

No Cliques Allowed: The Next Step Towards BDD/FC Conjecture

LUCAS LARROQUE, Inria, DI ENS, ENS, CNRS, PSL University, France

PIOTR OSTROPOLSKI-NALEWAJA, University of Wrocław, Poland; TU Dresden, Germany

MICHAËL THOMAZO, Inria, DI ENS, ENS, CNRS, PSL University, France

This paper addresses one of the fundamental open questions in the realm of existential rules: the conjecture on the finite controllability of bounded derivation depth rule sets ($\text{bdd} \Rightarrow \text{fc}$). We take a step toward a positive resolution of this conjecture by demonstrating that universal models generated by bdd rule sets cannot contain arbitrarily large tournaments (arbitrarily directed cliques) without entailing a loop query, $\exists x E(x, x)$. This simple yet elegant result narrows the space of potential counterexamples to the ($\text{bdd} \Rightarrow \text{fc}$) conjecture.

CCS Concepts: • **Theory of computation** → **Constraint and logic programming**; **Finite Model Theory**; **Automated reasoning**; **Logic and databases**.

Additional Key Words and Phrases: Chase, Existential Rules, Finite unification sets, Bounded Derivation Depth Property, Tuple Generating Dependencies, TGDs, BDD, FUS, Finite Controllability, FC/BDD Conjecture, FO-Rewritability

ACM Reference Format:

Lucas Larroque, Piotr Ostropolski-Nalewaja, and Michaël Thomazo. 2025. No Cliques Allowed: The Next Step Towards BDD/FC Conjecture. *Proc. ACM Manag. Data* 3, 2 (PODS), Article 101 (May 2025), 20 pages. <https://doi.org/10.1145/3725238>

1 Introduction

The last decade has witnessed a fruitful interplay between the database and the knowledge representation communities, aiming to facilitate the integration and querying of legacy databases. The general idea is to allow a user to formulate queries using a logical formalism, whose predicates' meaning is constrained through first-order formulas. This leads to a core reasoning problem, coined *ontology based query answering* (OBQA), that takes as input a database $\mathcal{I}$, a rule set $\mathcal{R}$ and a Boolean conjunctive query q , and asks whether $\mathcal{I}, \mathcal{R} \models q$, where $\models$ denotes the entailment relation of first-order logic. In other terms, is that true that any model (finite or infinite) of $\mathcal{I}$ and $\mathcal{R}$ is a model of q ? Rules are often expressed as *existential rules*¹ which are formulas of the shape $\forall \bar{x}, \bar{y} B(\bar{x}, \bar{y}) \rightarrow \exists \bar{z} H(\bar{y}, \bar{z})$, where B and H are conjunctions of atoms respectively called the *body* and the *head* of the rule. It is well known that when no further constraints are put on rules, OBQA is an undecidable problem.

¹Existential rules are also referred to as a *tuple-generating dependencies* [1], *conceptual graph rules* [28], *Datalog[±]* [19], and *$\forall\exists$ -rules* [2] in the literature.

Authors' Contact Information: Lucas Larroque, Inria, DI ENS, ENS, CNRS, PSL University, Paris, France, lucas.larroque@inria.fr; Piotr Ostropolski-Nalewaja, University of Wrocław, Wrocław, Poland; and TU Dresden, Dresden, Germany, postropolski@cs.uni.wroc.pl; Michaël Thomazo, Inria, DI ENS, ENS, CNRS, PSL University, Paris, France, michael.thomazo@inria.fr.



This work is licensed under a Creative Commons Attribution 4.0 International License.

© 2025 Copyright held by the owner/author(s).

ACM 2836-6573/2025/5-ART101

<https://doi.org/10.1145/3725238>

Decidability of OBQA. A significant research effort has thus been devoted to design conditions on $\mathcal{R}$ that ensure the decidability and sometimes the tractability of Boolean conjunctive query answering under $\mathcal{R}$. An important tool to understand these conditions is the *chase* [24]. In a nutshell, the chase is an algorithm that adds fresh terms and new atoms to ensure that each mapping of a rule body can be extended into a mapping of a rule head. The result of this possibly infinite process is a specific model of $\mathcal{I}$ and $\mathcal{R}$, which has a universal property [11]: it homomorphically maps to any model of $\mathcal{I}$ and $\mathcal{R}$. When the chase is finite (which is for instance the case when the rule set fulfills some kind of acyclicity [8, 10, 13, 16, 23]), this provides a decision procedure for query answering. It is also the case when the chase is not finite, but is a structure of bounded treewidth — this is typically the case for guarded rules [5].

Finally, and of great interest for our work, is the case of UCQ-rewritability. A rule set $\mathcal{R}$ is said to be UCQ-rewritable if conjunctive queries (CQs) can always be rewritten. Specifically, for every CQ q , there exists a union of conjunctive queries (UCQ) $Q_{\mathcal{R}}$ such that, for any database $\mathcal{I}$:

$$\mathcal{I}, \mathcal{R} \models q \iff \mathcal{I} \models Q_{\mathcal{R}}.$$

Given that the chase can produce an infinite structure, the UCQ-rewritability of a rule set ensures decidable OBQA, as the rewriting can be computed and subsequently used to query the database. Unsurprisingly, UCQ-rewritability has been the subject of extensive research, leading to the introduction of numerous subclasses over the past few decades, including: - *Linear theories*, which permit at most a single atom in rule bodies [6]; *Guarded bdd theories*, which generalize linear theories [3, 9]; *Sticky theories*, where strong restrictions are imposed on joins [7].

UCQ-rewritability is also studied independently under a number of different names — highlighting the fundamental nature of the property — with notable cases of finite unification sets class (fus) and bounded derivation depth property (bdd). The fus class is closely tied to the concept of the backward chaining procedure, which is essentially a process of “chasing in reverse.” In this approach, to answer the entailment question, one starts with a query and attempts to derive a substructure of the database by applying rules in reverse. The bdd class, formally introduced in the next section, has a much older origin, dating back to the 1980s. It stems from the classical notion of the boundedness of a Datalog program [15].

OBQA in the finite. Databases, however, are finite structures, and under the understanding that $\mathcal{I}$ is a partial description of the real world, and that $\mathcal{R}$ are constraints that should hold on the finite real world, the classical notion of entailment provided by first-order logic does not correspond to the desired notion of entailment. Indeed, one should be more interested in knowing whether “ q holds in any *finite* database that contains $\mathcal{I}$ and is a model of $\mathcal{R}$ ”. This semantics is clearly related with the previous one, as a consequence of $\mathcal{I}$ and $\mathcal{R}$ in the unrestricted semantics is also one in the finite one. The converse is however not true, as the following prototypical example witnesses.

Example 1. Let us consider $\mathcal{I} = \{E(a, b)\}$, and $\mathcal{R}$ containing the two rules $\forall x \forall y E(x, y) \rightarrow \exists z E(y, z)$ and $\forall x \forall y \forall z E(x, y) \wedge E(y, z) \rightarrow E(x, z)$. The query $\exists x E(x, x)$ is not a consequence of $\mathcal{I}, \mathcal{R}$ under the unrestricted semantics, as witnessed by the chase. In any finite model however, there must be a cycle, and hence a loop because of the transitivity enforced by the second rule.

Finite Controllability. Finite reasoning provides a user with a semi-decision procedure for non-entailment (enumerating finite interpretations, and checking whether there exists one that is not a model of Q), but entailment is not semi-decidable anymore, as a finite universal model may not exist. An interesting question, already raised by Rosati [27], is, given a rule set $\mathcal{R}$, whether for any database $\mathcal{I}$ and any query Q the unrestricted and the finite semantics coincide. If so, the rule set $\mathcal{R}$ is said to be *finitely controllable* (fc). Note that whenever a rule set is fc, query answering becomes decidable. Indeed, the classical mathematical tools for semi-deciding first order entailment are still

available, while non-entailment can be witnessed by finite structures. Rosati showed that inclusion dependencies, a very restricted form of existential rules, enjoy finite controllability [27]. This has been generalized in two ways, for guarded rules [4], as well as for sticky rules [18]. Note that both classes generalize inclusion dependencies in various ways, guarded rules enjoying the bounded treewidth property, while sticky rules enjoy the bounded derivation depth property.

There is a well-known conjecture by Gogacz and Marcinkowski [17] stating that rule sets with the bounded derivation depth (bdd) property are finitely controllable ($\text{bdd} \Rightarrow \text{fc}$). Moreover, they demonstrated that the conjecture holds in a specific restricted setting where bdd rule sets are defined over a binary signature and have heads containing only a single atom [17]. However, this result applies to a highly limited case, and little progress has been made towards resolving the conjecture in its general form. The broader question of whether finite controllability extends to all bdd rule sets remains an important open problem.

Contributions. Our main contribution is to show that bdd rule sets enjoy certain model-theoretic property. We write TOURNAMENTS_E to denote the query: *for all integers k , there is a set $\{x_1, \dots, x_k\}$ of elements in the given instance such that $E(x_i, x_j)$ or $E(x_j, x_i)$ holds in the instance for all $i \neq j$* , and write LOOP_E to denote the query $\exists x E(x, x)$. Then, the following is our main result:

Theorem 1. *For every bdd rule set $\mathcal{R}$ and every instance I we have:*

$$(I, \mathcal{R}) \models \text{TOURNAMENTS}_E \Rightarrow (I, \mathcal{R}) \models \text{LOOP}_E. \quad (\hat{\Delta})$$

In addition to providing insights into the structural properties of universal models of bdd rule sets, this result serves as an important step towards proving the ($\text{bdd} \Rightarrow \text{fc}$) conjecture. Specifically, it narrows the space of potential counterexamples to the conjecture by eliminating the most natural ones. We briefly explain why this is the case. First note that any model that entails TOURNAMENTS_E and not LOOP_E is infinite, as the terms $x_1, \dots, x_k$ in TOURNAMENTS_E must be distinct for LOOP_E not to be entailed. Thus, TOURNAMENTS_E does imply LOOP_E in the finite setting, and any bdd rule set for which this implication does not hold (in the unrestricted setting) would disprove ($\text{bdd} \Rightarrow \text{fc}$).

Note however that Example 1 does not constitute such a counterexample, as its rule set is simply not bdd, since the transitivity rule has to be applied at least as many times as the distance between a and b to entail $E(a, b)$, which cannot be bounded independently of the size of the database. Expanding on that example, one could try to mimic the behavior using bdd rules by replacing the transitivity rule with $\forall x \forall x' \forall y \forall y' E(x, x') \wedge E(y, y') \rightarrow E(x, y')$, which entails it. With this change, the rule set is finally bdd, and the chase entails TOURNAMENTS_E . However, this new rule triggers the entailment of $\exists x E(x, x)$ as soon as $\exists x \exists y E(x, y)$ is entailed, as expected due to Property $(\hat{\Delta})$.

