Solutions and query rewriting in data exchange

Citation for published version:
Arenas, M, Barcelo, P, Fagin, R & Libkin, L 2013, 'Solutions and query rewriting in data exchange', *Information and Computation*, vol. 228-229, pp. 28-61. https://doi.org/10.1016/j.ic.2013.06.002

Digital Object Identifier (DOI):
10.1016/j.ic.2013.06.002

Published In:
Information and Computation

Solutions and Query Rewriting in Data Exchange[☆]

Marcelo Arenas[a], Pablo Barceló[b], Ronald Fagin[*,c], Leonid Libkin[d]

[a]*Department of Computer Science, Pontificia Universidad Católica, Av. Vicuña Mackenna 4860, Santiago, Chile*

[b]*Department of Computer Science, Universidad de Chile, Avda. Blanco Encalada 2120, Santiago, Chile*

[c] *IBM Research – Almaden, 650 Harry Road, San Jose, CA 95120-6099*

[d]*Laboratory for Foundations of Computer Science, University of Edinburgh, Informatics Forum, Crichton Street, Edinburgh, EH8 9AB, UK*

Abstract

Data exchange is the problem of taking data structured under a source schema and creating an instance of a target schema. Given a source instance, there may be many solutions – target instances that satisfy the constraints of the data exchange problem. Previous work has identified two classes of desirable solutions: canonical universal solutions, and their cores. Query answering in data exchange amounts to rewriting a query over the target schema to another query that, over a materialized target instance, gives the result that is semantically consistent with the source (specifically, the "certain answers"). Basic questions are then: (1) how do these solutions compare in terms of query rewriting? and (2) how can we determine whether a query is rewritable over a particular solution?

Our goal is to answer these questions. Our first main result is that, in terms of rewritability by relational algebra queries, the core is strictly less expressive than the canonical universal solution, which in turn is strictly less expressive than the source. To develop techniques for proving queries nonrewritable, we establish structural properties of solutions; in fact they are derived from the technical machinery developed in the rewritability proofs. Our second result is that both the canonical universal solution and the core preserve the local structure of the data, and that every target query rewritable over any of these solutions cannot distinguish tuples whose neighborhoods in the source are similar. This gives us a first simple tool for checking whether a query is nonrewritable over the canonical universal solution or over the core. We also show that these tools generalize to arbitrary transformations that preserve the local structure of the data, and investigate an alternative semantics of query answering in data exchange.

Key words: data exchange, universal solutions, query answering, query rewriting, locality

1. Introduction

Data exchange is the problem of materializing an instance that adheres to a target schema, given an instance of a source schema and a specification of the relationship between the source and the target.

[☆]This is the full version of the conference paper [ABFL04].

[*]Corresponding author. Phone number: 1-408-927-1726 E-mail address: fagin@us.ibm.com Complete postal address: IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120-6099

Email addresses: marenas@ing.puc.cl (Marcelo Arenas), pbarcelo@dcc.uchile.cl (Pablo Barceló), fagin@us.ibm.com (Ronald Fagin), libkin@inf.ed.ac.uk (Leonid Libkin)

This is a very old problem (cf. Housel et al. [HTGL77]) that arises in many tasks where data must be transferred between independent applications that do not have the same data format. The need for data exchange has steadily increased over the years. With the proliferation of web data in various formats and with the emergence of e-business applications that need to communicate data yet remain autonomous, data exchange has become even more important. Recent surveys summarizing the state of the art in data exchange are [ABLM10, Bar09, Kol05].

A data exchange setting is a triple ($\mathbf{S}$, $\mathbf{T}$, Σ_{st}), where $\mathbf{S}$ is the source schema, $\mathbf{T}$ is the target schema, and Σ_{st} is a set of *source-to-target dependencies* that express the relationship between $\mathbf{S}$ and $\mathbf{T}$ (some papers also add a set Σ_t of dependencies that expresses constraints on $\mathbf{T}$). Such a setting gives rise to the following *data exchange problem*: given an instance I over the source schema $\mathbf{S}$, find an instance J over the target schema $\mathbf{T}$ such that I together with J satisfy the source-to-target dependencies Σ_{st} (when target dependencies are used, J must also satisfy them). Such an instance J is called a *solution* for I in the data exchange setting. In general, there may be many different solutions for a given source instance I.

One of the basic assumptions in data exchange is that the target instance is *autonomous* from the source. That is, once a solution J has been materialized, target queries have to be evaluated over J only (since the source is no longer available). For a data exchange system, the two key issues are the choice of a solution to be materialized, and query answering over that materialized solution.

The work of Fagin, Kolaitis, Miller, and Popa [FKMP05, FKP05] started a systematic investigation of these issues for data exchange settings in which $\mathbf{S}$ and $\mathbf{T}$ are relational schemas. They isolated a class of solutions, called *universal solutions*, possessing good properties that justify selecting them as the best solutions in data exchange. Universal solutions are the most general among all solutions and, in a precise sense, they represent the entire space of solutions. It was shown by Fagin et al. [FKMP05] that under fairly general conditions, universal solutions exist, and a particular universal solution can be found in polynomial time. This solution, that can be computed by applying the classical chase procedure (cf. [BV84, MMS79]), is called the *canonical* universal solution.

Since universal solutions need not be unique, this raises the question of which universal solution to materialize. A possible answer is based on using *minimality* as a key criterion for what constitutes the "best" universal solution [FKP05]. Although universal solutions come in different sizes, all of them share a unique (up to isomorphism) common "part", which is nothing else but the *core* of each of them, when viewed as relational structures. By definition, the core of a structure is the smallest substructure that is also a homomorphic image of the structure. The concept of the core originated in graph theory, where a number of its properties have been established [HN92]. It has been shown [FKP05] that under fairly general conditions the core of the universal solutions for I is itself a solution for I. Hence, the core of the universal solutions for I is the *smallest* universal solution for I, and thus an ideal candidate for the "best" solution, at least in terms of the space required to materialize it. Furthermore, in many cases there is a polynomial-time algorithm that computes the core [GN08].

Given a source instance and a data exchange setting, what is the meaning of the "answer" to a query Q over the target schema? Since there may be multiple solutions to the data exchange problem, the standard approach is to define it to be the set of *certain answers* [IL84], that is, those tuples that appear in $Q(J)$ for every solution J. The goal of query answering in data exchange is to find these certain answers based on just *one* materialized target instance.

While for some classes of queries (e.g., unions of conjunctive queries), certain answers to Q could be obtained by evaluating Q itself over the canonical universal solution or the core [FKMP05, FKP05], this fails already for conjunctive queries with inequalities. In fact, it is known that there is a Boolean conjunctive query Q with inequalities such that $Q(J)$ does not give the certain answers, no matter which universal solution J is selected, but for some other first-order query Q' (a *rewriting* of Q), the certain answers for Q are given by $Q'(J)$, where J is the canonical universal solution [FKMP05]. But query rewritability is not a general phenomenon either, as there is a Boolean conjunctive query Q with inequalities for which there is no such rewriting Q' [FKMP05].

This leads to the following natural questions:

1. Can the canonical universal solution and the core be compared in terms of the queries that can be rewritten over each one of them? In other words, is every query rewritable over one of these solutions also rewritable over the other?
2. Is there a simple way to prove that a query does not have a rewriting over the canonical universal solution (resp., over the core)?

Our main goal is to answer them. We now give a summary of the main results in the paper.

Query rewritability We investigate queries that can be rewritten by relational algebra, or first order queries. We look at three kinds of rewritability: over the core, over the canonical universal solution, and over the source data itself, and prove the following strict inclusions:

$$\begin{array}{c} \text{rewritable over the} \\ \text{core} \end{array} \subsetneq \begin{array}{c} \text{rewritable over the} \\ \text{canonical universal solution} \end{array} \subsetneq \begin{array}{c} \text{rewritable over the} \\ \text{source} \end{array}$$

Tools for proving non-rewritability We use the techniques we developed in the process of comparing rewritability of queries to show that every rewritable query is *locally source-dependent*: that is, the certain answers to Q cannot distinguish tuples that have the same small neighborhoods in the source. We demonstrate how this tool can be used to give simple proofs that some queries are not rewritable.

Structural properties of solutions Locality is a standard tool in the study of expressibility [Gai82, Han65, FSV95, Lib04], and we investigate its applicability in query rewritability and data exchange transformations. Specifically, we show that solutions such as the core and the canonical universal solution preserve the local character of the data. In some cases they preserve isomorphism types of local neighborhoods of tuples, and more generally, they preserve their logical types (that describe sets of definable first-order properties of neighborhoods).

Extensions We look at a different, universal solutions semantics, and show that, even though rewritability under this semantics is different from the usual rewritability, it continues to be locally source-dependent and thus our non-rewritability tools are applicable to it.

Organization. Basic notions related to data exchange, universal solutions, and cores, are presented in Section 2. In Section 3, we define query rewriting, and show the relationship between query rewriting over the core, the canonical universal solution, and the source. In Section 4 we define the notion of being locally source-dependent, prove it for the core and the canonical universal solution and show how it can be used to prove non-rewritability of queries. In Section 5 we study the structural properties of both the canonical universal solution and the core, and show that these solutions preserve the local structure of the data: up to isomorphism in some restricted settings, and up to logical equivalence in the general setting. In Section 6, we show that our results are robust, in the sense that they can also be applied in the context of a different semantics for query answering in data exchange. We finish the paper with some conclusions and suggestions for future work.

2. Preliminaries

A *schema* $\mathbf{R}$ is a finite sequence $\langle R_1, \ldots, R_m \rangle$ of relation symbols, with each R_i having a fixed arity n_i. An *instance* I *of* $\mathbf{R}$ assigns to each relation symbol R_i of $\mathbf{R}$ a finite n_i-ary relation $I(R_i)$. The *domain* $\text{dom}(I)$ of instance I is the set of all elements that occur in any of the relations $I(R_i)$.[1] The set of all instances of schema $\mathbf{R}$ is denoted by $\mathsf{Inst}(\mathbf{R})$. An instance J of $\mathbf{R}$ is a *subinstance* of I if $J(R_i) \subseteq I(R_i)$, for every i. If at least one of the inclusions is proper, we refer to J as a *proper subinstance* of I.

We will sometimes abuse the notation and use R_i to denote both the relation symbol and the relation $I(R_i)$ that interprets it. Given a tuple t occurring in a relation R, we denote by $R(t)$ the association between t and R and call it a *fact*. We often represent an instance by its set of facts.

An *m-ary query* Q over schema $\mathbf{R}$, where $m \geq 0$, is a map that associates with each $I \in \mathsf{Inst}(\mathbf{R})$ a subset of $\text{dom}(I)^m$, denoted by $Q(I)$, such that Q is closed under isomorphisms (that is, if f is an isomorphism from I_1 to I_2, and if $t \in Q(I_1)$, then $f(t) \in Q(I_2)$).[2] We assume that 0-ary queries (or *Boolean* queries) are maps from $\mathsf{Inst}(\mathbf{R})$ to the Boolean values `true` and `false`. From now on, we assume all queries to be specified in first-order logic (FO), with which we assume familiarity.

Ehrenfeucht-Fraïssé games. The *quantifier rank* of an FO formula ϕ, denoted by $\text{qr}(\phi)$, is the maximum depth of quantifier nesting in it.

Many proofs in this paper make use of *Ehrenfeucht-Fraïssé* (EF) games. This game is played in two instances, I_1 and I_2, of the same schema, by two players, the *spoiler* and the *duplicator*. There is a fixed nonnegative integer k of *rounds*. In round i the spoiler selects either an element c_i in $\text{dom}(I_1)$ or an element e_i in $\text{dom}(I_2)$; if the spoiler selects c_i in $\text{dom}(I_1)$ (resp., e_i in $\text{dom}(I_2)$) then the duplicator selects e_i in $\text{dom}(I_2)$ (resp., c_i in $\text{dom}(I_1)$). The duplicator *wins* if $\{(c_i, e_i) \mid i \leq k\}$ defines a partial isomorphism between I_1 and I_2. If the duplicator has a winning strategy, to win no matter how the spoiler plays, we write $I_1 \equiv_k I_2$. A classical result states that $I_1 \equiv_k I_2$ iff I_1 and I_2 agree on all FO sentences of quantifier rank at most k [Ehr61, Fra54] (cf. [Lib04]). Also, if $\bar{a}$ is an m-tuple in $\text{dom}(I_1)$ and $\bar{b}$ is an m-tuple in $\text{dom}(I_2)$, where $m \geq 0$, we write $(I_1, \bar{a}) \equiv_k (I_2, \bar{b})$ whenever the duplicator has a winning strategy, to win in k rounds no matter how the spoiler plays, but starting from position $(\bar{a}, \bar{b})$. Then $(I_1, \bar{a}) \equiv_k (I_2, \bar{b})$ iff $I_1 \models \phi(\bar{a}) \Leftrightarrow I_2 \models \phi(\bar{b})$ for every FO formula $\phi(\bar{x})$ of quantifier rank at most k.

It is well-known (cf. [Lib04]) that for a given schema, there are only finitely many FO formulae of quantifier rank k, up to logical equivalence. The *rank-k type* of an m-tuple $\bar{a}$ in an instance I is the set of all formulae $\phi(\bar{x})$ of quantifier rank at most k such that $I \models \phi(\bar{a})$. Given the above, there are only finitely many rank-k types, and each one of them is definable by an FO formula $\tau_k^{(I,\bar{a})}(\bar{x})$ of quantifier rank at most k (this formula $\tau_k^{(I,\bar{a})}(\bar{x})$ is the conjunction of the formulas $\phi(\bar{x})$ in its type).

EF games provide us with a useful tool to prove inexpressibility results for first-order logic. In fact, from previous remarks one can show that an m-ary query Q is not expressible in FO if and only if for every $k \geq 0$ there exist instances I_1 and I_2, and m-tuples $\bar{a}$ and $\bar{b}$ in $\text{dom}(I_1)$ and $\text{dom}(I_2)$, resp., such that $(I_1, \bar{a}) \equiv_k (I_2, \bar{b})$, the m-tuple $\bar{a}$ belongs to $Q(I_1)$, but the m-tuple $\bar{b}$ does not belong to $Q(I_2)$.

[1] An instance can be viewed as a special case of an $\mathbf{R}$-structure $\mathfrak{A}$ defined as $(A, R_1^{\mathfrak{A}}, \ldots, R_m^{\mathfrak{A}})$, where A is a set (the *universe*), and $R_i^{\mathfrak{A}} \subseteq A^{n_i}$ for each i. Thus, in the case of arbitrary structures, the universe may contain elements that are not present in any of the relations.

[2] If $t = (t_1, \ldots, t_m)$, then $f(t)$ is defined to be $(f(t_1), \ldots, f(t_m))$.

Data exchange setting. Let $\mathbf{S} = \langle S_1, \ldots, S_n \rangle$ and $\mathbf{T} = \langle T_1, \ldots, T_m \rangle$ be two schemas with no relation symbols in common. We refer to $\mathbf{S}$ as the *source* schema and to the S_i's as the source relation symbols. We refer to $\mathbf{T}$ as the *target* schema and to the T_j's as the target relation symbols. We denote by $\langle \mathbf{S}, \mathbf{T} \rangle$ the schema $\langle S_1, \ldots, S_n, T_1, \ldots, T_m \rangle$. Instances over $\mathbf{S}$ will be called source instances, while instances over $\mathbf{T}$ will be called target instances. If I is a source instance and J is a target instance, then (I, J) denotes an instance K over $\langle \mathbf{S}, \mathbf{T} \rangle$ such that $K(S_i) = I(S_i)$ and $K(T_j) = J(T_j)$, for $i \in [1, n]$ and $j \in [1, m]$.

A *source-to-target dependency* (std) is a sentence of the form

$$\forall \bar{x} \big(\phi_{\mathbf{S}}(\bar{x}) \to \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y}) \big), \tag{1}$$

where $\phi_{\mathbf{S}}(\bar{x})$ is an FO formula over $\mathbf{S}$, and $\psi_{\mathbf{T}}(\bar{x}, \bar{y})$ is a conjunction of atomic formulae over $\mathbf{T}$. We often write this std simply as $\phi_{\mathbf{S}}(\bar{x}) \to \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})$. For simplicity, we do not allow constants to occur anywhere inside an std. Many papers (e.g., [FKMP05, FKP05, GN08]) put an additional restriction that the formula $\phi_{\mathbf{S}}$ be a conjunction of atomic formulas (that contains all of the variables of $\bar{x}$); we do not impose this restriction here. However, we do impose a safety condition on stds. Recall that an FO-sentence is domain independent if its truth value in an instance I depends only on the tuples of I, and not on the underlying domain (see [Fag82] for a formal definition). In this paper, we assume that every std is domain independent.

Example 2.1. The formula $\forall x(\neg S(x) \to T(x))$ is not domain independent, while the formula $\forall x(R(x) \wedge \neg S(x) \to T(x))$ is domain independent. In fact, the latter is an example of an std that is not equivalent to any finite set of stds where the left-hand sides are conjunctions of atomic formulas. $\square$

Definition 2.2 (Data exchange setting). *A* data exchange setting *is a triple $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$, where $\mathbf{S}$ is a source schema, $\mathbf{T}$ is a target schema, and Σ_{st} is a set of source-to-target dependencies. The* data exchange problem *associated with $\mathcal{M}$ is the following: Given a source instance I, find a target instance J such that (I, J) satisfies each std in Σ_{st}. Such a J is called a* solution *for I under $\mathcal{M}$, or simply a* solution *for I if $\mathcal{M}$ is clear from the context.*

We denote by Const an infinite set of all values that may occur in source instances, and, following the data exchange terminology [FKMP05, FKP05], we call those values *constants*. We shall denote constants by $a, b, c, \ldots$, and tuples of constants by $\bar{a}, \bar{b}$, etc. In addition, we also assume an infinite set Var of elements, disjoint from Const. Elements of Var are called *nulls* [FKMP05, FKP05], and they are used to help populate target instances. Nulls are often denoted by $n, n_1, n_2, \ldots$, and tuples of nulls by $\bar{n}, \bar{n}_1$, etc. Sometimes we also denote nulls by uppercase letters $X, Y, Z, \ldots$, in order to clearly distinguish them from constants. That is, the domain of a target instance comes from $\mathsf{Const} \cup \mathsf{Var}$. If I is an instance with values in $\mathsf{Const} \cup \mathsf{Var}$, then $\mathsf{Const}(I)$ denotes the set of all constants occurring in relations in I, and $\mathsf{Var}(I)$ denotes the set of nulls occurring in relations in I. From now on, we assume that there is a way to distinguish constants from nulls. For example, we may assume that the target schema $\mathbf{T}$ contains an auxiliary predicate C whose interpretation is $\mathsf{Const}(I)$ (which can be thought as the `IS NOT NULL` condition in SQL). This assumption is used in many of the proofs presented in the paper.

Two important subclasses of data exchange settings have been identified in the literature [FKMP05, FKP05], inspired by the *local-as-view (LAV)* and *global-as-view (GAV)* classes of data integration problems (cf. Lenzerini [Len02]):

- *LAV setting*: Each dependency in Σ_{st} is of the form $S(\bar{x}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})$, where S is some relation symbol in the source schema $\mathbf{S}$, and, as before, $\psi_{\mathbf{T}}(\bar{x}, \bar{y})$ is a conjunction of atomic formulae over the target schema $\mathbf{T}$.

- *GAV setting*: Each dependency in Σ_{st} is of the form $\phi_{\mathbf{S}}(\bar{x}) \rightarrow T(\bar{x})$, where T is some relation symbol in the target schema $\mathbf{T}$. If $\phi_{\mathbf{S}}(\bar{x})$ is a conjunctive query (that is, a conjunction of atomic formulas), we speak of the GAV(CQ) setting.

Universal solutions, canonical universal solution, and core. The next example shows that there can be more than one solution for a source instance I in a data exchange setting.

Example 2.3. Consider a LAV data exchange setting in which $\mathbf{S} = \{M(\cdot,\cdot), N(\cdot,\cdot)\}$, $\mathbf{T} = \{P(\cdot,\cdot,\cdot), Q(\cdot,\cdot)\}$ and Σ_{st} contains the following stds:

$$\begin{array}{rcl} M(x,y) & \rightarrow & \exists w \exists z (P(x,y,z) \wedge Q(w,z)), \\ N(x,y) & \rightarrow & \exists z P(x,y,z). \end{array}$$

Notice that the stds in Σ_{st} do not completely specify the target. Suppose we have a source instance $I = \{M(a,b),\ N(a,b)\}$. One possible solution is:

$$J \;=\; \{P(a,b,n_1), P(a,b,n_2), Q(n_3,n_1)\},$$

where $n_1, n_2, n_3 \in \mathsf{Var}$. Another solution, containing no nulls, is

$$J' \;=\; \{P(a,b,a), Q(b,a)\}.$$

Solution J' can be considered to be less general than J, as it gives the same value to variables x and z even though this requirement is not imposed by the specification. $\square$

It has been argued in the literature [FKMP05], that the solution J' in the previous example should not be used for data exchange. In contrast, J is a better possible solution, as it contains no more and no less than what the specification requires. We formalize this intuition next.

Universal solutions. Here we provide an algebraic specification that selects, among all possible solutions, a special class of solutions called *universal*. These universal solutions have several good properties that justify why they are the preferred solutions in data exchange. Before presenting the main definition of this section, we introduce the useful concept of *homomorphism* between instances with values in $\mathsf{Const} \cup \mathsf{Var}$.

Definition 2.4 (Homomorphisms). *Let J, J' be two instances of $\mathbf{R}$ with values in $\mathsf{Const} \cup \mathsf{Var}$. A* homomorphism *$h : J \rightarrow J'$ is a mapping from $\mathsf{Const}(J) \cup \mathsf{Var}(J)$ to $\mathsf{Const}(J') \cup \mathsf{Var}(J')$ such that*

1. *$h(c) = c$ for all $c \in \mathsf{Const}(J)$, and*
2. *$\bar{t} = (t_1, \ldots, t_k) \in J(R)$ implies $h(\bar{t}) = (h(t_1), \ldots, h(t_k)) \in J'(R)$ for all $R \in \mathbf{R}$.*

Furthermore, we say that J and J' are homomorphically equivalent *if there are homomorphisms $h : J \rightarrow J'$ and $h' : J' \rightarrow J$.*

Now we are ready to introduce the notion of universal solution [FKMP05].

Definition 2.5 (Universal solution). *If I is a source instance in a data exchange setting, then a* universal solution *for I is a solution J for I such that for every solution J' for I, there exists a homomorphism $h : J \to J'$.*

Example 2.6. The solution J' in Example 2.3 is not universal, since there is no homomorphism from J' to J. This fact makes precise our earlier intuition that J' is less general than J. On the other hand, it can be easily shown that J has homomorphisms to all solutions, and thus, is universal. □

It has been shown that universal solutions possess good properties that justify selecting them (as opposed to arbitrary solutions) for the semantics of the data exchange problem [FKMP05]. A universal solution is more general than an arbitrary solution because, by definition, it can be homomorphically mapped into that solution. This implies that all universal solutions are homomorphically equivalent. Thus, in a certain sense, each universal solution precisely embodies the space of solutions.

Note that checking the condition in Definition 2.5 requires implicitly the ability to check the (infinite) space of solutions. Thus, it is not clear at first hand, to what extent the notion of universal solution is a computable one. Next we introduce two particular universal solutions that can be computed efficiently.

Canonical universal solution. To deal with the problem of computing universal solutions, Fagin et al. [FKMP05] proposed to compute a special kind of universal solution, called a *canonical* universal solution. The algorithm is based on the classical *chase* procedure [BV84, MMS79], but we shall define canonical universal solutions directly.

In the following we show how to compute the canonical universal solution of a source instance I in a data exchange setting $(\mathbf{S}, \mathbf{T}, \Sigma_{st})$. For each std

$$\phi_{\mathbf{S}}(\bar{x}, \bar{y}) \;\rightarrow\; \exists \bar{w}\, (R_1(\bar{x}_1, \bar{w}_1) \wedge \cdots \wedge R_k(\bar{x}_k, \bar{w}_k)) \;\in \Sigma_{st},$$

where all R_i's are in $\mathbf{T}$, the variables in $\bar{w}$ are exactly the variables that appear in some $\bar{w}_i$, and the variables in $\bar{x}$ are exactly the variables that appear in some $\bar{x}_i$, and for each tuple $\bar{a}$ of length $|\bar{x}|$, find all tuples $\bar{b}_1, \ldots, \bar{b}_m$ such that $I \models \phi_{\mathbf{S}}(\bar{a}, \bar{b}_i)$, $i \in [1, m]$. Then choose m tuples $\bar{n}_1, \ldots, \bar{n}_m$ of length $|\bar{w}|$ of fresh distinct null values over Var. Suppose that (i) σ is the mapping from the variables in $\bar{x}$ to the constants in $\bar{a}$ such that $\sigma(\bar{x}) = \bar{a}$, and (ii) for each j with $1 \leq j \leq m$, it is the case that σ_j is the mapping from the variables in $\bar{w}$ to the null values in $\bar{n}_j$ such that $\sigma_j(\bar{w}) = \bar{n}_j$. Relation R_i in the canonical universal solution for I, $i \in [1, k]$, contains tuples

$$(\sigma(\bar{x}_i), \sigma_j(\bar{w}_i)), \text{ for each } j \in [1, m]\,.$$

Furthermore, the relation R_i in the canonical universal solution for I contains only tuples that are obtained by applying this algorithm.

Example 2.7. The canonical universal solution (up to isomorphism) for I in Example 2.3, is $J = \{P(a, b, n_1), P(a, b, n_2), Q(n_3, n_1)\}$. □

Definition 2.8 (Transformation $\mathfrak{F}_{\text{univ}}$). *For a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$, we denote by $\mathfrak{F}^{\mathcal{M}}_{\text{univ}}$ (or simply by $\mathfrak{F}_{\text{univ}}$ if $\mathcal{M}$ is clear from the context) the transformation from $\mathsf{Inst}(\mathbf{S})$ to $\mathsf{Inst}(\mathbf{T})$, such that $\mathfrak{F}^{\mathcal{M}}_{\text{univ}}(I)$ is the canonical universal solution for I under $\mathcal{M}$.*

This definition differs from the standard one in the literature [FKMP05], where a canonical universal solution is obtained by using the classical chase procedure. Since the result of the chase is not necessarily unique (it depends on the order in which the chase steps are applied [FKMP05]), there may be multiple non-isomorphic canonical universal solutions under such a definition. On the other hand, it is clear that under our definition of canonical universal solution, the canonical universal solution is unique up to a renaming of nulls.

The next proposition states some useful properties of the transformation $\mathfrak{F}_{\text{univ}}$.

Proposition 2.9. *[FKMP05] Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a fixed data exchange setting. Then $\mathfrak{F}^{\mathcal{M}}_{\text{univ}}(I)$ is a universal solution for I, for each instance I of $\mathbf{S}$. Furthermore, there is a polynomial-time algorithm that computes $\mathfrak{F}^{\mathcal{M}}_{\text{univ}}(I)$ for each instance I of $\mathbf{S}$.*

The core. A reason one wants to compute a specific solution for the data exchange problem is to be able to evaluate queries over the target schema. It is easy to see that universal solutions need not be isomorphic, and thus any decision to choose any particular one is somewhat arbitrary. To deal with this problem, it has been proposed to use the *core* of the universal solutions [FKP05].

Definition 2.10 (Core). *A subinstance J of an instance I is called a* core *of I if there is a homomorphism from I to J, but there is no homomorphism from I to a proper subinstance of J.*

It follows from the work of Hell and Nešetřil [HN92, HN04] that every instance has a unique core (up to isomorphism). It is shown, in particular, that we can take the homomorphism h that maps an instance I into a core J' of I to be a *retraction*, that is, a homomorphism where $h(x) = x$ for x in $\text{dom}(J)$. It is also shown that every core J^* of J is an *induced* subinstance of J, that is, the restriction of J to the facts that consist uniquely of elements in J^* is precisely J^*. Furthermore, it is possible to prove that every universal solution has the same core up to isomorphism [FKP05].

