorem (1949):
The set of finitely valid first-order sentences is undecidable.

Corollary: For first-order queries, the query containment problem is undecidable.
Conjunctive Queries and their Extensions

Extensive study of the query containment problem for conjunctive queries and their extensions.

- Conjunctive queries: the most frequently asked queries. They are the `SELECT-PROJECT-JOIN` queries.
- Unions of conjunctive queries.
- Conjunctive queries with inequalities ≠ and arithmetic comparisons ≤ and ≥.
Conjunctive Queries and Their Extensions

• **Conjunctive Query:**
  - $Q(x_1,\ldots,x_k): \exists z_1 \ldots \exists z_m \varphi(x_1,\ldots,x_k,z_1,\ldots,z_m)$,
    where $\varphi$ is a conjunction of atoms.
  - Example:
    $\text{TAUGHT-BY}(x,y): \exists z (\text{ENROLLS}(x,z) \land \text{TEACHES}(y,z))$

• **Union of Conjunctive Queries**
  - Example: Path of length at most 2:
    $Q(x,y): E(x,y) \lor \exists z (E(x,z) \land E(z,y))$

• **Conjunctive Query with $\neq$**
  - Example: At least two different paths of length 2:
    $Q(x,y): \exists z \exists w (E(x,z) \land E(z,y) \land E(x,w) \land E(w,y) \land z \neq w)$. 
Conjunctive Queries and SQL

Fact: SQL provides direct support for conjunctive queries

Example: Consider the conjunctive query

- TAUGHT-BY(x,y): \( \exists z(\text{ENROLLS}(x,z) \land \text{TEACHES}(y,z)) \)
- SQL expression for this query:
  ```sql
  SELECT student, instructor
  FROM ENROLLS, TEACHES
  WHERE ENROLLS.course = TEACHES.course
  ```
Complexity of Conjunctive Query Containment

- **Theorem:** Chandra and Merlin – 1977
  For conjunctive queries, the containment problem is NP-complete.

- **Note:**
  - NP-hardness: reduction from 3-Colorability
  - Membership in NP is a consequence of the following result.
Complexity of Conjunctive Query Containment

Theorem: Chandra and Merlin – 1977
For Boolean conjunctive queries $Q_1$ and $Q_2$, the following are equivalent:

- $Q_1 \subseteq Q_2$.
- There is a homomorphism $h : D[Q_2] \rightarrow D[Q_1]$, where $D[Q_i]$ is the canonical database of $Q_i$.

Example: Conjunctive query and canonical database

- $Q$: $\exists x \exists y \exists z (E(x,y) \land E(y,z) \land E(z,x))$
- $D[Q] = \{ E(X,Y), E(Y,Z), E(Z,Y) \}$
Unions of Conjunctive Queries

**Theorem:** Sagiv & Yannakakis - 1980
The query containment problem for unions of conjunctive queries is NP-complete.

**Theorem:** Klug - 1988, van der Meyden – 1992
The query containment problem for conjunctive queries with $\neq$, $\leq$, $\geq$ is $\Pi_2^p$-complete.
## Complexity of Query Containment

<table>
<thead>
<tr>
<th>Class of Queries</th>
<th>Complexity of Query Containment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conjunctive Queries</td>
<td>NP-complete</td>
</tr>
<tr>
<td></td>
<td>Chandra &amp; Merlin – 1977</td>
</tr>
<tr>
<td>Unions of Conjunctive Queries</td>
<td>NP-complete</td>
</tr>
<tr>
<td></td>
<td>Sagiv &amp; Yannakakis - 1980</td>
</tr>
<tr>
<td>Conjunctive Queries with $\neq$, $\leq$, $\geq$</td>
<td>$\Pi_2^p$-complete</td>
</tr>
<tr>
<td></td>
<td>Klug 1988, van der Meyden -1992</td>
</tr>
<tr>
<td>First-Order (SQL) queries</td>
<td>Undecidable</td>
</tr>
<tr>
<td></td>
<td>Trakhtenbrot - 1949</td>
</tr>
</tbody>
</table>
Complexity of Query Containment

- So, the complexity of query containment for conjunctive queries and their variants is well understood.

**Caveat:**

- All preceding results assume set semantics, i.e., queries take sets as inputs and return sets as output (duplicates are eliminated).