While Theorem 1 does not entail the ($\text{bdd} \Rightarrow \text{fc}$) conjecture, as a different type of counterexample could in principle exist, we are confident that the insights and carefully curated toolkit presented in the following sections represent an important milestone in the effort to settle the ($\text{bdd} \Rightarrow \text{fc}$) conjecture in the general setting.

Organization of the paper. We first provide the necessary preliminaries, before stating our main result in Section 3. The proof is performed in two main steps. The first step (Section 4) involves a series of rule set “surgeries”. While some of these are well-known, the properties we require are tailored to the specific case at hand and need a separate formal proof. The second step (Section 5) primarily consists of a complex model-theoretic argument, which demonstrates that Property $(\hat{\Delta})$ holds for the transformed rules.

2 Preliminaries

We assume that the reader is familiar with first-order logic and basic concepts from graph theory. We only provide a brief recap of these topics in this section.

2.1 First-Order Logic

We define Preds, and Vars to be mutually disjoint and countably infinite sets of predicates, and variables, respectively. We associate every $P \in \text{Preds}$ with some *arity* $\text{ar}(P) \geq 0$. For every $n \geq 0$, the set of all n -ary predicates is also countably infinite. A *signature* is a set of predicates. We write tuples $(x_1, \dots, x_n)$ of variables as $\bar{x}$ and often treat these tuples as sets. The *support* of a tuple of variables is the set of variables that occur in it. A tuple of variables $\bar{y} = (y_1, \dots, y_m)$ is *compatible* with another tuple $\bar{x} = (x_1, \dots, x_n)$ if $m = n$, and for all $i, j \leq n$, we have $y_i = y_j$ whenever $x_i = x_j$. If, furthermore, for all i , either $y_i = x_i$ or $y_i = y_j = x_j$ for some j , then $\bar{y}$ is a *specialization* of $\bar{x}$. Any two tuples $\bar{x} = (x_1, \dots, x_n)$ and $\bar{t} = (t_1, \dots, t_n)$ such that $\bar{t}$ is compatible with $\bar{x}$ define a substitution $[\bar{x} \mapsto \bar{t}] : x_i \mapsto t_i$.

An *atom* over a predicate P , or P -atom, is a first-order formula of the form $P(\bar{x})$ with $\bar{x}$ a tuple of variables and P a $|\bar{x}|$ -ary predicate. For a nullary predicate P , we write P as a shortcut for the nullary atom $P()$. An *instance* over signature $\mathbb{S}$ is a set of atoms over predicates in $\mathbb{S}$. For simplicity of presentation, we assume that all instances contain a nullary fact $\top$. The *active domain* of an atom or an instance is the set of variables that occur in it. The active domain of an atom (or set thereof) S is denoted with $\text{adom}(S)$. The *disjoint union* $\mathcal{I}_1 \dot{\cup} \mathcal{I}_2$ of two instances $\mathcal{I}_1$ and $\mathcal{I}_2$ is $\mathcal{I}_1 \cup \sigma(\mathcal{I}_2)$, where σ is a function mapping variables of $\mathcal{I}_2$ to fresh variables, that do not occur in $\mathcal{I}_1$. For a first-order formula F and a tuple $\bar{x}$ of variables, we write $F(\bar{x})$ to indicate that $\bar{x}$ is the set of all free variables occurring in F .

Existential rules. An (*existential*) *rule* ρ is an FO formula of the form $\forall \bar{x}, \bar{y} B(\bar{x}, \bar{y}) \rightarrow \exists \bar{z} H(\bar{y}, \bar{z})$ where B and H are non-empty atom conjunctions. We refer to $\bar{y}$ as the *frontier* of ρ , and to $B(\bar{x}, \bar{y})$ and $\exists \bar{z} H(\bar{y}, \bar{z})$ as the *body* and the *head* of ρ , respectively. Such a rule is *Datalog* if it does not feature existential variables; that is, if $\bar{z}$ is empty. Given a rule ρ , the sets $\text{body}(\rho)$, $\text{head}(\rho)$ and $\text{fr}(\rho)$ denote the body, the head and the frontier of ρ , respectively. We refer to elements of $\text{body}(\rho)$ (resp. $\text{head}(\rho)$) as *body-atoms* (resp. *head-atoms*) of ρ . We omit $\forall$ quantifiers when writing rules. A rule set is over a signature $\mathbb{S}$ if all the atoms occurring in any of its rules are over a predicate in $\mathbb{S}$.

Homomorphisms. A *substitution* π is a function from Vars to Vars. For an atom $P(\bar{t})$, let $\pi(P(\bar{t}))$ be the atom that results from replacing all occurrences of every variable x in $P(\bar{t})$ with $\pi(x)$ if the latter is defined. A *homomorphism* π from an instance $\mathcal{A}$ to an instance $\mathcal{B}$ is a substitution such that $\pi(\mathcal{A}) \subseteq \mathcal{B}$. An *isomorphism* π from $\mathcal{A}$ to $\mathcal{B}$ is an injective and surjective homomorphism from $\mathcal{A}$ to $\mathcal{B}$. Two atom sets $\mathcal{A}$ and $\mathcal{B}$ are *homomorphically equivalent* if there is a homomorphism from $\mathcal{A}$ to $\mathcal{B}$ and a homomorphism from $\mathcal{B}$ to $\mathcal{A}$. We denote homomorphic equivalence with $\leftrightarrow$ symbol.

Queries. A *conjunctive query* (CQ) $q(\bar{x})$ is a pair of a first-order formula of the form $\exists \bar{z} B(\bar{x}, \bar{z})$, where B is a non-empty conjunction of atoms, and a tuple $\bar{x}$ of *answer variables* whose support is the set of free variables of the formula. A CQ is *Boolean* if it features no answer variable. A *union of conjunctive queries* (UCQ) $Q(\bar{x})$ is a pair of disjunction of CQs, that we often see as a set of CQs $\{q_1(\bar{x}_1), \dots, q_n(\bar{x}_n)\}$, and a tuple of answer variables $\bar{x}$ such that for all i , $\bar{x}_i$ is a specialization $\bar{x}$. Given a UCQ $Q(\bar{x}, \bar{y})$ and a tuple of variables $\bar{x}'$ compatible with $\bar{x}$, we denote with $Q(\bar{x}', \bar{y})$ the UCQ $Q[\bar{x} \mapsto \bar{x}'](\bar{x}', \bar{y})$. Under standard first-order semantics, an instance $\mathcal{I}$ entails a UCQ $Q(\bar{x}) = \{q_1(\bar{x}_1), \dots, q_n(\bar{x}_n)\}$ for a given tuple $\bar{t}$ of elements of $\text{adom}(\mathcal{I})$ compatible with $\bar{x}$ if there is a homomorphism from $q_k(\bar{x}_k[\bar{x} \mapsto \bar{t}])$ to $\mathcal{I}$ for some $1 \leq k \leq n$. If, furthermore, this homomorphism is injective, then $\mathcal{I}$ *injectively entails* Q . We denote these facts with $\mathcal{I} \models Q(\bar{t})$ and $\mathcal{I} \models_{\text{inj}} Q(\bar{t})$, resp. For a rule set $\mathcal{R}$, an instance $\mathcal{I}$ and a UCQ Q , we write $\langle \mathcal{R}, \mathcal{I} \rangle \models Q(\bar{t})$ to indicate that $\mathcal{R} \cup \mathcal{I}$ entails (over unrestricted models) $Q(\bar{x})$ for $\bar{t}$, and $\langle \mathcal{R}, \mathcal{I} \rangle \models_{\text{inj}} Q(\bar{t})$ when $\mathcal{R} \cup \mathcal{I}$ injectively entails (over unrestricted models) $Q(\bar{x})$ for $\bar{t}$.

2.2 The Chase

Given a rule set $\mathcal{R}$, an $(\mathcal{R})$ -trigger over some instance I is a pair $\langle \rho, h \rangle$ of a rule $\rho \in \mathcal{R}$ and a homomorphism h from $\text{body}(\rho)$ to some subset of I . The *output* of a trigger $\tau = \langle \rho, h \rangle$ over I is the instance $\text{output}(\tau) = h'(\text{head}(\rho))$, where h' is a homomorphism extending h by mapping existentially quantified variables of $\text{head}(\rho)$ to fresh variables. With $\text{triggers}(I, \mathcal{R})$ we denote the set of all triggers for I and $\mathcal{R}$. Step k of the (oblivious) chase [6, 21] is denoted with $Ch_k(I, \mathcal{R})$, and defined by

$$Ch_0(I, \mathcal{R}) = I \quad Ch_{n+1}(I, \mathcal{R}) = Ch_n(I, \mathcal{R}) \cup \bigcup_{\tau \in T_n} \text{output}(\tau) \quad (\text{for } n \in \mathbb{N}_{>0})$$

where $T_n = \text{triggers}(Ch_n(I, \mathcal{R}), \mathcal{R}) \setminus \text{triggers}(Ch_{n-1}(I, \mathcal{R}), \mathcal{R})$. The *result of the chase* from $\langle I, \mathcal{R} \rangle$, denoted with $Ch(I, \mathcal{R})$, is the union of $Ch_n(I, \mathcal{R})$ over all n . It is well known that $Ch(I, \mathcal{R})$ is a universal model of I and $\mathcal{R}$. To ease the reading, we denote $Ch_n(\{\top\}, \mathcal{R})$ as $Ch_n(\mathcal{R})$ and $Ch(\{\top\}, \mathcal{R})$ as $Ch(\mathcal{R})$.

Given an instance I and a rule set $\mathcal{R}$, a *chase term* is an element of $\text{adom}(Ch(I, \mathcal{R})) \setminus \text{adom}(I)$. Thus, every chase term is created by applying some trigger. The *frontier* of a chase term t created by some trigger $\langle \rho, h \rangle$ is the set $h(\text{fr}(\rho))$.

2.3 UCQ-rewritability

Definition 2. A rule set $\mathcal{R}$ is *UCQ-rewritable* if for all CQs $q(\bar{x})$, there is a UCQ $Q(\bar{x})$ such that for all instances I and tuples $\bar{t}$ of elements $\text{adom}(I)$ compatible with $\bar{x}$, we have

$$\langle I, \mathcal{R} \rangle \models q(\bar{t}) \iff I \models Q(\bar{t}).$$

The UCQ $Q(\bar{x})$ is a (UCQ)-*rewriting* of $q(\bar{x})$ and is denoted with $\text{rew}(q(\bar{x}), \mathcal{R})$.

