Corrigendum to "Enumeration Complexity of Conjunctive Queries with Functional Dependencies"

Nofar Carmeli

Abstract

We claimed that there is an exact reduction between any Conjunctive Query (CQ) and its FD-extension [1, 2], but our proof fails for CQs with self-joins. The main conclusion of our paper can still be shown in an alternative way for CQs with self-joins: a CQ (with or without self-joins) is tractable iff it is FD-free-connex.

1 The Mistake

The proof we provided for the following Theorem [2, Theorem 2] and its extensions is wrong in the case the CQ has self-joins: Let Q be a CQ over a schema $\mathcal{S} = (\mathcal{R}, \Delta)$, and let Q^+ be its FD-extended query. Then, $\textsc{Enum}_{\Delta}\langle Q\rangle \equiv_e \textsc{Enum}_{\Delta_{Q^+}}\langle Q^+\rangle$.

Consider for example the query $Q(x,y,z) \leftarrow R(x,y), S(x,y), R(x,z)$ with the FD $S : 1 \rightarrow 2$, and the database instance I with $R^I = \{(a,b),(a,c)\}$ and $S^I = \{(a,b)\}$. Clearly, $Q(I)$ has answers: (a,b,c) and (a,b,b). Now let us consider the FD-extension. We treat the dependency as between variables ($x \rightarrow y$), and obtain the extension $Q^+(x,y,z) \leftarrow R^+(x,y,t), S^+(x,y), R^+(x,z,y)$ with the FDs $S^+ : 1 \rightarrow 2$, $R^+ : 1 \rightarrow 2$ and $R^+ : 1 \rightarrow 3$. As part of the reduction that shows $\textsc{Enum}_{\Delta}\langle Q\rangle \leq_e \textsc{Enum}_{\Delta_{Q^+}}\langle Q^+\rangle$, we construct an instance to Q^+, where the first stage is cleaning, and it removes (a,c) from R^I in order to make sure the construction does not violate the newly introduced dependencies. Then, after the extension phase, we obtain the instance I^+ with $R^{+I^+} = \{(a,b,b)\}$ and $S^{+I^+} = \{(a,c)\}$. Over this construction, $Q^+(I^+) = \emptyset$, and so contrary to what we want, $Q(I) \neq Q^+(I^+)$.

The mistake in the proof itself is in the induction base. Given an answer to the original CQ over the original instance, we claimed that the same assignment is an answer to the original CQ over the cleaned instance. Indeed, in the self-join-free case, since we are given an answer over the original instance, there exist tuples, one of each atom of the query, that agree on the values of the variables of the FD (these tuples assign the variables with the same values μ assigns them). In the self-join-free case, these tuples are not removed during cleaning because

every variable-wise dependency agrees with them. In the case with self-joins, as the example demonstrates, this is no longer the case.

A similar mistake appears in the opposite direction of this claim: the proof we provided for $\textsc{Enum}_{\Delta_{Q^+}}\langle Q^+\rangle \leq_e \textsc{Enum}_{\Delta}\langle Q\rangle$ does not hold when the CQ contains self-joins, as the following example demonstrates. Consider the query $Q(v,w,x,y,z) \leftarrow R(x,y,z), R(v,w,x), S(x,y)$ with the dependency $S : 1 \to 2$, and the database instance I^+ with $R^{I^+} = \{(a,b,c,d),(e,f,a,b)\}$ and $S^I = \{(a,b),(e,g)\}$. The cleaning phase removes (e,f,a,b) from R^{I+}, and so this construction fails.

2 Corrections to past-made statements

Similar constructions also appears in the extensions of this Theorem [2, Theorem 2] to cardinality dependencies [2, Lemma 6 and Lemma 7], CQs with disequalities [2, Lemma 8 and Lemma 10], and their combination [2, Lemma 11]. All of these results still hold as long as we restrict the statement to **only apply for self-join-free CQs**. Some of these results also appear in the conference version of this article [1, Theorem 7 and Lemma 24], and these too still apply for self-join-free CQs.

