Data exchange: On the complexity of answering queries with inequalities.

Aleksander Mądry

Institute of Computer Science, University of Wrocław.

Key words: data exchange, query answering, certain answers, computational complexity

1 Introduction

The *data exchange problem* arises when we require data (called *source instance*) structured under one schema (*source schema*) to be translated into an instance (*target instance*) of a different schema (*target schema*), reflecting the source instance as accurately as possible, i.e. satisfying *source-to-target dependences*. Of course, in most cases there are many valid target instances corresponding to a particular problem - we can think of the solution of this problem as an indefinite database consisting of all possible target instances that are valid. So we can adapt the notion of certain answers used in the theory of incomplete databases for the semantics of query answering in data exchange, i.e. we focus only on those positive answers that hold in every target instance that satisfies the source-to-target dependences. In this paper we consider existential Boolean queries and the *local-as-view* (LAV) setting, which is a widely used, special case of the general data exchange problem setting.

We investigate the complexity of answering such queries. Fagin, Kolaitis, Miller and Popa [3] proved that the certain answers for queries being unions of conjunctive queries can be computed in polynomial time [1]. But when we allow inequalities within queries, then the situation changes.

It was shown by Abiteboul and Duschka [1] that computing the certain answers of unions of conjunctive queries with inequalities is in co-NP. This paper con-

Email address: a.madry@psz.pl (Aleksander Mądry).

[1] We assume that we have a fixed source schema, a target schema and source-to-target dependences. Complexity is measured with respect to the size of a given source instance.

tains also a false statement establishing co-NP-hardness of this problem even for conjunctive queries with one inequality. The correct proof [2] requires six inequalities. It has been shown [3] that there is a polynomial-time algorithm for computing the certain answers of unions of conjunctive queries with at most one inequality per disjunct. Thus the minimal number of inequalities giving rise to co-NP-hardness of this problem is at least two (unless P=NP) and six at most. So an interesting problem of precise determining this number arises. Fagin, Kolaitis, Miller and Popa [3] stated the following conjecture:

Conjecture 1 *In the LAV setting, computing the certain answers of boolean conjunctive queries with two inequalities is co-NP-complete.*

This paper presents a proof of this conjecture, closing the gap in the complexity of computing the certain answers of this kind of queries.

2 Preliminaries

We first define formally the general data exchange problem.

Definition 2 *A* **schema** $\mathbf{R}= \{R_i\}_{i=1,\cdots,k}$ *is a finite collection of relation symbols. An* **instance** I **over the schema** $\mathbf{R}$ *is a function associating a relation* $I(R_i)$ *with each relation symbol* $R_i \in \mathbf{R}$.

Let $\mathbf{S}$ *and* $\mathbf{T}$ *be two schemas.* $\mathbf{S}$ *is called a* **source schema** *and* $\mathbf{T}$ *- a* **target schema**. *An instance* I *over* $\mathbf{S}$ *is called a* **source instance** *and an instance* J *over* $\mathbf{T}$ *- a* **target instance**.

A formula of the form $\forall x(\phi_S(x) \rightarrow \exists y \chi_T(x,y))$, *where* $\phi_S(x)$ *is a formula of some logical formalism over* $\mathbf{S}$ *and* $\chi_T(x,y)$ *is a formula of (possibly different) logical formalism over* $\mathbf{T}$, *is called a* **source-to-target dependency**. Σ_{ST} *is a set of source-to-target dependences. Similarly, a formula in some logical formalism over* $\mathbf{T}$ *is called a* **target dependency**. Σ_T *is a set of target dependences.*

A schema $\mathbf{R}$ specifies a type of a relational database. An instance I over $\mathbf{R}$ defines a particular realization of the database of type $\mathbf{R}$. Each source-to-target dependency can be interpreted as a restriction on properties of a target database implied by properties of a source database. And target dependency defines the rules that constraints the shape of target database.