Example 2.11. In Example 2.3, $J^* = \{P(a, b, n_1),\ Q(n_3, n_1)\}$ is the unique core (up to isomorphism) of the universal solutions. In fact, J^* is a subinstance of the canonical universal solution $J = \{P(a, b, n_1), P(a, b, n_2), Q(n_3, n_1)\}$, and h defined as the identity on all elements but n_2, and $h(n_2) = n_1$, is a homomorphism from J to J^*. Furthermore, there is no homomorphism from J to a proper subinstance of J^*. $\square$

Definition 2.12 (Transformation $\mathfrak{F}_{\text{core}}$). *For a data exchange setting $(\mathbf{S}, \mathbf{T}, \Sigma_{st})$, we denote by $\mathfrak{F}^{\mathcal{M}}_{\text{core}}$ (or simply by $\mathfrak{F}_{\text{core}}$ if $\mathcal{M}$ is clear from the context) the transformation from $\mathsf{Inst}(\mathbf{S})$ to $\mathsf{Inst}(\mathbf{T})$, such that $\mathfrak{F}^{\mathcal{M}}_{\text{core}}(I)$ is the core of the universal solutions for I.*

The next proposition states some useful properties of the transformation $\mathfrak{F}_{\text{core}}$.

Proposition 2.13. *[FKP05] Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a fixed data exchange setting. Then $\mathfrak{F}^{\mathcal{M}}_{\text{core}}(I)$ is a universal solution for I, for each instance I of $\mathbf{S}$. Furthermore, there is a polynomial-time algorithm that computes $\mathfrak{F}^{\mathcal{M}}_{\text{core}}(I)$ for each instance I of $\mathbf{S}$.*

It can also be proved that the core has the *smallest* size among all universal solutions [FKP05]. It is argued that this property confirms that the core is an ideal candidate for the "best" universal solution, at least in terms of the space required to materialize it.

3. Query Rewriting

In this section, we study query rewriting in data exchange. Suppose we have a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$, and a query Q over the *target* schema $\mathbf{T}$. Since there are many possible solutions to the data exchange problem, the standard approach is to define the semantics of Q in terms of *certain answers.*

Definition 3.1 (Certain answers). *Let $(\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting. Let Q be an m-ary query, where $m \geq 0$, over schema $\mathbf{T}$, and I a source instance. The* certain answers of Q with respect to I under $\mathcal{M}$, *denoted by $\underline{\text{certain}}_{\mathcal{M}}(Q, I)$, is the set of all m-tuples $\bar{a}$ such that for every solution J for I under $\mathcal{M}$, we have that $\bar{a} \in Q(J)$. Equivalently,*

$$\underline{\text{certain}}_{\mathcal{M}}(Q, I) \quad = \bigcap_{J \text{ is a solution for } I} Q(J).$$

If Q is a Boolean query, then $\underline{\text{certain}}_{\mathcal{M}}(Q, I) = \texttt{true}$ if and only if $Q(J) = \texttt{true}$ for each solution J for I.

Note that if a tuple belongs to the set of certain answers of a query, then it does not contain any null values.

The definition of $\underline{\text{certain}}_{\mathcal{M}}(Q, I)$ refers to potentially infinitely many solutions. However, we need to compute it based on some specific solution. That is, given a transformation $\mathfrak{F} : \mathsf{Inst}(\mathbf{S}) \to \mathsf{Inst}(\mathbf{T})$, we would like to be able to compute $\underline{\text{certain}}_{\mathcal{M}}(Q, I)$ as $Q'(\mathfrak{F}(I))$ for some query Q'. This is precisely the notion of rewritability of Q over $\mathfrak{F}$. Formally, it is defined as follows.

Definition 3.2 (Query rewriting). *Given a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$, a mapping $\mathfrak{F} : \mathsf{Inst}(\mathbf{S}) \to \mathsf{Inst}(\mathbf{T})$ and an m-ary query Q over $\mathbf{T}$, we say that Q is* rewritable over $\mathfrak{F}$ under $\mathcal{M}$ *(or simply rewritable over $\mathfrak{F}$ if $\mathcal{M}$ is clear from the context) if there exists an* FO *formula $\phi(x_1, \ldots, x_m)$ over $\mathbf{T}$ such that $\underline{\text{certain}}_{\mathcal{M}}(Q, I) = \{(a_1, \ldots, a_m) \mid \mathfrak{F}(I) \models \phi(a_1, \ldots, a_m)\}$, for every instance I of $\mathbf{S}$.*

We shall refer to a query as being rewritable over the canonical universal solution if it is rewritable over $\mathfrak{F}_{\text{univ}}$, and rewritable over the core if it is rewritable over $\mathfrak{F}_{\text{core}}$.

In the first case, we refer to ϕ as a *rewriting of Q over the canonical universal solution*, and in the second case we refer to ϕ as a *rewriting of Q over the core.*

As the next proposition shows, rewritability of queries over these transformations is undecidable in general.

Proposition 3.3. *Given a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ and a query Q over $\mathbf{T}$ specified in* FO, *it is undecidable whether Q is rewritable over the canonical universal solution (resp., over the core) under $\mathcal{M}$.*

We delay the proof of Proposition 3.3 to Section 4.4, where we will be able to use locality tools in order to prove that a query is not rewritable over the canonical universal solution or the core. The application of those tools will allow us to simplify the proof of Proposition 3.3 considerably.

For some classes of queries we know rewritability results. For example, if $Q(x_1,\ldots,x_m)$ is a disjunction of conjunctive queries with free variables $x_1,\ldots,x_m$, and if we let $Q'(x_1,\ldots,x_m)$ be the rewriting $Q(x_1,\ldots,x_m)\wedge\bigwedge_{i\in[1,m]}C(x_i)$, then we have $\underline{\text{certain}}(Q,I) = Q'(J)$, where J is an arbitrary universal solution (recall that C is the predicate whose interpretation is $\mathsf{Const}(J)$). Thus, unions of conjunctive queries are rewritable over both $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ [FKMP05]. However, it has been shown that if inequalities of the form $x \neq y$ are allowed in conjunctive queries, rewritings need not exist even in LAV settings [FKMP05].

While a lot of attention has been paid to identifying cases when the canonical universal solution and the core can be obtained in polynomial time [FKMP05, FKP05, GN08], not much is known on how these two solutions compare in terms of query rewriting. Even less is known about how rewritings over these solutions compare with a different class of rewriting; namely, rewriting over the source. This type of rewriting is common in data integration and consistent query answering over inconsistent databases (e.g., see [ABC99, DL97]). Formally, given a data exchange setting $\mathcal{M} = (\mathbf{S},\mathbf{T},\Sigma_{st})$, and an m-ary query Q over $\mathbf{T}$, we say that Q is *rewritable over the source under* $\mathcal{M}$ if there exists an m-ary query Q' over $\mathbf{S}$ specified in FO, such that $\underline{\text{certain}}_{\mathcal{M}}(Q,I) = Q'(I)$ for every instance I of $\mathbf{S}$. Similarly to before, we refer to Q' as a *rewriting of* Q *over the source.*

Our main result describes precise relationships between these classes. Recall that we assume queries to be specified in FO.

Theorem 3.4. *The following holds:*

- *The class of queries rewritable over the core is strictly contained in the class of queries rewritable over the canonical universal solution.*
- *The class of queries rewritable over the canonical universal solution is strictly contained in the class of queries rewritable over the source.*

In the rest of the section we prove this result. The first item is proved in Section 3.1 and the second in Section 3.2.

3.1. Core versus canonical universal solution

Here we prove that the class of queries rewritable over the core is strictly contained in the class of queries rewritable over the canonical universal solution. This is shown by a theorem and a proposition below.

Theorem 3.5. *Every query that is rewritable over the core is also rewritable over the canonical universal solution.*

The next proposition says that the converse does not hold.

Proposition 3.6. *There is a query specified in* FO *that is rewritable over the canonical universal solution, but not rewritable over the core.*

In the rest of the section we prove these results. We start with Theorem 3.5, which we shall show follows from Lemma 3.7 below.

Lemma 3.7. *Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting and $\phi(\bar{x})$ an FO formula over $\mathbf{T}$. Then there exists an FO formula $\phi'(\bar{x})$ such that for every instance I of $\mathbf{S}$ and every $\bar{a} \in \text{dom}(\mathfrak{F}_{\text{univ}}(I))^m$ $(m \geq 0)$, it is the case that $\mathfrak{F}_{\text{univ}}(I) \models \phi'(\bar{a})$ if and only if there is a core J' of $\mathfrak{F}_{\text{univ}}(I)$ such that $\bar{a} \in \text{dom}(J')^m$ and $J' \models \phi(\bar{a})$.*

We now show that Theorem 3.5 follows from Lemma 3.7. Let $\phi(\bar{x})$ be a rewriting of an m-ary query Q, where $m \geq 0$, over the core. Then, from the lemma above, there is an FO formula $\phi'(\bar{x})$ such that for every instance I of $\mathbf{S}$ and every $\bar{a} \in \text{dom}(\mathfrak{F}_{\text{univ}}(I))^m$, it is the case that $\mathfrak{F}_{\text{univ}}(I) \models \phi'(\bar{a})$ if and only if there is a core J' of $\mathfrak{F}_{\text{univ}}(I)$ such that $\bar{a} \in \text{dom}(J')^m$ and $J' \models \phi(\bar{a})$. But since tuples belonging to the certain answers of a query contain no null values, if $\mathfrak{F}_{\text{core}}(I) \models \phi(\bar{a})$ then $\bar{a}$ contains only elements in Const. Furthermore, all cores of an instance are isomorphic. Since isomorphisms have to be the identity on constants, we can prove that for every tuple $\bar{a}$ of constants,

$$\mathfrak{F}_{\text{core}}(I) \models \phi(\bar{a}) \;\Leftrightarrow\; \text{there is a core } J' \text{ of } \mathfrak{F}_{\text{univ}}(I) \text{ such that } J' \models \phi(\bar{a}) \;\Leftrightarrow\; \mathfrak{F}_{\text{univ}}(I) \models \phi'(\bar{a}).$$

We conclude that $\phi'(\bar{x}) \wedge \bigwedge_{x \text{ appears in } \bar{x}} C(x)$ is a rewriting of Q over the canonical universal solution.

In order to prove Lemma 3.7, the key ingredient is proving Claim 3.9 which states the following: There exists a FO formula $\text{Core}_{\mathsf{Var}}(x_1, \ldots, x_m)$ over the target schema $\mathbf{T}$ such that, for every source instance I and tuple $(n_1, \ldots, n_m) \in \text{dom}(\mathfrak{F}_{\text{univ}}(I))^m$, it is the case that $\mathfrak{F}_{\text{univ}}(I) \models \text{Core}_{\mathsf{Var}}(n_1, \ldots, n_m)$ if and only if $(n_1, \ldots, n_m)$ is a tuple of null values that belongs to some core of $\mathfrak{F}_{\text{univ}}(I)$. The proof of this fact uses induction on the number $m \geq 1$ of variables of the formula. The basis case $m = 1$ is handled in Claim 3.8, where we provide a characterization in terms of a set of properties – which are all FO definable – of when a null value in a universal solution J belongs to a core of J.

Fix a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$. We first need a necessary and sufficient condition for a null value to be in a core of a universal solution. We show such a condition in Claim 3.8. In this claim we use the following terminology. Let J be an instance of a target schema $\mathbf{T}$. Then we define the *Gaifman graph of the nulls of* J [FKMP05] to be an undirected graph in which (1) the nodes are all the nulls of J, and (2) there is an edge between two nulls whenever the nulls belong to the same tuple of some relation in J. Given a target structure J and a null value X of J, we define $\mathsf{block}(X)$ as:

$$\begin{aligned}\mathsf{block}(X) \;=\; \{Y \mid Y &\text{ is a null of } J \text{ that is in the} \\ &\text{connected component of } X \text{ in the Gaifman graph of the nulls of } J\},\end{aligned}$$

and we define the null-extension of X given J, denoted by $\mathsf{null\text{-}ext}(X, J)$, as the following subinstance of J. For every relation symbol R in $\mathbf{T}$ and tuple $t \in J(R)$, we have that t belongs to the interpretation of R in $\mathsf{null\text{-}ext}(X, J)$ if t contains a null in $\mathsf{block}(X)$. We note that if J is a canonical universal solution, then for every null X of J, the number of tuples in $\mathsf{null\text{-}ext}(X, J)$ is bounded by the maximum number of conjuncts that appear in the conclusions of the members of Σ_{st}. Also, given a subinstance J' of a target instance J, we define $\mathsf{null\text{-}ext}(J', J)$ as:

$$\mathsf{null\text{-}ext}(J', J) \;=\; J' \cup \bigcup_{\{X \,\mid\, X \text{ is a null of } J'\}} \mathsf{null\text{-}ext}(X, J).$$

Claim 3.8. *Let I be a source instance, J a universal solution for I and X a null value of J. Then there exists a core of J containing X if and only if for every pair J', J'' of target instances, there exists a homomorphism $h' : J'' \to J$ such that X is a null of $h'(J'')$ whenever*

1. *$J' \subseteq J$ and $|J'| \leq |\mathsf{null\text{-}ext}(X, J)|$,*

2. *there exists a homomorphism* h : $\mathsf{null\text{-}ext}(X, J) \rightarrow J'$ *such that* X *is not a null of* $h(\mathsf{null\text{-}ext}(X, J))$*, and*
3. $J' \subseteq J'' \subseteq \mathsf{null\text{-}ext}(J', J)$.

Proof: ($\Leftarrow$) We show that X is in some core of J. Assume that J^* is a core of J. As noted earlier, J^* is an induced subinstance of J. If X is a null value of J^*, then there is nothing to prove. Thus, assume also that X is not a null of J^*. The main idea behind the rest of the proof is showing that there exists a substructure J_A of J^* that can be "replaced" by an isomorphic substructure J_C of J that contains X, in order to obtain a new core of J in which X appears.

Since J^* is a core of J, there exists a homomorphism $h^* : J \rightarrow J^*$ that is the identity on J^* (here we are taking advantage of the fact, noted earlier, that the homomorphism mapping J to J^* can be taken to be a retraction). Define J' as $h^*(\mathsf{null\text{-}ext}(X, J))$ and J'' as $\mathsf{null\text{-}ext}(J', J^*)$, where $\mathsf{null\text{-}ext}(J', J^*)$ is well-defined since J' is contained in J^*. Then

(1) $J' \subseteq J$ and $|J'| \leq |\mathsf{null\text{-}ext}(X, J)|$;

(2) the restriction of h^* to $\mathsf{null\text{-}ext}(X, J)$ is a homomorphism from $\mathsf{null\text{-}ext}(X, J)$ to J' such that X is not a null of $h^*(\mathsf{null\text{-}ext}(X, J))$ (since X is not a null of J^*); and

(3) $J' \subseteq J'' \subseteq \mathsf{null\text{-}ext}(J', J)$ (where the first inclusion follows from the definition of $\mathsf{null\text{-}ext}(J', J^*)$, and the second inclusion holds since $J^* \subseteq J$, and so $J'' = \mathsf{null\text{-}ext}(J', J^*) \subseteq \mathsf{null\text{-}ext}(J', J)$).

By the hypothesis, these three conditions imply that there exists a homomorphism $h' : J'' \rightarrow J$ such that X is a null of $h'(J'')$.

Let A be the set of constants and null values of J'', and let J_A be the subinstance of J induced by A. Now J'' is a subinstance of J^*, because $J'' = \mathsf{null\text{-}ext}(J', J^*)$, and it is always true by definition that $\mathsf{null\text{-}ext}(J_1, J_2)$ is a subinstance of J_2. Moreover, we now show that J_A is also a subinstance of J^*. In fact, we just noted that J'' is a subinstance of the core J^* of J, which is an induced substructure of J. Hence every fact in J_A, which is the subinstance of J induced by the elements in J'', must also belong to J^*. Next we show that there is a homomorphic image J_C of J_A such that (i) J_C is a subinstance of J, (ii) J_C contains the null value X, and (iii) J_A and J_C are isomorphic. We shall show that J_A can be replaced in J^* by J_C to construct a core containing null X (as suggested at the beginning of the proof).

Consider the homomorphism $h' : J'' \rightarrow J$. Clearly, h' is well-defined on J_A since every element of J_A belongs to A, and hence to J''. We now prove that h' is also a homomorphism from J_A into J, i.e. $h'(J_A)$ is a subinstance of J. We start by proving that if t is a fact in $J_A \setminus J''$, then t does not mention any null, i.e. it consists only of constants. Indeed, let t be a fact of $J_A \setminus J''$. Assume for the sake of contradiction that t mentions a null Y. Then Y also belongs to J'' (because J_A is the substructure of J induced by the elements of J''). Since $J'' = \mathsf{null\text{-}ext}(J', J^*)$, we have by definition that every fact in J^* that mentions a null in J'' also belongs to J''. But we proved above that J_A is a subinstance of J^*, and hence since t belongs to J_A (and so to J^*) and mentions null Y it also belongs to J'', which is a contradiction.

We now show that $h'(J_A)$ is a subinstance of J. Since h' is a homomorphism from J'' to J, we only have to show that if t is a fact in $J_A \setminus J''$ then the image of t under h' belongs to J. Let t be an arbitrary fact in $J_A \setminus J''$. Then, as we proved above, t consists only of constants. Thus, the image of t under h' is precisely t. By definition, this fact belongs to J.

Let us define J_C as the homomorphic image of J_A under h'. Thus, (i) J_C is a subinstance of J and (ii) J_C contains the null value X (since X belongs to $h'(J'')$ and J'' is a subinstance of J_A). We prove next that J_A and J_C are isomorphic, and later that this implies that J_A can be replaced in J^* by J_C to construct a core containing null X (as suggested before).

We start by proving that the homomorphism $h^* : J \to J^*$ maps $J_C = h'(J_A)$ into J_A. On the contrary, assume that this is not the case. We shall show that this leads to a contradiction.

Let us define a function $f : J^* \to J^*$ as follows:

$$f(x) = \begin{cases} h^*(h'(x)) & \text{if } x \text{ is an element of } J_A, \\ x & \text{if } x \text{ is an element of } J^* \setminus J_A. \end{cases} \tag{2}$$

By $J^* \setminus J_A$, we mean the result of removing those facts from J^* that are in J_A. We observe that f is the identity outside J_A and is the composition $h^* \circ h'$ on J_A. Furthermore, f is well-defined since if an element x appears in some fact of J_A and in some fact of $J^* \setminus J_A$, then as we now show, x is a constant and, therefore, $h^*(h'(x)) = x$. The reason why x is a constant is the following. Assume that x is a null value in J_A. Then x is in J'', by construction of J_A. However, no null value in J'' can also belong to a fact in $J^* \setminus J''$. This is, as before, because $J'' = \mathsf{null\text{-}ext}(J', J^*)$, and hence, by definition, every fact in J^* that mentions a null in J'' also belongs to J''.

In the search for our contradiction we show that $f : J^* \to J^*$ is a homomorphism. Let t be a fact of J^*. If t is a fact of $J^* \setminus J_A$, then $f(t) = t$, and, therefore, $f(t)$ is a fact of J^*. Otherwise, t is a fact of J_A and, hence, $f(t) = h^*(h'(t))$. Since h' is a homomorphism from J_A to J, we have that $h'(t)$ is a fact of J. Thus, given that h^* is a homomorphism from J to J^*, we have that $h^*(h'(t))$ is a fact of J^*. We conclude that $f(t) = h^*(h'(t))$ is a fact of J^*. So indeed, $f : J^* \to J^*$ is a homomorphism.

In the search for our contradiction we also prove that every constant in $J_C = h'(J_A)$ belongs to J_A. Assume, on the contrary, that there is a constant c in J_C that does not belong to J_A. Then there is a null Y in J_A that is mapped to c by h'. As we just noted, the mapping f defined in (2) is a homomorphism from J^* to J^*. Then $f(J_A) = h^*(h'(J_A))$ contains strictly fewer nulls than J_A, and hence f maps J^* into a subinstance of itself with strictly fewer nulls. This contradicts the fact that J^* is a core.

Recall that we are assuming, for the sake of contradiction, that h^* is not a homomorphism from J_C to J_A. As we now show, this implies that there exists a null value Y of J_C such that $h^*(Y)$ is not an element of J_A. Indeed, assume on the contrary that every null value in $J_C = h'(J_A)$ is mapped into J_A by h^*. We prove then that h^* is a homomorphism from J_C to J_A. Let $t = R(p_1, \ldots, p_n)$ be a fact of J_C. We now show that $h^*(p_i)$ belongs to J_A for each i with $1 \leq i \leq n$. In fact, if p_i is a null this holds by hypothesis, and if p_i is a constant then $h^*(p_i) = p_i$, and we noted before that each constant that appears in $J_C = h'(J_A)$ also appears in J_A. We prove next that $h^*(t)$ belongs to J_A. But this is easy since $h^*(t)$ belongs to J^* (because $h^* : J \to J^*$ is a homomorphism), every element of $h^*(t)$ belongs to J_A, and by definition J_A is an induced substructure of J (and hence also of J^*).

Thus, there exists a null value Y of J_C such that $h^*(Y)$ is not an element (constant or null) of J_A. We know that there exists a null value Z of J_A such that $h'(Z) = Y$ (because every element of J_C is the image under h' of an element in J_A), and, hence, it must be the case that $f(Z) = h^*(h'(Z)) = h^*(Y)$ is not an element of J_A. Let us denote $f(Z)$ by W. Since W is not in J_A, we have $f(W) = W$, by definition of f. Hence, $f(Z) = f(W)$, and $Z \neq W$ since Z is in J_A and W is not. Therefore f is not one-to-one, which gives us our desired contradiction, because f is a homomorphism from the core J^* into itself.

We prove next that J_A and J_C are isomorphic. We start by showing that J_A is a core. In fact, we have explained above that no fact in $J^* \setminus J_A$ can share a null value with a fact of J_A. Or, in other words, the only elements shared by facts in J_A and facts in $J^* \setminus J_A$ are constants. Thus, if J_A were not a core, that is, if there were a homomorphism h'' mapping J_A to a proper instance of itself, then the mapping $h''' : J^* \to J^*$, defined as $h'''(x) = h''(x)$ if x is in J_A and $h'''(x) = x$ if x is in $J^* \setminus J_A$, would also be a homomorphism from J^* into a proper subinstance of itself. This is a contradiction because J^* is a core.

The fact that J_A is a core implies that h' sends distinct null values in J_A to distinct null values in J_C. Assume otherwise. Then, given that h^* is a homomorphism from J_C into J_A, it must be the case that $h^* \circ h'$ maps J_A into a subinstance of itself with strictly fewer nulls, which contradicts the fact that J_A is a core. Since J_C is a homomorphic image of J_A, we conclude that J_A and J_C are isomorphic. We show next that this implies that J_A can be replaced in J^* by J_C, in order to construct a core of J that contains null value X.

We start by noticing that no null value in $J_C = h'(J_A)$ belongs to $J^* \setminus J_A$. Assume for the sake of contradiction that null value Y in J_C is mentioned in a fact of $J^* \setminus J_A$. By definition, Y is the homomorphic image under h' of some null value Z in J_A. Recall that we proved before that the mapping $f : J^* \to J^*$ defined in (2) is a homomorphism. Since Y is in $J^* \setminus J_A$, and Z is in J_A, and since we showed that no null is in both $J^* \setminus J_A$ and J_A, it follows that Y and Z are distinct. Thus, f maps the distinct null values Y and Z to the same element Y, and, hence, $f(J^*)$ has strictly fewer null values than J^*. This contradicts the fact that J^* is a core.

Finally, let J^*_X be the instance that is obtained from J^* by removing all facts in J_A and then adding all facts in $J_C = h'(J_A)$. Clearly, X appears in J^*_X. We prove next that J^*_X is a core of J, which finishes the first direction of the proof. In order to prove this it is enough to prove that J^* and J^*_X are isomorphic, which is what we do next.

Consider the mapping $f^* : J^* \to J$ defined as:

$$f^*(x) = \begin{cases} h'(x) & \text{if } x \text{ is an element of } J_A, \\ x & \text{if } x \text{ is an element of } J^* \setminus J_A. \end{cases}$$

First of all, f^* is well-defined since we have proved that the only elements that are shared by facts in J_A and $J^* \setminus J_A$ are constants (and, thus, $h'(x) = x$ for each such element x). In addition, it is easy to see that f^* is a homomorphism, and, further, $f^*(J^*) = J^*_X$. Thus, the homomorphic image of J^* under f^* is J^*_X. We prove next that no two distinct null values in J^* are mapped to the same element in J^*_X by f^*. Assume otherwise. Then there are distinct nulls Y and Z in J^* such that $f^*(Y) = f^*(Z)$. We consider four cases:

1. Z and Y belong to J_A. Then $h'(Z) = h'(Y)$, which contradicts the fact that h' is one-to-one.
2. Z and Y belong to $J^* \setminus J_A$. But f^* is the identity on $J^* \setminus J_A$, which contradicts the fact that $Y \neq Z$.
3. Y belongs to J_A and Z belongs to $J^* \setminus J_A$. But then $h'(Y) = Z$, which contradicts the fact proved above that no null value in J_C belongs to $J^* \setminus J_A$ (since $h'(Y)$ is in J_C and Z is in $J^* \setminus J_A$).
4. Z belongs to J_A and Y belongs to $J^* \setminus J_A$. This case is analogous to the previous one.

We conclude that f^* is an isomorphism from J^* to J^*_X (since it maps each constant to itself, each null to a distinct null, and is onto) and hence that J^*_X is a core. This finishes the first direction of the proof of the claim.

($\Rightarrow$) By contradiction, assume that there exists a core J^* of J containing null value X and that there exist target structures J', J'' such that:

1. $J' \subseteq J$ and $|J'| \leq |\text{null-ext}(X, J)|$,
2. there exists a homomorphism $h : \text{null-ext}(X, J) \to J'$ such that X is not a null of $h(\text{null-ext}(X, J))$,
3. $J' \subseteq J'' \subseteq \text{null-ext}(J', J)$, and
4. there is no homomorphism $h' : J'' \to J$ such that X is a null of $h'(J'')$.

Let h^* be a homomorphism from J to J^* that is the identity on J^* (again, we are taking advantage of the fact, noted earlier, that the homomorphism mapping J to J^* can be taken to be a retraction). Define function $f : J^* \to J^*$ as follows:

$$f(x) = \begin{cases} h^*(h(x)) & \text{if } x \text{ is an element of null-ext}(X, J^*), \\ x & \text{if } x \text{ is an element of } J^* \setminus \text{null-ext}(X, J^*). \end{cases}$$

We observe that f is the identity outside $\text{null-ext}(X, J^*)$ and is the composition of h and h^* on this extension. Furthermore, f is well-defined since we shall prove that if an element x is mentioned in some fact of $\text{null-ext}(X, J^*)$ and in some fact of $J^* \setminus \text{null-ext}(X, J^*)$, then x is a constant and, therefore, $h^*(h(x)) = x$. Indeed, by definition of $\text{null-ext}(X, J^*)$, every fact t of J^* that mentions a null that is also mentioned in a fact t' of $\text{null-ext}(X, J^*)$ must belong to $\text{null-ext}(X, J^*)$, and hence t cannot belong to $J^* \setminus \text{null-ext}(X, J^*)$.

Now we show that $f : J^* \to J^*$ is a homomorphism. Let t be a fact of J^*. If t is a fact of $J^* \setminus \text{null-ext}(X, J^*)$, then $f(t) = t$ and, therefore, $f(t)$ is a fact of J^*. Otherwise, t is a fact of $\text{null-ext}(X, J^*)$ and, hence, $f(t) = h^*(h(t))$. Since h is a homomorphism from $\text{null-ext}(X, J)$ to J' and $\text{null-ext}(X, J^*) \subseteq \text{null-ext}(X, J)$, we have that $h(t)$ is a fact of J'. Thus, given that J' is contained in J and h^* is a homomorphism from J to J^*, we have that $h^*(h(t))$ is a fact of J^*. We conclude that $f(t) = h^*(h(t))$ is a fact of J^*.