- DBMS, however, use bag (multiset) semantics, since they return bags (duplicates are not eliminated).
A Real Conjunctive Query

- Consider the following SQL query:
  Table EMPLOYEE has columns (attributes) named salary, dept, …,
  ```sql
  SELECT salary
  FROM EMPLOYEE
  WHERE dept = 'CS'
  ```

- SQL keeps duplicates, because:
  - Duplicates are important for aggregate queries.
  - Duplicate elimination adds extra computational cost.
### Query Evaluation under Bag Semantics

<table>
<thead>
<tr>
<th>Operation</th>
<th>Multiplicity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Union</td>
<td>$m_1 + m_2$</td>
</tr>
<tr>
<td>$R_1 \cup R_2$</td>
<td></td>
</tr>
<tr>
<td>Intersection</td>
<td>min($m_1$, $m_2$)</td>
</tr>
<tr>
<td>$R_1 \cap R_2$</td>
<td></td>
</tr>
<tr>
<td>Product</td>
<td>$m_1 \times m_2$</td>
</tr>
<tr>
<td>$R_1 \times R_2$</td>
<td></td>
</tr>
<tr>
<td>Projection and Selection</td>
<td>Duplicates are not eliminated</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$R_1$</th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$R_2$</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$(R_1 \bowtie R_2)$</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>5</td>
</tr>
</tbody>
</table>
Bag Semantics

Chaudhuri & Vardi – 1993
Optimization of Real Conjunctive Queries

- Called for a re-examination of conjunctive-query optimization under bag semantics.

- In particular, they initiated the study of the containment problem for conjunctive queries under bag semantics.
Bag Semantics vs. Set Semantics

- For bags $R_1, R_2$:
  $$R_1 \subseteq_{\text{BAG}} R_2 \text{ if } m(a,R_1) \leq m(a,R_2), \text{ for every tuple } a.$$  
- $Q_{\text{BAG}}(D)$: Result of evaluating $Q$ on (bag) database $D$.
- $Q_1 \subseteq_{\text{BAG}} Q_2$ if for every (bag) database $D$, we have that
  $$Q_{1\text{BAG}}(D) \subseteq_{\text{BAG}} Q_{2\text{BAG}}(D).$$

Fact:
- $Q_1 \subseteq_{\text{BAG}} Q_2$ implies $Q_1 \subseteq Q_2$.
- The converse does not always hold.
Bag Semantics vs. Set Semantics

Fact: $Q_1 \subseteq Q_2$ does not imply that $Q_1 \subseteq_{BAG} Q_2$.

Example:
- $Q_1(x) :- P(x), T(x)$
- $Q_2(x) :- P(x)$
- $Q_1 \subseteq Q_2$ (obvious from the definitions)
- $Q_1 \not\subseteq_{BAG} Q_2$
- Consider the (bag) instance $D = \{P(a), T(a), T(a)\}$. Then:
  - $Q_1(D) = \{a, a\}$
  - $Q_2(D) = \{a\}$, so $Q_1(D) \not\subseteq_{BAG} Q_2(D)$. 
Query Containment under Bag Semantics

• Chaudhuri & Vardi - 1993 stated (without proof) that:
  Under bag semantics, the containment problem for conjunctive queries is $\Pi_2^p$-hard.

• Problem:
  – What is the exact complexity of the containment problem for conjunctive queries under bag semantics?
  – Is this problem decidable?
Query Containment Under Bag Semantics

• 27 years have passed since the containment problem for conjunctive queries under bag semantics was raised.

• Several attacks to solve this problem have failed.

• At least two technically flawed PhD theses on this problem have been produced.

• Chaudhuri and Vardi have withdrawn the stated $\Pi_2^p$-hardness of this problem; no proof found so far.
The containment problem for conjunctive queries under bag semantics remains open to date.

However, progress has been made towards the containment problem under bag semantics for the two main extensions of conjunctive queries:

- Unions of conjunctive queries
- Conjunctive queries with ≠
Unions of Conjunctive Queries

Theorem: Ioannidis & Ramakrishnan – 1995
Under bag semantics, the containment problem for unions of conjunctive queries is **undecidable**.

Hint of Proof:
Reduction from Hilbert’s 10th Problem.
Hilbert’s 10th Problem

- Hilbert’s 10th Problem – 1900
  (10th in Hilbert’s list of 23 problems)
  Find an algorithm for the following problem:
  Given a polynomial \( p(x_1,\ldots,x_n) \) with integer coefficients, does it have an all-integer solution?

- Theorem: Matiyasevich – 1971
  (building on Davis, Putnam, and Robinson)
  Hilbert’s 10th Problem is undecidable, hence no such algorithm exists.
Hilbert’s 10th Problem

• **Fact:** The following variant of Hilbert’s 10th Problem is undecidable:
  
  – Given two polynomials $p_1(x_1,\ldots,x_n)$ and $p_2(x_1,\ldots,x_n)$ with positive integer coefficients and no constant terms, is it true that $p_1 \leq p_2$?
    
    In other words, is it true that $p_1(a_1,\ldots,a_n) \leq p_2(a_1,\ldots,a_n)$, for all positive integers $a_1,\ldots,a_n$?