Note that $\text{rew}(q(\bar{x}), \mathcal{R})$ is not unique; however, we can take the minimal one, which is uniquely defined up to bijective renaming of variables [22]. Moreover, the notion of UCQ rewriting can easily be lifted from CQs to UCQs.

Definition 3. A rule set has *bounded derivation depth* (*bdd*) if for every CQ q , there is some $k \geq 0$ such that for all instances I , we have $\langle I, \mathcal{R} \rangle \models q$ if and only if $Ch_k(I, \mathcal{R}) \models q$. We use *bdd* to denote the class of rule sets with *bdd*. For a given q , we call the minimal k as above the *bdd-constant* of q and denote it with $\text{bdd}(q, \mathcal{R})$.

Interestingly the notions coincide. For a full proof of this result, see Gottlob et al. [20].

Proposition 4. A rule set is UCQ-rewritable if and only if it is *bdd*.

Lemma 5. Let $\mathcal{R}_1$ and $\mathcal{R}_2$ be *bdd*. If for every instance I it holds that $Ch(Ch(I, \mathcal{R}_1), \mathcal{R}_2)$ is homomorphically equivalent to $Ch(I, \mathcal{R}_1 \cup \mathcal{R}_2)$ then $\mathcal{R}_1 \cup \mathcal{R}_2$ is *bdd*.

PROOF. Let Q be a rewriting of a CQ q against $\mathcal{R}_2$ and then Q' be a rewriting of Q against $\mathcal{R}_1$. Note that Q' is a rewriting of q against $\mathcal{R}_1 \cup \mathcal{R}_2$ as $Ch(Ch(I, \mathcal{R}_1), \mathcal{R}_2) \leftrightarrow Ch(I, \mathcal{R}_1 \cup \mathcal{R}_2)$. $\square$

Later, we will need injective rewritings, meaning we would like to have Definition 2 rephrased:

Definition 2 (rephrased). A rule set $\mathcal{R}$ is *UCQ-rewritable* if for all CQs $q(\bar{x})$, there is a UCQ $Q(\bar{x})$ such that for all instances I and tuples $\bar{t}$ of elements $\text{adom}(I)$ compatible with $\bar{x}$, we have:

$$\langle I, \mathcal{R} \rangle \models q(\bar{t}) \iff I \models_{\text{inj}} Q(\bar{t})$$

The UCQ $Q(\bar{x})$ is an *injective rewriting* of $q(\bar{x})$ and is denoted with $\text{rew}_{\text{inj}}(q(\bar{x}), \mathcal{R})$.

The above is indeed an equivalent definition due to the following:

Proposition 6. *For every UCQ Q there exists another UCQ Q_{inj} s.t. for every instance I and tuple $\bar{a}$:*

$$I \models Q(\bar{a}) \iff \exists q \in Q_{\text{inj}} \quad I \models_{\text{inj}} q(\bar{a}) \iff I \models Q_{\text{inj}}(\bar{a}).$$

PROOF. We will show the case when Q is a CQ, the proof easily generalizes to the UCQ case. Let q be a CQ and $\bar{x}$ be the tuple of its variables. Note that whenever q maps to some instance I via a homomorphism h we have that there exists a specialization $\bar{y}$ of $\bar{x}$ and an injective homomorphism h' such that $[\bar{x} \mapsto \bar{y}] \circ h' = h$. Then the required UCQ is simply a disjunction of $q[\bar{x} \mapsto \bar{y}]$ for all specializations $\bar{y}$ of $\bar{x}$. To get the second equivalence note the above construction is idempotent. $\square$

2.4 Other Useful Notions

Graph Theoretical Notions. A (directed) graph is a pair $\langle V, E \rangle$ composed of a set V of vertices, and a subset E of $V \times V$ whose elements are called edges. The size of a graph is the number of its vertices. We often say that an edge $\langle v, w \rangle \in E$ goes from v to w . A loop is an edge from a node to itself. A subgraph $\langle V', E' \rangle$ of a graph $\langle V, E \rangle$ is a graph such that $V' \subseteq V$ and $E \subseteq E'$. A subgraph $\langle V', E' \rangle$ of $\langle V, E \rangle$ is induced if for all $v, w \in V'$, if $\langle v, w \rangle \in E$, then $\langle v, w \rangle \in E'$. A path between two vertices s and t is a sequence of vertices $v_0, \dots, v_n$ such that $v_0 = s$, $v_n = t$ and for all $i < n$, $\langle v_i, v_{i+1} \rangle \in E$. A cycle is a path from a node to itself. A graph is acyclic if it does not contain any cycle. A tournament is a graph such that for all distinct vertices v, w , we have $\langle v, w \rangle \in E$ or $\langle w, v \rangle \in E$. A k -coloring of the edges of a graph $\langle V, E \rangle$ is a function from E to $\{1, \dots, k\}$.³ Given a graph G and a k -coloring C of its edges, a subgraph $\langle V', E' \rangle$ of G is colored with i for some $i \leq k$ if for all $e \in E$, we have $C(e) = i$. The following is a rephrasing of the Ramsey's theorem for directed graphs.

Theorem 7. *For any integers $s_1, \dots, s_k \geq 1$, there is some integer $R(s_1, \dots, s_k)$ such that for any tournament T of size at least $R(s_1, \dots, s_k)$ and any k -coloring of the edges of T , there is a subgraph of T that is a tournament of size s_i colored with some i .*

Multisets. A multiset over a set D is a function M from D to $\mathbb{N}$. A multiset is finite if $\{x \in D \mid M(x) > 0\}$ is finite. The size of a finite multiset M is $|M| = \sum_{x \in D} M(x)$. The empty multiset is $\emptyset_m : x \mapsto 0$. Given a finite list $[x_1, \dots, x_n]$ of elements of D , we denote the multiset $x \mapsto |\{i \mid x_i = x\}|$ with $\{x_1, \dots, x_n\}_m$. Given a function f from some finite set E to D , we denote the multiset $x \mapsto |f^{-1}(x)|$ with $\{f(x) \mid x \in E\}_m$. The union of two multisets M and N is the multiset $M \cup_m N : x \mapsto M(x) + N(x)$, their intersection is $M \cap_m N : x \mapsto \min(M(x), N(x))$, and their difference is $M \setminus_m N : x \mapsto \max(M(x) - N(x), 0)$.

Consider a strict order $<$ over D . The maximum of a multiset M over D is $\max_m(M) = \max\{x \in D \mid M(x) > 0\}$ whenever it is defined, and undefined otherwise. The strict lexicographical order $<_{\text{lex}}$ over multisets is defined inductively as follows: for all non-empty multisets M , $\emptyset_m <_{\text{lex}} M$, and given two non-empty multisets M and N , we have $M <_{\text{lex}} N$ if and only if $\max_m(M) < \max_m(N)$, or if $\max_m(M) = \max_m(N)$ and $(M \setminus_m \{\max_m(M)\})_m <_{\text{lex}} (N \setminus_m \{\max_m(N)\})_m$. The lexicographical order $\leq_{\text{lex}}$ is defined by $M \leq_{\text{lex}} N$ if and only if $M <_{\text{lex}} N$ or $M = N$ as usual.

Lemma 8. *If $<$ is well-founded on D , then for all k , the order $<_{\text{lex}}$ is well-founded on the set of multisets over D of size at most k .*

PROOF. We proceed by induction on k . As the only multiset of size 0 is $\emptyset_m$, the case $k = 0$ is trivial. Then, consider a set $\mathbb{S}_k$ of multisets of size at most k over D . Then, since all the elements of $\mathbb{S}_k$ are finite, they all admit a maximum. Since $<$ is well-founded over D , the set $\{\max_m(M) \mid M \in \mathbb{S}_k\}$ admits a minimum m . Consider the set $\mathbb{S}_{k-1} = \{M \mid M \cup_m \{m\} \in \mathbb{S}_k \wedge \max_m(M \cup_m \{m\}) = m\}$. As all the multisets in this set have size at most $k - 1$, by induction hypothesis, it admits a minimum

²Note that this “or” is inclusive, contrary to the usual definition of a tournament.

³Note that we do not impose adjacent vertices to have different colors.

$M'_{\min}$. Then, we claim that $M_{\min} = M'_{\min} \cup_m \{m\}_m$ is the minimum of $\mathbb{S}_k$. Consider an element M of $\mathbb{S}_k$. If $M \setminus_m \{m\}_m \notin \mathbb{S}_{k-1}$, then $\max_m(M) > m = \max_m(M_{\min})$ by definition of $\mathbb{S}_{k-1}$, so $M_{\min} \leq_{\text{lex}} M$. Otherwise, if $M \setminus_m \{m\}_m \in \mathbb{S}_{k-1}$, then $\max_m(M) = m = \max_m(M_{\min})$, and $M_{\min} \setminus_m \{m\}_m = M'_{\min} \leq_{\text{lex}} M \setminus_m \{m\}_m$ as $M'_{\min}$ is the minimum of $\mathbb{S}_{k-1}$. Thus, $M_{\min} \leq_{\text{lex}} M$, which concludes the proof. $\square$

3 The Main Result

Definition 9. Given a rule set $\mathcal{R}$, an instance I and a binary predicate E , we write TOURNAMENTS_E to denote the following query:⁴ *for all integers k , there exists a E -tournament of size k in the given instance.*

Definition 10. We define the E -loop query $\exists x E(x, x)$ and denote it with LOOP_E .

For the rest of the paper, let us fix a binary predicate E for use in both TOURNAMENTS_E and LOOP_E .

Theorem 1 (restated). *For every signature $\mathbb{S}$, every UCQ-rewritable rule set $\mathcal{R}$ over $\mathbb{S}$, and every instance I we have:*

$$\text{Ch}(I, \mathcal{R}) \models \text{TOURNAMENTS}_E \quad \Rightarrow \quad \text{Ch}(I, \mathcal{R}) \models \text{LOOP}_E. \quad (\hat{\Delta})$$

In the context of this theorem and statements with a similar structure we call $(\hat{\Delta})$ the *consequence* while the preceding part is the *assumption*. We shall devote the rest of the paper to the proof of the above, main result.