Next we show that f maps J^* into a proper subinstance of J^*, which contradicts the fact that J^* is a core. Since $\text{null-ext}(X, J^*) \subseteq \text{null-ext}(X, J)$ and h is a homomorphism from $\text{null-ext}(X, J)$ to J', we conclude that $h(\text{null-ext}(X, J^*)) \subseteq J'$. Given that $h^* : J \to J^*$ is a homomorphism and J'' is a subinstance of J (because $J' \subseteq J$ and $J'' \subseteq \text{null-ext}(J', J)$), the restriction of h^* to J'' is also a homomorphism from J'' to J (because $J^* \subseteq J$ and h^* is a homomorphism from J to J^*). Thus, by the fourth condition mentioned above, we have that X is not a null of $h^*(J'')$. Therefore, X is not a null of $h^*(h(\text{null-ext}(X, J^*)))$ since, by hypothesis, $h(\text{null-ext}(X, J^*)) \subseteq J' \subseteq J''$. Thus, given that $h^*(h(\text{null-ext}(X, J^*))) = f(\text{null-ext}(X, J^*))$ and f is the identity on $J^* \setminus \text{null-ext}(X, J^*)$, we have that X is not in the range of f. We conclude that f maps J^* into a proper subinstance of J^*, which is our desired contradiction. □

The second step to prove Lemma 3.7 is to show that for every $m \geq 1$, there exists an FO formula $\text{Core}(x_1, \ldots, x_m)$ that can be used to check whether an m-tuple $\bar{n}$ of nulls belongs to some core. More precisely, given a canonical universal solution J of some instance I of $\mathbf{S}$ and $m \geq 1$, the predicate $\text{Core}_{\mathsf{Var}}(x_1, \ldots, x_m)$ is defined as follows. For every $\{n_1, \ldots, n_m\} \subseteq \mathsf{Var}(J)$, it is the case that $J \models \text{Core}_{\mathsf{Var}}(n_1, \ldots, n_m)$ if and only if there exists a core J^* of J such that $\{n_1, \ldots, n_m\} \subseteq \mathsf{Var}(J^*)$. We prove that for each $m \geq 1$, the predicate $\text{Core}_{\mathsf{Var}}(x_1, \ldots, x_m)$ is definable in FO.

Claim 3.9. *For every $m \geq 1$, there exists an* FO *formula* $\text{Core}_{\mathsf{Var}}(x_1, \ldots, x_m)$ *over the target schema* $\mathbf{T}$ *such that, for every source instance I and tuple $(n_1, \ldots, n_m) \in \text{dom}(\mathfrak{F}_{\text{univ}}(I))^m$, it is the case that $\mathfrak{F}_{\text{univ}}(I) \models \text{Core}_{\mathsf{Var}}(n_1, \ldots, n_m)$ if and only if $(n_1, \ldots, n_m)$ is a tuple of null values that belongs to some core of $\mathfrak{F}_{\text{univ}}(I)$.*

Proof: By induction on m. For $m = 1$ we define $\text{Core}_{\mathsf{Var}}(x)$ by establishing that x does not belong to the interpretation of the predicate C that distinguishes constants and then using Claim 3.8. In fact, since we assume a fixed data exchange setting, all conditions in Claim 3.8 can be reduced to: (1) comparing cardinalities of subinstances up to a fixed size, (2) checking containment of instances of bounded size, and (3) checking the existence of homomorphisms between subinstances of bounded size, the predicate $\text{Core}_{\mathsf{Var}}(x)$ is clearly definable in FO (even though it is cumbersome to write the actual FO formula that defines it).

Assume now for the inductive step that $\text{Core}_{\mathsf{Var}}(x_1, \ldots, x_m)$ is definable in FO, where $m \geq 1$. We prove that $\text{Core}_{\mathsf{Var}}(x_1, \ldots, x_{m+1})$ is also definable in FO. The proof relies on the following crucial property:

(†) For each tuple $(n_1, \ldots, n_{m+1}) \in \mathsf{Var}(\mathfrak{F}_{\text{univ}}(I))^{m+1}$ such that $n_{m+1} \neq n_j$, for each j with $1 \leq j \leq m$, there exists a core of $\mathfrak{F}_{\text{univ}}(I)$ that contains $n_1, \ldots, n_{m+1}$ if and only if (1) there exists a core of $\mathfrak{F}_{\text{univ}}(I)$ that contains $n_1, \ldots, n_m$, and (2) if J' is the instance obtained from $\mathfrak{F}_{\text{univ}}(I)$ by replacing nulls $n_1, \ldots, n_m$ with fresh constants $c_1, \ldots, c_m$, respectively, then n_{m+1} is in a core of J'.

Before proving property (†) we show that $\text{Core}_{\mathsf{Var}}(x_1, \ldots, x_{m+1})$ can be defined in FO with the help of this property. In fact, if $\text{Core}_{\mathsf{Var}}(x)|_{(x_1,\ldots,x_m)}$ is the FO formula obtained from $\text{Core}_{\mathsf{Var}}(x)$ by replacing every occurrence of an atomic formula $C(y)$ with $C(y) \vee \bigvee_{i\in[1,m]} y = x_i$, then $\text{Core}_{\mathsf{Var}}(x_1, \ldots, x_{m+1})$ is defined by the following FO formula:

$$\Big(\big(\bigvee_{1\leq j\leq m} x_{m+1} = x_j\big) \wedge \text{Core}_{\mathsf{Var}}(x_1, \ldots, x_m)\Big) \vee$$
$$\Big(\big(\bigwedge_{1\leq j\leq m} x_{m+1} \neq x_j\big) \wedge \text{Core}_{\mathsf{Var}}(x_1, \ldots, x_m) \wedge \text{Core}_{\mathsf{Var}}(x_{m+1})|_{(x_1,\ldots,x_m)}\Big).$$

To finish the proof of Claim 3.9 we need to show that property (†) is true. In order to do that we first need to prove the following simple but important lemma. Recall that J' is the instance that is obtained from $\mathfrak{F}_{\text{univ}}(I)$ by replacing nulls $n_1, \ldots, n_m$ with fresh constants $c_1, \ldots, c_m$, respectively.

Lemma 3.10. *Assume that J_1 and J_2 are instances of $\mathbf{T}$ such that nulls $n_1, \ldots, n_m$ belong to both J_1 and J_2. Let J_1' and J_2' be instances obtained from J_1 and J_2, respectively, by replacing nulls $n_1, \ldots, n_m$ with fresh constants $c_1, \ldots, c_m$, respectively. Then:*

- *If there is a homomorphism $h : J_1 \to J_2$ that is the identity on $n_1, \ldots, n_m$, there is also a homomorphism $h' : J_1' \to J_2'$.*
- *If there is a homomorphism $h' : J_1' \to J_2'$, there is also a homomorphism $h : J_1 \to J_2$ that is the identity on $n_1, \ldots, n_m$.*

Proof: We now prove the first item. Assume that $h : J_1 \to J_2$ is a homomorphism that is the identity on $n_1, \ldots, n_m$. Let $h' : J_1' \to J_2'$ be the mapping such that $h'(x)$ is defined as follows for each element x in J_1': If $x = c_i$ for some i with $1 \leq i \leq m$, then $h'(x) = c_i$. Otherwise, $h'(x) = h(x)$, if $h(x) \neq n_i$ for each i with $1 \leq i \leq m$, and $h'(x) = c_i$, if $h(x) = n_i$ for some i with $1 \leq i \leq m$. Notice that h' is well-defined since each element x in J_1' that is not of the form c_i, for $1 \leq i \leq m$, belongs to J_1, and thus to the domain of h. Furthermore, h' is the identity on constants. We prove next that $h' : J_1' \to J_2'$ is a homomorphism. Assume that the fact $R(\bar{t})$ belongs to J_1'. Let $\bar{t}'$ be the tuple that is obtained from $\bar{t}$ by replacing each occurrence of the constant c_i, $1 \leq i \leq m$, with the null n_i. Then, by construction of J_1', it is the case that $R(\bar{t}')$ belongs to J_1, which implies that $R(h(\bar{t}'))$ belongs to J_2

(because $h : J_1 \to J_2$ is a homomorphism). Let $\bar{u}$ be the tuple that is obtained from $h(\bar{t}')$ by replacing each occurrence of the null n_i, for $1 \leq i \leq m$, with the constant c_i. Again, by construction, it must be the case that $R(\bar{u})$ belongs to J_2'. We prove below that $\bar{u} = h'(\bar{t})$, and hence that $R(h'(\bar{t}))$ belongs to J_2'.

Assume that $|\bar{u}| = p$. It is enough to prove that if u and t are the j-th projection $(1 \leq j \leq p)$ of $\bar{u}$ and $\bar{t}$, respectively, then $u = h'(t)$. We consider three cases: (1) $t = n_i$, for some i with $1 \leq i \leq m$. Then $h'(t) = c_i$. On the other hand, by construction the j-th projection of $\bar{t}'$ is n_i, and hence the j-th projection of $h(\bar{t}')$ is also n_i (because h is the identity on $n_1, \dots, n_m$). Finally, by construction the j-th projection u of $\bar{u}$ is c_i, which is precisely $h'(t)$, since $t = c_i$. (2) t belongs to J_1 and $h(t) \neq c_i$, for each i with $1 \leq i \leq m$. Then $h'(t) = h(t)$. On the other hand, by construction the j-th projection of $\bar{t}'$ is t (because $t \neq c_i$ for each i with $1 \leq i \leq m$), and hence the j-th projection of $h(\bar{t}')$ is $h(t)$. Finally, by construction the j-th projection u of $\bar{u}$ is $h(t)$ (because $h(t) \neq n_i$, for each i with $1 \leq i \leq m$), which is precisely $h'(t)$. (3) t belongs to J_1 and $h(t) = n_i$, for some i with $1 \leq i \leq m$. Then $h'(t) = c_i$. On the other hand, by construction the j-th projection of $\bar{t}'$ is t (because $t \neq c_i$ for each i with $1 \leq i \leq m$), and hence the j-th projection of $h(\bar{t}')$ is $h(t) = n_i$. Finally, by construction the j-th projection u of $\bar{u}$ is c_i (because $h(t) = n_i$), which is precisely $h'(t)$. Thus, $\bar{u} = h'(\bar{t})$ and, therefore, $R(h'(\bar{t}))$ is in J_2'. We conclude that $h'' : J_1' \to J_2'$ is a homomorphism.

For the second item assume that $h' : J_1' \to J_2'$ is a homomorphism. Let $h : J_1 \to J_2$ be the mapping such that $h(x)$ is defined as follows for each element x in J_1: If $x = n_i$ for some i with $1 \leq i \leq m$, then $h(x) = n_i$. Otherwise $h(x) = h'(x)$, if $h'(x) \neq c_i$ for each i with $1 \leq i \leq m$, and $h(x) = n_i$, if $h'(x) = c_i$ for some i with $1 \leq i \leq m$. Notice that h is well-defined since each element x in J_1 that is not of the form n_i, for $1 \leq i \leq m$, belongs to J_1', and thus to the domain of h'. Furthermore, h is the identity on $n_1, \dots, n_m$. We prove next that h is the identity on constants. Let d be a constant in J_1. Then d belongs to J_1' and $h'(d) \neq c_i$, for each i with $1 \leq i \leq m$, because h' is the identity on constants, and $d \neq c_i$, for each i with $1 \leq i \leq m$. Thus, $h(d) = h'(d) = d$. The proof that $h : J_1 \to J_2$ is a homomorphism is analogous to the one for the first item and left to the reader. $\square$

With the help of Lemma 3.10 we can now prove property (†). We start by proving ($\Rightarrow$). Assume that for some tuple $(n_1, \dots, n_{m+1}) \in \mathsf{Var}(\mathfrak{F}_{\text{univ}}(I))^{m+1}$ such that $n_{m+1} \neq n_j$, for each j with $1 \leq j \leq m$, it is the case that there exists a core J of $\mathfrak{F}_{\text{univ}}(I)$ that contains $n_1, \dots, n_{m+1}$. Then, clearly, condition (1) holds since J is a core of $\mathfrak{F}_{\text{univ}}(I)$ and it contains $n_1, \dots, n_m$. We prove next that condition (2) also holds. That is, we prove that if J' is the instance obtained from $\mathfrak{F}_{\text{univ}}(I)$ by replacing nulls $n_1, \dots, n_m$ with fresh constants $c_1, \dots, c_m$, respectively, then n_{m+1} is in a core of J'. Let J'' be the instance obtained from J by replacing nulls $n_1, \dots, n_m$ with constants $c_1, \dots, c_m$, respectively. Clearly, J'' contains element n_{m+1} because $n_{m+1} \neq n_j$, for each j with $1 \leq j \leq m$. We prove below that, in addition, J'' is a core of J'.

We start by proving that there exists a homomorphism from J' to J''. Since J is a core of $\mathfrak{F}_{\text{univ}}(I)$ there is a homomorphism $h : \mathfrak{F}_{\text{univ}}(I) \to J$ that is a retraction, i.e. h is the identity on J, and, in particular, on $n_1, \dots, n_m$. Then, by applying the first part of Lemma 3.10 to $J_1 = \mathfrak{F}_{\text{univ}}(I)$ and $J_2 = J$, we obtain that there exists a homomorphism h' from $J_1' = J'$ to the subinstance J_2' of J' that is obtained from $J_2 = J$ by replacing each occurrence of nulls $n_1, \dots, n_m$ with constants $c_1, \dots, c_m$, respectively. By definition, $J_2' = J''$, and, thus, h' is a homomorphism from J' to J''.

We prove next that J'' is a core. Assume otherwise. Then there is a homomorphism g from J'' into a proper subinstance J''' of itself. Since $c_1, \dots, c_m$ are constants in J'', they appear in J'''. Let J^* be the subinstance of J that is obtained from J''' by replacing constants $c_1, \dots, c_m$ with nulls $n_1, \dots, n_m$, respectively. Then, by applying the second part of Lemma 3.10 to $J_1 = J$, $J_2 = J^*$, $J_1' = J''$ and $J_2' = J'''$, we obtain that there is a homomorphism from $J_1 = J$ into $J_2 = J^*$. We prove next that J^*

is a proper subinstance of J, which is our desired contradiction since J is a core.

Since J''' is a proper subinstance of J'', there is a fact $R(\bar{t})$ in J'' but not in J'''. Let $\bar{t}'$ be the tuple that is obtained from $\bar{t}$ by replacing each occurrence of constant c_i, for $1 \leq i \leq m$, with null n_i. By definition, $R(\bar{t}')$ belongs to J but not to J^*. We conclude that J^* is properly contained in J.

We prove next the ($\Leftarrow$) direction of property (†). Assume that for some tuple $(n_1, \ldots, n_{m+1}) \in \mathsf{Var}(\mathfrak{F}_{\text{univ}}(I))^{m+1}$ such that $n_{m+1} \neq n_j$, for each j with $1 \leq j \leq m$, the following holds: (1) there exists a core J of $\mathfrak{F}_{\text{univ}}(I)$ that contains $n_1, \ldots, n_m$, and (2) if J' is the instance obtained from $\mathfrak{F}_{\text{univ}}(I)$ by replacing nulls $n_1, \ldots, n_m$ with fresh constants $c_1, \ldots, c_m$, respectively, then n_{m+1} belongs to some core J'' of J'. Since $c_1, \ldots, c_m$ are constants in J', they appear in J''. Let J^* be the instance obtained from J'' by replacing constants $c_1, \ldots, c_m$ with nulls $n_1, \ldots, n_m$, respectively. Clearly, J^* is a subinstance of $\mathfrak{F}_{\text{univ}}(I)$ that contains $n_1, \ldots, n_{m+1}$. We now prove that, in addition, J^* is a core of $\mathfrak{F}_{\text{univ}}(I)$.

We start by showing that J and J^* are homomorphically equivalent and that homomorphisms in both directions can be assumed to be the identity on $n_1, \ldots, n_m$. First of all, J is a core of $\mathfrak{F}_{\text{univ}}(I)$, and hence there is a homomorphism h from $\mathfrak{F}_{\text{univ}}(I)$ to J that is a retraction, i.e. it is the identity on elements of J, and, in particular, on $n_1, \ldots, n_m$. Since J^* is a subinstance of $\mathfrak{F}_{\text{univ}}(I)$, the mapping h is also a homomorphism from J^* to J that is the identity on $n_1, \ldots, n_m$. On the other hand, let J''' be the subinstance of J' that is obtained from J by replacing nulls $n_1, \ldots, n_m$ with constants $c_1, \ldots, c_m$, respectively. Since J'' is a core of J' there is a homomorphism from J' to J'', and, therefore, from J''' to J'' (because J''' is a subinstance of J'). By definition, both J and J^* contain nulls $n_1, \ldots, n_m$. Then, by applying the second part of Lemma 3.10 to $J_1 = J$, $J_2 = J^*$, $J_1' = J'''$ and $J_2' = J''$, we conclude that there is a homomorphism h' from J to J^* that is the identity on $n_1, \ldots, n_m$.

We prove next that $|\text{dom}(J)| = |\text{dom}(J^*)|$. Assume otherwise. Suppose first that $|\text{dom}(J^*)| < |\text{dom}(J)|$. Then clearly $h \circ h'$ is a homomorphism from J into a proper subinstance of itself, which is a contradiction since J is a core. Suppose now that $|\text{dom}(J)| < |\text{dom}(J^*)|$. Then $h' \circ h$ is a homomorphism from J^* into a proper subinstance K of itself. But, as we noted before, both h and h' are the identity on $n_1, \ldots, n_m$, and, therefore, $h' \circ h$ is a homomorphism from J^* to K that is the identity on $n_1, \ldots, n_m$. Let K' be the subinstance of J'' that is obtained from K by replacing nulls $n_1, \ldots, n_m$ with constants $c_1, \ldots, c_m$, respectively. Then, by applying the first part of Lemma 3.10 to $J_1 = J^*$ and $J_2 = K$, we conclude that there is a homomorphism from J'' into K'. In addition, it is not hard to prove that K' is a proper subinstance of J''. We conclude that there is a homomorphism from J'' into a proper subinstance of itself, which contradicts the fact that J'' is a core.

We prove finally that J^* is a core of $\mathfrak{F}_{\text{univ}}(I)$. Clearly, $h' \circ h$ is a homomorphism from $\mathfrak{F}_{\text{univ}}(I)$ to J^*. Assume for the sake of contradiction that J^* is a not a core, and hence that there is a homomorphism h'' from J^* into a proper subinstance K of itself. We prove next that the mapping $h \circ h'' \circ h'$ is a homomorphism from J into a proper subinstance of itself. In fact, we know that h' maps J into J^*, that h'' maps J^* into K, and that h maps J^*, and, thus, K, into J. Notice that the image of J under $h \circ h'' \circ h'$ contains at most $|\text{dom}(K)|$ elements. Recall that $|\text{dom}(J)| = |\text{dom}(J^*)|$, and, thus, $|\text{dom}(K)| < |\text{dom}(J)|$ (because K contains strictly fewer elements than J^*). Therefore, $h \circ h'' \circ h'$ maps J into a subinstance of itself with strictly fewer elements. We conclude that J is not a core, which is a contradiction. This finishes the proof of the claim. □

Recall that our goal is to construct an FO formula $\text{Core}(x_1, \ldots, x_n)$ that checks whether an n-tuple of nulls belongs to some core. From $\text{Core}_{\mathsf{Var}}(x_1, \ldots, x_n)$ we define formula $\text{Core}(x_1, \ldots, x_n)$ as follows:

$$\bigvee_{V \subseteq [1,n]} \bigg(\Big(\bigwedge_{i \in ([1,n] \setminus V)} C(x_i) \Big) \wedge \text{Core}_{\mathsf{Var}}(\bar{x}_V) \bigg),$$

where $\bar{x}_V$ is the tuple that contains all x_i with $i \in V$. Intuitively, for a canonical universal solution J and a tuple $\bar{a} \in \text{dom}(J)$, it is the case that $J \models \text{Core}(\bar{a})$ iff there is a core J^* of J that contains all the elements in $\bar{a}$.

Proof of Lemma 3.7: We finally have all the ingredients needed for the proof of Lemma 3.7. Take an arbitrary FO formula $\phi(\bar{x})$, where $|\bar{x}| = m$. We first transform $\phi(\bar{x})$ into an equivalent FO formula in prenex normal form, and then we transform the latter into an equivalent formula $\phi^*(\bar{x})$ of the form:

$$(\neg)\exists y_1 (\neg)\exists y_2 \ldots (\neg)\exists y_p\, \psi(\bar{x}, y_1, \ldots, y_p),$$

where ψ is a quantifier-free formula and $(\neg)$ means that the negation may or may not occur in front of the existential quantifier. We claim that $\phi'(\bar{x})$ defined as

$$\begin{aligned}&\text{Core}(\bar{x}) \,\wedge\, (\neg)\exists y_1 \big(\text{Core}(\bar{x}, y_1) \,\wedge\, (\neg)\exists y_2 \,(\text{Core}(\bar{x}, y_1, y_2) \,\wedge\, \ldots \\ &\qquad\qquad \wedge\, (\ldots \,\wedge\, (\neg)\exists y_p \,(\text{Core}(\bar{x}, y_1, \ldots, y_p) \,\wedge\, \psi(\bar{x}, y_1 \ldots, y_p)) \ldots))\big)\end{aligned}$$

satisfies the property that for every instance I of $\mathbf{S}$ and every $\bar{a} \in \text{dom}(\mathfrak{F}_{\text{univ}}(I))^m$, we have $\mathfrak{F}_{\text{univ}}(I) \models \phi'(\bar{a})$ if and only if there exists a core J' of $\mathfrak{F}_{\text{univ}}(I)$ such that $\bar{a} \in \text{dom}(J')^m$ and $J' \models \phi^*(\bar{a})$. We prove this by induction on p. This also proves the lemma as ϕ^* and ϕ are equivalent.

Let I be an arbitrary source instance. Assume that $J = \mathfrak{F}_{\text{univ}}(I)$ and $\bar{a} \in \text{dom}(J)^m$. For $p = 0$ we have that $\phi'(\bar{x}) = \text{Core}(\bar{x}) \wedge \psi(\bar{x})$. Assume first $J \models \text{Core}(\bar{a}) \wedge \psi(\bar{a})$. Since $J \models \text{Core}(\bar{a})$, there is a core J' of J that contains all the elements in $\bar{a}$. Moreover, since ψ is a boolean combination of atomic formulas, and the restriction of J to the elements in $\bar{a}$ is the same as the restriction of J' to the elements in $\bar{a}$ (because J' is an core of J, and, thus, an induced subinstance), it must be the case that $J' \models \psi(\bar{a})$. On the other hand, assume there is a core J' of J such that $J' \models \phi(\bar{a})$ and all the elements of $\bar{a}$ are in J'. Since the restriction of J to the elements in $\bar{a}$ is the same as the restriction of J' to the elements in $\bar{a}$, and ψ is a boolean combination of atomic formulas, it is the case that $J \models \text{Core}(\bar{a}) \wedge \psi(\bar{a})$.

For the inductive case, we consider two cases depending on whether $\neg$ occurs in front of $\exists y_1$. First, assume that $\exists y_1$ occurs positively in $\phi^*(\bar{x})$ and that $\bar{a}$ is a tuple in $\text{dom}(J)^m$. If

$$\begin{aligned}&J \models \text{Core}(\bar{a}) \,\wedge\, \exists y_1 \big(\text{Core}(\bar{a}, y_1) \,\wedge\, \ldots \\ &\qquad\qquad \wedge\, (\ldots \,\wedge\, (\neg)\exists y_{p+1} \,(\text{Core}(\bar{a}, y_1, y_2, \ldots, y_{p+1}) \,\wedge\, \psi(\bar{a}, y_1, y_2, \ldots, y_{p+1})) \ldots))\big),\end{aligned}$$

then for some $c \in \text{dom}(J)$,

$$\begin{aligned}&J \models \text{Core}(\bar{a}, c) \,\wedge\, (\neg)\exists y_2 \,(\text{Core}(\bar{a}, c, y_2) \,\wedge\, \ldots \\ &\qquad\qquad \wedge\, (\ldots \,\wedge\, (\neg)\exists y_{p+1} \,(\text{Core}(\bar{a}, c, y_2, \ldots, y_{p+1}) \,\wedge\, \psi(\bar{a}, c, y_2, \ldots, y_{p+1})) \ldots)).\end{aligned}$$

Thus, by induction hypothesis, there is a core J' of J such that J' contains all the elements in $\bar{a}c$ and $J' \models (\neg)\exists y_2 \ldots (\neg)\exists y_{p+1}\, \psi(\bar{a}, c, y_2 \ldots, y_{p+1})$, which implies that $J' \models \exists y_1 (\neg)\exists y_2 \ldots (\neg)\exists y_{p+1}\, \psi(\bar{a}, y_1, y_2 \ldots, y_{p+1})$.

On the other hand, if there is a core J' of J such that J' contains all the elements in $\bar{a} \in \text{dom}(J)^m$ and $J' \models \exists y_1 (\neg)\exists y_2 \ldots (\neg)\exists y_{p+1}\, \psi(\bar{a}, y_1, y_2 \ldots, y_{p+1})$, then for some $c \in \text{dom}(J')$, we have that $J' \models (\neg)\exists y_2 \ldots (\neg)\exists y_{p+1}\, \psi(\bar{a}, c, y_2 \ldots, y_{p+1})$. Given that J' is a core of J, we have that $\text{dom}(J') \subseteq \text{dom}(J)$ and, therefore, $c \in \text{dom}(J)$. Thus, by the induction hypothesis:

$$\begin{aligned}&J \models \text{Core}(\bar{a}, c) \,\wedge\, (\neg)\exists y_2 \,(\text{Core}(\bar{a}, c, y_2) \,\wedge\, \ldots \\ &\qquad\qquad \wedge\, (\ldots \,\wedge\, (\neg)\exists y_{p+1} \,(\text{Core}(\bar{a}, c, y_2, \ldots, y_{p+1}) \,\wedge\, \psi(\bar{a}, c, y_2, \ldots, y_{p+1})) \ldots)),\end{aligned}$$

and, hence,

$$J \models \mathrm{Core}(\bar{a}) \wedge \exists y_1 \big(\mathrm{Core}(\bar{a}, y_1) \wedge \ldots \\ \wedge (\ldots \wedge (\neg)\exists y_{p+1} (\mathrm{Core}(\bar{a}, y_1, y_2, \ldots, y_{p+1}) \wedge \psi(\bar{a}, y_1, y_2, \ldots, y_{p+1})) \ldots))\big).$$

Second, assume that $\exists y_1$ occurs negatively in $\phi^*(\bar{x})$ and that $\bar{b}$ is a tuple in $\mathrm{dom}(J)^m$. If

$$J \models \mathrm{Core}(\bar{b}) \wedge \neg\exists y_1 \big(\mathrm{Core}(\bar{b}, y_1) \wedge \ldots \\ \wedge (\ldots \wedge (\neg)\exists y_{p+1} (\mathrm{Core}(\bar{b}, y_1, y_2, \ldots, y_{p+1}) \wedge \psi(\bar{b}, y_1, y_2, \ldots, y_{p+1})) \ldots))\big),$$

then for every $c \in \mathrm{dom}(J)$, it is the case that $J \models \mathrm{Core}(\bar{b})$ and

$$J \not\models \mathrm{Core}(\bar{b}, c) \wedge (\neg)\exists y_2 (\mathrm{Core}(\bar{b}, c, y_2) \wedge \ldots \\ \wedge (\ldots \wedge (\neg)\exists y_{p+1} (\mathrm{Core}(\bar{b}, c, y_2, \ldots, y_{p+1}) \wedge \psi(\bar{b}, c, y_2, \ldots, y_{p+1})) \ldots)).$$

Take J' to be a core of J that contains all the elements in $\bar{b}$. By the induction hypothesis, for every $c \in \mathrm{dom}(J')$, we have that $J' \not\models (\neg)\exists y_2 \ldots (\neg)\exists y_{p+1}\, \psi(\bar{b}, c, y_2 \ldots, y_{p+1})$, which implies that $J' \models \neg\exists y_1 (\neg)\exists y_2 \ldots (\neg)\exists y_{p+1}\, \psi(\bar{b}, y_1, y_2 \ldots, y_{p+1})$.

On the other hand, if

$$J \not\models \mathrm{Core}(\bar{b}) \wedge \neg\exists y_1 \big(\mathrm{Core}(\bar{b}, y_1) \wedge \ldots \\ \wedge (\ldots \wedge (\neg)\exists y_{p+1} (\mathrm{Core}(\bar{b}, y_1, y_2, \ldots, y_{p+1}) \wedge \psi(\bar{b}, y_1, y_2, \ldots, y_{p+1})) \ldots))\big).$$

and there is a core J' of J such that $\bar{b} \in \mathrm{dom}(J')^m$ (that is, $J \models \mathrm{Core}(\bar{b})$), then

$$J \not\models \neg\exists y_1 \big(\mathrm{Core}(\bar{b}, y_1) \wedge (\neg)\exists y_2 (\mathrm{Core}(\bar{b}, y_1, y_2) \wedge \ldots \\ \wedge (\ldots \wedge (\neg)\exists y_{p+1} (\mathrm{Core}(\bar{b}, y_1, y_2, \ldots, y_{p+1}) \wedge \psi(\bar{b}, y_1, y_2, \ldots, y_{p+1})) \ldots))\big).$$

Thus, there is $c \in \mathrm{dom}(J)$ such that

$$J \models \mathrm{Core}(\bar{b}, c) \wedge (\neg)\exists y_2 (\mathrm{Core}(\bar{b}, c, y_2) \wedge \ldots \\ \wedge (\ldots \wedge (\neg)\exists y_{p+1} (\mathrm{Core}(\bar{b}, c, y_2, \ldots, y_{p+1}) \wedge \psi(\bar{b}, c, y_2, \ldots, y_{p+1})) \ldots)).$$

Hence, by induction hypothesis, there is a core J' of J such that J' contains all elements in $\bar{b}c$ and $J' \models (\neg)\exists y_2 \ldots (\neg)\exists y_{p+1}\, \psi(\bar{b}, c, y_2 \ldots, y_{p+1})$, which implies that $J' \not\models \neg\exists y_1 (\neg)\exists y_2 \ldots (\neg)\exists y_{p+1}\, \psi(\bar{b}, y_1, y_2 \ldots, y_{p+1})$. This concludes the proof of the lemma. □

We finish this section by proving Proposition 3.6.