• Thus, there is no algorithm for deciding questions like:
  
  – Is $3x_1^4x_2x_3 + 2x_2x_3 \leq x_1^6 + 5x_2x_3$?
Unions of Conjunctive Queries

Theorem: Ioannidis & Ramakrishnan – 1995
Under bag semantics, the containment problem for unions of conjunctive queries is undecidable.

Hint of Proof:
- Reduction from the previous variant of Hilbert’s 10th Problem:
  - Use joins of unary relations to encode monomials (products of variables).
  - Use unions to encode sums of monomials.
Unions of Conjunctive Queries

Example: Consider the polynomial $3x_1^4x_2x_3 + 2x_2x_3$

- The monomial $x_1^4x_2x_3$ is encoded by the conjunctive query $P_1(w), P_1(w), P_1(w), P_1(w), P_2(w), P_3(w)$.

- The monomial $x_2x_3$ is encoded by the conjunctive query $P_2(w), P_3(w)$.

- The polynomial $3x_1^4x_2x_3 + 2x_2x_3$ is encoded by the union consisting of:
  - three copies of $P_1(w), P_1(w), P_1(w), P_1(w), P_2(w), P_3(w)$ and
  - two copies of $P_2(w), P_3(w)$. 


## Complexity of Query Containment

<table>
<thead>
<tr>
<th>Class of Queries</th>
<th>Complexity – Set Semantics</th>
<th>Complexity – Bag Semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conjunctive queries</td>
<td>NP-complete</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CM – 1977</td>
<td></td>
</tr>
<tr>
<td>Unions of conj. queries</td>
<td>NP-complete</td>
<td>Undecidable</td>
</tr>
<tr>
<td></td>
<td>SY - 1980</td>
<td>IR - 1995</td>
</tr>
<tr>
<td>Conj. queries with $\neq, \leq, \geq$</td>
<td>$\Pi_2^p$-complete</td>
<td></td>
</tr>
<tr>
<td></td>
<td>vdM - 1992</td>
<td></td>
</tr>
<tr>
<td>First-order (SQL) queries</td>
<td>Undecidable</td>
<td>Undecidable</td>
</tr>
<tr>
<td></td>
<td>Trakhtenbrot - 1949</td>
<td></td>
</tr>
</tbody>
</table>
Conjunctive Queries with ≠

Theorem: Jayram, K …, Vee – 2006
Under bag semantics, the containment problem for conjunctive queries with ≠ is undecidable.

In fact, this problem is undecidable even if
- the queries use only a single relation of arity 2;
- the number of inequalities in the queries is at most some fixed (albeit huge) constant.
Proof Outline

Proof is carried out in three steps.

**Step 1:** Only consider DBs of a **special** form. Show how to use conjunctive queries to encode polynomials and reduce Hilbert’s 10th Problem to conjunctive query containment over databases of **special** form (no inequalities are used!)

**Step 2:** Arbitrary databases
Use inequalities ≠ in the queries to achieve the following:
• If a database D is of **special** form, then we are back to the previous case.
• If a database D is not of **special** form, then $Q_1(D) \subseteq_{\text{bag}} Q_2(D)$.

**Step 3:** Show that we only need a **single** relation of arity 2.
# Complexity of Query Containment

<table>
<thead>
<tr>
<th>Class of Queries</th>
<th>Complexity – Set Semantics</th>
<th>Complexity – Bag Semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conjunctive queries</td>
<td>NP-complete</td>
<td>Open</td>
</tr>
<tr>
<td></td>
<td>CM – 1977</td>
<td></td>
</tr>
<tr>
<td>Unions of conj. queries</td>
<td>NP-complete</td>
<td>Undecidable, IR - 1995</td>
</tr>
<tr>
<td></td>
<td>SY - 1980</td>
<td></td>
</tr>
<tr>
<td>Conj. queries with ≠ , ≤, ≥</td>
<td>$\Pi_2^p$-complete</td>
<td>Undecidable, JKV - 2006</td>
</tr>
<tr>
<td></td>
<td>vdM - 1992</td>
<td></td>
</tr>
<tr>
<td>First-order (SQL) queries</td>
<td>Undecidable</td>
<td>Undecidable</td>
</tr>
<tr>
<td></td>
<td>Trakhtenbrot - 1949</td>
<td></td>
</tr>
</tbody>
</table>
Recent Developments

• Identification of restricted classes of conjunctive queries for which the containment problem under bag semantics is decidable.

• Discovery of tight connections between information theory and the conjunctive query containment problem under bag semantics.
Theorem: Konstantinidis and Mogavero – 2019
The containment problem $Q_1 \subseteq_{\text{BAG}} Q_2$ for conjunctive queries is decidable when $Q_1$ is projection-free (e.g., $Q_1(x,y):= E(x,y) \land P(y,y)$)
Moreover, this problem is NP-hard and in $\Pi_2^P$.