4 Reducing the Main Theorem

The goal of this section is to reduce the assumption of the main theorem of the paper to a more controlled one in which the considered instance is always $\{\top\}$ and the rule set satisfies a number of properties. This overarching goal is reflected in Theorem 28 which - as we show in this section - is equivalent to Theorem 1. The challenge here lies in showing that Theorem 28 implies Theorem 1 as the other direction is trivial. The proof goes via a number of steps - each showing equivalence to some intermediate statement (Lemmas 11, 18 and 23) with progressively stronger assumptions. The toolkit of the section consists of a number of rule set surgeries of varying degrees of complexity.

All the parts of the proof (of equivalence of Theorem 1 and Theorem 28), will follow the same formula, which we will discuss alongside the next step.

4.1 Encoding Instances in Rule Sets

We begin with the first intermediate statement, enforcing the instance in Theorem 1 to be $\{\top\}$.

Lemma 11. *For every signature $\mathbb{S}$, and every UCQ-rewritable rule set $\mathcal{R}$ over $\mathbb{S}$ we have:*

$$\text{Ch}(\mathcal{R}) \models \text{TOURNAMENTS}_E \quad \Rightarrow \quad \text{Ch}(\mathcal{R}) \models \text{LOOP}_E. \quad (\hat{\Delta})$$

The goal of this step is to show that Lemma 11 is equivalent to Theorem 1. As we are restricting the set of potential instances, the $(\Rightarrow)$ direction of the proof is trivial, that is Theorem 1 implies Lemma 11.

In this and the following steps of the proof, we shall always prove the $(\Leftarrow)$ direction via contraposition. Therefore, assume signature $\mathbb{S}$, rule set $\mathcal{R}$, and instance I form a counterexample to Theorem 1, now we shall construct $\mathcal{R}'$ such that $\mathcal{R}'$ and $\mathbb{S}$ constitute a counterexample to Lemma 11.

⁴In order for $\models$ to make sense with TOURNAMENTS_E we can define the query via the following second order logic formula: $\forall X \text{ finite}, \exists Y \forall x, y \exists x', y' (X(x) \wedge X(y) \rightarrow Y(x, x') \wedge Y(y, y') \wedge E(x', y'))$. The finiteness of X is expressed by stating that all functions from X to X are injective if and only if they are surjective.

Rule Set Surgery. The counterexample construction will always consist of some rule set surgery.

Definition 12. For any instance $\mathcal{J}$ we denote with $\top \rightarrow \mathcal{J}$ the following rule

$$\top \rightarrow \exists f(\text{adom}(\mathcal{J})) \bigwedge_{A(\bar{t}) \in \mathcal{J}} A(f(\bar{t}))$$

where f is a bijective renaming of terms to fresh variables.

Observation 13. Let ρ be any rule with $\top$ being its body, then for any rule set $\mathcal{S}$ and any instance $\mathcal{J}$ we have:

$$\text{Ch}(\mathcal{J}, \mathcal{S} \cup \{\rho\}) \leftrightarrow \text{Ch}(\text{Ch}(\mathcal{J}, \{\rho\}), \mathcal{S}).$$

PROOF. Note that ρ always triggers, and triggers only once. $\square$

We simply let $\mathcal{R}' = \mathcal{R} \cup \{\top \rightarrow \mathcal{I}\}$. With $\mathcal{R}'$ constructed we shall show now that the newly constructed rule set $\mathcal{R}'$ and the original $\mathcal{S}$ form a counterexample to Lemma 11. This will always consist of two parts: showing that the assumption holds - in this case that $\mathcal{R}'$ is UCQ-rewritable - and that the consequence does not hold - which in this case amounts to showing:

$$\text{Ch}(\mathcal{R}') \models \text{TOURNAMENTS}_{\mathcal{E}} \wedge \text{Ch}(\mathcal{R}') \not\models \text{LOOP}_{\mathcal{E}}$$

Main Properties of the Surgery. As in the following steps we present here the main properties of the surgery. The raison d'être of such subsections is to provide a modular and reusable formulation of the main building blocks of the proof. As noted in the introduction, some of these techniques are well-known in the community. However, they often lack proper formulation or formal statements. Importantly, the properties of most of the techniques with respect to UCQ-rewritability are far from being well-known and are themselves results of separate interest to the community. Finally, we note that we will usually state two important properties - one used for the “Falsifying the Consequence” parts and one for the “Proving the Assumption” ones.

Observation 14. For every pair of instances $\mathcal{J}$ and $\mathcal{J}'$ and a rule set $\mathcal{S}$ we have that $\text{Ch}(\mathcal{J} \cup \mathcal{J}', \mathcal{S})$ is homomorphically equivalent to $\text{Ch}(\mathcal{J}, \mathcal{S} \cup \{\top \rightarrow \mathcal{J}'\})$.

PROOF. Consider the following sequence of equivalences:

$$\begin{aligned} \text{Ch}(\mathcal{J} \cup \mathcal{J}', \mathcal{S}) &\leftrightarrow \text{Ch}(\text{Ch}(\mathcal{J}, \{\top \rightarrow \mathcal{J}'\}), \mathcal{S}) && \text{(Definition of } \{\top \rightarrow \mathcal{J}'\}) \\ &\leftrightarrow \text{Ch}(\mathcal{J}, \mathcal{S} \cup \{\top \rightarrow \mathcal{J}'\}) && \text{(From Observation 13)} \quad \square \end{aligned}$$

The following allows us to encapsulate the first property of the surgery:

Corollary 15. For any instance $\mathcal{J}$ and any rule set $\mathcal{S}$ we have that:

$$\text{Ch}(\mathcal{J}, \mathcal{S}) \leftrightarrow \text{Ch}(\{\top\}, \mathcal{S} \cup \{\top \rightarrow \mathcal{J}\}).$$

and the following is its second property:

Observation 16. Given a rule set $\mathcal{S}$ that is UCQ-rewritable and an instance $\mathcal{J}$ we have that $\mathcal{S} \cup \{\top \rightarrow \mathcal{J}\}$ is UCQ-rewritable.

PROOF. Using Lemma 5 and Observation 13 we get that $\mathcal{S} \cup \{\top \rightarrow \mathcal{J}\}$ is UCQ-rewritable. $\square$

Falsifying the Consequence: ($\hat{\Delta}$).

Lemma 17. $\text{Ch}(\mathcal{R}') \models \text{TOURNAMENTS}_{\mathcal{E}} \wedge \text{Ch}(\mathcal{R}') \not\models \text{LOOP}_{\mathcal{E}}$.

PROOF. Follows from:

- $\text{Ch}(\mathcal{I}, \mathcal{R}) \leftrightarrow \text{Ch}(\mathcal{R}')$ (Corollary 15)
- $\text{Ch}(\mathcal{I}, \mathcal{R}) \models \text{TOURNAMENTS}_{\mathcal{E}} \wedge \text{Ch}(\mathcal{I}, \mathcal{R}) \not\models \text{LOOP}_{\mathcal{E}}$ (from assumption). $\square$

Proving the Assumption of Lemma 11. In this case we have to show that $\mathcal{R}'$ is UCQ-rewritable. It follows, from Observation 16, as $\mathcal{R}$ is UCQ-rewritable.

4.2 Reducing to Binary Signatures

Lemma 18. *For every binary signature $\mathbb{S}$, and every UCQ-rewritable rule set $\mathcal{R}$ over $\mathbb{S}$ we have:*

$$Ch(\mathcal{R}) \models \text{TOURNAMENTS}_{\mathbb{S}} \Rightarrow Ch(\mathcal{R}) \models \text{LOOP}_{\mathbb{S}} \quad (\clubsuit)$$

Now we show that Lemma 11 is equivalent to Lemma 18. The $(\Rightarrow)$ direction is trivial, we show $(\Leftarrow)$ via the contraposition. Let $\mathbb{S}$ be a signature and $\mathcal{R}$ be a rule set that together form a counterexample to Lemma 11.

Reification. Given a relational symbol A of arity $\text{ar}(A) > 2$ we define $\text{reify}(A)$ as the following set $\{A_1, \dots, A_{\text{ar}(A)}\}$ of binary predicates. Given a finite signature $\mathbb{S} = \mathbb{S}_{\leq 2} \uplus \mathbb{S}_{\geq 3}$ divided into at-most-binary and higher-arity predicates, we define the *reified version* of $\mathbb{S}$ as the binary signature $\text{reify}(\mathbb{S}) = \mathbb{S}_{\leq 2} \uplus \bigcup_{A \in \mathbb{S}_{\geq 3}} \text{reify}(A)$. Given an atom $\alpha = A(x_1, \dots, x_n)$ we define $\text{reify}(\alpha)$ as $\{A_i(x_i, x_\alpha) \mid 1 < i \leq n\}$ for $n > 2$ and let reify be the identity on atoms of arity less than three. We lift reify to instances, rules, and queries in the natural way.

We prove that $\mathcal{R}' = \text{reify}(\mathcal{R})$ and $\mathbb{S}' = \text{reify}(\mathbb{S})$ form a counterexample to Lemma 18.

Main Properties of Reification. We take our first property from Feller et al. [14] (Property (ii) of reification):

Lemma 19. *For any instance $\mathcal{J}$ and any rule set $\mathcal{S}$ we have that*

$$Ch(\text{reify}(\mathcal{J}), \text{reify}(\mathcal{S})) \leftrightarrow \text{reify}(Ch(\mathcal{J}, \mathcal{S})).$$

The relation of reification to UCQ-rewritability was to our knowledge not explored before:

Lemma 20. *For any UCQ-rewritable rule set $\mathcal{S}$ we have that $\text{reify}(\mathcal{S})$ is UCQ-rewritable as well.*

PROOF. Given a symbol A of signature Ξ of $\mathcal{S}$ whose arity is greater than two, let ρ_A be a rule

$$A(x_1, \dots, x_n) \rightarrow \exists z \bigwedge_{A_i \in \text{reify}(A)} A_i(x_i, z)$$

that is ρ_A is a rule that simply *projects* A to its reified variant. Let $\mathcal{S}'$ be $\mathcal{S}$ enriched with rules ρ_A for all relevant A of signature of $\mathcal{S}$. Note that $\mathcal{S}'$ is UCQ-rewritable - as $\mathcal{S}$ is UCQ-rewritable and the added rules cannot be fired recursively or cannot trigger rules from $\mathcal{S}$.