Definition 3 *The* **data exchange problem** *associated with a fixed source schema* $\mathbf{S}$, *target schema* $\mathbf{T}$, *a set of source-to-target dependences* Σ_{ST} *and a set of target dependences* Σ_T *is stated as follows: given a finite source instance* I, *find a finite target instance* J *such that* $<I,J>$ *satisfies all formulas in* Σ_{ST}

and J satisfies all formulas in Σ_T. Such a J is called a **solution for** *I. We denote such instance of the data exchange problem as a tuple* $<\mathbf{S},\mathbf{T},\Sigma_{ST},\Sigma_T,I>$.

So, the data exchange problem consists of finding a target database such that all restrictions resulting from the properties of a given source database hold. It was already mentioned that there can be more than one such database (solution) - so one can consider the problem of checking whether in every such database an additional property is preserved. This leads to the problem of computing the certain answers for Boolean queries.

Definition 4 *Consider a data exchange problem instance* $<\mathbf{S},\mathbf{T},\Sigma_{ST},\Sigma_T,I>$. *Let q be an existential boolean query over the target schema $\mathbf{T}$, and let J be a solution for I. We write $q(J) = true$ if q holds in J. Otherwise we write $q(J) = false$.*

If $q(J) = true$ for every solution for I, we say that the **certain answer** *for q with respect to I is true and write $certain(q, I) = true$. We say that the* **certain answer** *for q with respect to I is $false$ (denoted by $certain(q, I) = false$) if there exists a solution J' for I such that $q(J') = false$.*

Definition 5 *We say that a data exchange problem instance* $<\mathbf{S},\mathbf{T},\Sigma_{ST},\Sigma_T,I>$ **obeys LAV setting's restrictions** *if $\Sigma_T = \emptyset$ and each formula in Σ_{ST} is of the form $\forall x(\phi_S(x) \to \exists y \chi_T(x, y))$, where $\phi_S(x)$ is a conjunction of atomic formulas over $\mathbf{S}$ and $\chi_T(x, y)$ is a conjunction of atomic formulas over $\mathbf{T}$.*

In the LAV setting *we consider only those instances of data exchange problem that obey above restrictions.*

We can state now the main theorem.

Theorem 6 *In the LAV setting, computing the certain answers of boolean conjunctive queries with two or more inequalities is co-NP-complete.*

3 Proof

As it was already said, the membership of this problem in co-NP was established in [1]. We shall establish its co-NP-hardness by reduction to the complement of 3SAT.

Reduction 1 *Let ϕ be a given 3CNF-formula of the form $\bigwedge_{i=1}^{m} \psi_i$ with variables $x_1, \ldots, x_n$. The source schema $\mathbf{S}$ consists of binary relations P,L,R,N and the target schema $\mathbf{T}$ consists of binary relations P',N'. We consider the following source-to-target dependences:*

$$
\begin{aligned}
&P(x,y) \rightarrow P'(x,y) && (1)\\
&L(x,y) \rightarrow \exists z(P'(x,z) \wedge P'(z,y)) && (2)\\
&R(x,y) \rightarrow (\exists u,v,t)(P'(x,u) \wedge P'(u,v) \wedge P'(v,t) \wedge P'(t,y)) && (3)\\
&N(x,y) \rightarrow N'(x,y), && (4)
\end{aligned}
$$

a query q:

$$
\begin{aligned}
q: -(\exists x, y_1, y_2, y_3)(P'(x,y_1) \wedge P'(x,y_2) \wedge P'(x,y_3) \wedge \\
\wedge N'(y_1,y_2) \wedge y_1 \neq y_3 \wedge y_2 \neq y_3),
\end{aligned}
$$

and a source instance I with elements:

- T,F,F',
- t_j,f_j *and* v_j, *for each variable* x_j *of* ϕ,
- *for each clause* ψ_i *of* ϕ, *three elements* x_1^i, x_2^i, x_3^i *representing each literal of* ψ_i *respectively.*

and relations:

$$
\begin{aligned}
I(P) = &\{(x_1^i, x_2^i), (x_2^i, x_3^i), (x_3^i, x_1^i) : i = 1, \ldots, m\} \cup \\
&\cup \{(v_j, t_j), (v_j, f_j), (t_j, t_j), (f_j, f_j) : j = 1, \ldots, n\} \cup \\
&\cup \{(x_k^i, t_j) : k\text{-th literal of } \psi_i \text{ is } x_j\} \cup \\
&\cup \{(x_k^i, f_j) : k\text{-th literal of } \psi_i \text{ is } \neg x_j\} \cup \\
&\cup \{(F,F), (F,F'), (F',F), (F',F')\} \\
I(L) = &\{(v_j, F) : j = 1, \ldots, n\} \\
I(R) = &\{(x_k^i, T) : i = 1, \ldots, m;\ k = 1,2,3\} \\
I(N) = &\{(x,y) : x,y \text{ are distinct elements of } I\}.
\end{aligned}
$$

Intuitively, we construct a digraph using the relation P. This digraph is a subgraph of the digraph defined by relation P' in the target instance, in which for each pair x,y of elements being in relation L (R), there exists a directed path of length two (four) from x to y (this path may have cycles). The relations N,N' simulate inequalities between elements of the source instance and the target instance respectively (it should be clear that relation N' can also claim an element to be 'different' from itself). The query q asks whether there exists in P'-digraph a node whose out-degree is greater than two (where two of three out-neighbours of this node are 'different' in the sense of relation N'). Some ideas of this construction are taken from [2].

Claim 7 *ϕ is satisfiable if and only if $certain(q, I) = false$.*

PROOF. In order to prove the left-to-right implication, assume ϕ is satisfiable by some truth assignment v. Choose a target instance J such that:

$$\begin{aligned} J(P') &= I(P) \cup \{(t_j, T), (f_j, F) : v(x_j) = true\} \cup \\ &\quad \cup \{(f_j, T), (t_j, F) : v(x_j) = false\} \\ J(N') &= I(N). \end{aligned}$$

It is easy to see that J satisfies source-to-target dependences (1), (2), (4) and $q(J) = false$. In order to prove that (3) is satisfied, consider a tuple $R(x_k^i, T)$. Since ψ_i is satisfied by v, there exists a literal x_l^i such that for some j $x_l^i = x_j$ ($x_l^i = \neg x_j$), $P'(x_l^i, t_j)$ ($P'(x_l^i, f_j)$) holds and $v(x_j) = true$ ($v(x_j) = false$). Thus $P'(t_j, T)$ ($P'(f_j, T)$) holds and a directed path from x_k^i through x_l^i to T exists. Its length is four at most and by adding a proper number of arcs $P'(T, T)$ we can construct a path that satisfies (3). Thus $certain(q, I) = false$.

To prove the second implication, assume $certain(q, I) = false$. Then there exists a target instance J such that $q(J) = false$. The desired implication is derived from the following properties of J.

Definition 8 *Let an element t_j (f_j) ($1 \leqslant j \leqslant n$) be such that for some x_k^i ($1 \leqslant i \leqslant m; 1 \leqslant k \leqslant 3$) there exists a directed path in the P'-digraph of J that satisfies (3), includes t_j (f_j) and moreover x_j ($\neg x_j$) is a literal in ψ_i. Such element is called a* **T-witness**. *We also say that x_k^i* **has T-witness** *t_j (f_j), and that t_j (f_j)* **T-witnesses a path from** *x_k^i.*

Definition 9 *Let an element t_j (f_j) ($1 \leqslant j \leqslant n$) be such that $P'(t_j, F)$ ($P'(f_j, F)$) holds in J. We call such an element a* **F-witness**.