Proof of Proposition 3.6: Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting, where $\mathbf{S} = \{R(\cdot,\cdot),\ S(\cdot,\cdot),\ N(\cdot),\ M(\cdot)\}$, $\mathbf{T} = \{S'(\cdot,\cdot),\ N'(\cdot),\ M'(\cdot),\ P(\cdot),\ T(\cdot)\}$ and Σ_{st} is defined as follows. Predicates S', N' and M' are defined by means of the following rules:

$$\begin{array}{rcl} S(x,y) & \rightarrow & S'(x,y), \\ N(x) & \rightarrow & N'(x), \\ M(x) & \rightarrow & M'(x). \end{array}$$

Predicate P plays the following role:

$$
\begin{array}{rcl}
\exists x R(x,x) & \rightarrow & \exists u P(u), \\
\exists x \exists y (\neg R(x,y) \wedge \neg R(y,x) \wedge x \neq y) & \rightarrow & \exists u P(u), \\
\exists x \exists y \exists z (R(x,y) \wedge R(y,z) \wedge \neg R(x,z)) & \rightarrow & \exists u P(u), \\
\neg \exists x \exists y R(x,y) & \rightarrow & \exists u P(u), \\
\neg \forall x \forall y (S(x,y) \leftrightarrow R(x,y) \wedge \neg \exists z (R(x,z) \wedge R(z,y))) & \rightarrow & \exists u P(u), \\
\exists x (N(x) \wedge M(x)) & \rightarrow & \exists u P(u).
\end{array}
$$

Intuitively, if R is not a linear order on the domain of an instance I of $\mathbf{S}$, or R is empty, or S does not correspond to the successor relation of R, or there is an element in the intersection of N and M, then P contains at least one element in every solution for I. Finally, predicate T plays the following role:

$$
\begin{array}{rcl}
\exists x \exists y (R(x,y) \wedge N(x) \wedge M(y)) & \rightarrow & \exists u T(u), \\
\exists x \exists y \, S(x,y) & \rightarrow & \exists u T(u).
\end{array}
$$

Intuitively, if in the source instance I (i) there is a pair (a,b) of elements in R such that a belongs to M and b belongs to N, or (ii) there are at least two elements that are related by S, then the interpretation of T is nonempty in every solution for I. Notice that if both (i) and (ii) hold in I, then $\mathfrak{F}_{\text{univ}}(I)$ contains exactly two nulls in T, while $\mathfrak{F}_{\text{core}}(I)$ contains only one.

Let Q be the following query:

$$
\exists x P(x) \vee \exists x \exists y (N'(x) \wedge S'(x,y) \wedge \neg N'(y)) \ \vee \ \exists x (N'(x) \wedge M'(x)). \tag{3}
$$

We will prove that Q is rewritable over $\mathfrak{F}_{\text{univ}}$ and that Q is not rewritable over $\mathfrak{F}_{\text{core}}$.

Let Q' be the query $\exists x P(x) \vee \exists x \exists y (T(x) \wedge T(y) \wedge x \neq y)$. We show next that Q' is a rewriting of Q over the canonical universal solution, that is, for every instance I of $\mathbf{S}$ with canonical universal solution J, it is the case that $Q'(J)$ holds if and only if $\underline{\text{certain}}_{\mathcal{M}}(Q,I) = \texttt{true}$.

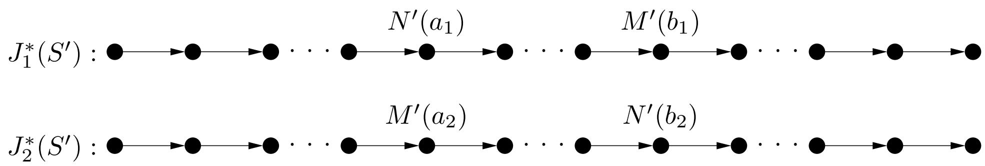
- Assume that $Q'(J)$ holds. If $J \models \exists x P(x)$, then every solution J' for I satisfies this sentence, since there is a homomorphism from J to J', and, therefore, $\underline{\text{certain}}_{\mathcal{M}}(Q,I) = \texttt{true}$. Thus, assume that $J \not\models \exists x P(x)$ and $J \models \exists x \exists y (T(x) \wedge T(y) \wedge x \neq y)$. Then $I(R)$ is a linear order, $I(S)$ is the successor relation of this order, and there exists $a, b \in \text{dom}(I)$ such that $R(a,b)$, $N(a)$ and $M(b)$. Let J' be a solution for I. To prove that $\underline{\text{certain}}_{\mathcal{M}}(Q,I) = \texttt{true}$, we show that $J' \models \exists x \exists y (N'(x) \wedge S'(x,y) \wedge \neg N'(y)) \ \vee \ \exists x (N'(x) \wedge M'(x))$. Assume that $J' \not\models \exists x \exists y (N'(x) \wedge S'(x,y) \wedge \neg N'(y))$ and, hence, $J' \models \forall x \forall y (N'(x) \wedge S'(x,y) \rightarrow N'(y))$. Then, $N'(b)$ is in J' since $N'(a)$ is in J' and a' appears before b' in the order R. We conclude that $J' \models \exists x (N'(x) \wedge M'(x))$ since $M'(b)$ is in J'.
- Assume that $\underline{\text{certain}}_{\mathcal{M}}(Q,I) = \texttt{true}$ and that $Q'(J)$ does not hold. Then $J \not\models \exists x P(x)$, $J \models \exists x \exists y (N'(x) \wedge S'(x,y) \wedge \neg N'(y)) \vee \exists x (N'(x) \wedge M'(x))$ and $J \not\models \exists x \exists y (T(x) \wedge T(y) \wedge x \neq y)$. Hence, $I(R) \neq \emptyset$, $I(R)$ is a linear order, $I(S)$ is the successor relation of this order, all the elements in $I(M)$ appear before all the elements in $I(N)$ in the order $I(R)$, and there is no element in the intersection of $I(N)$ and $I(M)$. Let $\{J_n\}_{n \geq 0}$ be a sequence of solutions for I recursively defined as follows: $J_0 := J$ and

$$
J_{n+1} \ := \ J_n \cup \{N'(b) \mid \text{there exists } a \in \text{dom}(J) \text{ such that } N'(a) \text{ is in } J_n \text{ and } S'(a,b) \text{ is in } J\}.
$$

Then $J' = \bigcup_{n\geq 0} J_n$ is an instance (this is because the union $\bigcup_{n\geq 0} J_n$ is finite simply because J is finite). Furthermore, J' is a solution for I such that $J' \not\models \exists x \exists y (N'(x) \wedge S'(x,y) \wedge \neg N'(y))$. Clearly, J' does not contain an element in the intersection of N' and M', that is, $J' \not\models \exists x (N'(x) \wedge M'(x))$, which contradicts the fact that $\underline{\text{certain}}_{\mathcal{M}}(Q, I) = \texttt{true}$.

Now we prove that Q is not rewritable over $\mathfrak{F}_{\text{core}}$. On the contrary, assume that there exists an FO sentence ϕ such that for every instance I of $\mathbf{S}$ with core solution J^*, it is the case that $J^* \models \phi$ if and only if $\underline{\text{certain}}_{\mathcal{M}}(Q, I) = \texttt{true}$. Let k be the quantifier rank of ϕ. Define instances I_1, I_2 of $\mathbf{S}$ as follows: $I_i(S)$ is the successor relation of a linear order $I_i(R)$ $(i = 1,2)$ containing k' elements, where k' is a function of k (to be defined later), $I_1(N) = \{a_1\}$, $I_1(M) = \{b_1\}$, $I_2(N) = \{b_2\}$ and $I_2(M) = \{a_2\}$. Furthermore, $(a_i, b_i) \in I_i(R)$ $(i = 1,2)$, the distance between a_i and b_i is $k'/2$ $(i = 1,2)$ and the distance between the first point of $I_i(R)$ and a_i is $k'/4$ $(i = 1,2)$. It is not hard to see that $\underline{\text{certain}}_{\mathcal{M}}(Q, I_1) = \texttt{true}$ and $\underline{\text{certain}}_{\mathcal{M}}(Q, I_2) = \texttt{false}$.

The core solutions J_1^*, J_2^* for I_1, I_2 are as follows:



Furthermore, $J_1^*(P)$, $J_2^*(P)$ are empty and $J_1^*(T)$, $J_2^*(T)$ contain only one null value. Thus, if k' is big enough, then $J_1^* \equiv_k J_2^*$ and, therefore, $J_1^* \models \phi$ if and only if $J_2^* \models \phi$. We conclude that $\underline{\text{certain}}_{\mathcal{M}}(Q, I_1) = \underline{\text{certain}}_{\mathcal{M}}(Q, I_2)$, which leads to a contradiction. (Notice, on the other hand, that $J_1 \not\equiv_k J_2$, where J_1, J_2 are the canonical universal solutions for I_1, I_2, since J_1 contains two null values in T while J_2 contains only one). $\square$

3.2. Canonical universal solution versus source

Here we prove that the class of queries rewritable over the canonical universal solution is strictly contained in the class of queries rewritable over the source. We start by proving that:

Theorem 3.11. *Every query that is rewritable over the canonical universal solution is also rewritable over the source.*

And then we show that the converse does not hold:

Proposition 3.12. *There is a query specified in* FO *that is rewritable over the source, but not rewritable over the canonical universal solution.*

In the proof of Theorem 3.11, we use the following terminology. Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting. Then

$$\begin{aligned} \ell_{\mathbf{S}} &= \max\{\mathrm{qr}(\phi_{\mathbf{S}}) \mid \phi_{\mathbf{S}}(\bar{x}) \rightarrow \exists \bar{y} \psi_{\mathbf{T}}(\bar{x}, \bar{y}) \in \Sigma_{st}\} \\ m_{\mathbf{S}} &= \max\{1\} \cup \{|\bar{x}| \mid \phi_{\mathbf{S}}(\bar{x}) \rightarrow \exists \bar{y} \psi_{\mathbf{T}}(\bar{x}, \bar{y}) \in \Sigma_{st}\} \end{aligned}$$

For a source instance I, if $J = \mathfrak{F}_{\text{univ}}(I)$ and n is a null in J, then the presence of n in J can be unambiguously identified with the instantiation

$$\phi_{\mathbf{S}}(\bar{c}) \;\rightarrow\; \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{c}, \bar{y}) \tag{4}$$

of some std of the form $\phi_{\mathbf{S}}(\bar{x}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})$ in Σ_{st}, where $\phi_{\mathbf{S}}(\bar{c})$ holds in I and n is the null used in J to witness a variable in $\bar{y}$ in order to satisfy (4). We call $\bar{c}$ the *witness* of n in I, and $\phi_{\mathbf{S}}(\bar{x}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})$ the *carrier* of n in I. In the same way, we can define for a fact $T(\bar{a})$ in J a pair containing its witness and carrier in I, as the presence of $T(\bar{a})$ in J can be identified with the instantiation

$$\phi_{\mathbf{S}}(\bar{c}) \;\rightarrow\; \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{c}, \bar{y}) \tag{5}$$

of an std of the form $\phi_{\mathbf{S}}(\bar{x}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})$ in Σ_{st} (the carrier), where $\phi_{\mathbf{S}}(\bar{c})$ holds in I ($\bar{c}$ is the witness, and it contains all constants in $\bar{a}$), and $T(\bar{a})$ is used in J in order to satisfy $\exists \bar{y} \psi_{\mathbf{T}}(\bar{c}, \bar{y})$. Note that if $\bar{a}$ contains only constants, then there is not necessarily a unique pair of witness and carrier for $T(\bar{a})$ in I.

The following theorem will be used in the proof of Theorem 3.11. In [FK12], a transformation $\mathfrak{F}$ is called *FO-local* if for every k, there is k' such that whenever $I_1 \equiv_{k'} I_2$, then $\mathfrak{F}(I_1) \equiv_k \mathfrak{F}(I_2)$. It follows immediately from the next theorem (by letting $\bar{a}$ and $\bar{b}$ be empty) that $\mathfrak{F}_{\text{univ}}$ is FO-local. The paper [FK12] made use of our result that $\mathfrak{F}_{\text{univ}}$ is FO-local.

Theorem 3.13. *For every $k \geq 0$ there exists $k' \geq 0$ such that, for all source instances I_1 and I_2, and tuples of constants $\bar{a} \in \text{dom}(\mathfrak{F}_{\text{univ}}(I_1))^m$ and $\bar{b} \in \text{dom}(\mathfrak{F}_{\text{univ}}(I_2))^m$, where $m \geq 0$, if $(I_1, \bar{a}) \equiv_{k'} (I_2, \bar{b})$ then $(\mathfrak{F}_{\text{univ}}(I_1), \bar{a}) \equiv_k (\mathfrak{F}_{\text{univ}}(I_2), \bar{b})$.*

Proof: Let I_1 and I_2 be arbitrary source instances, and denote $\mathfrak{F}_{\text{univ}}(I_1)$ by J_1 and $\mathfrak{F}_{\text{univ}}(I_2)$ by J_2. Fix $k \geq 0$. We choose k' to be $m_{\mathbf{S}} \cdot (k+1) + \ell_{\mathbf{S}}$. For each round i, where $0 < i \leq k$, of the k-round game on $(J_1, \bar{a})$ and $(J_2, \bar{b})$, the duplicator's response q_i in J_2 (resp., p_i in J_1) to an element p_i in J_1 (resp., q_i in J_2) played by the spoiler, is defined by looking at the duplicator's response $\bar{e}_i$ in I_2 (resp., $\bar{c}_i$ in I_1), in rounds $m_{\mathbf{S}} \cdot (i-1) + 1$ to $m_{\mathbf{S}} \cdot i$ of the k'-round game on $(I_1, \bar{a})$ and $(I_2, \bar{b})$, to a tuple $\bar{c}_i$ in I_1 (resp., $\bar{e}_i$ in I_2) of length $m_{\mathbf{S}}$, according to a winning strategy provided by the fact that $(I_1, \bar{a}) \equiv_{k'} (I_2, \bar{b})$.

Assume that for round i with $0 < i < k$ the elements played by following this strategy are (1) $(p_1, \ldots, p_i)$ in J_1, (2) $(q_1, \ldots, q_i)$ in J_2, (3) $(\bar{c}_1, \ldots, \bar{c}_i)$ in I_1, and (4) $(\bar{e}_1, \ldots, \bar{e}_i)$ in I_2. Since we assume that the $\bar{c}_i$'s and $\bar{e}_i$'s are played according to a winning strategy of the duplicator in the k'-round game on $(I_1, \bar{a})$ and $(I_2, \bar{b})$, it is the case that

$$(I_1, \bar{a}, \bar{c}_1, \ldots, \bar{c}_i) \equiv_{m_{\mathbf{S}} \cdot (k-i+1) + \ell_{\mathbf{S}}} (I_2, \bar{b}, \bar{e}_1, \ldots, \bar{e}_i).$$

Also, assume without loss of generality that for round $i+1$ of the game on $(J_1, \bar{a})$ and $(J_2, \bar{a})$, the spoiler picks an element p_{i+1} in J_1 (the case when he picks an element q_{i+1} in J_2 is completely symmetric). The duplicator response q_{i+1} in J_2 is defined as follows:

- Assume first that p_{i+1} is a null value. Let $\bar{c}$ and $\phi_{\mathbf{S}}(\bar{x}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})$ be the witness and carrier for p_{i+1} in I_1, respectively. Also, let $y \in \bar{y}$ be the variable that is witnessed by p_{i+1} in J_1. Assume that $\bar{c} = (c_1, \ldots, c_r)$, where $r \leq m_{\mathbf{S}}$. Then we define $\bar{c}_{i+1}$ to be the tuple $(c_1, \ldots, c_r, c_1, \ldots, c_1)$ of length $m_{\mathbf{S}}$ (thus, we pad the tuple to be of length $m_{\mathbf{S}}$).

The duplicator response q_{i+1} to p_{i+1} (in the k-round game on $(J_1, \bar{a})$ and $(J_2, \bar{b})$) is defined by looking at the duplicator response $\bar{e}_{i+1}$ to $\bar{c}_{i+1}$ in rounds $m_{\mathbf{S}} \cdot i + 1$ to $m_{\mathbf{S}} \cdot (i+1)$ of the k'-round game on $(I_1, \bar{a})$ and $(I_2, \bar{b})$. More precisely, since $\mathrm{qr}(\phi_{\mathbf{S}}) \leq \ell_s$, and

$$(I_1, \bar{a}, \bar{c}_1, \ldots, \bar{c}_i, \bar{c}_{i+1}) \equiv_{m_{\mathbf{S}} \cdot (k-i) + \ell_{\mathbf{S}}} (I_2, \bar{b}, \bar{e}_1, \ldots, \bar{e}_i, \bar{e}_{i+1}),$$

we conclude that $\phi_{\mathbf{S}}(\bar{e})$ holds in I_2, where $\bar{e}$ is the tuple that contains the first r elements of $\bar{e}_{i+1}$. Hence, there is a null value that is used in J_2 to witness the variable $y \in \bar{y}$ in order to satisfy $\exists \bar{y} \psi_{\mathbf{T}}(\bar{e}, \bar{y})$. The duplicator response q_{i+1} to p_{i+1} is set to be this null value.

- If p_{i+1} is a constant, it also belongs to I_1. Let us define $\bar{c}_{i+1}$ to be a tuple of length $m_{\mathbf{S}}$ that only contains the element p_{i+1}. Again, the duplicator response q_{i+1} to p_{i+1} in the k-round game on $(J_1, \bar{a})$ and $(J_2, \bar{b})$ is defined by looking at the duplicator response $\bar{e}_{i+1}$ to $\bar{c}_{i+1}$ in rounds $(m_{\mathbf{S}} \cdot i + 1)$ to $m_{\mathbf{S}} \cdot (i+1)$ of the k'-round game on $(I_1, \bar{a})$ and $(I_2, \bar{b})$. In this case, q_{i+1} is set to be the only element that belongs to $\bar{e}_{i+1}$. To finish this part of the proof, we have to show that our previous choice is correct, that is, we need to show that q_{i+1} belongs to J_2.

  Indeed, the presence of p_{i+1} in J_1 can be identified with the instantiation

  $$\phi_{\mathbf{S}}(\bar{c}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{c}, \bar{y})$$

  of an std of the form $\phi_{\mathbf{S}}(\bar{x}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})$ in Σ_{st}, where there is a tuple $\bar{c}$ of constant symbols that includes p_{i+1} such that $\phi_{\mathbf{S}}(\bar{c})$ holds in the source instance I_1. Since $|\bar{c}| \leq m_{\mathbf{S}}$ and $i < k$, there exists $\bar{e}$ in I_2 such that,

  $$(I_1, \bar{a}, \bar{c}_1, \ldots, \bar{c}_i, \bar{c}_{i+1}, \bar{c}) \equiv_{m_{\mathbf{S}} \cdot (k-i-1) + \ell_{\mathbf{S}}} (I_2, \bar{b}, \bar{e}_1, \ldots, \bar{e}_i, \bar{e}_{i+1}, \bar{e}).$$

  Thus, it is the case that $\phi_{\mathbf{S}}(\bar{e})$ holds in I_2 (since $\mathrm{qr}(\phi_{\mathbf{S}}) \leq \ell_{\mathbf{S}}$), and therefore, q_{i+1} belongs to J_2.

Note that p_{i+1} is a null if and only if q_{i+1} is a null.

In order to prove that $(J_1, \bar{a}) \equiv_k (J_2, \bar{b})$, it is enough to prove the following by induction on $i \leq k$: if $(p_1, \ldots, p_i)$ and $(q_1, \ldots, q_i)$ are the moves played in J_1 and J_2, resp., by using the strategy defined above, then $\big((\bar{a}, p_1, \ldots, p_i), (\bar{b}, q_1, \ldots, q_i)\big)$ is a partial isomorphism between J_1 and J_2. We do this next.

For $i = 0$ the proof is as follows. Assume that $T(\bar{a}_0)$ is in J_1, for some $T \in \mathbf{T}$ and tuple $\bar{a}_0$ of elements in $\bar{a}$. Let $\bar{c}$ and $\phi_{\mathbf{S}}(\bar{x}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})$ be a pair of witness and carrier for $T(\bar{a}_0)$ in I_1. Since $(I_1, \bar{a}) \equiv_{\ell_{\mathbf{S}} + m_{\mathbf{S}}} (I_2, \bar{b})$, there is a tuple $\bar{e}$ of constants in I_2 such that $(I_1, \bar{a}, \bar{c}) \equiv_{\ell_{\mathbf{S}}} (I_2, \bar{b}, \bar{e})$. Then, since $\mathrm{qr}(\phi_{\mathbf{S}}) \leq \ell_s$, it is also the case that $\phi_{\mathbf{S}}(\bar{e})$ holds in I_2, and hence $T(\bar{b}_0)$ holds in J_2, where $\bar{b}_0$ is the tuple corresponding to $\bar{a}_0$ in $\bar{b}$.

Assume now that the property holds for $i < k$. Next we prove it for $i + 1$. Let $(p_1, \ldots, p_i)$ and $(q_1, \ldots, q_i)$ be the i moves played in J_1 and J_2, resp., by following the strategy. Then, by induction hypothesis, $\big((\bar{a}, p_1, \ldots, p_i), (\bar{b}, q_1, \ldots, q_i)\big)$ is a partial isomorphism between J_1 and J_2. Assume without loss of generality that in round $i+1$ the spoiler picks an element p_{i+1} in J_1 (the case when he picks an element q_{i+1} in J_2 is completely symmetric). We show that if q_{i+1} in J_2 is the response defined by the strategy, then $\big((\bar{a}, p_1, \ldots, p_i, p_{i+1}), (\bar{b}, q_1, \ldots, q_i, q_{i+1})\big)$ is a partial isomorphism between J_1 and J_2.

Assume, for the sake of contradiction, that this is not the case. Then, without loss of generality, there is a tuple $T(\bar{a}')$ in J_1, where $T \in \mathbf{T}$ and all the elements of $\bar{a}'$ belong to $(\bar{a}, p_1, \ldots, p_{i+1})$, such that $T(\bar{b}')$ is not in J_2, where $\bar{b}'$ is the corresponding tuple in $(\bar{b}, q_1, \ldots, q_{i+1})$. We have, by induction hypothesis, that p_{i+1} belongs to $\bar{a}'$, and q_{i+1} belongs to $\bar{b}'$. Furthermore, $T \neq C$, where C is the unary relation symbol that distinguishes constants over the target, since p_{i+1} is a null value if and only if q_{i+1} is a null value. Let $\bar{c}$ and $\phi_{\mathbf{S}}(\bar{x}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})$ be a pair of witness and carrier for $T(\bar{a}')$ in I_1. We consider two cases depending on whether p_{i+1} is a null or a constant.

Assume first that p_{i+1} is a null value. Then $\bar{c}$ corresponds to the projection of $\bar{c}_{i+1}$ over its first $|\bar{c}|$ elements. Therefore, since

$$(I_1, \bar{a}, \bar{c}_1, \ldots, \bar{c}_i, \bar{c}_{i+1}) \equiv_{m_{\mathbf{S}} \cdot (k-i) + \ell_{\mathbf{S}}} (I_2, \bar{b}, \bar{e}_1, \ldots, \bar{e}_i, \bar{e}_{i+1}),$$

it is the case that if $\bar{e}$ corresponds to the projection of $\bar{e}_{i+1}$ over its first $|\bar{c}|$ elements, then $\phi_{\mathbf{S}}(\bar{e})$ holds in I_2 (because $\mathrm{qr}(\phi_{\mathbf{S}}) \leq \ell_{\mathbf{S}}$). Furthermore, all nulls in $\bar{a}'$ have witness $\bar{c}_{i+1}$ in I_1, and all nulls in $\bar{b}'$ have witness $\bar{e}_{i+1}$ in I_2. Then by the way the duplicator strategy is defined, in particular from the facts that (1) constants are preserved from the game on $(I_1, \bar{a})$ and $(I_2, \bar{b})$ to the game on $(J_1, \bar{a})$ and $(J_2, \bar{b})$, and (2) for each $j \leq i+1$ such that p_j and q_j are nulls, both p_j and q_j witness the same variable $y \in \bar{y}$ in their carriers, we can conclude that $T(\bar{b}')$ is in J_2, which is a contradiction.

Assume otherwise that p_{i+1} is a constant (in this case $\bar{c}$ is not necessarily equal to $\bar{c}_{i+1}$). Then, since $|\bar{c}| \leq m_{\mathbf{S}}$ and $i < k$, there exists $\bar{e}$ in I_2 such that,

$$(I_1, \bar{a}, \bar{c}_1, \ldots, \bar{c}_i, \bar{c}_{i+1}, \bar{c}) \equiv_{m_{\mathbf{S}} \cdot (k-i-1) + \ell_{\mathbf{S}}} (I_2, \bar{b}, \bar{e}_1, \ldots, \bar{e}_i, \bar{e}_{i+1}, \bar{e}).$$

Thus, it is the case that $\phi_{\mathbf{S}}(\bar{e})$ holds in I_2 (since $\mathrm{qr}(\phi_{\mathbf{S}}) \leq \ell_{\mathbf{S}}$). Furthermore, all nulls in $\bar{a}'$ have witness $\bar{c}$, and all nulls in $\bar{b}'$ have witness $\bar{e}$. Then, by the way the duplicator strategy is defined, in particular from the facts that (1) constants are preserved from the game on $(I_1, \bar{a})$ and $(I_2, \bar{b})$ to the game on $(J_1, \bar{a})$ and $(J_2, \bar{b})$, and (2) for each $j \leq i+1$ such that p_j and q_j are nulls, both p_j and q_j witness the same variable $y \in \bar{y}$ in their carriers, we can conclude that $T(\bar{b}')$ is in J_2, which is again a contradiction. This concludes the proof. $\square$

We are finally ready to prove Theorem 3.11.

Proof of Theorem 3.11: From Theorem 3.13 we conclude that for every FO formula $\phi(\bar{x})$ over $\mathbf{T}$, there is an FO formula $\psi(\bar{x})$ over $\mathbf{S}$ such that, for every instance I of $\mathbf{S}$ and tuple $\bar{a}$ of constants in $\mathrm{dom}(\mathfrak{F}_{\mathrm{univ}}(I))^{|\bar{x}|}$,

$$I \models \psi(\bar{a}) \iff \mathfrak{F}_{\mathrm{univ}}(I) \models \phi(\bar{a}).$$

Indeed, let $\mathrm{qr}(\phi)$ be $k \geq 0$. From the lemma above, there is $k' \geq 0$ such that, for all source instances I_1 and I_2, and tuples $\bar{a} \in \mathrm{dom}(\mathfrak{F}_{\mathrm{univ}}(I_1))^{|\bar{x}|}$ and $\bar{b} \in \mathrm{dom}(\mathfrak{F}_{\mathrm{univ}}(I_2))^{|\bar{x}|}$ of constants, if $(I_1, \bar{a}) \equiv_{k'} (I_2, \bar{b})$ then $(\mathfrak{F}_{\mathrm{univ}}(I_1), \bar{a}) \equiv_k (\mathfrak{F}_{\mathrm{univ}}(I_2), \bar{b})$. Then it is not hard to see that $\psi(\bar{x})$ can be defined as:

$$\bigvee_{\{(I,\bar{a}) \mid \mathfrak{F}_{\mathrm{univ}}(I) \models \phi(\bar{a})\}} \tau_{k'}^{(I,\bar{a})}(\bar{x}).$$

Recall that $\tau_{k'}^{(I,\bar{a})}(\bar{x})$ is the rank-k' FO type of $(I, \bar{a})$. Note that k' (which is the quantifier rank of ψ) depends only on k and Σ_{st}. $\square$

We conclude this section by proving Proposition 3.12.