The restriction of $Ch(\mathcal{J}, \mathcal{S}')$ to reified signature $\text{reify}(\Xi)$ is isomorphic to $\text{reify}(Ch(\mathcal{J}, \mathcal{R}))$ for any instance $\mathcal{J}$. By Lemma 19, it is also isomorphic to $Ch(\text{reify}(\mathcal{J}), \text{reify}(\mathcal{R}))$. From this we conclude that if we take any CQ Q over $\text{reify}(\Xi)$ and let Φ be its rewriting against $\mathcal{S}'$, we have that $\text{reify}(\Phi)$ is the rewriting of Q against $\text{reify}(\mathcal{S})$ - which implies that $\text{reify}(\mathcal{S})$ is UCQ-rewritable. $\square$

Falsifying the Consequence: $(\clubsuit)$. Using Lemma 19 we have that $\text{reify}(Ch(\mathcal{R}))$ is homomorphically equivalent to $Ch(\mathcal{R}')$; hence if the consequence $(\clubsuit)$ does not hold, then $(\clubsuit)$ does not hold as well.

Proving the Assumption of Lemma 18. As $\mathbb{S}'$ is binary and we have Lemma 20 we have shown that assumption of Lemma 18 holds for $\mathcal{R}'$ and signature $\mathbb{S}'$ which ends the proof.

4.3 Streamlining the Heads

We now introduce two definitions that allow us to have a convenient structure in the chase as long as only non-Datalog rules have been applied: atoms are endowed with an orientation, and each term has a unique predecessor by atoms of a given predicate.

Definition 21 (Forward-existential). A rule ρ is *forward-existential* iff for each its head-atoms $A(x, y)$ we have that x is a frontier variable and y is an existential variable. A rule set is forward-existential iff each of its non-Datalog rule is forward-existential.

Definition 22 (Predicate-unique). A rule set $\mathcal{R}$ over binary signature $\mathbb{S}$ is *predicate-unique* iff for every non-Datalog rule ρ of $\mathcal{R}$ every predicate $E \in \mathbb{S}$ appears at most once in the head of ρ .

Note, that the above is not equivalent with single-head rule sets as

$$A(x), B(y) \rightarrow \exists z D(x, z), E(y, z)$$

is a predicate-unique, forward-existential rule.

Lemma 23. For every binary signature $\mathbb{S}$ and every UCQ-rewritable, forward-existential, and predicate-unique rule set $\mathcal{R}$ over $\mathbb{S}$ we have:

$$Ch(\mathcal{R}) \models \text{TOURNAMENTS}_{\mathbb{E}} \Rightarrow Ch(\mathcal{R}) \models \text{LOOP}_{\mathbb{E}}. \quad (\boxtimes)$$

Now we show that Lemma 18 is equivalent to Lemma 23. The $(\Rightarrow)$ direction is trivial, we show $(\Leftarrow)$ via the contraposition. Let $\mathbb{S}$ be a signature and $\mathcal{R}$ be a rule set that together form a counterexample to Lemma 18.

Rule Set Surgery. Let A and B (with indices) be fresh relational symbols. Given a rule

$$\rho = B(\bar{x}, \bar{y}) \rightarrow \exists \bar{z} H(\bar{y}, \bar{z})$$

we define:

- the existential rule ρ_{init} as

$$B(\bar{x}, \bar{y}) \rightarrow \exists w A_0^\rho(w) \wedge \bigwedge_{y \in \bar{y}} A_{y,w}^\rho(y, w);$$

- the existential rule $\rho_{\exists}$ as

$$A_0^\rho(w) \wedge \bigwedge_{y \in \bar{y}} A_{y,w}^\rho(y, w) \rightarrow \exists \bar{z} \bigwedge_{y' \in \bar{y} \cup \{w\}} \bigwedge_{z \in \bar{z}} B_{y',z}^\rho(y', z);$$

- the Datalog rule ρ_{DL} as

$$\bigwedge_{y' \in \bar{y} \cup \{w\}} \bigwedge_{z \in \bar{z}} B_{y',z}^\rho(y', z) \rightarrow H(\bar{y}, \bar{z}).$$

Given a rule set $\mathcal{S}$ let $\nabla(\mathcal{S})$ be the rule set containing ρ_{init} , $\rho_{\exists}$ and ρ_{DL} for all rules in $\mathcal{S}$. Now let $\mathcal{R}' = \nabla(\mathcal{R})$, and let $\mathbb{S}'$ contain all predicates that appear in $\mathcal{R}'$. We prove that $\mathcal{R}'$ and $\mathbb{S}'$ form a counterexample to Lemma 23.

Main Properties of Streamlining. As the rules of $\nabla(\mathcal{S})$ simply introduce intermediate steps to the chase — note there is no room for unwanted interplay between the rules — we have the following:

Lemma 24. For every rule signature $\mathbb{S}$, rule set $\mathcal{S}$ and instance $\mathcal{J}$ over $\mathbb{S}$, we have that $Ch(\mathcal{J}, \mathcal{S})$ is homomorphically equivalent to $Ch(\mathcal{J}, \nabla(\mathcal{S}))$ when restricted to $\mathbb{S}$.

PROOF. The proof can be found in Appendix A. □

Lemma 25. Given a rule set $\mathcal{S}$ we have that $\nabla(\mathcal{S})$ is forward-existential and predicate-unique. Moreover, if $\mathcal{S}$ is UCQ-rewritable then $\nabla(\mathcal{S})$ is UCQ-rewritable as well.

PROOF. The proof can be found in Appendix B □

Falsifying the Consequence: (⚡). From Lemma 24 we have that the chase of $\{\top\}$ and $\mathcal{R}$ and the chase of $\{\top\}$ and $\mathcal{R}'$ are homomorphically equivalent when restricted to signature $\mathbb{S}$. Therefore, if the consequence (⚡) does not hold then (⚡) does not hold as well.

Proving the Assumption of Lemma 23. Given $\mathcal{R}$ is UCQ-rewritable we have that $\mathcal{R}'$ is as required due to Lemma 25.

4.4 Rewriting Bodies

In this section we will use the quickness property from [26], inspired from [25].

Definition 26 (Quick). A rule set $\mathcal{R}'$ is *quick* iff for every instance I and every atom β of $Ch(I, \mathcal{R}')$ if all frontier terms of β appear in $\text{adom}(I)$ then $\beta \in Ch_1(I, \mathcal{R}')$.

Definition 27 (Regal). A rule set $\mathcal{R}$ over binary signature is *regal* iff it is UCQ-rewritable, quick, forward-existential, and predicate-unique.

We can now state Theorem 28, which is the statement that we prove in Section 5.

Theorem 28. For every binary signature $\mathbb{S}$ and every regal rule set $\mathcal{R}$ over $\mathbb{S}$ we have:

$$Ch(\mathcal{R}) \models \text{TOURNAMENTS}_E \Rightarrow Ch(\mathcal{R}) \models \text{LOOP}_E. \quad (\heartsuit)$$

We now show that Lemma 23 is equivalent to Theorem 28. The $(\Rightarrow)$ direction is trivial, we show $(\Leftarrow)$ via the contraposition. Let $\mathbb{S}$ be a signature and $\mathcal{R}$ be a rule set that together form a counterexample to Lemma 23.

Rule Set Surgery. Following [26] consider the next definition:

Definition 29 (Body Rewriting). Given a rule set $\mathcal{S}$ and an existential rule $\rho \in \mathcal{S}$ of the form $B(\bar{x}, \bar{y}) \rightarrow \exists \bar{z} H(\bar{y}, \bar{z})$, let $\text{rew}(\rho, \mathcal{S})$ be the rule set:

$$\left\{ q(\bar{x}', \bar{y}') \rightarrow \exists \bar{z} H(\bar{y}', \bar{z}) \mid \exists \bar{x} q(\bar{x}', \bar{y}') \in \text{rew}(\exists \bar{x} B(\bar{x}, \bar{y}), \mathcal{S}) \right\}.$$

Finally, let $\text{rew}(\mathcal{S}) = \mathcal{S} \cup \bigcup_{\rho \in \mathcal{S}} \text{rew}(\rho, \mathcal{S})$.

Let $\mathcal{R}'$ be $\text{rew}(\mathcal{R})$. We now show that $\mathcal{R}'$ and $\mathbb{S}$ form a counterexample to Theorem 28.

Main Properties of Body Rewriting. We take the following (Lemma 42, [26]) as our first property of the surgery:

Lemma 30 ([26]). For any bdd rule set $\mathcal{S}$ and any instance $\mathcal{I}$ we have $Ch(\mathcal{I}, \mathcal{S}) \leftrightarrow Ch(\mathcal{I}, \text{rew}(\mathcal{S}))$.

As an easy consequence of Lemma 30, we have the following.

Lemma 31. The rew rule set surgery preserves UCQ-rewritable, predicate-unique, and forward-existential properties of rule sets.

PROOF. Let $\mathcal{S}$ be a rule set that is:

- *UCQ-rewritable:* As chases of $\mathcal{S}$ and $\text{rew}(\mathcal{S})$ are homomorphically equivalent (Lemma 30), the rewritings against $\mathcal{S}$ can simply serve as rewritings against $\text{rew}(\mathcal{S})$.
- *Predicate-unique or forward-existential:* Note that the heads of rules of $\text{rew}(\mathcal{S})$ are unchanged – except potential identification of frontier variables – when compared to $\mathcal{S}$. Therefore, as $\mathcal{S}$ is predicate-unique and forward-existential we have that $\text{rew}(\mathcal{S})$ is as well. $\square$

Moreover, we take the following (Lemma 28, [26])

Lemma 32 ([26]). For any UCQ-rewritable rule set $\mathcal{S}$, $\text{rew}(\mathcal{S})$ is quick.

4.4.1 Main Properties of Regality. In this section we note all relevant properties of regal rule sets that we use in the following section. Let $\mathcal{S}$ be a regal rule set and $\mathcal{S}^{DL}$ and $\mathcal{S}^\exists$ be its subsets of Datalog rules and non-Datalog rules, respectively. We start by noticing that, due to the quickness of $\mathcal{S}$, $Ch(\mathcal{S})$ can be obtained by applying only Datalog rules on top of $Ch(\mathcal{S}^\exists)$.