Hint of Proof:
Reduction to a decidable case of Hilbert’s 10th Problem, namely, to the solvability of a system of linear inequalities with integer coefficients.

Note: Extends an earlier result by Afrati, Damigos, Gergatsoulis in 2010, for the case in which both $Q_1$ and $Q_2$ are projection-free.
Information Theory Basics

- $W$ = random variable with $n$ outcomes
- Entropy of $W$
  $$H(W) = - \sum p(w) \log(p(w)) \leq \log(n)$$
- Join Entropy of $W_1, \ldots, W_k$
  $$H(W_1, \ldots, W_k) = -\sum p(w_1, \ldots, w_k) \log(p(w_1, \ldots, w_k))$$

Basic Shannon Inequalities
- $H(X) \geq 0$
- $H(XY) \geq H(X)$ (monotonicity)
- $H(XY) + H(XZ) \geq H(XYZ) + H(X)$ (submodularity)
Entropic Inequalities

Basic Shannon Inequalities
- $H(X) \geq 0$
- $H(XY) \geq H(X)$ (monotonicity)
- $H(XY) + H(XZ) \geq H(XYZ) + H(X)$ (submodularity)

Pippenger (1986):
Valid entropic inequalities are “the laws of information theory”.

Zhang and Yeung (1998): There are valid entropic inequalities that are not implied by Shannon’s basic inequalities

Decision Problem: Given an entropic inequality, is it valid?

Note: It is not known if this problem is decidable or undecidable.
Information Theory and Bag Containment

Kopparty & Rossman – 2011:
Applied information theory to bag containment

Example:
Boolean Queries $Q_1$ and $Q_2$

- $Q_1 () : \exists x, y, z \ (E(x, y) \land E(y, z) \land E(z, x))$
- $Q_2 () : \exists x, y, z \ (E(x, y) \land E(x, z))$

- $Q_1 \subseteq_{BAG} Q_2$ follows from the \textit{max entropic inequality}:

\[
\max(2H(XY) - H(X), 2H(YZ) - H(Y), 2H(ZX) - H(Z)) \geq H(XYZ)
\]
The Containment Problem for Boolean Queries

- **Note:**
  For Boolean conjunctive queries, the containment problem under bag semantics is equivalent to the Homomorphism Domination Problem.

- **The Homomorphism Domination Problem for graphs**
  Given two graphs $G$ and $H$, is it true that $\text{hom}(G,T) \leq \text{hom}(H,T)$, for every graph $T$?
  Here,
  - $\text{hom}(G,T) =$ number of homomorphisms from $G$ to $T$
  - $\text{hom}(H,T) =$ number of homomorphisms from $H$ to $T$. 
The Homomorphism Domination Problem

**Theorem:** Kopparty and Rossman - 2011

- There is an algorithm to decide, given a **chordal** graph $G$ and a **series-parallel** graph $H$, whether or not $\text{hom}(G,T) \leq \text{hom}(H,T)$, for all directed graphs $T$.

Equivalently,

- The containment problem $Q_1 \subseteq_{\text{BAG}} Q_2$ is decidable for Boolean conjunctive queries $Q_1$ and $Q_2$ such that the canonical database $D[Q_1]$ is a **chordal** graph and the canonical database $D[Q_2]$ is a **series-parallel** graph.

**Note:**

Proof uses entropy and linear programming.
The query containment problem $Q_1 \subseteq_{BAG} Q_2$ with $Q_2$ acyclic is Turing equivalent to determining if a given max entropic inequality is valid.

Note:
- Acyclic queries: well-behaved conjunctive queries
- Not known if max entropic inequalities are harder than entropic inequalities.

Take-home message: Tight connection between information theory and bag containment.
Acyclic Conjunctive Queries

\[ Q(\ ) : \exists x \ y \ z \ u \ v \ w \]
\[ (A(x,y,z) \land B(y,v) \land C(y,z,v) \land D(z,u,v) \land F(u,v,w)) \]
Concluding Remarks

• Twenty seven years after it was first raised and in spite of considerable efforts, the containment problem for conjunctive queries under bag semantics remains open.

• Information theory brings a new perspective to this problem.

• Let us hope that this problem will be settled some time in the not-too-distant future.

• But let us also recall a piece of wisdom by Piet Hein.
PROBLEMS

Problems worthy of attack prove their worth by hitting back.

in: *Grooks* by Piet Hein (1905-1996)
Selected References

- Chaudhuri and Vardi, Optimization of *Real* Conjunctive Queries, PODS 1993
- Ioannidis and Ramakrishnan, Containment of Conjunctive Queries: Beyond Relations as Sets, ACM TODS 1995
- Konstantinidis and Mogavero, Attacking Diophantus: Solving a Special Case of Bag Containment, PODS 2019
- Abo Khamis, Ngo, Kolaitis, Suciu: Bag Query Containment and Information Theory, PODS 2020