Proof of Proposition 3.12: We use the same data exchange setting as in the proof of Proposition 3.6, but with all references to predicate $T \in \mathbf{T}$ removed. We also use the same query Q as that in the proof of Proposition 3.6:

$$\exists x P(x) \vee \exists x \exists y (N'(x) \wedge S'(x,y) \wedge \neg N'(y)) \;\vee\; \exists x (N'(x) \wedge M'(x)).$$

Let ϕ be the disjunction of all formulae that appear in the left-hand side of an std of the form $\psi \rightarrow \exists u P(u)$ in Σ_{st}, and Q' be the following FO sentence over $\mathbf{S}$:

$$\phi \vee \exists x \exists y (R(x,y) \wedge N(x) \wedge M(y)).$$

We show next that Q' is a rewriting of Q over the source, that is, for every instance I of $\mathbf{S}$ it is the case that $Q'(I)$ holds if and only if $\underline{\text{certain}}_{\mathcal{M}}(Q,I) = \texttt{true}$.

- Assume that $Q'(I)$ holds. If $I \models \phi$, then every solution J for I satisfies the sentence $\exists x P(x)$, and hence $\underline{\text{certain}}_{\mathcal{M}}(Q,I) = \texttt{true}$. If $I \not\models \phi$ then $I(R)$ is a linear order and $I(S)$ is the successor relation of this order. Given that $I \not\models \phi$ but $Q'(I)$ holds, it is the case that $I \models \exists x \exists y (R(x,y) \wedge N(x) \wedge M(y))$. Thus, there exists $a, b \in \text{dom}(I)$ such that $R(a,b)$, $N(a)$ and $M(b)$. Let J be a solution for I. To prove that $\underline{\text{certain}}_{\mathcal{M}}(Q,I) = \texttt{true}$, we show that $J \models \exists x \exists y (N'(x) \wedge S'(x,y) \wedge \neg N'(y)) \;\vee\; \exists x (N'(x) \wedge M'(x))$. Assume that $J \not\models \exists x \exists y (N'(x) \wedge S'(x,y) \wedge \neg N'(y))$ and, hence, $J \models \forall x \forall y (N'(x) \wedge S'(x,y) \rightarrow N'(y))$. Then, $N'(b)$ is in J since $N'(a)$ is in J and a' appears before b' in the order R. We conclude that $J \models \exists x (N'(x) \wedge M'(x))$ since $M'(b)$ is in J.
- Assume that $\underline{\text{certain}}_{\mathcal{M}}(Q,I) = \texttt{true}$ and that $Q'(I)$ does not hold. Let J be the canonical universal solution of I. Then $I \not\models \phi$, $J \models \exists x \exists y (N'(x) \wedge S'(x,y) \wedge \neg N'(y)) \;\vee\; \exists x (N'(x) \wedge M'(x))$ and $I \not\models \exists x \exists y (R(x,y) \wedge N(x) \wedge M(y))$. Hence, $I(R) \neq \emptyset$, $I(R)$ is a linear order, $I(S)$ is the successor relation of this order, all the elements in $I(M)$ appear before all the elements in $I(N)$ in the order $I(R)$, and there is no element in the intersection of $I(N)$ and $I(M)$. Let $\{J_n\}_{n \geq 0}$ be a sequence of solutions for I recursively defined as follows: $J_0 := J$ and

  $$J_{n+1} \;:=\; J_n \cup \{N'(b) \mid \text{there exists } a \in \text{dom}(J) \text{ such that } N'(a) \text{ is in } J_n \text{ and } S'(a,b) \text{ is in } J\}.$$

  Then $J' = \bigcup_{n \geq 0} J_n$ is an instance (this is because the union $\bigcup_{n \geq 0} J_n$ is finite simply because J is finite). Furthermore, J' is a solution for I such that $J' \not\models \exists x \exists y (N'(x) \wedge S'(x,y) \wedge \neg N'(y))$. Since there is no element in the intersection of $I(N)$ and $I(M)$, it follows that there is no element in the intersection of $J'(N')$ and $J'(M')$, that is, $J' \not\models \exists x (N'(x) \wedge M'(x))$. Also, $J' \not\models \exists x P(x)$, since $I \not\models \phi$. This contradicts the fact that $\underline{\text{certain}}_{\mathcal{M}}(Q,I) = \texttt{true}$.

To prove that Q is not rewritable over the canonical universal solution, we can use a similar technique that the one used in the proof of Proposition 3.6, when showing that Q was not rewritable over the core. In fact, exactly the same source instances I_1 and I_2 considered in such proof can be used here, as $\underline{\text{certain}}_{\mathcal{M}}(Q,I_1) = \texttt{true}$, $\underline{\text{certain}}_{\mathcal{M}}(Q,I_2) = \texttt{false}$, and the absence of predicate T does not allow to distinguish between $\mathfrak{F}_{\text{univ}}(I_1)$ and $\mathfrak{F}_{\text{univ}}(I_2)$ in k rounds of the Ehrenfeucht-Fraïssé game. $\square$

4. Tools for Proving Non-rewritability

In view of the fact that rewriting is the main approach for obtaining certain answers for target queries, and that the canonical universal solution and the core are the preferred solutions in data exchange, it becomes crucial to develop tools that help to decide whether a query admits a rewriting over these solutions.

In this section, we present such a tool, following the standard approach developed in finite model theory for proving inexpressibility results (cf. [Lib04]). The main tool for proving inexpressibility is Ehrenfeucht-Fraïssé games, but since games often involve nontrivial combinatorial arguments, one often looks for sufficient conditions that guarantee a win for the duplicator. Most commonly used conditions for FO involve the notions of locality, described in Section 4.2 below.

We follow the same approach here: we first present a simple game-based criterion, and then use results of the previous section to introduce a locality-based criterion, and give examples of its applicability. Finally, we use those results to prove Proposition 3.3, that states that the notion of rewritability is undecidable.

4.1. A game-based tool

The following is the main technical tool for proving non-rewritability results in data exchange.

Proposition 4.1. *Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting, and assume that $\mathfrak{F}$ is either $\mathfrak{F}_{\text{univ}}$ or $\mathfrak{F}_{\text{core}}$. An m-ary query Q, for $m \geq 0$, over schema $\mathbf{T}$ is not rewritable over $\mathfrak{F}$ under $\mathcal{M}$ if and only if for all $k \geq 0$ there are instances I_1 and I_2 of $\mathbf{S}$ and m-tuples $\bar{a} \in \text{dom}(I_1)^m$ and $\bar{b} \in \text{dom}(I_2)^m$, such that $(\mathfrak{F}(I_1), \bar{a}) \equiv_k (\mathfrak{F}(I_2), \bar{b})$, $\bar{a} \in \underline{\text{certain}}_{\mathcal{M}}(Q, I_1)$, but $\bar{b} \notin \underline{\text{certain}}_{\mathcal{M}}(Q, I_2)$.*

Proof: To prove the proposition we consider its "positive" version: An m-ary query Q over schema $\mathbf{T}$ is rewritable over $\mathfrak{F}$ if and only if there exists $k \geq 0$ such that for every pair of instances I_1 and I_2 of $\mathbf{S}$ and m-tuples $\bar{a} \in \text{dom}(I_1)^m$ and $\bar{b} \in \text{dom}(I_2)^m$, if $(\mathfrak{F}(I_1), \bar{a}) \equiv_k (\mathfrak{F}(I_2), \bar{b})$ and $\bar{a} \in \underline{\text{certain}}_{\mathcal{M}}(Q, I_1)$, then $\bar{b} \in \underline{\text{certain}}_{\mathcal{M}}(Q, I_2)$.

The "only if" part of the positive version of the proposition is proved by taking k to be the quantifier rank of the rewriting Q' of Q. In fact, assume that $(\mathfrak{F}(I_1), \bar{a}) \equiv_{\text{qr}(Q')} (\mathfrak{F}(I_2), \bar{b})$ and $\bar{a} \in \underline{\text{certain}}_{\mathcal{M}}(Q, I_1)$, where I_1 and I_2 are instances of $\mathbf{S}$, $\bar{a} \in \text{dom}(I_1)^m$, and $\bar{b} \in \text{dom}(I_2)^m$. Since Q' is a rewriting of Q over $\mathfrak{F}$, the fact that $\bar{a} \in \underline{\text{certain}}_{\mathcal{M}}(Q, I_1)$ implies that $\bar{a} \in Q'(\mathfrak{F}(I_1))$. But then $\bar{b} \in Q'(\mathfrak{F}(I_2))$, because $(\mathfrak{F}(I_1), \bar{a}) \equiv_{\text{qr}(Q')} (\mathfrak{F}(I_2), \bar{b})$. We conclude that $\bar{b} \in \underline{\text{certain}}_{\mathcal{M}}(Q, I_2)$.

For the "if" part, we prove that Q admits the following rewriting Q': $\bigvee_{\bar{a} \in \underline{\text{certain}}_{\mathcal{M}}(Q,I)} \tau_k^{(\mathfrak{F}(I),\bar{a})}(\bar{x})$ (recall that $\tau_k^{(\mathfrak{F}(I),\bar{a})}(\bar{x})$ is the rank-k FO type of $(\mathfrak{F}(I), \bar{a})$). Assume first that $\bar{a}_1 \in \underline{\text{certain}}_{\mathcal{M}}(Q, I_1)$. Then $\bar{a}_1 \in Q'(\mathfrak{F}(I_1))$, because $\mathfrak{F}(I_1) \models \tau_k^{(\mathfrak{F}(I_1),\bar{a}_1)}(\bar{a}_1)$. Assume, on the other hand, that $\bar{a}_1 \in Q'(\mathfrak{F}(I_1))$. This means that $\mathfrak{F}(I_1) \models \tau_k^{(\mathfrak{F}(I),\bar{a})}(\bar{a}_1)$, for some instance I of $\mathbf{S}$ and tuple $\bar{a} \in \text{dom}(I)^m$ such that $\bar{a} \in \underline{\text{certain}}_{\mathcal{M}}(Q, I)$. Thus, $(\mathfrak{F}(I), \bar{a}) \equiv_k (\mathfrak{F}(I_1), \bar{a}_1)$, and, therefore, $\bar{a}_1 \in \underline{\text{certain}}_{\mathcal{M}}(Q, I_1)$. $\square$

This result provides us with both necessary and sufficient conditions to obtain non-rewritability results in data exchange. However, to apply this proposition is often a nontrivial task as it involves not only playing Ehrenfeucht-Fraïssé games between structures, but also finding two structures I_1 and I_2 over schema $\mathbf{S}$ such that their transformations $\mathfrak{F}(I_1)$ and $\mathfrak{F}(I_2)$ over the *different* schema $\mathbf{T}$ still allow the duplicator to win the game (notice that these transformations could "create" null values).

It would be nice then to have a tool for proving non-rewritability results in data exchange such that (1) it does not involve playing Ehrenfeucht-Fraïssé games between structures, and (2) it does not require inspecting the transformation $\mathfrak{F}(I)$ but only the source instance I. We present such tool below, based on the results of Sections 3.1 and 3.2. As usual, we make use of locality notions as a simple way to find winning duplicator strategies.

4.2. A locality-based tool

Before presenting our locality-based tool for proving non-rewritability results in data exchange, we introduce the basic concepts and techniques related to locality as a tool to prove inexpressibility results for FO.

Neighborhoods and locality. Though EF games are the main tool to prove inexpressibility results for FO, finding winning duplicator strategies in such games becomes nontrivial for fairly simple cases. But often we can avoid those intricate combinatorial arguments by using rather simple sufficient conditions that guarantee a winning strategy for the duplicator [Fag97]. Most of such conditions are based on the idea of *locality.*

Notions of locality [Han65, Gai82, HLN99] have been widely used to prove easy inexpressibility results for first-order logic and some of its counting extensions [Nur96, Ete97, Nur00]. The intuition underlying those notions of locality is that FO cannot express properties that involve nontrivial recursive computations (such as connectivity, cyclicity, etc). The setting of locality is as follows. The *Gaifman graph* $\mathcal{G}(I)$ of an instance I of $\mathbf{R}$ is the graph whose nodes are the elements of $\mathrm{dom}(I)$, and such that there exists an edge between a and b in $\mathcal{G}(I)$ if and only if a and b belong to the same tuple of a relation $I(R)$, for some $R \in \mathbf{R}$. For example, if I is an undirected graph, then $\mathcal{G}(I)$ is I itself. The distance between two elements a and b in I, denoted by $\Delta_I(a,b)$ (or $\Delta(a,b)$, if I is understood), is the distance between them in $\mathcal{G}(I)$. We define $\Delta(\bar{a},b)$ as the minimum value of $\Delta(a,b)$ where a is an element of $\bar{a}$.

Given a tuple $\bar{a} = (a_1, \ldots, a_m) \in \mathrm{dom}(I)^m$, we define the instance $N_d^I(\bar{a})$, called the *d-neighborhood of $\bar{a}$ in I*, as the restriction of I to the elements at distance at most d from $\bar{a}$, with the members of $\bar{a}$ treated as distinguished elements. That is, if two neighborhoods $N_d^I(\bar{a})$ and $N_d^I(\bar{b})$ are isomorphic (written as $N_d^I(\bar{a}) \cong N_d^I(\bar{b})$), then there is an isomorphism $f : N_d^I(\bar{a}) \to N_d^I(\bar{b})$ such that $f(a_i) = b_i$, for $1 \leq i \leq m$.

For two instances I_1 and I_2 of the same schema, and m-tuples $\bar{a}$ in $\mathrm{dom}(I_1)$ and $\bar{b}$ in $\mathrm{dom}(I_2)$, we write

$$(I_1, \bar{a}) \leftrightarrows_d (I_2, \bar{b})$$

if there is a bijection $f : \mathrm{dom}(I_1) \to \mathrm{dom}(I_2)$ such that $N_d^{I_1}(\bar{a}c) \cong N_d^{I_2}(\bar{b}f(c))$ for every $c \in \mathrm{dom}(I_1)$. Note that $(I_1, \bar{a}) \leftrightarrows_d (I_2, \bar{b})$ implies that the domains of I_1 and I_2 have the same cardinality.

As we mentioned, the notion of locality allows one to find simple sufficient conditions that ensure a winning strategy for the duplicator in the Ehrenfeucht-Fraïssé game. This is summarized in the following theorem:

Theorem 4.2. (see [Gai82, FSV95, HLN99]) *The following holds:*

- *For every number $k \geq 0$ there exists a number $d \geq 0$ such that $N_d^I(\bar{a}) \cong N_d^I(\bar{b})$ implies $(I, \bar{a}) \equiv_k (I, \bar{b})$, for every instance I and tuples $\bar{a}$ and $\bar{b}$ of elements in I.*
- *For every number $k \geq 0$ there exists a number $d \geq 0$ such that $(I_1, \bar{a}) \leftrightarrows_d (I_2, \bar{b})$ implies $(I_1, \bar{a}) \equiv_k (I_2, \bar{b})$, for every pair I_1, I_2 of instances and tuples $\bar{a}$ in $\mathrm{dom}(I_1)$ and $\bar{b}$ in $\mathrm{dom}(I_2)$.*

The first part of Theorem 4.2 is directly related to a property of FO known as *Gaifman-locality* [Gai82, HLN99], while the second part is related to a property known as *Hanf-locality* [FSV95, HLN99]. Notice that Gaifman-locality deals with a single instance I at a time, whereas the notion of Hanf-locality deals with a pair I_1, I_2 of instances at a time.

The first part of Theorem 4.2 can be further refined. In fact, Gaifman's theorem [Gai82] implies the following:

Theorem 4.3. *For every $k \geq 0$ there are numbers $d, \ell \geq 0$ such that if $N_d^I(\bar{a}) \equiv_\ell N_d^I(\bar{b})$, then $(I, \bar{a}) \equiv_k (I, \bar{b})$, for every instance I and tuples $\bar{a}$ and $\bar{b}$ in I.*

Locality tools for proving non-rewritability results in data exchange. We introduce two new notions of locality below, that are modifications of the standard notions of locality presented above, and can be applied in the data exchange scenario to prove non-rewritability results.

Definition 4.4. *Given a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ and an m-ary query Q over $\mathbf{T}$, where $m \geq 0$, we say that:*

- *Q is* locally source-dependent in Gaifman-sense under $\mathcal{M}$ *if there is $d \geq 0$ such that for every instance I of $\mathbf{S}$ and for every $\bar{a}, \bar{b} \in \mathrm{dom}(I)^m$, if $N_d^I(\bar{a}) \cong N_d^I(\bar{b})$ then ($\bar{a} \in \underline{\mathrm{certain}}_{\mathcal{M}}(Q, I) \Leftrightarrow \bar{b} \in \underline{\mathrm{certain}}_{\mathcal{M}}(Q, I)$).*
- *Q is* locally source-dependent in Hanf-sense under $\mathcal{M}$ *if there is $d \geq 0$ such that for every pair of instances I_1 and I_2 of $\mathbf{S}$ and for every $\bar{a} \in \mathrm{dom}(I_1)^m$ and $\bar{b} \in \mathrm{dom}(I_2)^m$, if $(I_1, \bar{a}) \leftrightarrows_d (I_2, \bar{b})$ then ($\bar{a} \in \underline{\mathrm{certain}}_{\mathcal{M}}(Q, I_1) \Leftrightarrow \bar{b} \in \underline{\mathrm{certain}}_{\mathcal{M}}(Q, I_2)$).*

We next show that these notions apply to all queries rewritable over the core or the canonical universal solution.

Theorem 4.5. *Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting, and Q a query over $\mathbf{T}$. Assume that Q is rewritable over transformation $\mathfrak{F}_{\mathrm{univ}}$ or $\mathfrak{F}_{\mathrm{core}}$. Then Q is locally source-dependent in both Hanf- and Gaifman-sense under $\mathcal{M}$.*

Proof: Recall that $\tau_k^{(\mathfrak{F}_{\mathrm{core}}(I), \bar{a})}(\bar{x})$ is the FO formula that defines the rank-k FO type of $(\mathfrak{F}_{\mathrm{core}}(I), \bar{a})$. Then from Lemma 3.7 we have that for every $k \geq 0$, and for every source instance I_1 and m-tuple $\bar{a} \in \mathrm{dom}(\mathfrak{F}_{\mathrm{univ}}(I_1))^m$ of constants, there is an FO formula $\phi(\bar{x})$ over schema $\mathbf{T}$ such that for every source instance I_2 and m-tuple $\bar{b}$ of constants in $\mathfrak{F}_{\mathrm{univ}}(I_2)^m$,

$$\mathfrak{F}_{\mathrm{univ}}(I_2) \models \phi(\bar{b}) \iff \mathfrak{F}_{\mathrm{core}}(I_2) \models \tau_k^{(\mathfrak{F}_{\mathrm{core}}(I_1), \bar{a})}(\bar{b}). \tag{6}$$

As we show next, the latter implies that for every $k \geq 0$ the following holds:

$$(\mathfrak{F}_{\mathrm{univ}}(I_1), \bar{a}) \equiv_{\mathrm{qr}(\phi)} (\mathfrak{F}_{\mathrm{univ}}(I_2), \bar{b}) \implies (\mathfrak{F}_{\mathrm{core}}(I_1), \bar{a}) \equiv_k (\mathfrak{F}_{\mathrm{core}}(I_2), \bar{b}), \tag{7}$$

for every pair of source instances I_1 and I_2 and m-tuples of constants $\bar{a} \in \mathrm{dom}(\mathfrak{F}_{\mathrm{core}}(I_1))^m$ and $\bar{b} \in \mathrm{dom}(\mathfrak{F}_{\mathrm{core}}(I_2))^m$. This is because $\mathfrak{F}_{\mathrm{core}}(I_1) \models \tau_k^{(\mathfrak{F}_{\mathrm{core}}(I_1), \bar{a})}(\bar{a})$, and hence from (6) we have that $\mathfrak{F}_{\mathrm{univ}}(I_1) \models \phi(\bar{a})$. Thus, if $(\mathfrak{F}_{\mathrm{univ}}(I_1), \bar{a}) \equiv_{qr(\phi)} (\mathfrak{F}_{\mathrm{univ}}(I_2), \bar{b})$ it must be the case that $\mathfrak{F}_{\mathrm{univ}}(I_2) \models \phi(\bar{b})$, and hence, again from (6), that $\mathfrak{F}_{\mathrm{core}}(I_2) \models \tau_k^{(\mathfrak{F}_{\mathrm{core}}(I_1), \bar{a})}(\bar{b})$. The latter implies that $(\mathfrak{F}_{\mathrm{core}}(I_1), \bar{a}) \equiv_k (\mathfrak{F}_{\mathrm{core}}(I_2), \bar{b})$.

On the other hand, from Theorem 3.13 we have that for every $k \geq 0$ there exists $k' \geq 0$ such that

$$(I_1, \bar{a}) \equiv_{k'} (I_2, \bar{b}) \implies (\mathfrak{F}_{\mathrm{univ}}(I_1), \bar{a}) \equiv_k (\mathfrak{F}_{\mathrm{univ}}(I_2), \bar{b}), \tag{8}$$

for every pair of source instances I_1 and I_2 and m-tuples of constants $\bar{a} \in \mathrm{dom}(\mathfrak{F}_{\mathrm{univ}}(I_1))^m$ and $\bar{b} \in \mathrm{dom}(\mathfrak{F}_{\mathrm{univ}}(I_2))^m$.

Thus, using (7) and (8) together with Proposition 4.1, one can immediately derive the following:

Claim 4.6. *Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting, and assume that $\mathfrak{F}$ is either $\mathfrak{F}_{\text{univ}}$ or $\mathfrak{F}_{\text{core}}$. An m-ary query Q, for $m \geq 0$, over schema $\mathbf{T}$ is not rewritable over $\mathfrak{F}$ under $\mathcal{M}$ if for all $k \geq 0$ there exist instances I_1 and I_2 of $\mathbf{S}$ and m-tuples $\bar{a} \in \text{dom}(I_1)^m$ and $\bar{b} \in \text{dom}(I_2)^m$, such that $(I_1, \bar{a}) \equiv_k (I_2, \bar{b})$, $\bar{a} \in \underline{\text{certain}}_{\mathcal{M}}(Q, I_1)$, but $\bar{b} \notin \underline{\text{certain}}_{\mathcal{M}}(Q, I_2)$.*

Now we conclude the proof of Theorem 4.5. Recall from Theorem 4.2 that for every $k \geq 0$ there exists a number d such that $N_d^I(\bar{a}) \cong N_d^I(\bar{b})$ implies $(I, \bar{a}) \equiv_k (I, \bar{b})$, and that for every $k \geq 0$ there exists a number d such that $(I_1, \bar{a}) \leftrightarrows_d (I_2, \bar{b})$ implies $(I_1, \bar{a}) \equiv_k (I_2, \bar{b})$. Then the theorem follows directly from Claim 4.6. □

4.3. Applications of the tools

In order to prove that a query is not rewritable over the core or the canonical universal solution, it suffices to show that it is not locally source dependent in either Hanf- or Gaifman-sense. We now apply Theorem 4.5 to prove non-rewritability results in extremely simple data exchange settings. We call a data exchange setting *copying* if $\mathbf{S}$ and $\mathbf{T}$ are two copies of the same schema (that is, $\mathbf{S} = \{R_1, \ldots, R_n\}, \mathbf{T} = \{R'_1, \ldots, R'_n\}$, and R_i and R'_i have the same arity), and $\Sigma_{st} = \{R_i(\bar{x}) \rightarrow R'_i(\bar{x}) \mid i = 1, \ldots, n\}$. Note that a copying setting is both LAV and GAV.

We have the following simple lemma about rewritability for copying settings.

Lemma 4.7. *Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a copying data exchange setting. A query Q over $\mathbf{T}$ is rewritable over the canonical universal solution under $\mathcal{M}$ if and only if it is rewritable over the core under $\mathcal{M}$ if and only if it is rewritable over the source $\mathcal{M}$.*

Proof: The source, canonical universal solution, and the core are all identical (up to a renaming of relation symbols) for a copying data exchange setting. So rewritability for one gives rewritability for them all. □

Corollary 4.8. *There is a copying data exchange setting $\mathcal{M}$ and an FO formula, with a single free variable, that is neither rewritable over the canonical universal solution, nor over the core, nor over the source under $\mathcal{M}$.*

Proof: Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be the data exchange setting such that $\mathbf{S} = \langle G(\cdot, \cdot),\ R(\cdot)\rangle$, $\mathbf{T} = \langle G'(\cdot, \cdot),\ R'(\cdot)\rangle$ and $\Sigma_{st} = \{G(x, y) \rightarrow G'(x, y),\ R(x) \rightarrow R'(x)\}$. Define a unary query Q over the target schema as:

$$R'(x) \ \vee\ \exists y \exists z (R'(y) \wedge G'(y, z) \wedge \neg R'(z)).$$

By Lemma 4.7, we need only show that Q is not rewritable over $\mathfrak{F}_{\text{univ}}$. Assume that Q is rewritable over $\mathfrak{F}_{\text{univ}}$ under $\mathcal{M}$. Then it is locally source-dependent in Gaifman-sense under $\mathcal{M}$, that is, there exists $d \geq 0$ such that for every source instance I and every $a, b \in \text{dom}(I)$, we have $a \in \underline{\text{certain}}_{\mathcal{M}}(Q, I)$ if and only if $b \in \underline{\text{certain}}_{\mathcal{M}}(Q, I)$ whenever $N_d^I(a) \cong N_d^I(b)$.

Define a source instance I as shown in Figure 1: $I(G)$ is the disjoint union of two cycles of length $2d + 2$, and $I(R) = \{c\}$. Then $N_d^I(a) \cong N_d^I(b)$, which implies that $a \in \underline{\text{certain}}_{\mathcal{M}}(Q, I)$ if and only if $b \in \underline{\text{certain}}_{\mathcal{M}}(Q, I)$. However, we now show that $a \in \underline{\text{certain}}_{\mathcal{M}}(Q, I)$ while $b \notin \underline{\text{certain}}_{\mathcal{M}}(Q, I)$. Indeed, if J is an arbitrary solution for I, then $J \models R'(a) \vee \exists y \exists z (R'(y) \wedge G'(y, z) \wedge \neg R'(z))$ (if J does not

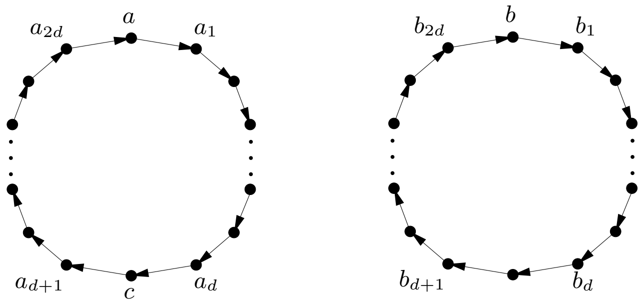


Figure 1: Instance I of Corollary 4.8.

satisfy the second disjunct, then $J \models \forall y \forall z (R'(y) \wedge G'(y,z) \rightarrow R'(z))$ and, hence, $J \models R'(a)$ since $R'(c)$ is true in every solution, and a and c are on the same cycle). Furthermore, if J_0 is a target instance such that $J_0(G') = I(G)$ and $J_0(R')$ includes exactly all the points in the cycle containing a, then J_0 is a solution for I. However, $J_0 \not\models Q(b)$, and thus $b \notin \underline{\text{certain}}_{\mathcal{M}}(Q,I)$. This contradiction shows that Q is not rewritable over the canonical universal solution, as desired. $\square$

In the previous proof we used our locality tool in Gaifman-sense to prove that a query with one free-variable does not admit a rewriting. In order to prove that a Boolean query (i.e. a query without free variables) is not rewritable over $\mathfrak{F}_{\text{univ}}$ or $\mathfrak{F}_{\text{core}}$, we need to use the locality tool in Hanf-sense, as shown in the proof of the following corollary:

Corollary 4.9. *There is a copying data exchange setting $\mathcal{M}$ and a Boolean FO formula that is neither rewritable over the canonical universal solution, nor over the core, nor over the source under $\mathcal{M}$.*

Proof: Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be the data exchange setting such that S consists of binary relation E and unary relations A and B, the schema $\mathbf{T}$ consists of binary relation G and unary relations P and R, and Σ_{st} consists of the st-tgds:

$$\begin{array}{rcl} E(x,y) & \rightarrow & G(x,y) \\ A(x) & \rightarrow & P(x) \\ B(x) & \rightarrow & R(x) \end{array}$$

Clearly, $\mathcal{M}$ is copying. The query Q over $\mathbf{T}$ is defined as:

$$\exists x \exists y\, (P(x) \wedge R(y) \wedge G(x,y)) \;\vee\; \exists x \exists y \exists z\, (G(x,z) \wedge G(z,y) \wedge \neg G(x,y)).$$

This is the union of a conjunctive query and a conjunctive query with a single negated relational atom. By Lemma 4.7, we need only show that Q is not rewritable over the canonical universal solution.