Lemma 33. *$Ch(\mathcal{S})$ and $Ch(Ch(\mathcal{S}^\exists), \mathcal{S}^{DL})$ are homomorphically equivalent.*

PROOF. The existence of a homomorphism from $Ch(Ch(\mathcal{S}^\exists), \mathcal{S}^{DL})$ to $Ch(\mathcal{S})$ is clear, as there is a homomorphism from $Ch(\mathcal{S}^\exists)$ to $Ch(\mathcal{S})$ (because $\mathcal{S}^\exists \subseteq \mathcal{S}$), and $\mathcal{S}^{DL} \subseteq \mathcal{S}$. Conversely, we prove by induction on i the existence of h_i (extending h_{i-1} if $i > 0$) from the terms of $Ch_i(\mathcal{S})$ to the terms of $Ch_i(\mathcal{S}^\exists)$ such that h_i is a homomorphism from $Ch_i(\mathcal{S})$ to $Ch(Ch_i(\mathcal{S}^\exists), \mathcal{S}^{DL})$. For $i = 0$, the result holds trivially. If the result holds for $i \geq 0$, let us consider a term t that belongs to $Ch_{i+1}(\mathcal{S})$ but not to $Ch_i(\mathcal{S})$. It must have been created by a non-Datalog rule ρ whose body maps by h to $Ch_i(\mathcal{S})$. Thus $h_i \circ h$ is a homomorphism from the body of ρ to $Ch(Ch_i(\mathcal{S}^\exists), \mathcal{S}^{DL})$, and all frontier terms of the created atoms belong to $Ch_i(\mathcal{S}^\exists)$. By quickness of $\mathcal{S}^{DL} \cup \mathcal{S}^\exists$, these atoms can be created by a single application of a rule ρ , which must belong to $\mathcal{S}^\exists$, as it is non-Datalog. We extend h_i by mapping t to the corresponding term created by that application of ρ . Regarding Datalog rule applications, let $\langle \rho, h \rangle$ be a trigger applied at step $i + 1$. $\pi(\text{body}(\rho)) \subseteq Ch_i(\mathcal{S})$, hence $\langle \rho, h_i \circ \pi \rangle$ is a trigger applicable on $Ch(Ch_i(\mathcal{S}^\exists), \mathcal{S}^{DL})$, and its result thus belongs to $Ch(Ch_i(\mathcal{S}^\exists), \mathcal{S}^{DL})$. $\square$

However, $Ch(\mathcal{S}^\exists)$ has a much nicer structure than $Ch(\mathcal{S})$, that we will exploit shortly. To describe this structure, we associate with each term a timestamp.

Definition 34. Given a term t of $Ch(\mathcal{S}^\exists)$ we define the *timestamp* $TS(t)$ of t to be the smallest integer i such that $t \in \text{adom}(Ch_i(\mathcal{S}^\exists))$. Given a set of terms T of $Ch(\mathcal{S}^\exists)$, we define $TS_m(T)$ as the multiset $\{TS(t) \mid t \in T\}_m$.

Given that the signature of $\mathcal{S}$ is binary, we can see instances (and queries) as directed graphs.

Observation 35. *$Ch(\mathcal{S}^\exists)$ is a directed acyclic graph.*

PROOF. As $\mathcal{S}$ is forward-existential (and the database does not contain any terms), for all atoms $A(t, u)$ in $Ch(\mathcal{S}^\exists)$, we have $TS(t) < TS(u)$. Thus, if some term t is part of a cycle, we get that $TS(t) < TS(t)$, which is a contradiction. $\square$

Falsifying the Consequence: ($\bowtie$). From Lemma 30 we have that the chase of $\{\top\}$ and $\mathcal{R}$ and the chase of $\{\top\}$ and $\mathcal{R}'$ are homomorphically equivalent. Therefore, if the consequence ($\bowtie$) does not hold then ($\bowtie$) does not hold as well.

Proving the Assumption of Theorem 28. Knowing that rew preserves UCQ-rewritable, predicate-unique, and forward-existential properties of rule sets (Lemma 31) and that $\mathcal{R}'$ is quick (Lemma 32), we know that $\mathcal{R}'$ is regal.

5 Proving Theorem 28

Let us fix a regal rule set $\mathcal{R}_{\bowtie}$. Let $\mathcal{R}_{\bowtie}^\exists$ define its non-Datalog part and $\mathcal{R}_{\bowtie}^{DL}$ denote the set of Datalog rules of $\mathcal{R}_{\bowtie}$. We shall prove Theorem 28 by contradiction. Assume TOURNAMENTS_E holds in $Ch(\mathcal{R}_{\bowtie})$ but LOOP_E does not. By Lemma 33, TOURNAMENTS_E also holds in $Ch(Ch(\mathcal{R}_{\bowtie}^\exists), \mathcal{R}_{\bowtie}^{DL})$, but LOOP_E does not. Let K_n , for $n \in \mathbb{N}$, be an E-tournament of size n in $Ch(Ch(\mathcal{R}_{\bowtie}^\exists), \mathcal{R}_{\bowtie}^{DL})$.

The idea behind the proof is as follows. Due to the regality of $\mathcal{R}_{\bowtie}$, for each $E(s, t)$ of K_n , the injective rewriting $Q_{\bowtie}$ of E against $\mathcal{R}_{\bowtie}$ holds for s and t in $Ch(\mathcal{R}_{\bowtie}^\exists)$. In Section 5.1 we prove that arbitrary large tournaments can be witnessed by a single disjunct of $Q_{\bowtie}$ of a specific shape (called a *valley query*). Finally, in Section 5.2, we prove that a single valley query can only generate cliques of size 3 without creating loops, which will conclude the proof.

5.1 Valley Queries

Let $Q_{\sqcup}$ be the injective rewriting of $E(x, y)$ against $\mathcal{R}_{\sqcup}$, which exists due to Proposition 6 and because $\mathcal{R}_{\sqcup}$ is UCQ-rewritable.

Definition 36. Given s and t such that $E(s, t) \in Ch(Ch(\mathcal{R}_{\sqcup}^{\exists}), \mathcal{R}_{\sqcup}^{DL})$, we define the set of *witnesses* of $E(s, t)$ in $Ch(\mathcal{R}_{\sqcup}^{\exists})$ as $\mathcal{W}(s, t) = \{ q \in Q_{\sqcup} \mid Ch(\mathcal{R}_{\sqcup}^{\exists}) \models_{\text{inj}} q(s, t) \}$.

Observation 37. For all s and t such that $E(s, t) \in Ch(Ch(\mathcal{R}_{\sqcup}^{\exists}), \mathcal{R}_{\sqcup}^{DL})$, the set $\mathcal{W}(s, t)$ is not empty.

PROOF. Follows from Proposition 6, and the fact that $Q_{\sqcup}$ is an injective rewriting of $E(x, y)$ against $\mathcal{R}_{\sqcup}$, which contains $\mathcal{R}_{\sqcup}^{DL}$. $\square$

The acyclicity of $Ch(\mathcal{R}_{\sqcup}^{\exists})$ allows us to find queries of a special shape within $\mathcal{W}(s, t)$.

Definition 38. Given an instance $\mathcal{I}$ (or a CQ) that is a directed acyclic graph, we define a strict partial order $<_{\mathcal{I}}$ over terms of $\mathcal{I}$ as follows: given two terms s, t of $\mathcal{I}$ we have $s <_{\mathcal{I}} t$ iff there exists a directed path from s to t . We define $\leq_{\mathcal{I}}$ as the reflexive closure of $<_{\mathcal{I}}$. We use $<_{\sqcup}$ and $\leq_{\sqcup}$ as shorthands for $<_{Ch(\mathcal{R}_{\sqcup}^{\exists})}$ and $\leq_{Ch(\mathcal{R}_{\sqcup}^{\exists})}$ respectively.

Definition 39. A *valley query* $q(x, y)$ is a binary conjunctive query that is a DAG and whose only $<_q$ -maximal variables are x and y .

Lemma 40 (Peak Removing Argument). For all s and t such that $E(s, t) \in Ch(Ch(\mathcal{R}_{\sqcup}^{\exists}), \mathcal{R}_{\sqcup}^{DL})$, the set $\mathcal{W}(s, t)$ contains a valley query.

PROOF. Assume towards contradiction that this is not the case. Consider a non-valley query $q \in \mathcal{W}(s, t)$ and an injective homomorphism h from q to $Ch(\mathcal{R}_{\sqcup}^{\exists})$ such that $h(x) = s$, $h(y) = t$, and $\text{TS}_m(h(q))$ is $\leq_{\text{lex}}$ -minimal among all such queries and homomorphisms. Such a pair exists by Observation 37 and the fact that $\leq_{\text{lex}}$ is well-founded over multisets of size at most $\max\{|q'| \mid q' \in Q_{\sqcup}\}$, by Lemma 8.

Then, consider a variable $z \in \text{Evars}(q)$ that is $\leq_q$ -maximal. Such a variable exists as q is not a valley query. Since h is injective, $h(z)$ does not have any outgoing edge in $h(q)$. Denote the set of atoms containing the variable z in q with Z , and let $\langle \rho, \pi \rangle$ be the trigger that created $h(z)$. Since $\mathcal{R}_{\sqcup}$ is forward-existential and z is $\leq_q$ -maximal, then $h(Z) \subseteq \pi(\text{head}(\rho))$. We thus define $I = h(q) \setminus h(Z) \cup \pi(\text{body}(\rho))$. Note that by this definition, $I \subseteq Ch(\mathcal{R}_{\sqcup}^{\exists})$.

The trigger $\langle \rho, \pi \rangle$ is applicable on I and yields $\pi(\text{head}(\rho))$ which contains $h(Z)$, up to the renaming of $h(z)$. Thus, since z is existentially quantified in q , we have $Ch(I, \mathcal{R}_{\sqcup}) \models q(s, t)$. Thus, $Ch(I, \mathcal{R}_{\sqcup}) \models E(s, t)$, which by the fact that $Q_{\sqcup}(x, y)$ is an injective rewriting of $E(x, y)$ entails that $I \models_{\text{inj}} q'(s, t)$ for some $q' \in Q_{\sqcup}$. Consider the injective homomorphism h' that witnesses this fact.

First note that as $h'(q') \subseteq I$, we have $\text{TS}_m(h'(q')) \leq_{\text{lex}} \text{TS}_m(I)$. Then, since $h(z)$ is created by the trigger $\langle \rho, \pi \rangle$, by definition of the chase, we have $\text{TS}(x) < \text{TS}(h(z))$ for all $x \in \pi(\text{body}(\rho))$. Thus, $\text{TS}_m(I) <_{\text{lex}} \text{TS}_m(h(q))$, which entails that $\text{TS}_m(h'(q')) <_{\text{lex}} \text{TS}_m(h(q))$, contradicting the minimality of $\langle q, h \rangle$. $\square$

Proposition 41. If there exists E-tournaments of arbitrary size in $Ch(\mathcal{R}_{\sqcup})$, there exists tournaments of arbitrary size defined by a single valley query over $Ch(\mathcal{R}_{\sqcup}^{\exists})$.