Property 1 ($\forall i = 1, \ldots, m; k = 1, 2, 3; j = 1, \ldots, n$) *in the P'-digraph of J, the elements F,F',x_k^i,v_j have out-degree two.*

PROOF. Assume for the sake of contradiction that an element F has out-degree greater than two in the P'-digraph of J. By (1), F and F' are its out-neighbours, moreover by (4) $N'(F, F')$ holds in J. Since F has out-degree at least three, there exists a third element X such that $F \neq X$, $F' \neq X$ and $P'(F, X)$. But this implies that $q(J) = true$, which contradicts the assumption that $q(J) = false$. Proof for other elements is similar. □

Property 2 *In the P'-digraph of J, there is no directed path from F to T.*

PROOF. This is an obvious consequence of Property 1. □

Property 3 $(\forall j = 1, \ldots, n)$ *either t_j or f_j is a F-witness.*

PROOF. By (2) there exists a two-arc directed path from v_j to F. Since out-degree of v_j is two (Property 1), this path leads through t_j or f_j, and reaches F immediately after t_j (f_j). So, the desired property holds. $\Box$

Property 4 $(\forall i = 1, \ldots, m; k = 1, 2, 3)$ *x_k^i has a T-witness.*

PROOF. Consider any x_k^i. Since (3), there exists a directed path leading from x_k^i to T. Thereby this path must finally leave the cycle consisting of x_1^i, x_2^i and x_3^i. But since Property 1 holds, this is possible only through edge that leads to one of potential T-witnesses. $\Box$

Property 5 $(\forall j = 1, \ldots, n)$ *t_j (f_j) can't be both a T-witness and an F-witness.*

PROOF. Assume for the sake of contradiction that this property fails and there exists an element being both a T-witness and an F-witness. Consider the case when this element is t_j for some j. Then $P'(t_j, F)$ holds and the path that t_j T-witnesses must finally leave t_j and lead out to some X ($X \neq t_j$) such that $X \neq F$ (by Property 2). But since $P'(t_j, t_j) \wedge P'(t_j, F) \wedge N'(t_j, F)$ holds (this is a consequence of (1) and (4)), we conclude that $q(J) = true$, which contradicts the assumption that $q(J) = false$. The result holds symmetrically for f_j. $\Box$

Consider now the following truth assignment:

$$v(x_j) = \begin{cases} true & \text{if } f_j \text{ is F-witness} \\ false & \text{otherwise} \end{cases}$$

Lemma 10 *ϕ is satisfied by v.*

PROOF. Assume for the sake of contradiction that some clause ψ_i of ϕ is not satisfied by v. By Property 4, x_1^i has a T-witness, say t_j (f_j). By the definition of T-witness, the literal x_j ($\neg x_j$) belongs to the clause ψ_i. Moreover, since Property 3 holds, either t_j or f_j is an F-witness and by Property 5 it must be f_j (t_j). But that implies that $v(x_j) = true$ ($v(x_j) = false$) which contradicts the assumption that v does not satisfies ψ_i. $\Box$

Lemma 10 implies that ϕ is satisfiable, so Claim 7 is proved. $\Box$

Since Claim 7 holds, reduction given above is valid and establishes co-NP-hardness of our problem. Therefore the theorem is proved.

It should be noted that, although in this theorem we restricted the form of existential boolean queries only to those being conjunctive, this result affects also existential boolean queries being unions of conjunctive ones. This, combined with already mentioned result of Fagin, Kolaitis, Miller and Popa [3] about polynomial complexity of computing the certain answers of unions of conjunctive queries with at most one inequality per disjunct, gives a tight characterization of complexity of computing the certain answers for all queries of such type.

References

[1] S. Abiteboul and O. M. Duschka. Complexity of Answering Queries Using Materialized Views. In *Proceedings of the ACM Symposium on Principles of Database Systems (PODS)*, Seattle, WA, 1998, pages 254-263.

[2] S. Abiteboul and O. M. Duschka. Complexity of Answering Queries Using Materialized Views. Unpublished full version of [1], 2000.

[3] R. Fagin, P. G. Kolaitis, R. J. Miller, L. Popa. Data Exchange: Semantics and Query Answering. Proc. *International Conference on Database Theory (ICDT)*, 2003, pp. 207-224. To appear in *Theoretical Computer Science.*