Assume otherwise. We prove below that Q is not locally source-dependent in Hanf-sense under $\mathcal{M}$, which directly contradicts Theorem 4.5. Recall that for this we need to construct, for every $d \geq 0$, two source instances I_1 and I_2 such that $I_1 \leftrightarrows_d I_2$ but $\underline{\text{certain}}_{\mathcal{M}}(Q,I_1) \neq \underline{\text{certain}}_{\mathcal{M}}(Q,I_2)$.

Define source instances I_1 and I_2 as shown in Figure 2: $I_1(E)$ is a cycle of length $4d+4$ with elements

$$a_1, b_1, c_1, c_2, \ldots, c_{4d+2},$$

$I_2(E)$ is the disjoint union of two cycles of length $2d+2$, the first one with elements

$$a_2, e_1, \ldots, e_{2d+1},$$

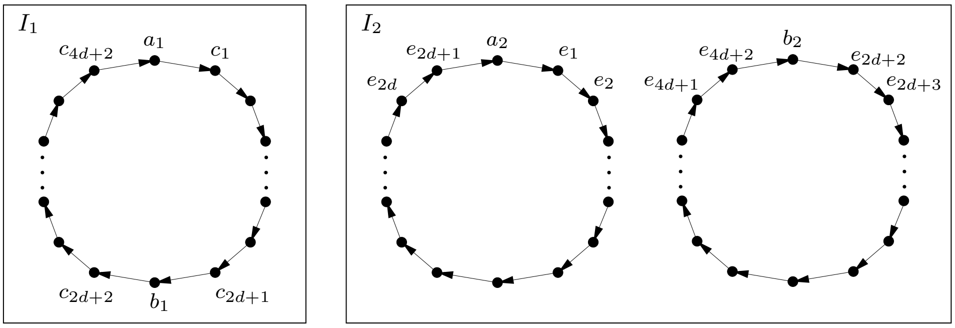


Figure 2: Instances I_1 and I_2 of Corollary 4.9.

and the second one with elements

$$b_2, e_{2d+2}, \ldots, e_{4d+2},$$

$I_1(A) = \{a_1\}$, $I_2(A) = \{a_2\}$, $I_1(B) = \{b_1\}$ and $I_2(B) = \{b_2\}$.

Let $f : \text{dom}(I_1) \to \text{dom}(I_2)$ be defined as $f(a_1) = a_2$, $f(b_1) = b_2$ and

$$\begin{cases} f(c_i) = e_i & \text{if } 1 \le i \le d+1 \text{ or } 2d+2 \le i \le 3d+2 \\ f(c_i) = e_{2d+1+i} & \text{if } d+2 \le i \le 2d+1 \\ f(c_i) = e_{i-2d-1} & \text{if } 3d+3 \le i \le 4d+2 \end{cases}$$

Clearly, f is a bijection from $\text{dom}(I_1)$ into $\text{dom}(I_2)$. Furthermore, a simple case analysis proves that for every $v \in \text{dom}(I_1)$ it is the case that $N_d^{I_1}(v) \cong N_d^{I_2}(f(v))$. This implies that $I_1 \leftrightarrows_d I_2$. However, we prove below that $\underline{\text{certain}}_{\mathcal{M}}(Q, I_1) = \texttt{true}$ while $\underline{\text{certain}}_{\mathcal{M}}(Q, I_2) = \texttt{false}$.

Let us consider first I_1. Let J_1 be the canonical universal solution for I_1. Notice that J_1 is just a "copy" of I_1 over the target; that is, $I_1(E) = J_1(G)$, $I_1(A) = J_1(P)$ and $I_1(B) = J_1(R)$. Let J_1' be an arbitrary solution for I_1 that does not satisfy the second disjunct $\exists x \exists y \exists z (G(x,z) \wedge G(z,y) \wedge \neg G(x,y))$ of Q. Then it must be the case that the transitive closure of $I_1(E)$ is contained in $J_1'(G)$, and hence that J_1' satisfies the first disjunct $\exists x \exists y (P(x) \wedge R(y) \wedge G(x,y))$ of Q. This is because $a_1 \in J_1'(P)$, $b_1 \in J_1'(R)$ and (a_1, b_1) belongs to the transitive closure of $I_1(E)$. We conclude that $\underline{\text{certain}}_{\mathcal{M}}(Q, I_1) = \texttt{true}$.

Let us consider now I_2. Again, the canonical universal solution J_2 for I_2 is a "copy" of I_2 over the target. Let J_2' be the solution for I_2 that is obtained from J_2 by extending the interpretation of G with every tuple that belongs to the transitive closure of $J_2(G)$. Then clearly $J_2' \not\models \exists x \exists y \exists z (G(x,z) \wedge G(z,y) \wedge \neg G(x,y))$. Moreover, since a_2 is the only element in J_2' that belongs to the interpretation of P, and b_2 is the only element in J_2' that belongs to the interpretation of R, and a_2 and b_2 belong to different connected components of the graph induced by $J_2(G)$, it is the case that $J_2' \not\models \exists x \exists y (P(x) \wedge R(y) \wedge G(x,y))$. We conclude that $J_2' \not\models Q$, and hence that $\underline{\text{certain}}_{\mathcal{M}}(Q, I_2) = \texttt{false}$. $\square$

4.4. Rewritability is undecidable

In this section we prove Proposition 3.3, with the help of the locality tools and the non-rewritability results we have just obtained. Recall that we want to prove the following: Given a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ and a query Q over $\mathbf{T}$ specified in FO, it is undecidable whether Q is rewritable over the canonical universal solution (resp., over the core) under $\mathcal{M}$. Interestingly, it follows from the proof that this continues to hold even if $\mathcal{M}$ is a copying setting.

Proof of Proposition 3.3: Recall that Trakhtenbrot's theorem states that the following problem is undecidable: Given a Boolean FO formula ϕ over schema $\mathbf{R}$, is there an instance K of $\mathbf{R}$ such that $K \models \phi$? The proof of this theorem is by a reduction from the halting problem for Turing machines on the empty input, that is, for each Turing machine M one constructs a Boolean FO formula ϕ such that M halts on the empty input if and only if there is an instance K such that $K \models \phi$. However, it is not hard to notice that Trakhtenbrot's theorem can be proved in a way that all quantification in ϕ is restricted to the active domain (cf. [Lib04]). This proves that the following problem is also undecidable: Given a *domain-independent* Boolean FO formula, is there an instance K of $\mathbf{R}$ such that $K \models \phi$? In order to prove Proposition 3.3, we reduce the latter problem to the complement of the problem of checking whether a Boolean FO formula is rewritable over the canonical universal solution or the core. That is, we show that for each schema $\mathbf{R}$ and domain-independent Boolean FO formula ϕ over $\mathbf{R}$, one can compute a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ and a Boolean FO formula θ over $\mathbf{T}$ such that the following two facts are equivalent:

1. There is an instance K of $\mathbf{R}$ such that $K \models \phi$.
2. θ is *not* rewritable over the canonical universal solution (resp., the core) under $\mathcal{M}$.

Take an arbitrary schema $\mathbf{R}$ and a domain-independent Boolean FO formula ϕ over $\mathbf{R}$. We show how to construct $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ and θ from ϕ and $\mathbf{R}$. First of all, it follows from the proof of Corollary 4.9 that if $\mathcal{M}' = (\mathbf{S}', \mathbf{T}', \Sigma'_{st})$ is the data exchange setting such that $\mathbf{S}'$ consists of binary relation E and unary relations A and B, the schema $\mathbf{T}'$ consists of binary relation G and unary relations P and R, and Σ'_{st} consists of the copying stds:

$$\begin{aligned} E(x,y) &\rightarrow G(x,y) \\ A(x) &\rightarrow P(x) \\ B(x) &\rightarrow R(x), \end{aligned}$$

then the query Q:

$$\exists x \exists y \, (P(x) \wedge R(y) \wedge G(x,y)) \vee \exists x \exists y \exists z \, (G(x,z) \wedge G(z,y) \wedge \neg G(x,y))$$

is rewritable neither over the canonical universal solution nor over the core under $\mathcal{M}'$. Notice, in addition, that Q is domain independent. Without loss of generality we assume that $\mathbf{S}'$ and $\mathbf{R}$ contain no relation symbols in common. With the help of $\mathcal{M}' = (\mathbf{S}', \mathbf{T}', \Sigma'_{st})$ and Q we can construct $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ and θ as follows:

- $\mathbf{S}$ consists of all the relation symbols that are either in $\mathbf{R}$ or in $\mathbf{S}'$ (recall that $\mathbf{R}$ and $\mathbf{S}'$ are assumed to be disjoint).
- $\mathbf{T}$ consists of all the relation symbols that are either in $\mathbf{R}'$ or in $\mathbf{T}'$, where $\mathbf{R}'$ is a disjoint copy of $\mathbf{R}$; that is, if $\mathbf{R} = \{R_1, \ldots, R_n\}$ then $\mathbf{R}' = \{R'_1, \ldots, R'_n\}$ and R_i and R'_i have the same arity. Furthermore, we assume without loss of generality that $\mathbf{R}'$ has no relation symbols in common with either $\mathbf{T}'$ or $\mathbf{S}'$.
- Σ_{st} is defined as
$$\Sigma'_{st} \;\cup\; \{R_i(\bar{x}) \;\rightarrow\; R'_i(\bar{x}) \mid 1 \leq i \leq n\}.$$
That is, Σ_{st} consists of the st-tgds in Σ'_{st}, that relate source relation symbols in $\mathbf{S}'$ with target relation symbols in $\mathbf{T}'$ in a way that is consistent with $\mathcal{M}'$, plus a set of *copying* st-tgds that transfer the source content of each relation symbol in $\mathbf{R}$ into the corresponding target relation symbol in $\mathbf{R}'$.

- Finally, θ is defined as $\phi' \rightarrow Q$, where ϕ' is the Boolean FO formula over $\mathbf{R}'$ that is obtained from ϕ by replacing each occurrence of the relation symbol R_i with R'_i, for every i with $1 \leq i \leq n$.

We prove next that there exists an instance K of $\mathbf{R}$ such that $K \models \phi$ if and only if θ is not rewritable over the canonical universal solution (resp., over the core) under $\mathcal{M}$.

Assume that for every instance K of $\mathbf{R}$ it is the case that $K \not\models \phi$. Then for every source instance I of $\mathbf{S}$, and solution J for I under $\mathcal{M}$, we have $J \not\models \phi'$. Assume otherwise. Then the restriction J' of J to the relation symbols in $\mathbf{R}'$ also satisfies ϕ' (this is because ϕ, and hence ϕ', is domain independent; and so, its evaluation does not depend of the interpretation of the relation symbols in $\mathbf{R}'$ that are not mentioned in ϕ'), and, therefore, the instance K of $\mathbf{R}$ that is obtained from J' by setting $K(R_i)$ to coincide with the interpretation of R_i in J', for each $1 \leq i \leq n$, satisfies ϕ, which is a contradiction. This implies that θ is rewritable over the canonical universal solution and over the core simply by the Boolean constant `true`.

Assume, on the other hand, that for at least some instance K of $\mathbf{R}$ it is the case that $K \models \phi$. We prove that $\phi' \rightarrow Q$ is not rewritable over the canonical universal solution, nor over the core, under $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$. To prove this it is sufficient to prove that $\phi' \rightarrow Q$ is not locally source-dependent under $\mathcal{M}$. In the present case, for each $d \geq 0$ we construct source instances I'_1 and I'_2 such that:

- The restriction of both I'_1 and I'_2 to schema $\mathbf{R}$ corresponds to the instance K, and
- the restriction of I'_i, for $i = 1$ or $i = 2$, to $\mathbf{S}'$ corresponds to the instance I_i used in the proof of Corollary 4.9.

We now show that $I'_1 \leftrightarrows_d I'_2$. This is because I'_i, for $i = 1$ or $i = 2$, essentially consists of the disjoint union of I_i and K, and we know from the proof of Corollary 4.9 that $I_1 \leftrightarrows_d I_2$. We show below that $\underline{\text{certain}}_{\mathcal{M}}(\phi' \rightarrow Q, I'_1) \neq \underline{\text{certain}}_{\mathcal{M}}(\phi' \rightarrow Q, I'_2)$.

Let us consider first I'_1. Consider an arbitrary solution J'_1 for I'_1 under $\mathcal{M}$. Then the restriction J_1 of J'_1 to the relation symbols in $\mathbf{T}'$ is a solution for I_1 under $\mathcal{M}'$. From the proof of Corollary 4.9 it holds that $\underline{\text{certain}}_{\mathcal{M}'}(Q, I_1) = \texttt{true}$, and hence $J_1 \models Q$. Since Q is domain independent, it follows that $J'_1 \models Q$. We conclude that $\underline{\text{certain}}_{\mathcal{M}}(\phi' \rightarrow Q, I'_1) = \texttt{true}$.

Let us consider now I'_2. Since $\underline{\text{certain}}_{\mathcal{M}'}(Q, I_2) = \texttt{false}$ from the proof of Corollary 4.9, there exists a solution J_2 for I_2 under $\mathcal{M}'$ such that $J_2 \not\models Q$. Consider now the instance J'_2 of schema $\mathbf{T}$ that consists of the disjoint union of J_2 (over schema $\mathbf{T}'$) and a "copy" of K (over schema $\mathbf{R}'$). Clearly, J'_2 is a solution for I'_2 under $\mathcal{M}$. The restriction of J'_2 to the relation symbols in $\mathbf{T}'$ (which is J_2) does not satisfy Q. But since Q is domain independent, we also have $J'_2 \not\models Q$. Furthermore, the restriction of J'_2 to $\mathbf{R}$ is a "copy" of K, which satisfies ϕ. Since ϕ is a domain-independent query over $\mathbf{R}$, it must be the case that $J'_2 \models \phi$. We conclude that $J'_2 \models \phi \wedge \neg Q$, and hence that $\underline{\text{certain}}_{\mathcal{M}}(\phi \rightarrow Q, I'_2) = \texttt{false}$. This completes the proof. $\square$

Notice that the formula $\theta = \phi' \rightarrow Q$ used in the previous proof is domain independent, since both ϕ' and Q are domain independent. It follows then that the notion of rewritability over the canonical universal solution, or the core, is undecidable even for queries specified as Boolean and domain-independent FO formulas.

5. Structural Properties of Transformations

As another application of the results presented in Sections 3.1 and 3.2, we give some nice structural properties of the transformations $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$. In particular, we show that they preserve the *local* character of the data. That is, tuples with similar neighborhoods in I have similar neighborhoods in $\mathfrak{F}(I)$. We call this property *local consistency.* Based on the types of logical formulae used in the data exchange settings, we establish different types of local consistency exhibited in data exchange.

5.1. Local consistency

We now introduce the notion of local consistency of transformations from $\mathsf{Inst}(\mathbf{S})$ to $\mathsf{Inst}(\mathbf{T})$. The first notion says that neighborhoods around elements common to the input and output instances are preserved. Informally, if tuples $\bar{a}$ and $\bar{b}$ in $\text{dom}(I)^m$, for $m > 0$, are present in the domain of the resulting instance J of $\mathbf{T}$, then the isomorphism of sufficiently large neighborhoods of $\bar{a}$ and $\bar{b}$ in I guarantees that their neighborhoods are isomorphic in J as well. Formally, we define this as follows.

Definition 5.1 (Local consistency). *A mapping $\mathfrak{F} : \mathsf{Inst}(\mathbf{S}) \to \mathsf{Inst}(\mathbf{T})$ is* locally consistent *if for every $m, d \geq 0$ there exists $d' \geq 0$ such that, for every instance I of $\mathbf{S}$ and tuples $\bar{a}, \bar{b} \in \text{dom}(I)^m$, if $N^I_{d'}(\bar{a}) \cong N^I_{d'}(\bar{b})$, then*

1. $\bar{a} \in \text{dom}(\mathfrak{F}(I))^m \;\Leftrightarrow\; \bar{b} \in \text{dom}(\mathfrak{F}(I))^m$, *and*
2. $N^{\mathfrak{F}(I)}_d(\bar{a}) \cong N^{\mathfrak{F}(I)}_d(\bar{b})$.

The next theorem guarantees the local consistency of both $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ in the LAV setting.

Theorem 5.2 ($\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ are locally consistent for LAV settings). *In the LAV setting, both the canonical universal solution transformation $\mathfrak{F}_{\text{univ}}$ and the core transformation $\mathfrak{F}_{\text{core}}$ are locally consistent.*

The proof of this theorem follows from the two consecutive lemmas below.

Lemma 5.3. *In the LAV setting, the canonical universal solution transformation $\mathfrak{F}_{\text{univ}}$ is locally consistent.*

Proof: Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a LAV setting, and fix $d \geq 0$. We show that for every source instance I and tuples $\bar{a}$ and $\bar{b}$ in $\text{dom}(I)^m$, if $N^I_{d+1}(\bar{a}) \cong N^I_{d+1}(\bar{b})$ then

(1.) $\bar{a} \in \text{dom}(\mathfrak{F}_{\text{univ}}(I))^m \;\Leftrightarrow\; \bar{b} \in \text{dom}(\mathfrak{F}_{\text{univ}}(I))^m$, and

(2.) $N^{\mathfrak{F}_{\text{univ}}(I)}_d(\bar{a}) \cong N^{\mathfrak{F}_{\text{univ}}(I)}_d(\bar{b})$.

Let I be an arbitrary source instance, and denote $\mathfrak{F}_{\text{univ}}(I)$ by J. We prove (1.) first. Let $\bar{a}$ be $(a_1, \ldots, a_m)$ and $\bar{b}$ be $(b_1, \ldots, b_m)$. We show that for each $i \in [1, m]$, if $a_i \in \text{dom}(J)$ then $b_i \in \text{dom}(J)$. Assume that a_i belongs to $\text{dom}(J)$, for some $i \in [1, m]$. This is caused by an std in Σ_{st} of the form

$$S(\bar{x}) \;\to\; \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y}), \tag{9}$$

where S is a source relation symbol, and where there is a tuple $\bar{c}$ of constant symbols that includes a_i such that $S(\bar{c})$ holds in the source instance I. It follows that every member of $\bar{c}$ belongs to $N^I_{d+1}(\bar{a})$. Let f be a bijection witnessing $N^I_{d+1}(\bar{a}) \cong N^I_{d+1}(\bar{b})$. Then $S(f(\bar{c}))$ holds in the source instance I, and $f(a_i) = b_i$ belongs to $\mathrm{dom}(J)$. In the same way we can show that for each $i \in [1, m]$, if $b_i \in \mathrm{dom}(J)$ then $a_i \in \mathrm{dom}(J)$. This proves that $a_i \in \mathrm{dom}(J)$ if and only if $b_i \in \mathrm{dom}(J)$, for each $i \in [1, m]$. Hence, $\bar{a} \in \mathrm{dom}(J)^m$ if and only if $\bar{b} \in \mathrm{dom}(J)^m$.

Now we show that $N^I_{d+1}(\bar{a}) \cong N^I_{d+1}(\bar{b})$ implies $N^J_d(\bar{a}) \cong N^J_d(\bar{b})$. We require the following claim.

Claim 5.4. *All constants in $N^J_d(\bar{a})$ also belong to $N^I_d(\bar{a})$.*

Proof of the claim: Recall that for each instance I', we denote the Gaifman graph of I' by $\mathcal{G}(I')$. Let c_1 and c_2 be distinct constants. Assume now that there is a path in $\mathcal{G}(J)$ between c_1 and c_2 where every point except the endpoints c_1 and c_2 are nulls. This is caused by an std in Σ_{st} of the form (9), where there is a tuple $\bar{c}$ of constant symbols that includes c_1 and c_2 such that $S(\bar{c})$ holds in the source instance I. Hence, c_1 and c_2 are adjacent in $\mathcal{G}(I)$.

We now show that the constants in $N^J_d(\bar{a})$ also belong to $N^I_d(\bar{a})$. Let c be a constant different from all constants in the tuple $\bar{a}$, and such that $c \in N^J_d(\bar{a})$. Then there are $b_0, b_1, \ldots, b_d$ with b_0 a constant in tuple $\bar{a}$ and $b_d = c$, where b_i and b_{i+1} are adjacent in J, for $0 \leq i < d$. Assume that the number of b_i's that are constant is t (we know $2 \leq t \leq d + 1$). Let $b_{i_1}, \ldots, b_{i_t}$ be the b_i's that are constant, with $i_1 < \cdots < i_t$. From what we showed earlier, b_{i_j} and $b_{i_{j+1}}$ are adjacent in I, for $1 \leq j < t$. So $b_{i_1}, b_{i_2}, \ldots, b_{i_t}$ gives a path between $\bar{a}$ and c in I. Hence, the distance between $\bar{a}$ and c in I is at most $t - 1$, and c belongs to $N^I_d(\bar{a})$, as desired. □

Now we continue with the proof of Lemma 5.3. Let $\bar{s}$ be a tuple of one or more constants and zero or more nulls such that $T(\bar{s})$ holds in J for some target relation symbol T, and such that every member of $\bar{s}$ is in $N^J_d(\bar{a})$. This is caused by an std in Σ_{st} of the form (9), where there is a tuple $\bar{c}$ of constants that includes all of the constants of $\bar{s}$, such that $S(\bar{c})$ holds in the source I. But $\bar{s}$ contains at least one constant, and every constant in $\bar{s}$ belongs to $N^J_d(\bar{a})$. So $\bar{s}$ contains some constant in $N^J_d(\bar{a})$, and hence (from the claim above), in $N^I_d(\bar{a})$. Since $S(\bar{c})$ holds in I, and $\bar{c}$ contains some constant in $N^I_d(\bar{a})$, it follows that every member of $\bar{c}$ belongs to $N^I_{d+1}(\bar{a})$.

We just showed that the fact that $T(\bar{s})$ holds in $N^J_d(\bar{a})$ is caused by an std in Σ_{st} of the form (9), where there is a tuple $\bar{c}$ of constants in $N^I_{d+1}(\bar{a})$ such that $S(\bar{c})$ holds in I. Therefore, $N^I_{d+1}(\bar{a})$ completely *determines* $N^J_d(\bar{a})$. It follows that $N^I_{d+1}(\bar{a}) \cong N^I_{d+1}(\bar{b})$ implies $N^J_d(\bar{a}) \cong N^J_d(\bar{b})$, which finishes the proof of the lemma. □

Lemma 5.5. *In the LAV setting, the core transformation $\mathfrak{F}_{\text{core}}$ is locally consistent.*

Proof: By Lemma 5.3, the canonical universal solution transformation $\mathfrak{F}_{\text{univ}}$ of a LAV setting is locally consistent. We shall show that the mapping that maps the canonical universal solution onto the core is locally consistent. Since the composition of locally consistent transformations is locally consistent, and constants are preserved from canonical universal solutions to their cores, this is enough to prove the lemma. The proof proceeds by making use of the algorithm given for computing the core in the LAV setting [FKP05].

Let J be an instance with nulls. Recall that the Gaifman graph of the nulls of J is an undirected graph in which (1) the nodes are all the nulls of J, and (2) there is an edge between two nulls whenever the nulls belong to the same tuple of some relation in J. A *block* of nulls is the set of nulls in a connected component of the Gaifman graph of nulls. If v is a null of J, then we may refer to the block of nulls that contains v as the *block of* v. Note that, by the definition of blocks, the set of all nulls of J is partitioned into disjoint blocks.

Let h be a homomorphism of an instance J. Denote the result of applying h to J by $h(J)$. If $h(J)$ is a subinstance of J, then we call h an *endomorphism* of J. An endomorphism h of J is *useful* if $h(J) \neq J$ (i.e., $h(J)$ is a proper subinstance of J).

Let J and J' be two instances such that the nulls of J' form a subset of the nulls of J. Let h be some endomorphism of J', and let B be a block of nulls of J. We say that h is *J-local for B* if $h(x) = x$ whenever $x \notin B$. (Since all the nulls of J' are among the nulls of J, it makes sense to consider whether or not a null x of J' belongs to the block B of J.) We say that h is *J-local* if it is J-local for B, for some block B of J.

We now present an algorithm for computing the core of the universal solutions, when given the canonical universal solution J [FKP05].

1. Compute the blocks of J, and initialize J' to be J.
2. Check whether there exists a useful J-local endomorphism h of J'. If not, stop with result J'.
3. Update J' to be $h(J')$, and return to Step 2.

Let b be the maximal number of existentially quantified variables over all stds in Σ_{st}. It follows easily from the construction of the canonical universal solution J (by chasing with Σ_{st}) that b is an upper bound on the size of a block in J (see e.g. [FKP05]).

Let I be a source database, let J be a canonical universal solution, and let J_0 be the core of J. Assume $m \geq 0$ and $d \geq 1$ (we do not allow $d = 0$ for technical convenience, and it is clear that this restriction is unimportant). We need only show that whenever $\bar{a}$ and $\bar{b}$ are m-tuples with isomorphic $(d+b-1)$-neighborhoods in the canonical universal solution J, then $\bar{a}$ and $\bar{b}$ have isomorphic d-neighborhoods in the core. To simplify the wording, let us phrase this by saying that we need only show that the $(d+b-1)$-neighborhood $N^J_{d+b-1}(\bar{a})$ determines the d-neighborhood $N^{J_0}_d(\bar{a})$.

Since we are assuming a LAV setting, the stds in Σ_{st} are of the form $R(\bar{x}) \to \exists \bar{y} \varphi_{\mathbf{T}}(\bar{x}, \bar{y})$, where $R(\bar{x})$ is an atomic formula, and where $\varphi_{\mathbf{T}}(\bar{x}, \bar{y})$ is a conjunction of atomic formulae. Let n be a null in $N^J_d(\bar{a})$. Then there is such an std σ in Σ_{st} and there is a tuple $\bar{c}$ of constants where $R(\bar{c})$ holds in the source instance I, such that n, and every member of its block, is generated by chasing σ starting with $R(\bar{c})$. If every member of $\bar{c}$ were at least distance d from $\bar{a}$ in I (that is, outside of $N^I_d(\bar{a})$) then it is easy to see that n would be at least distance $d+1$ from $\bar{a}$ in J. But this is false, since $n \in N^J_d(\bar{a})$. So some member of $\bar{c}$ is in $N^I_{d-1}(\bar{a})$, and hence every member of $\bar{c}$ is in $N^I_d(\bar{a})$. We see from the definition of b that chasing with σ causes every member of the block of n to be in $N^J_{d+b-1}(\bar{a})$. Hence, if h is an endomorphism of J, then every member of the block B of v is mapped into $N^J_{d+b-1}(\bar{a})$ (this is because paths of length, say, m, that begin with a member a of $\bar{a}$ are mapped by each endomorphism into paths of length at most m that begin with a). So every J-local endomorphism maps the nulls of B (each of which is in $N^J_{d+b-1}(\bar{a})$) into points in $N^J_{d+b-1}(\bar{a})$.

We now show that there is enough information in $N^J_{d+b-1}(\bar{a})$ to produce $N^{J_0}_d(\bar{a})$. Hence, the $(d+b-1)$-neighborhood $N^J_{d+b-1}(\bar{a})$ determines the d-neighborhood $N^{J_0}_d(\bar{a})$, which as we noted is sufficient to prove the lemma.

Let us call a block *special* if it contains a null in $N_d^J(\bar{a})$. Consider a modified version of the algorithm for producing the core where the special blocks are selected first. Thus, the modified version of the algorithm selects a non-special block with a useful J-local endomorphism in Step 2 only when there is no special block with a useful J-local endomorphism. We can think of this algorithm as consisting of two phases. In the first phase, only special blocks are selected, and in the second phase, non-special blocks are selected. Since, as we showed, every J-local endomorphism maps the nulls of a special block B (each of which is in $N_{d+b-1}^J(\bar{a})$) into points in $N_{d+b-1}^J(\bar{a})$, it follows easily that there is enough information in $N_{d+b-1}^J(\bar{a})$ to carry out the first phase of the algorithm. Let C be the neighborhood about $\bar{a}$ of radius d in J' at the end of the first phase (J' is as defined in the algorithm). It is fairly easy to see that C is also the neighborhood about $\bar{a}$ of radius d in J' at the end of the second phase (intuitively, no changes take place in C in the second phase, because a useful J-local endomorphism can only remove tuples, not add tuples.) Since J' at the end of the second phase is the core, it follows that C is $N_d^{J_0}(\bar{a})$. So indeed, there is enough information in $N_{d+b-1}^J(\bar{a})$ to produce $N_d^{J_0}(\bar{a})$, which was to be shown. $\square$

The previous results stating the local consistency of both $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ do not extend to the GAV setting, even when restricted to conjunctive queries.