PROOF. First, as noticed earlier, by Lemma 33, if there are E-tournaments of arbitrary size in $Ch(\mathcal{R}_{\sqcup})$, there are E-tournaments of arbitrary size in $Ch(Ch(\mathcal{R}_{\sqcup}^{\exists}), \mathcal{R}_{\sqcup}^{DL})$. We then make use of Ramsey's theorem. For a given E-tournament in $Ch(Ch(\mathcal{R}_{\sqcup}^{\exists}), \mathcal{R}_{\sqcup}^{DL})$, we color each edge $E(s, t)$ by an arbitrary valley query in $\mathcal{W}(s, t)$. This possible due to Lemma 40. Ramsey theorem ensures that for any n , if the original tournament is larger than $R(n, \dots, n)$ (with as many arguments as elements of $Q_{\sqcup}$), there is a tournament of size n defined by a single valley query. $\square$

5.2 Tournaments Defined by a Single Valley Query

We now analyze the cliques that can be defined by a single valley query. Our main tool here is that the image of a valley query in $Ch(\mathcal{R}_{\boxminus}^{\exists})$ is entirely determined by the image of its answer variables. This strong property holds for valley queries because $\mathcal{R}_{\boxminus}$ is both predicate-unique and forward-existential.

Lemma 42. *Let $q(x, \bar{y})$ be a CQ such that $y <_q x$ for every $y \in \bar{y}$ then the set*

$$\{ \langle s, \bar{t} \rangle \in \text{adom}(Ch(\mathcal{R}_{\boxminus}^{\exists})) \times \text{adom}(Ch(\mathcal{R}_{\boxminus}^{\exists}))^{|\bar{y}|} \mid Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q(s, \bar{t}) \}$$

is a function.

PROOF. We first show this result in the case where $|\bar{y}| = 1$ (and y is the only variable in $\bar{y}$), and $q(x, y)$ is a path from y to x . We prove by induction on the length of the path from y to x in $q(x, y)$ that for all $s \in \text{adom}(Ch(\mathcal{R}_{\boxminus}^{\exists}))$, there is at most one $t \in \text{adom}(Ch(\mathcal{R}_{\boxminus}^{\exists}))$ such that $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q(s, t)$. The case where this path has length 0 being trivial, we assume it has at least size one. Thus, $q(x, y) = q'(x, z) \wedge A(y, z)$, where A is a predicate in $\mathbb{S}$, and $q'(x, z)$ is a path query of length $|q(x, y)| - 1$. Thus, by induction hypothesis, for all $s \in \text{adom}(Ch(\mathcal{R}_{\boxminus}^{\exists}))$, there is at most one $u \in \text{adom}(Ch(\mathcal{R}_{\boxminus}^{\exists}))$ such that $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q'(s, u)$.

Then, assume that there are some terms s, t and t' such that $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q(s, t) \wedge q(s, t')$. Then, there is a unique u such that $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q'(s, u)$, so the atoms $A(t, u)$ and $A(t', u)$ are both present in $Ch(\mathcal{R}_{\boxminus}^{\exists})$. Let $\langle \rho, h \rangle$ be the trigger that introduces u in $Ch(\mathcal{R}_{\boxminus}^{\exists})$. Since $\mathcal{R}_{\boxminus}^{\exists}$ is forward-existential, t and t' are both images of frontier terms of ρ . However, since $\mathcal{R}_{\boxminus}^{\exists}$ is frontier-functional, there is at most one A -atom in $\text{head}(\rho)$. Thus, it must be that $t = t'$, which concludes the induction.

The full result then follows from this case. Indeed, for all $y \in \bar{y}$, consider a path from y to x in $q(x, \bar{y})$. This path defines a path query $q'_y(x, y)$, so for all $s \in \text{adom}(Ch(\mathcal{R}_{\boxminus}^{\exists}))$, there is at most one $t \in \text{adom}(Ch(\mathcal{R}_{\boxminus}^{\exists}))$ such that $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q'_y(s, t)$. We then get the full result from the fact that for all s and $\bar{t}$, $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q(s, \bar{t})$ implies $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q'_y(s, t)$. $\square$

Proposition 43. *If a valley query defines an E-tournament of size 4, it also defines an E-loop.*

PROOF. Let q be a valley query.

- If $q(x, y)$ is disconnected, then $q(x, y) = q_1(x) \wedge q_2(y) \wedge q_3$, with q_i and q_j having disjoint sets of variables if $i \neq j$. As there is an edge between k_1 and k_2 , and they are symmetric, we can assume without loss of generality that $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q_1(k_1) \wedge q_2(k_2)$. There is also an edge between k_3 and k_4 , so we assume (again without loss of generality) that $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q_1(k_3) \wedge q_2(k_4)$. Finally, there is an edge between k_1 and k_3 , so either $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q_2(k_1)$, or $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q_2(k_3)$. In both cases, $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models \exists u q_1(u) \wedge q_2(u) \wedge q_3$, so $Ch(\mathcal{R}_{\boxminus}^{\exists}) \models \exists u E(u, u)$, hence an E-loop is defined. We thus assume in all other cases that q is connected.
- If $q(x, y)$ contains a single $\leq_q$ -maximal vertex, say x for instance (as the case where y is $\leq_q$ -maximal is symmetric). Thus, $y <_{\boxminus} x$, so by Lemma 42, the set $\{ \langle s, t \rangle \in \text{adom}(Ch(\mathcal{R}_{\boxminus}^{\exists}))^2 \mid Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q(s, t) \}$ is a function. Thus, every vertex in K has out-degree at most one, which contradicts the fact that K is a tournament of size 4.
- If both x and y are $\leq_q$ -maximal in $q(x, y)$, then let $\bar{v}$ be the tuple of existential variables of q that are smaller than both x and y (for $\leq_q$). Then, $q(x, y) = \exists \bar{v} q_x(x, \bar{v}) \wedge q_y(\bar{v}, y)$, where q_x (resp. q_y) contains all the variables smaller than x (resp. y). Then, by Lemma 42, the sets $\{ \langle x, \bar{v} \rangle \mid Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q_x(x, \bar{v}) \}$ and $\{ \langle y, \bar{v} \rangle \mid Ch(\mathcal{R}_{\boxminus}^{\exists}) \models q_y(\bar{v}, y) \}$ are functions, that we denote with f_x and f_y , respectively. Since K is a tournament over 4 elements, it contains, up to renaming of elements, the following set: $\{ E(k_1, k_2), E(k_1, k_3), E(k_2, k_3) \}$.

We then show that $Ch(\mathcal{R}_{\sqcup}) \models E(k_2, k_2)$. First, observe that if $q(s, t) \in Ch(\mathcal{R}_{\sqcup}^{\exists})$, then $f_x(s) = f_y(t)$. Thus, we get the three following equalities: $f_x(k_1) = f_y(k_2)$, $f_x(k_1) = f_y(k_3)$ and $f_x(k_2) = f_y(k_3)$. Thus, by composing these equalities, we get that $f_x(k_2) = f_y(k_2)$, so $Ch(\mathcal{R}_{\sqcup}^{\exists}) \models q_x(k_2, f_x(k_2)) \wedge q_y(f_x(k_2), k_2)$. Thus, $Ch(\mathcal{R}_{\sqcup}^{\exists}) \models q(k_2, k_2)$, and $Ch(\mathcal{R}_{\sqcup}) \models E(k_2, k_2)$, which concludes the proof. $\square$

6 Discussion

Tournament Definition

While the main result focuses on tournaments over a fixed relation E , our results trivially extend to any relation definable by a binary UCQ. Consider an arbitrary binary UCQ $Q(x, y) = \bigvee_{i=1}^k q_i(x, y)$, where each q_i is a CQ. One can simply add, for each i , the following rule to the considered rule set:

$$q_i(x, y) \rightarrow E(x, y).$$

This addition does not affect the UCQ-rewritability of the rule set when E is a fresh predicate symbol.

Arbitrary Colorability

A natural next step in this line of research is to establish that with UCQ-rewritable rule sets, one cannot define structures of arbitrarily high chromatic number without entailing the LOOP_E query:

Conjecture 44. *For every signature $\mathbb{S}$, every UCQ-rewritable rule set $\mathcal{R}$ over $\mathbb{S}$, and every instance I , we have:*

$$Ch(I, \mathcal{R})|_E \text{ cannot be colored with a finite number of colors} \Rightarrow Ch(I, \mathcal{R}) \models \text{LOOP}_E.$$

We believe that this conjecture, if proven, would constitute an elegant model-theoretic property of UCQ-rewritable rule sets and represent a significant step toward proving the $(\text{bdd} \Rightarrow \text{fc})$ conjecture. Let us elaborate. Similar to the results presented in this paper, if Conjecture 44 holds, it would eliminate a substantial portion of the potential counterexample space: any structure that cannot be colored with a finite number of colors cannot be homomorphically embedded into a finite structure that does not entail LOOP_E .

Our hope is that the tools developed throughout this paper will serve as a foundation for the proof of Conjecture 44. However, we note that such a proof would not be a straightforward extension of the techniques in Section 5. The current proof aims at showing existence of four-tournament witnessed by a single valley query. We know, however, that there exists structures of arbitrarily high chromatic numbers that do not contain the four-clique:

Theorem 45 (Erdős [12]). *There exist graphs with arbitrarily high girth and chromatic number.*

Tournament Size Bounds

Of separate, but related interest, is the study of the following question:

Question 46. *Given a UCQ-rewritable rule set $\mathcal{R}$ such that $Ch(\{\top\}, \mathcal{R}) \not\models \text{LOOP}_E$, what is the maximal n such that $Ch(I, \mathcal{R})$ admits a tournament of size n ?*

A careful reader will note that an upper bound on the maximal size of a tournament of the form $N(4, \dots, 4)$ (with $|Q_{\sqcup}|$ arguments) can be extracted from the proof presented in previous sections — if a tournament of at least this size exists in the chase one can employ the machinery of Section 5.2 to show that LOOP_E is entailed.

We leave Question 46 as an open question, as it ties together various significant problems regarding existential rules, particularly in the UCQ-rewritable fragment.

Acknowledgments

Piotr Ostropolski-Nalewaja was supported by grant 2022/45/B/ST6/00457 from the Polish National Science Centre (NCN).