We can show the main results that use the broken reduction also for queries with self-joins using an intermediate step that eliminates self-joins. Let SJF be a function that assigns each atom with a different relation symbol. For example, if an atom $R(\vec{v})$ appears in the CQ for the kth time, we can replace it by the atom $R_k(\vec{v})$ where R_k is a new relation symbol. We denote the transformed CQ by $\text{SJF}(Q)$ and we also replace the symbols in the dependency set accordingly to obtain $\text{SJF}(\Delta)$. Whenever this self-join-free version is tractable, the original CQ is tractable too, as we can duplicate the original relations to construct relations for the new distinct symbols and get the same result set. Formally, this proves that $\textsc{Enum}_{\Delta}\langle Q\rangle \leq_e \textsc{Enum}_{\text{SJF}(\Delta)}\langle \text{SJF}(Q)\rangle$. As our problematic proof still holds for the self-join-free case, we also have $\textsc{Enum}_{\text{SJF}(\Delta)}\langle \text{SJF}(Q)\rangle \leq_e \textsc{Enum}_{\text{SJF}(\Delta)_{Q^+}}\langle \text{SJF}(Q)^+\rangle$. By combining these two facts, we get that $\textsc{Enum}_{\Delta}\langle Q\rangle \leq_e \textsc{Enum}_{\text{SJF}(\Delta)_{Q^+}}\langle \text{SJF}(Q)^+\rangle$. This proves the **following correction** to our corollary [2, Corollary 1][1, Corollary 8]: Let $\mathcal{C}$ be an enumeration class that is closed under exact reduction. Let Q be a CQ and let Q^+ be its FD-extension. If $\textsc{Enum}_{\text{SJF}(\Delta)_{Q^+}}\langle \text{SJF}(Q)^+\rangle \in \mathcal{C}$, then $\textsc{Enum}_{\Delta}\langle Q\rangle \in \mathcal{C}$.

3 Statements that still hold

Then, our following conclusion [2, Corollary 2][1, Corollary 9] remains unchanged: Let Q be a CQ over a schema $\mathcal{S} = (\mathcal{R}, \Delta)$. If Q is FD-free-connex, then $\textsc{Enum}_{\Delta}\langle Q\rangle \in \mathsf{DelayC_{lin}}$. To prove this, we take an intermediary step through the self-join-free version of the query, and we need the following claim.

Claim 1. *Q^+ is free-connex iff $\text{SJF}(Q)^+$ is free-connex.*

Proof. First note that Q and $\mathrm{SJF}(Q)$ differ only on the relation names, but they have the same sets of variables in their atoms. Then, note that the extension procedure mostly depends only on the variable-sets inside the atoms and the matching dependencies, but it has a small sensitivity to the relation names: in the case of self-joins, additional fresh variables are added to the extension; however, every such variable only appears in one atom. So the difference between Q^+ and $\mathrm{SJF}(Q)^+$ is only in the relation names and the fact that atoms in Q^+ may have additional variables, where each such variable appears only in one atom. Now note the following properties of free-connexity: (1) it is not affected by relation names, and (2) it is not affected by the addition or removal of a variable that appears only in a single atom. This proves that Q^+ is free-connex iff $\mathrm{SJF}(Q)^+$ is free-connex. □

We can now prove the corollary. If Q is FD-free-connex, then by definition Q^+ is free-connex. We just proved that this means $\mathrm{SJF}(Q)^+$ is free-connex, and therefore $\textsc{Enum}_{\mathrm{SJF}(\Delta)_{Q^+}}\langle \mathrm{SJF}(Q)^+ \rangle \in \mathsf{DelayC_{lin}}$. Following the corrected corollary, $\textsc{Enum}_{\Delta}\langle Q \rangle \in \mathsf{DelayC_{lin}}$. The extension of this corollary to CQs with disequalities and CDs [2, Theorem 7] similarly holds in general.

Note that the hardness results in the article are not affected by this mistake because these (including all of Sections 4 and 5) were already restricted only to self-join-free CQs in the original article.

Acknowledgements

I would like to thank Carsten Lutz for bringing this mistake to my attention.

References

[1] Nofar Carmeli and Markus Kröll. "Enumeration Complexity of Conjunctive Queries with Functional Dependencies". In: *21st International Conference on Database Theory*. 2018.

[2] Nofar Carmeli and Markus Kröll. "Enumeration complexity of conjunctive queries with functional dependencies". In: *Theory of Computing Systems* 64.5 (2020), pp. 828–860.