Proposition 5.6. *There are GAV(CQ) settings for which the corresponding transformations $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ are not locally consistent.*

Proof: In the GAV setting we have that $\mathfrak{F}_{\text{core}}(I) \cong \mathfrak{F}_{\text{univ}}(I)$, for each source instance I. So, we may assume in the following that $\mathfrak{F} = \mathfrak{F}_{\text{core}} = \mathfrak{F}_{\text{univ}}$. Consider a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$, where $\mathbf{S} = \langle E(\cdot,\cdot), U(\cdot)\rangle$, $\mathbf{T} = \langle R(\cdot,\cdot,\cdot)\rangle$ and Σ_{st} contains a single dependency $\{E(x,y) \wedge U(z) \rightarrow R(x,y,z)\}$. We will show that if $d = 2$, then for every $d' \geq 0$ there exists an instance I of $\mathbf{S}$ and elements $a, b \in \text{dom}(I)$ for which $N_{d'}^I(a) \cong N_{d'}^I(b)$ and $N_d^{\mathfrak{F}(I)}(a) \not\cong N_d^{\mathfrak{F}(I)}(b)$.

For a given $d' \geq 0$ set I to be the disjoint union of a point c under predicate U and two successor relations S_1 and S_2 under predicate E of length $2d'+2$ and $2d'+4$ respectively. Choose a to be the middle point of S_1 and b the middle point of S_2. Then $N_{d'}^I(a) \cong N_{d'}^I(b)$ but $N_d^{\mathfrak{F}(I)}(a) \not\cong N_d^{\mathfrak{F}(I)}(b)$ (because $|N_d^{\mathfrak{F}(I)}(a)| < |N_d^{\mathfrak{F}(I)}(b)|$). $\square$

5.2. Local consistency under logical equivalence

We have seen that mappings that arise in the LAV setting are locally consistent, but local consistency may fail even in some simple GAV settings. To overcome this, we introduce a notion of locality based on logical equivalence (in particular, FO-equivalence) rather than isomorphism of neighborhoods, and we prove that in general, the canonical universal solution transformation $\mathfrak{F}_{\text{univ}}$ and the core transformation $\mathfrak{F}_{\text{core}}$ are locally consistent under FO-equivalence. Here we make explicit use of the results in Sections 3.1 and 3.2.

Definition 5.7 (Local consistency under FO-equivalence). *A mapping $\mathfrak{F} : \mathsf{Inst}(\mathbf{S}) \rightarrow \mathsf{Inst}(\mathbf{T})$ is* locally consistent under FO-equivalence *if for every $m, d, k \geq 0$ there exist $d', k' \geq 0$ such that, for every instance I of $\mathbf{S}$ and tuples $\bar{a}, \bar{b} \in \text{dom}(I)^m$, if $N_{d'}^I(\bar{a}) \equiv_{k'} N_{d'}^I(\bar{b})$, then*

1. *$\bar{a} \in \text{dom}(\mathfrak{F}(I))^m \;\Leftrightarrow\; \bar{b} \in \text{dom}(\mathfrak{F}(I))^m$, and*
2. *$N_d^{\mathfrak{F}(I)}(\bar{a}) \equiv_k N_d^{\mathfrak{F}(I)}(\bar{b})$.*

Then we obtain the following variation of Theorem 5.2:

Theorem 5.8 ($\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ are locally consistent under FO-equivalence). *For an arbitrary data exchange setting, both the canonical universal solution transformation $\mathfrak{F}_{\text{univ}}$ and the core transformation $\mathfrak{F}_{\text{core}}$ are locally consistent under FO-equivalence.*

Again, this result follows from two consecutive lemmas.

Lemma 5.9. *For an arbitrary data exchange setting, the canonical universal solution transformation $\mathfrak{F}_{\text{univ}}$ is locally consistent under FO-equivalence.*

Proof: Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting. We define

$$\begin{aligned} \ell_{\mathbf{S}} &= \max\{\text{qr}(\phi_{\mathbf{S}}) \mid \phi_{\mathbf{S}}(\bar{x}) \to \exists \bar{y}\psi_{\mathbf{T}}(\bar{x}, \bar{y}) \in \Sigma_{st}\} \\ m_{\mathbf{S}} &= \max\{1\} \cup \{|\bar{x}| \mid \phi_{\mathbf{S}}(\bar{x}) \to \exists \bar{y}\psi_{\mathbf{T}}(\bar{x}, \bar{y}) \in \Sigma_{st}\} \end{aligned}$$

Fix $d, k \geq 0$ and $m > 0$. Let I be an arbitrary source instance. We denote $\mathfrak{F}_{\text{univ}}(I)$ by J. For each tuple $\bar{a}$ of constants in $\text{dom}(J)^m$, there is an FO formula that describes the rank-k type of $N_d^J(\bar{a})$, that is, an FO formula $\phi_{J,\bar{a}}^{d,k}(\bar{x})$ over $\mathbf{T}$ such that, for each instance J' of $\mathbf{T}$ and tuple $\bar{b}$ of constants in $\text{dom}(J')^m$,

$$J' \models \phi_{J,\bar{a}}^{d,k}(\bar{b}) \iff N_d^J(\bar{a}) \equiv_k N_d^{J'}(\bar{b}).$$

It is worth mentioning that $\text{qr}(\phi_{J,\bar{a}}^{d,k})$ depends only on k, d and m (and not on J and $\bar{a}$).

As we now show, it follows from Theorem 3.13 that there is an FO formula $\psi_{J,\bar{a}}^{d,k}(\bar{x})$ over $\mathbf{S}$ such that, for every source instance I' and tuple $\bar{b}$ of constants in $\text{dom}(\mathfrak{F}_{\text{univ}}(I'))^m$,

$$I' \models \psi_{J,\bar{a}}^{d,k}(\bar{b}) \iff \mathfrak{F}_{\text{univ}}(I') \models \phi_{J,\bar{a}}^{d,k}(\bar{b}). \tag{10}$$

In fact, Theorem 3.13 tells us that there exists $t \geq 0$ such that, for all source instances I_1 and I_2, and tuples of constants $\bar{a} \in \text{dom}(\mathfrak{F}_{\text{univ}}(I_1))^m$ and $\bar{b} \in \text{dom}(\mathfrak{F}_{\text{univ}}(I_2))^m$, where $m \geq 0$, if $(I_1, \bar{a}) \equiv_t (I_2, \bar{b})$ then $(\mathfrak{F}_{\text{univ}}(I_1), \bar{a}) \equiv_{\text{qr}(\phi_{J,\bar{a}}^{d,k})} (\mathfrak{F}_{\text{univ}}(I_2), \bar{b})$. This tells us that $\psi_{J,\bar{a}}^{d,k}(\bar{x})$ can be chosen to be

$$\bigvee_{\{(I',b)) \mid \mathfrak{F}_{\text{univ}}(I') \models \phi_{J,\bar{a}}^{d,k}(\bar{b})\}} \tau_t^{(I',\bar{b})}(\bar{x})$$

(recall that $\tau_t^{(I',\bar{b})}(\bar{x})$ is the rank-t FO type of $(I', \bar{b})$). Note that $\text{qr}(\psi_{J,\bar{a}}^{d,k})$ depends only on k, d and Σ_{st} (and not on J and $\bar{a}$).

Also, from Theorem 4.3, there are $s, \ell \geq 0$ such that for an arbitrary source instance I' and tuples $\bar{b}_1$ and $\bar{b}_2$ in $\text{dom}(I')$ of the same length $x \leq m_{\mathbf{S}} + m$,

$$N_s^{I'}(\bar{b}_1) \equiv_\ell N_s^{I'}(\bar{b}_2) \quad \text{implies} \quad (I', \bar{b}_1) \equiv_{\text{qr}(\psi_{J,\bar{a}}^{d,k}) + \ell_{\mathbf{S}} + m_{\mathbf{S}}} (I', \bar{b}_2). \tag{11}$$

Note that s and ℓ depend only on $\text{qr}(\psi_{J,\bar{a}}^{d,k})$, Σ_{st}, and m.

Set d' and k' to be s and ℓ, resp., as defined above. We show that for each source instance instance I and tuples $\bar{a}$ and tuple $\bar{b}$ in $\text{dom}(I)^m$, if $N_{d'}^I(\bar{a}) \equiv_{k'} N_{d'}^I(\bar{b})$, then

(a) $\bar{a} \in \mathrm{dom}(\mathfrak{F}_{\text{univ}}(I))^m \Leftrightarrow \bar{b} \in \mathrm{dom}(\mathfrak{F}_{\text{univ}}(I))^m$; and

(b) $N_d^{\mathfrak{F}_{\text{univ}}(I)}(\bar{a}) \equiv_k N_d^{\mathfrak{F}_{\text{univ}}(I)}(\bar{b})$.

Let I be an arbitrary source instance. We denote $\mathfrak{F}_{\text{univ}}(I)$ by J. We prove (a) first. Let $\bar{a}$ be $(a_1, \ldots, a_m)$ and $\bar{b}$ be $(b_1, \ldots, b_m)$. We show that for each $i \in [1, m]$, if $a_i \in \mathrm{dom}(J)$ then $b_i \in \mathrm{dom}(J)$. Assume that a_i belongs to $\mathrm{dom}(J)$, for some $i \in [1, m]$. The presence of a_i in J can be identified with the instantiation

$$\phi_{\mathbf{S}}(\bar{c}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{c}, \bar{y})$$

of an std of the form $\phi_{\mathbf{S}}(\bar{x}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})$ in Σ_{st}, where there is a tuple $\bar{c}$ of constant symbols that includes a_i such that $\phi_{\mathbf{S}}(\bar{c})$ holds in the source instance I. The length of $\bar{c}$ is at most $m_{\mathbf{S}}$. Let $x_{j_1}, \ldots, x_{j_t}$ be all variables in $\bar{x}$ such that the value of x_{j_p} in $\bar{c}$, for $p \in [1, t]$, does not belong to $\bar{a}$. Let $\phi'_{\mathbf{S}}(\bar{y})$ be the formula $\exists x_{j_1}, \ldots, x_{j_t} \phi_{\mathbf{S}}(\bar{x})$. Note that $\mathrm{qr}(\phi'_{\mathbf{S}}) \leq \ell_{\mathbf{S}} + m_{\mathbf{S}}$, and $I \models \phi'_{\mathbf{S}}(\bar{a}')$ for some tuple $\bar{a}'$ of elements in $\bar{a}$ of length at most $m_{\mathbf{S}}$. Since $N_{d'}^I(\bar{a}) \equiv_{k'} N_{d'}^I(\bar{b})$, it follows from remark (11) that $(I, \bar{a}) \equiv_{\ell_{\mathbf{S}} + m_{\mathbf{S}}} (I, \bar{b})$, and thus, $I \models \phi'_{\mathbf{S}}(\bar{b}')$, where $\bar{b}'$ is the tuple of elements in $\bar{b}$ that corresponds to $\bar{a}'$ in $\bar{a}$. It is not hard to see that this implies that $b_i \in \mathrm{dom}(J)$. In the same way we can show that for each $i \in [1, m]$, if $b_i \in \mathrm{dom}(J)$ then $a_i \in \mathrm{dom}(J)$. This proves that $a_i \in \mathrm{dom}(J)$ if and only if $b_i \in \mathrm{dom}(J)$, for each $i \in [1, m]$. Hence, $\bar{a} \in \mathrm{dom}(J)^m$ if and only if $\bar{b} \in \mathrm{dom}(J)^m$.

Now we prove (b). As we mentioned,

$$N_{d'}^I(\bar{a}) \equiv_{k'} N_{d'}^I(\bar{b}) \quad \text{implies} \quad (I, \bar{a}) \equiv_{\mathrm{qr}(\psi_{J,\bar{a}}^{d,k})} (I, \bar{b}).$$

But then,

$$N_{d'}^I(\bar{a}) \equiv_{k'} N_{d'}^I(\bar{b}) \quad \text{implies} \quad \big(I \models \psi_{J,\bar{a}}^{d,k}(\bar{a}) \;\Leftrightarrow\; I \models \psi_{J,\bar{a}}^{d,k}(\bar{b})\big).$$

From remark (10), we have that

$$N_{d'}^I(\bar{a}) \equiv_{k'} N_{d'}^I(\bar{b}) \quad \text{implies} \quad \big(J \models \phi_{J,\bar{a}}^{d,k}(\bar{a}) \;\Leftrightarrow\; J \models \phi_{J,\bar{a}}^{d,k}(\bar{b})\big),$$

but since it is always the case that $J \models \phi_{J,\bar{a}}^{d,k}(\bar{a})$, we also have that

$$N_{d'}^I(\bar{a}) \equiv_{k'} N_{d'}^I(\bar{b}) \quad \text{implies} \quad J \models \phi_{J,\bar{a}}^{d,k}(\bar{b}).$$

We conclude that

$$N_{d'}^I(\bar{a}) \equiv_{k'} N_{d'}^I(\bar{b}) \quad \text{implies} \quad N_d^J(\bar{a}) \equiv_k N_d^J(\bar{b}),$$

which was to be shown. This concludes the proof of the lemma. □

Lemma 5.10. *For an arbitrary data exchange setting, the core transformation $\mathfrak{F}_{\text{core}}$ is locally consistent under FO-equivalence.*

Proof: By Lemma 5.9, the canonical universal solution transformation $\mathfrak{F}_{\text{univ}}$ of a data exchange setting is locally consistent under FO-equivalence. We shall show that the mapping that maps the canonical universal solution onto the core is also locally consistent under FO-equivalence. Since the composition of locally consistent transformations under FO-equivalence is also locally consistent under FO-equivalence, and constants are preserved from canonical universal solutions to their cores, this is enough to prove the lemma.

Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting, and fix $m, d, k \geq 0$. Let I be an arbitrary source instance. We denote $\mathfrak{F}_{\text{core}}(I)$ by J. For each tuple $\bar{a}$ of constants in $\mathrm{dom}(J)^m$, let $\phi_{J,\bar{a}}^{d,k}(\bar{a})$ be the FO formula that describes the rank-k type of $N_d^J(\bar{a})$. Recall that this formula has the following property: For every tuple $\bar{b}$ of constants in $\mathrm{dom}(J)^m$,

$$J \models \phi^{d,k}_{J,\bar{a}}(\bar{b}) \iff N_d^J(\bar{a}) \equiv_k N_d^J(\bar{b}).$$

It is worth mentioning that $\mathrm{qr}(\phi^{d,k}_{J,\bar{a}})$ depends only on d and k (and not on J and $\bar{a}$). From Lemma 3.7, there is an FO formula $\psi^{d,k}_{J,\bar{a}}(\bar{x})$ over $\mathbf{T}$ such that, for every tuple $\bar{b}$ of constants in $\mathrm{dom}(\mathfrak{F}_{\mathrm{univ}}(I))^m$,

$$\mathfrak{F}_{\mathrm{univ}}(I) \models \psi^{d,k}_{J,\bar{a}}(\bar{b}) \iff J \models \phi^{d,k}_{J,\bar{a}}(\bar{b}).$$

Note that $\mathrm{qr}(\psi^{d,k}_{J,\bar{a}})$ depends only on k, d and Σ_{st} (and not on J and $\bar{a}$). Also, from Theorem 4.3, there are $s, \ell \geq 0$ such that for every tuples $\bar{a}$ and $\bar{b}$ of constants in $\mathrm{dom}(\mathfrak{F}_{\mathrm{univ}}(I))^m$,

$$N_s^{\mathfrak{F}_{\mathrm{univ}}(I)}(\bar{a}) \equiv_\ell N_s^{\mathfrak{F}_{\mathrm{univ}}(I)}(\bar{b}) \quad \text{implies} \quad \big(\mathfrak{F}_{\mathrm{univ}}(I) \models \psi^{d,k}_{J,\bar{a}}(\bar{a}) \Leftrightarrow \mathfrak{F}_{\mathrm{univ}}(I) \models \psi^{d,k}_{J,\bar{a}}(\bar{b})\big).$$

Note that s and ℓ depend only on $\mathrm{qr}(\psi^{r,k}_{J,\bar{a}})$, Σ_{st}, and m.

Let $d' = s$ and $k' = \ell$. Then

$$N_{d'}^{\mathfrak{F}_{\mathrm{univ}}(I)}(\bar{a}) \equiv_{k'} N_{d'}^{\mathfrak{F}_{\mathrm{univ}}(I)}(\bar{b}) \quad \text{implies} \; \big(J \models \phi^{d,k}_{J,\bar{a}}(\bar{a}) \Leftrightarrow J \models \phi^{d,k}_{J,\bar{a}}(\bar{b})\big).$$

Since it is always the case that $J \models \phi^{d,k}_{J,\bar{a}}(\bar{a})$, we obtain that

$$N_{d'}^{\mathfrak{F}_{\mathrm{univ}}(I)}(\bar{a}) \equiv_{k'} N_{d'}^{\mathfrak{F}_{\mathrm{univ}}(I)}(\bar{b}) \quad \text{implies} \quad J \models \phi^{d,k}_{J,\bar{a}}(\bar{b}),$$

and, thus,

$$N_{d'}^{\mathfrak{F}_{\mathrm{univ}}(I)}(\bar{a}) \equiv_{k'} N_{d'}^{\mathfrak{F}_{\mathrm{univ}}(I)}(\bar{b}) \quad \text{implies} \quad N_d^J(\bar{a}) \equiv_k N_d^J(\bar{b}),$$

which was to be shown. □

We conclude from Theorem 5.8 and Proposition 5.6 that local consistency under FO-equivalence properly extends local consistency for transformations $\mathfrak{F}_{\mathrm{univ}}$ and $\mathfrak{F}_{\mathrm{core}}$. We do not know whether this holds for arbitrary transformations.

5.3. Rewritable queries over locally consistent transformations

Here we show that queries that are rewritable over a transformation $\mathfrak{F}$ that is locally consistent (under FO-equivalence) are also locally source-dependent in Gaifman-sense. This strengthens the result in Corollary 4.8, as it implies that neither the canonical universal solution, nor the core, nor *any* other generated solution that has the property of being locally consistent (under FO-equivalence) supports rewriting for arbitrary first-order queries.

Theorem 5.11. *Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting, and Q a query over $\mathbf{T}$. Assume that Q is rewritable over a transformation $\mathfrak{F} : \mathsf{Inst}(\mathbf{S}) \to \mathsf{Inst}(\mathbf{T})$ under $\mathcal{M}$, where $\mathfrak{F}$ is either locally consistent, or locally consistent under FO-equivalence. Then Q is locally source-dependent in Gaifman-sense under $\mathcal{M}$.*

Proof: We only prove the theorem for the case when $\mathfrak{F}$ is locally consistent under FO-equivalence. The proof for the other case, that is, when $\mathfrak{F}$ is locally consistent, is analogous. Let Q' be a first-order rewriting of Q over $\mathfrak{F}$, that is, an m-ary query over $\mathbf{T}$ specified in FO such that for every instance I of $\mathbf{S}$, we have $\underline{\mathrm{certain}}_{\mathcal{M}}(Q, I) = Q'(\mathfrak{F}(I))$. By Theorem 4.3, there exist $d, k \geq 0$ such that for every instance J of $\mathbf{T}$ and tuples $\bar{a}, \bar{b}$ in $\mathrm{dom}(J)^m$, if $N_d^J(\bar{a}) \equiv_k N_d^J(\bar{b})$, then $\bar{a} \in Q'(J)$ if and only if $\bar{b} \in Q'(J)$. Given that $\mathfrak{F}$ is locally consistent under FO-equivalence, there exist $d', k' \geq 0$ such that for every instance I of $\mathbf{S}$ and tuples $\bar{a}$, $\bar{b}$ in $\mathrm{dom}(I)^m$, if $N_{d'}^I(\bar{a}) \equiv_{k'} N_{d'}^I(\bar{b})$, then

1. $\bar{a} \in \mathrm{dom}(\mathfrak{F}(I))^m \;\Leftrightarrow\; \bar{b} \in \mathrm{dom}(\mathfrak{F}(I))^m$, and
2. $N_d^{\mathfrak{F}(I)}(\bar{a}) \equiv_k N_d^{\mathfrak{F}(I)}(\bar{b})$.

From this we conclude that Q is locally source-dependent in Gaifman-sense, since for every instance I of $\mathbf{S}$ and tuples $\bar{a}$, $\bar{b}$ in $\mathrm{dom}(I)^m$,

$$\begin{aligned} N_{d'}^{I}(\bar{a}) \equiv_{k'} N_{d'}^{I}(\bar{b}) \;&\Rightarrow\; N_d^{\mathfrak{F}(I)}(\bar{a}) \equiv_k N_d^{\mathfrak{F}(I)}(\bar{b}) \\ &\Rightarrow\; (\bar{a} \in Q'(\mathfrak{F}(I)) \;\Leftrightarrow\; \bar{b} \in Q'(\mathfrak{F}(I))) \\ &\Rightarrow\; (\bar{a} \in \underline{\mathrm{certain}}_{\mathcal{M}}(Q, I) \;\Leftrightarrow\; \bar{b} \in \underline{\mathrm{certain}}_{\mathcal{M}}(Q, I)). \end{aligned}$$

This concludes the proof of the theorem. □

Thus, the unary query Q in the proof of Corollary 4.8 does not admit a rewriting over any data exchange solution that is locally consistent (under FO-equivalence).

5.4. Target dependencies

Several papers have considered an extension of the data exchange setting in which dependencies exist for the target schema as well [FKMP05, HLS11]. A solution is then required to satisfy those *target* dependencies. Based on familiar classes of dependencies (cf. [BV84]), we define extensions of the data exchange setting with *tuple-generating dependencies* (tgds) as well as *equality-generating dependencies* (egds). The tgds over $\mathbf{T}$ are of the form

$$\forall \bar{x}(\varphi_{\mathbf{T}}(\bar{x}) \rightarrow \exists \bar{y}\, \psi_{\mathbf{T}}(\bar{x}, \bar{y})),$$

where $\varphi_{\mathbf{T}}(\bar{x})$ and $\psi_{\mathbf{T}}(\bar{x}, \bar{y})$ are conjunctions of FO atomic formulae. (There is also a safety condition that every x in $\bar{x}$ that actually appears in $\psi_{\mathbf{T}}(\bar{x}, \bar{y})$ also actually appears in $\varphi_{\mathbf{T}}(\bar{x})$.)

The egds over $\mathbf{T}$ are of the form

$$\forall \bar{x}(\varphi_{\mathbf{T}}(\bar{x}) \rightarrow (x_1 = x_2)),$$

where $\varphi_{\mathbf{T}}(\bar{x})$ is a conjunction of atomic FO formulae, with free variables $\bar{x}$, and x_1, x_2 are in $\bar{x}$. If, furthermore, the data exchange setting is restricted to LAV or GAV, we shall speak of LAV+tgd settings, LAV+egd settings, etc.

A target instance J is a solution for a source instance I under a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st}, \Sigma_t)$, with stds Σ_{st} and target dependencies Σ_t, if (I, J) satisfies every std in Σ_{st}, and, in addition, J satisfies all target dependencies in Σ_t. As before, a solution for I under $\mathcal{M}$ is universal if it can be homomorphically mapped into every other solution. As opposed to the case without target dependencies, it has been noted [FKMP05] that (universal) solutions do not always exist for source instances in the presence of target dependencies. However, when solutions exist a particular canonical universal solution can be constructed by using the chase (for a definition see [FKMP05]). Furthermore, the core of such canonical universal solution is also a universal solution [FKP05]. We then define transformations $\mathfrak{F}_{\mathrm{univ}}$ and $\mathfrak{F}_{\mathrm{core}}$ as partial mappings that are defined only in those source instances for which a canonical universal solution exists. Furthermore, we say that these transformations are locally consistent (under FO-equivalence) if they are locally consistent (under FO-equivalence) when restricted to the source instances for which the transformation is well-defined.

The next proposition cover the four settings of LAV+tgd, GAV+tgd, LAV+egd, and GAV+egd. In fact, it is not hard to see that the results in the next proposition give us an answer about local consistency for all possible choices of LAV (resp., GAV) with tgds and/or egds. For example, GAV+egd+tgd is not necessarily locally consistent, since GAV+tgd is not necessarily locally consistent.

Proposition 5.12.

(a) The transformations $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ of LAV+tgd (or GAV+tgd) settings are not necessarily locally consistent (under FO-equivalence), even if there is only one target dependency.

(b) The transformations $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ of GAV+egd settings are locally consistent (under FO-equivalence).

(c) The transformations $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ of LAV+egd settings are not necessarily locally consistent (under FO-equivalence), even if all of the target dependencies are key dependencies.

Proof: (a) Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st}, \Sigma_t)$ be a LAV+tgd (GAV+tgd) setting defined as follows: $\mathbf{S} = \{S(\cdot,\cdot), M(\cdot)\}$, $\mathbf{T} = \{T(\cdot,\cdot), N(\cdot)\}$, $\Sigma_{st} = \{S(x,y) \to T(x,y),\ M(x) \to N(x)\}$ and $\Sigma_t = \{T(x,y) \wedge T(y,z) \to T(x,z)\}$. Let $m = 1$, $d = 1$ and $k = 1$. We will show that for these values there is no $d', k' \geq 0$ such that for every instance I of $\mathbf{S}$ and for every $a, b \in \text{dom}(I) \cap \text{dom}(\mathfrak{F}_{\text{univ}}(I))$, if $N^I_{d'}(a) \equiv_{k'} N^I_{d'}(b)$ then $N^{\mathfrak{F}_{\text{univ}}(I)}_d(a) \equiv_k N^{\mathfrak{F}_{\text{univ}}(I)}_d(b)$ (resp., if $N^I_{d'}(a) \cong N^I_{d'}(b)$) then $N^{\mathfrak{F}_{\text{univ}}(I)}_d(a) \cong N^{\mathfrak{F}_{\text{univ}}(I)}_d(b)$). On the contrary, assume that such d', k' exist and let I be an instance of $\mathbf{S}$ defined as the disjoint union of two successor relations of length $d' + 1$:

$$I(S) \;=\; \{(a_i, a_{i+1}) \mid 1 \leq i \leq d'\} \;\cup\; \{(b_i, b_{i+1}) \mid 1 \leq i \leq d'\}.$$

Furthermore, assume that $I(M) = \{a_{d'+1}\}$. Then $N^I_{d'}(a_1) \cong N^I_{d'}(b_1)$ and $N^I_{d'}(a_1) \equiv_{k'} N^I_{d'}(b_1)$. In this case the predicate T in $\mathfrak{F}_{\text{univ}}(I)$ is composed by the disjoint union of two linear orders:

$$\mathfrak{F}_{\text{univ}}(I)(T) \;=\; \{(a_i, a_j) \mid 1 \leq i < j \leq d' + 1\} \;\cup\; \{(b_i, b_j) \mid 1 \leq i < j \leq d' + 1\}.$$

Thus, $N^{\mathfrak{F}_{\text{univ}}(I)}_d(a_1) \not\equiv_k N^{\mathfrak{F}_{\text{univ}}(I)}_d(b_1)$ since $\mathfrak{F}_{\text{univ}}(I)(N) = \{a_{d'+1}\}$ and $a_{d'+1}$ is at distance 1 from a_1. For the same reason, $N^{\mathfrak{F}_{\text{univ}}(I)}_d(a_1) \not\cong N^{\mathfrak{F}_{\text{univ}}(I)}_d(b_1)$.

We note that the previous proof also shows that the transformation $\mathfrak{F}_{\text{core}}$ of LAV+tgd (GAV+tgd) settings is not necessarily locally consistent (under FO-equivalence), as in the setting shown above $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ coincide.

(b) Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st}, \Sigma_t)$ be a GAV+egd setting, where Σ_t is a set of equality generating dependencies over $\mathbf{T}$, and I an instance of $\mathbf{S}$. If I has a canonical universal solution J, then J is also a canonical universal solution of I in the GAV setting $\mathcal{M}' = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$. But $\mathfrak{F}^{\mathcal{M}'}_{\text{univ}}$ is locally consistent (under FO-equivalence) by Corollary 5.9, which proves that $\mathfrak{F}^{\mathcal{M}}_{\text{univ}}$ is also locally consistent (under FO-equivalence), since egds have no effect on constants.

Given that the transformations $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ of GAV+egd settings coincide, the previous proof also shows that the transformation $\mathfrak{F}_{\text{core}}$ of GAV+egd settings is locally consistent (under FO-equivalence).

(c) Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st}, \Sigma_t)$ be a LAV+egd setting, where $\mathbf{S} = \{E(\cdot,\cdot),\ V(\cdot)\}$, $\mathbf{T} = \{E'(\cdot,\cdot),\ V'(\cdot),\ R_1(\cdot,\cdot),\ R_2(\cdot,\cdot)\}$, Σ_{st} contains the following source-to-target dependencies:

$$\begin{aligned} E(x,y) &\;\to\; E'(x,y), \\ V(x) &\;\to\; V'(x), \\ E(x,y) &\;\to\; \exists u_1 \exists u_2 \exists u_3 (R_1(x,u_1) \wedge R_1(y,u_2) \wedge R_2(u_1,u_3) \wedge R_2(u_2,u_3)), \end{aligned}$$

and Σ_t contains the following key dependencies:

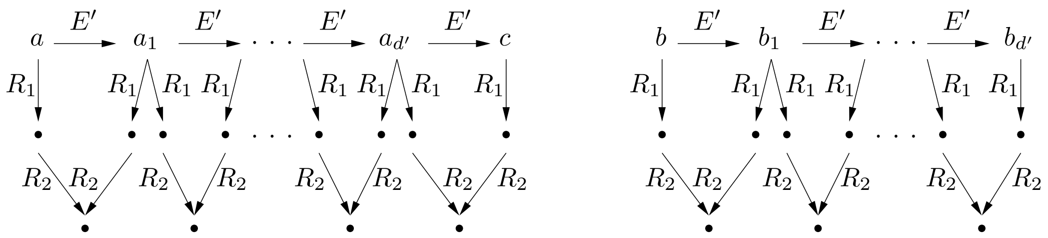
$$\begin{aligned} R_1(x,y) \wedge R_1(x,z) &\;\to\; y = z, \\ R_2(x,y) \wedge R_2(x,z) &\;\to\; y = z, \\ R_2(y,x) \wedge R_2(z,x) &\;\to\; y = z. \end{aligned}$$

Let $m = 1$, $d = 2$ and $k \geq 1$. We will show that for these values there is no $d', k' \geq 0$ such that for every instance I of $\mathbf{S}$ and for every $a, b \in \mathrm{dom}(I) \cap \mathrm{dom}(\mathfrak{F}_{\mathrm{univ}}(I))$, if $N^I_{d'}(a) \equiv_{k'} N^I_{d'}(b)$ then $N^{\mathfrak{F}_{\mathrm{univ}}(I)}_d(a) \equiv_k N^{\mathfrak{F}_{\mathrm{univ}}(I)}_d(b)$ (resp., if $N^I_{d'}(a) \cong N^I_{d'}(b)$ then $N^{\mathfrak{F}_{\mathrm{univ}}(I)}_d(a) \cong N^{\mathfrak{F}_{\mathrm{univ}}(I)}_d(b)$). On the contrary, assume that such d', k' exist. Define a database instance I with domain $\{a, a_1, \ldots, a_{d'}, b, b_1, \ldots, b_{d'}, c\}$ as follows: $I(V) = \{c\}$ and $I(E)$ contains the following tuples:

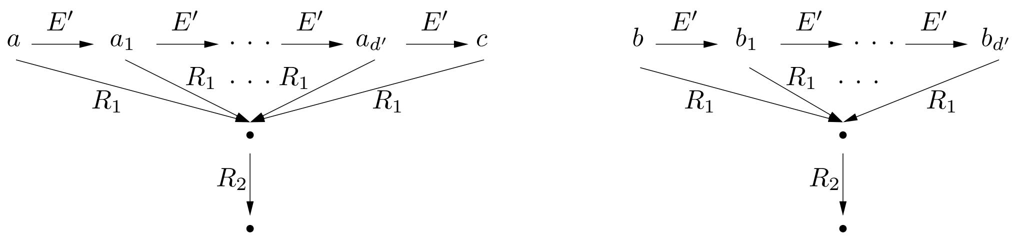
$$a \xrightarrow{E} a_1 \xrightarrow{E} \cdots \xrightarrow{E} a_{d'} \xrightarrow{E} c \qquad\qquad b \xrightarrow{E} b_1 \xrightarrow{E} \cdots \xrightarrow{E} b_{d'}$$

As shown in the figure, $I(E)$ is a union of two paths, one containing $d' + 2$ elements with first element a and last element c and another path containing $d' + 1$ elements with first element b. Observe that $N^I_{d'}(a) \cong N^I_{d'}(b)$ and $N^I_{d'}(a) \equiv_{k'} N^I_{d'}(b)$.

The canonical universal solution $\mathfrak{F}_{\mathrm{univ}}(I)$ of I can be constructed by first applying the set of source-to-target dependencies Σ_{st}:



(where each element in $\mathfrak{F}_{\mathrm{univ}}(I)$ that is not in $\mathrm{dom}(I)$ is a fresh null value), and then applying the set of key dependencies Σ_t:



In the figures shown above, the symbol $\bullet$ is used to represent null values. Observe that predicate V' in $(\mathfrak{F}_{\mathrm{univ}}(I))$ only contains the element c, since the only source-to-target dependency mentioning predicate V' is $V(x) \rightarrow V'(x)$. Thus, $N^{\mathfrak{F}_{\mathrm{univ}}(I)}_d(a) \not\equiv_k N^{\mathfrak{F}_{\mathrm{univ}}(I)}_d(b)$, since the distance between a and c is at most 2 and there is no a point c' in $N^{\mathfrak{F}_{\mathrm{univ}}(I)}_d(b)$ such that $V'(c')$ holds. For the same reason, $N^{\mathfrak{F}_{\mathrm{univ}}(I)}_d(a) \not\cong N^{\mathfrak{F}_{\mathrm{univ}}(I)}_d(b)$

We note that the previous proof also shows that the transformation $\mathfrak{F}_{\mathrm{core}}$ of LAV+egd settings is not necessarily locally consistent (under FO-equivalence) since for the instance I shown above, $\mathfrak{F}_{\mathrm{univ}}(I) = \mathfrak{F}_{\mathrm{core}}(I)$. $\square$

Thus, in most of the cases the transformations $\mathfrak{F}_{\mathrm{univ}}$ and $\mathfrak{F}_{\mathrm{core}}$ in the presence of target dependencies are not locally consistent (under FO-equivalence).

6. Universal Solutions Semantics

In this section, we show that many of our results extend to an alternative semantics that is based completely on the preferred solutions in data exchange: the universal solutions. This semantics was proposed [FKP05]. One motivation for it is that universal solutions have a special status in data exchange, and it is thus natural to consider them when defining the semantics of answers. A different motivation is that sometimes the usual semantics behaves in an unexpected way. For instance, the proposition below shows the following observation about the certain answers semantics. Suppose we have a Boolean query Q, and at least for one instance the certain answer to Q is true. Then, to compute the certain answer to $\neg Q$, one does not need to look at the data at all: it will always be false. We shall now state this precisely, and then demonstrate in Example 6.3 that the universal solutions semantics does not exhibit such a behavior.

Proposition 6.1. *Let $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ be a data exchange setting. Then for every Boolean query Q over $\mathbf{T}$, either $\underline{\text{certain}}_{\mathcal{M}}(Q, I) = \texttt{false}$ for all instances I of $\mathbf{S}$, or $\underline{\text{certain}}_{\mathcal{M}}(\neg Q, I) = \texttt{false}$ for all instances I of $\mathbf{S}$.*

Proof: Let Q be a Boolean query over $\mathbf{T}$, and assume that there exists an instance I_0 of $\mathbf{S}$ such that $\underline{\text{certain}}_{\mathcal{M}}(Q, I_0) = \texttt{true}$. Then we show that for every instance I of $\mathbf{S}$, $\underline{\text{certain}}_{\mathcal{M}}(\neg Q, I) = \texttt{false}$.

Let I be an instance of $\mathbf{S}$ and J a solution for I. Then given a solution J_0 for I_0, the instance J' defined as $J'(R) = J(R) \cup J_0(R)$, for every $R \in \mathbf{T}$, is a solution for both I and I_0. Since $\underline{\text{certain}}_{\mathcal{M}}(Q, I_0) = \texttt{true}$, $Q(J')$ is true and, therefore, there is a solution of I not satisfying $\neg Q$. We conclude that $\underline{\text{certain}}_{\mathcal{M}}(\neg Q, I) = \texttt{false}$. □

Let us take a closer look now to the concept of the certain answers of a query Q. It has been argued [FKMP05] that the universal solutions should be the preferred solutions to the data exchange problem, since in a precise sense they are the most general solutions and, thus, they represent the space of all solutions. This suggests that, in the context of data exchange, the notion of certain answers on universal solutions may be more fundamental and more meaningful than that of the certain answers. Thus, it is proposed in [FKP05] that the *certain answers on universal solutions* semantics, as defined next, should be used for query answering in data exchange, instead of $\underline{\text{certain}}(Q, I)$.

Definition 6.2 (Certain answers on universal solutions). *Given a data exchange setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$, an m-ary query Q over $\mathbf{T}$, and a source instance I, we define the* certain answers on universal solutions semantics *of Q under $\mathcal{M}$ as*

$$\underline{\text{u-certain}}_{\mathcal{M}}(Q, I) \quad = \bigcap_{J \text{ is a universal solution for } I} Q(J).$$

Clearly, $\underline{\text{certain}}_{\mathcal{M}}(Q, I) \subseteq \underline{\text{u-certain}}_{\mathcal{M}}(Q, I)$. The next example shows that the universal solution semantics avoids the problem shown in the previous proposition, that is, there exists a Boolean query Q such that $\underline{\text{u-certain}}_{\mathcal{M}}(Q, I_1) = \texttt{true}$ and $\underline{\text{u-certain}}_{\mathcal{M}}(\neg Q, I_2) = \texttt{true}$, for some instances I_1 and I_2.

Example 6.3. Given a copying data exchange setting with $\mathbf{S} = \{P(\cdot),\, R(\cdot)\}$, $\mathbf{T} = \{P'(\cdot),\, R'(\cdot)\}$ and $\Sigma_{st} = \{P(x) \to P'(x),\, R(x) \to R'(x)\}$, let Q be a Boolean query over $\mathbf{T}$ defined as $\exists x\, (P'(x) \wedge R'(x))$.

Define instances I_1, I_2 of $\mathbf{S}$ as $\{P(a), R(a)\}$ and $\{P(a), R(b)\}$, respectively. Then both $\underline{\text{u-certain}}(Q, I_1)$ and $\underline{\text{u-certain}}(\neg Q, I_2)$ are *true* (if J is a universal solution for I_2, then there is a homomorphism

$h : J \to \mathfrak{F}_{\text{univ}}(I_2) = \{P'(a), R'(b)\}$ and, hence, for every null value n in J it could not be the case that $P'(n)$ and $R'(n)$ are in J). $\square$

Given a mapping $\mathfrak{F} : \mathsf{Inst}(\mathbf{S}) \to \mathsf{Inst}(\mathbf{T})$, we say that the m-ary query Q over schema $\mathbf{T}$ is *rewritable over* $\mathfrak{F}$ *under the universal solutions semantics*, if there exists an m-ary query Q' specified in FO such that

$$\underline{\text{u-certain}}_{\mathcal{M}}(Q, I) \;=\; Q'(\mathfrak{F}(I)), \text{ for every instance } I \text{ of } \mathbf{S}.$$

While rewritings under the certain answer semantics may not exist even for queries in the class of conjunctive queries with inequalities, rewritings under the universal solutions semantics exist for a big class of queries that includes the latter. Indeed, every *existential* m-ary query Q, that is, a domain-independent query of the form $\exists \bar{y}\, \phi(\bar{x}, \bar{y})$, where $\bar{x} = (x_1, \ldots, x_m)$ is the tuple of free variables of Q and ϕ is a Boolean combination of atomic formulae, is rewritable over the *core* under the universal solution semantics by the formula $Q \wedge \bigwedge_{i \in [1,m]} C(x_i)$ [FKP05]. This can be seen as an advantage of the universal solution semantics against the certain answers semantics.

We say that Q is *locally source-dependent under the universal solution semantics in Gaifman-sense*, if there is $d \geq 0$ such that for every instance I of $\mathbf{S}$ and every $\bar{a}, \bar{b} \in \text{dom}(I)^m$, whenever $N_d^I(\bar{a}) \cong N_d^I(\bar{b})$ we have that

$$\big(\bar{a} \in \underline{\text{u-certain}}_{\mathcal{M}}(Q, I) \;\;\Leftrightarrow\;\; \bar{b} \in \underline{\text{u-certain}}_{\mathcal{M}}(Q, I)\big).$$

Also, we say that Q is *locally source-dependent under the universal solution semantics in Hanf-sense*, if there is $d \geq 0$ such that for every instances I_1 and I_2 of $\mathbf{S}$ and every $\bar{a} \in \text{dom}(I_1)^m$ and $\bar{b} \in \text{dom}(I_2)^m$, whenever $(I_1, \bar{a}) \leftrightarrows_d (I_2, \bar{b})$ we have that

$$\big(\bar{a} \in \underline{\text{u-certain}}_{\mathcal{M}}(Q, I) \;\;\Leftrightarrow\;\; \bar{b} \in \underline{\text{u-certain}}_{\mathcal{M}}(Q, I)\big).$$

We now show that the main results of Sections 3 and 4 are preserved when one considers the new semantics. First, Theorems 3.5 and 3.11 extend to the universal solutions semantics. That is,

Theorem 6.4. *Every query that is rewritable over the core under the universal solution semantics is also rewritable over the canonical universal solution under the universal solution semantics. Furthermore, every query that is rewritable over the canonical universal solution under the universal solution semantics is also rewritable over the source under the universal solution semantics*

Proof: Exactly the same proofs of Theorems 3.5 and 3.11 apply, as they do not rely on the underlying semantics. $\square$

We can also prove the following refinement of Theorem 4.5.

Theorem 6.5. *Let* $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$ *be a data exchange setting. Every query over* $\mathbf{T}$ *that is rewritable over the canonical universal solution, or over the core, under the universal solutions semantics, is locally source-dependent under the universal solutions semantics in both Hanf- and Gaifman-sense.*

Proof: By mimicking the proof of Theorem 4.5 in the context of the universal solution semantics. $\square$

Thus, Theorem 6.5 can be used as a tool for proving non-rewritability under the new semantics introduced in this section.

It is then natural to ask what the relationship between the notions of rewritability under the two semantics is. We now show that the two are incompatible.

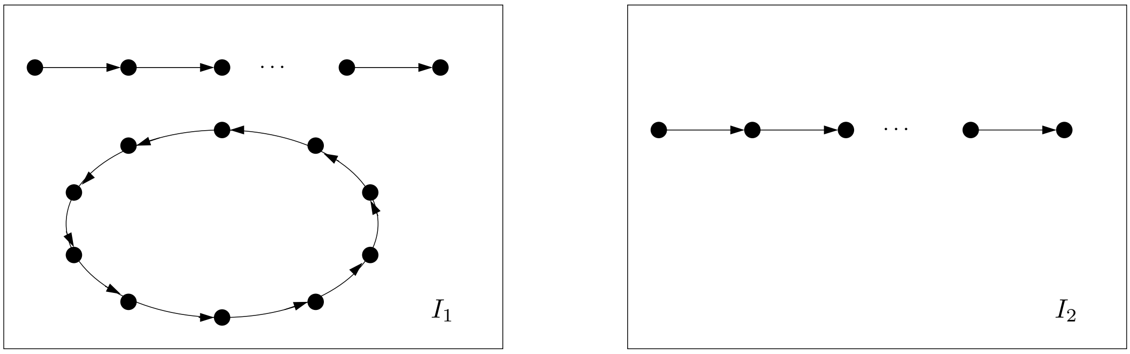


Figure 3: Instances I_1 and I_2 of Theorem 6.6.

Theorem 6.6. *Let $\mathfrak{F}$ be either $\mathfrak{F}_{\text{univ}}$ or $\mathfrak{F}_{\text{core}}$.*

1. *There is an FO-query Q that is rewritable over $\mathfrak{F}$ under the usual semantics, but is not rewritable over $\mathfrak{F}$ under the universal solutions semantics.*
2. *There is an FO-query Q that is rewritable over $\mathfrak{F}$ under the universal solutions semantics, but is not rewritable over $\mathfrak{F}$ under the usual semantics.*

Proof: We first prove part 1. Consider the LAV (and GAV) setting $\mathcal{M} = (\mathbf{S}, \mathbf{T}, \Sigma_{st})$, where $\mathbf{S} = \{E\}$, $\mathbf{T} = \{E', D\}$ and Σ_{st} consists of the following source-to-target dependencies

$$E(x, y) \to E'(x, y), \qquad E(x, y) \to D(x), \qquad E(x, y) \to D(y).$$

Furthermore, define a Boolean query Q as

$$\exists x \exists y (E'(x, y) \wedge \neg D(x) \wedge \neg D(y)) \;\; \to \;\; \exists u \exists v (E'(u, v) \wedge \neg D(u) \wedge \neg D(v) \;\wedge\; \forall z (E'(z, u) \;\to\; D(z))).$$

We will prove that Q is not rewritable over $\mathfrak{F}$, where $\mathfrak{F}$ is $\mathfrak{F}_{\text{univ}}$ or $\mathfrak{F}_{\text{core}}$, under the universal solution semantics. We concentrate on $\mathfrak{F}_{\text{univ}}$ since for the data exchange setting $\mathcal{M}$ presented above the transformations $\mathfrak{F}_{\text{univ}}$ and $\mathfrak{F}_{\text{core}}$ coincide.

In view of Theorem 6.5, to prove that Q is not rewritable over $\mathfrak{F}$ under the universal solution semantics it is enough to show that it is not locally source-dependent, under the universal solution semantics, in Hanf-sense. That is, we need to exhibit for every $d \geq 0$, two instances I_1 and I_2 such that $I_1 \leftrightarrows_d I_2$, $\underline{\text{u-certain}}_{\mathcal{M}}(Q, I_1) = \texttt{false}$, but $\underline{\text{u-certain}}_{\mathcal{M}}(Q, I_2) = \texttt{true}$. Let I_1 and I_2 be two source instances such that I_1 is a disjoint union of a directed cycle of length $2d+2$ and a successor relation of length $2d+2$, and I_2 is a successor relation of length $4d+4$ (see Figure. 3). It is not hard to see that $I_1 \leftrightarrows_d I_2$.

We have to prove now that $\underline{\text{u-certain}}_{\mathcal{M}}(Q, I_1) = \texttt{false}$ but $\underline{\text{u-certain}}_{\mathcal{M}}(Q, I_2) = \texttt{true}$. First we show that $\underline{\text{u-certain}}_{\mathcal{M}}(Q, I_1) = \texttt{false}$. Let J_1 be the canonical universal solution for I_1. Consider an instance J_1' such that $J_1(D) = J_1'(D)$ and $J_1'(E')$ is equal to $J_1(E')$ plus a (directed) cycle of null values with the same cardinality as the cycle in J_1. Then J_1' is a universal solution for I_1 since the function sending each null value in J_1' to a different element in the cycle of the constants (and that maintains adjacency) is a homomorphism from J_1' to J_1. Moreover, given that all the null values in J_1' (that is, all the elements that do not belong to the interpretation of predicate D) are in the cycle, it is not hard to see that the antecedent of Q is true but the consequent is false, and thus, $Q(J_1') = \texttt{false}$ and $\underline{\text{u-certain}}_{\mathcal{M}}(Q, I_1) = \texttt{false}$.

Next, we prove that $\underline{\text{u-certain}}_{\mathcal{M}}(Q, I_2) = \texttt{true}$. Assume on the contrary that $\underline{\text{u-certain}}_{\mathcal{M}}(Q, I_2) = \texttt{false}$. Hence, for some universal solution J_2' for I_2 it is the case that

$$J_2' \not\models \exists x \exists y (E'(x,y) \wedge \neg D(x) \wedge \neg D(y)) \;\rightarrow\; \exists u \exists v (E'(u,v) \wedge \neg D(u) \wedge \neg D(v) \wedge \forall z (E'(z,u) \rightarrow D(z))).$$

This implies that J_2' contains at least two elements a and b in the relation E' such that none of them belongs to the relation D, and

$$J_2' \models \forall u \forall v \big(\neg(\neg D(u) \wedge E'(u,v) \wedge \neg D(v)) \vee \exists z (E'(z,u) \wedge \neg D(z)) \big).$$

Since all constants in the canonical universal solution J_2 for I_2 belong to D, and each universal solution J for I_2 is an extension of I_2 such that each element that belongs to J but not to I_2 is a null value, we have that a and b are nulls.

Furthermore, since

$$J_2' \models \forall u \forall v \big(\neg(\neg D(u) \wedge E'(u,v) \wedge \neg D(v)) \vee \exists z (E'(z,u) \wedge \neg D(z)) \big),$$

we have that

$$J_2' \models \forall u \big((\neg D(u) \wedge \exists v (E'(u,v) \wedge \neg D(v))) \rightarrow \exists z (E'(z,u) \wedge \neg D(z)) \big).$$

The latter implies that there is a non-terminating backward chain in J_2' of elements not labeled in D (nulls) starting from a. But J_2' is finite, and hence it contains at least one cycle of null values. This shows that there is no homomorphism from J_2' to J_2, contradicting that J_2' is a universal solution. We conclude that Q is not rewritable over $\mathfrak{F}$ under the universal solutions semantics.

At the same time, it is not hard to see that under the usual semantics, $\underline{\text{certain}}_{\mathcal{M}}(Q, I) = \texttt{false}$ for every source instance I. Therefore, Q is rewritable over $\mathfrak{F}$ under the usual semantics.

Now we prove part 2. It is known [FKMP05] that there is a conjunctive query Q with one inequality that is not rewritable over $\mathfrak{F}$ under the usual semantics. But we mentioned that existential queries are rewritable over the core under the universal solution semantics and, hence, Q is rewritable over the core under this semantics. Thus, from Theorem 6.4 we conclude that Q is also rewritable over the canonical universal solution under the universal solution semantics. □

7. Conclusions and Future Work

We focused on query rewriting in data exchange, and showed how the two most studied data exchange solutions to date (the canonical universal solution and the core) compare in terms of query rewriting. From this, we developed an easy tool for proving non-existence of query rewritings over both the core and the canonical universal solution. Unlike isolated results on rewriting that exist in the literature, our results give easily applicable tools for studying the query rewriting problem.

Since the conference version of this paper was published [ABFL04], the issue of anomalous behavior of query answering semantics and of alternatives semantics in data exchange has received much attention, and different notions of query answering semantics have been proposed [LS11, HLS11, AK08]. It would be interesting to see how these newer results relate to the results presented here. It might be also interesting to study whether Theorem 3.5 still holds in the presence of target dependencies. The proof given here, based on Claim 3.8, does not work when target dependencies are allowed, as in such case the size of the block of a null in the canonical universal solution is not necessarily bounded.

References

[ABC99] Marcelo Arenas, Leopoldo Bertossi, and Jan Chomicki. Consistent query answers in inconsistent databases. In *Proceedings of the 18th ACM Symposium on Principles of Database Systems, PODS'99*, pages 68–79, 1999.

[ABFL04] Marcelo Arenas, Pablo Barceló, Ronald Fagin, and Leonid Libkin. Locally consistent transformations and query answering in data exchange. In *Proceedings of the 23rd ACM Symposium on Principles of Database Systems, PODS'04*, pages 229–240, 2004.

[ABLM10] Marcelo Arenas, Pablo Barceló, Leonid Libkin, and Filip Murlak. *Relational and XML Data Exchange*. Morgan&Claypool Publishers, 2010.

[AK08] Foto Afrati and Phokion Kolaitis. Answering aggregate queries in data exchange. In *Proceedings of the 27th ACM Symposium on Principles of Database Systems, PODS'08*, pages 129–138, 2008.

[Bar09] Pablo Barceló. Logical foundations of relational data exchange. *SIGMOD Record*, 38(1):49–58, 2009.

[BV84] Catriel Beeri and Moshe Vardi. A proof procedure for data dependencies. *Journal of the ACM*, 31(4):718–741, 1984.

[DL97] Oliver Duschka and Alon Y. Levy. Recursive plans for infomation gathering. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence, IJCAI'97*, pages 778–784, 1997.

[Ehr61] Andrzej Ehrenfeucht. An application of games to the completeness problem for formalized theories. *Fundamenta Mathematicae*, 49:129–141, 1961.

[Ete97] Kousha Etessami. Counting quantifiers, successor relations, and logarithmic space. *Journal of Computer and System Sciences*, 54(3):400–411, 1997.

[Fag82] Ronald Fagin. Horn clauses and database dependencies. *J. ACM*, 29(4):952–985, 1982.

[Fag97] Ronald Fagin. Easier ways to win logical games. In *Proceedings of the DIMACS Workshop on Finite Models and Descriptive Complexity*, pages 1–32. American Mathematical Society, 1997.

[FK12] Ronald Fagin and Phokion G. Kolaitis. Local transformations and conjunctive-query equivalence. In *Proceedings of the 31st ACM Symposium on Principles of Database Systems, PODS'12*, pages 179–190, 2012.

[FKMP05] Ronald Fagin, Phokion Kolaitis, Renee Miller, and Lucian Popa. Data exchange: semantics and query answering. *Theoretical Computer Science*, 336(1):89–124, 2005.

[FKP05] Ronald Fagin, Phokion Kolaitis, and Lucian Popa. Data exchange: getting to the core. *ACM Transactions on Database Systems*, 30(1):174–210, 2005.

[Fra54] Roland Fraïssé. Sur quelques classifications des systèmes de relations. *Publ. Sci. Univ. Alger. Sér. A*, 1:35–182, 1954.

[FSV95] Ronald Fagin, Larry Stockmeyer, and Moshe Vardi. On monadic NP vs monadic coNP. *Information and Computation*, 120(1):78–92, 1995.

[Gai82] Haim Gaifman. On local and non-local properties. In *Proceedings Herbrand Symposium Logic Colloquium, North Holland, 1982*, pages 105–135, 1982.

[GN08] Georg Gottlob and Alan Nash. Efficient core computation in data exchange. *Journal of the ACM*, 55(2), 2008.

[Han65] William P. Hanf. Model-theoretic methods in the study of elementary logic. In *The Theory of Models; Addison, Henkin, and Tarski, eds., North Holland 1965*, pages 132–145, 1965.

[HLN99] Lauri Hella, Leonid Libkin, and Juha Nurmonen. Notions of locality and their logical characterizations over finite models. *Journal of Symbolic Logic*, 64(4):1751–1773, 1999.

[HLS11] André Hernich, Leonid Libkin, and Nicole Schweikardt. Closed world data exchange. *ACM Transactions on Database Systems*, 36(2):14, 2011.

[HN92] Pavol Hell and Jaroslav Nešetřil. The core of a graph. *Discrete Mathematics*, 109(1-3):127–126, 1992.

[HN04] Pavol Hell and Jaroslav Nešetřil. *Graphs and Homomorphisms.* Oxford University Press, 2004.

[HTGL77] Barron Housel, Richard Taylor, Sakti Ghosh, and Vincent Y. Lum. EXPRESS: a data extraction, processing, and restructuring system. *ACM Transactions on Database Systems*, 2(2):134–174, 1977.

[IL84] Tomasz Imielinski and Witold Lipski. Incomplete information in relational databases. *Journal of the ACM*, 31(4):761–791, 1984.

[Kol05] Phokion Kolaitis. Schema mappings, data exchange, and metadata management. In *Proceedings of the 24th ACM Symposium on Principles of Database Systems, PODS'05*, pages 61–75, 2005.

[Len02] Maurizio Lenzerini. Data integration: a theoretical perspective. In *Proceedings of the 21st ACM Symposium on Principles of Database Systems, PODS'02*, pages 233–246, 2002.

[Lib04] Leonid Libkin. *Elements of Finite Model Theory.* Springer-Verlag, 2004.

[LS11] Leonid Libkin and Cristina Sirangelo. Data exchange and schema mappings in open and closed worlds. *Journal of Computer and System Sciences*, 77(3):542–571, 2011.

[MMS79] David Maier, Alberto Mendelzon, and Yehoshua Sagiv. Testing implications of data dependencies. *ACM Transactions on Database Systems*, 4(4):455–469, 1979.

[Nur96] Juha Nurmonen. On winning strategies with unary quantifiers. *Journal of Logic and Computation*, 6(6):779–798, 1996.

[Nur00] Juha Nurmonen. Counting modulo quantifiers on finite structures. *Information and Computation*, 160(1-2):62–87, 2000.