References

- [1] Serge Abiteboul, Richard Hull, and Victor Vianu. 1995. *Foundations of Databases*. Addison-Wesley. <http://webdam.inria.fr/Alice/>
- [2] Jean-François Baget, Michel Leclère, Marie-Laure Mugnier, and Eric Salvat. 2011. On rules with existential variables: Walking the decidability line. *Artificial Intelligence* 175, 9 (2011), 1620–1654. doi:10.1016/j.artint.2011.03.002
- [3] Pablo Barceló, Gerald Berger, Carsten Lutz, and Andreas Pieris. 2018. First-Order Rewritability of Frontier-Guarded Ontology-Mediated Queries. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*. 1707–1713. doi:10.24963/ijcai.2018/236
- [4] Vince Bárány, Georg Gottlob, and Martin Otto. 2010. Querying the Guarded Fragment. In *2010 25th Annual IEEE Symposium on Logic in Computer Science*. 1–10. doi:10.1109/LICS.2010.26
- [5] Andrea Cali, Georg Gottlob, and Michael Kifer. 2008. Taming the Infinite Chase: Query Answering under Expressive Relational Constraints. In *Proceedings of the Eleventh International Conference on Principles of Knowledge Representation and Reasoning (Sydney, Australia) (KR'08)*. AAAI Press, 70–80.
- [6] Andrea Cali, Georg Gottlob, and Michael Kifer. 2013. Taming the infinite chase: query answering under expressive relational constraints. *J. Artif. Int. Res.* 48, 1 (Oct. 2013), 115–174.
- [7] Andrea Cali, Georg Gottlob, and Andreas Pieris. 2010. Advanced processing for ontological queries. *Proceedings of the VLDB Endowment* 3 (2010). Issue 1. doi:10.14778/1920841.1920912
- [8] David Carral, Irina Dragoste, and Markus Krötzsch. 2017. Restricted Chase (Non)Termination for Existential Rules with Disjunctions. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017), Australia*, Carles Sierra (Ed.). ijcai.org, 922–928.
- [9] Cristina Civili and Riccardo Rosati. 2015. On the first-order rewritability of conjunctive queries over binary guarded existential rules (extended abstract).
- [10] Bernardo Cuenca Grau, Ian Horrocks, Markus Krötzsch, Clemens Kupke, Despoina Magka, Boris Motik, and Zhe Wang. 2013. Acyclicity Notions for Existential Rules and Their Application to Query Answering in Ontologies. *Journal of Artificial Intelligence Research (JAIR)* 47 (2013), 741–808.
- [11] Alin Deutsch, Alan Nash, and Jeff Remmel. 2008. The chase revisited. *Proceedings of the ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*. doi:10.1145/1376916.1376938
- [12] P. Erdős. 1959. Graph Theory and Probability. *Canadian Journal of Mathematics* 11 (1959), 34–38. doi:10.4153/CJM-1959-003-9
- [13] Ronald Fagin, Phokion G. Kolaitis, Renée J. Miller, and Lucian Popa. 2003. Data exchange: Semantics and query answering. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 2572 (2003). doi:10.1007/3-540-36285-1_14
- [14] Thomas Feller, Tim S. Lyon, Piotr Ostropolski-Nalewaja, and Sebastian Rudolph. 2023. Finite-Cliqueswidth Sets of Existential Rules: Toward a General Criterion for Decidable yet Highly Expressive Querying. In *26th International Conference on Database Theory (ICDT 2023) (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 255)*, Floris Geerts and Brecht Vandevoort (Eds.). Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 18:1–18:18. doi:10.4230/LIPIcs.ICDT.2023.18
- [15] Haim Gaifman, Harry Mairson, Yehoshua Sagiv, and Moshe Y. Vardi. 1993. Undecidable Optimization Problems for Database Logic Programs. *J. ACM* 40, 3 (July 1993), 683–713. doi:10.1145/174130.174142
- [16] Lukas Gerlach and David Carral. 2023. General Acyclicity and Cyclicity Notions for the Disjunctive Skolem Chase. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence, 2023, USA*, Brian Williams, Yiling Chen, and Jennifer Neville (Eds.). AAAI Press, 6372–6379. doi:10.1609/aaai.v37i5.25784
- [17] Tomasz Gogacz and Jerzy Marcinkowski. 2013. On the BDD/FC conjecture. In *Proceedings of the 32nd ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems (New York, New York, USA) (PODS '13)*. Association for Computing Machinery, New York, NY, USA, 127–138. doi:10.1145/2463664.2463668
- [18] Tomasz Gogacz and Jerzy Marcinkowski. 2017. Converging to the chase – A tool for finite controllability. *J. Comput. System Sci.* 83, 1 (2017), 180–206. doi:10.1016/j.jcss.2016.08.001
- [19] Georg Gottlob. 2009. Datalog+/-: A Unified Approach to Ontologies and Integrity Constraints. In *Proceedings of the 17th Italian Symposium on Advanced Database Systems, (SEBD'09)*, Valeria De Antonellis, Silvana Castano, Barbara Catania, and Giovanna Guerrini (Eds.). Edizioni Seneca, 5–6.
- [20] Georg Gottlob, Stanislav Kikot, Roman Kontchakov, Vladimir Podolskii, Thomas Schwentick, and Michael Zakhar'yashev. 2014. The price of query rewriting in ontology-based data access. *Artificial Intelligence* 213 (2014), 42–59.

doi:10.1016/j.artint.2014.04.004

- [21] D.S. Johnson and A. Klug. 1984. Testing containment of conjunctive queries under functional and inclusion dependencies. *J. Comput. System Sci.* 28, 1 (1984), 167–189. doi:10.1016/0022-0000(84)90081-3
- [22] Mélanie König, Michel Leclère, Marie-Laure Mugnier, and Michaël Thomazo. 2015. Sound, complete and minimal UCQ-rewriting for existential rules. *Semantic Web* 6, 5 (2015), 451–475. doi:10.3233/SW-140153
- [23] Markus Krötzsch and Sebastian Rudolph. 2011. Extending decidable existential rules by joining acyclicity and guardedness. *IJCAI International Joint Conference on Artificial Intelligence*. doi:10.5591/978-1-57735-516-8/IJCAI11-166
- [24] David Maier, Alberto O. Mendelzon, and Yehoshua Sagiv. 1979. Testing implications of data dependencies. *ACM Transactions on Database Systems (TODS)* 4 (1979). Issue 4. doi:10.1145/320107.320115
- [25] Piotr Ostropolski-Nalewaja, Jerzy Marcinkowski, David Carral, and Sebastian Rudolph. 2022. A Journey to the Frontiers of Query Rewritability. In *PODS '22: International Conference on Management of Data, Philadelphia, PA, USA, June 12 - 17, 2022*, Leonid Libkin and Pablo Barceló (Eds.). ACM, 359–367. doi:10.1145/3517804.3524163
- [26] Piotr Ostropolski-Nalewaja and Sebastian Rudolph. 2024. The Sticky Path to Expressive Querying: Decidability of Navigational Queries under Existential Rules. In *Proceedings of the 21st International Conference on Principles of Knowledge Representation and Reasoning*. 574–584. doi:10.24963/kr.2024/54
- [27] Riccardo Rosati. 2011. On the finite controllability of conjunctive query answering in databases under open-world assumption. *J. Comput. Syst. Sci.* 77, 3 (2011), 572–594. doi:10.1016/J.JCSS.2010.04.011
- [28] Eric Salvat and Marie-Laure Mugnier. 1996. Sound and Complete Forward and Backward Chainings of Graph Rules. In *Proceedings of the 4th International Conference on Conceptual Structures (ICCS'96) (LNCS, Vol. 1115)*, Peter W. Eklund, Gerard Ellis, and Graham Mann (Eds.). Springer, 248–262. doi:10.1007/3-540-61534-2_16

A Proof of Lemma 24

Let $\nabla_{\text{init}}(\mathcal{S})$ be the subset of $\nabla(\mathcal{S})$ containing only rules of form ρ_{init} , let $\nabla_{\exists}(\mathcal{S})$ and $\nabla_{\text{DL}}(\mathcal{S})$ be defined in the analogous way.

Observation 47. *For any rule set $\mathcal{S}$, the signature of $\nabla(\mathcal{S})$ can be partitioned into three disjoint signatures $\mathbb{S}$, $\mathbb{S}_A$ and $\mathbb{S}_B$, such that*

- $\mathbb{S}$ is the signature of heads of rules in $\nabla_{\text{DL}}(\mathcal{S})$ and bodies of rules in $\nabla_{\text{init}}(\mathcal{S})$,
- $\mathbb{S}_A$ is the signature of heads of rules in $\nabla_{\text{init}}(\mathcal{S})$ and bodies of rules in $\nabla_{\exists}(\mathcal{S})$,
- $\mathbb{S}_B$ is the signature of heads of rules in $\nabla_{\exists}(\mathcal{S})$ and bodies of rules in $\nabla_{\text{DL}}(\mathcal{S})$,

Lemma 48. *For every rule set $\mathcal{S}$, every instance $\mathcal{J}$ and every integer i , there is a homomorphism σ_i^∇ from $Ch_i(\mathcal{J}, \mathcal{S})$ to $Ch_{3i}(\mathcal{J}, \nabla(\mathcal{S}))$.*

PROOF. We proceed by induction on i . For the base case, we have $Ch_0(\mathcal{J}, \mathcal{S}) = Ch_0(\mathcal{J}, \nabla(\mathcal{S})) = \mathcal{J}$, so σ_0^∇ is the identity. Now, for the induction step, assume σ_i^∇ constructed, and consider a trigger $\tau = \langle \rho, h \rangle$ over $Ch_i(\mathcal{J}, \mathcal{S})$. Then, $\sigma_i^\nabla(h(\text{body}(\rho_{\text{init}}))) = \sigma_i^\nabla(h(\text{body}(\rho)))$ (as these bodies are equal), so the trigger $\langle \rho_{\text{init}}, \sigma_i^\nabla \circ h \rangle$ is a trigger over $Ch_{3i}(\mathcal{J}, \nabla(\mathcal{S}))$. Thus, the trigger $\langle \rho_{\exists}, \sigma_i^\nabla \circ h \rangle$ is a trigger over $Ch_{3i+1}(\mathcal{J}, \nabla(\mathcal{S}))$, as $\rho_{\exists}$'s body is ρ_{init} 's head. The same applies for $\langle \rho_{\text{DL}}, \sigma_i^\nabla \circ h \rangle$ at step $3i + 2$. Thus, there is an isomorphism between $\sigma_i^\nabla(\text{output}(\tau))$ and $\text{output}(\langle \rho_{\text{DL}}, \sigma_i^\nabla \circ h \rangle)$, so we can extend σ_i^∇ using these isomorphisms for all the triggers applied at step $i + 1$ in $Ch_{i+1}(\mathcal{J}, \mathcal{S})$, constructing σ_{i+1}^∇ as a homomorphism from $Ch_{i+1}(\mathcal{J}, \mathcal{S})$ to $Ch_{3i+3}(\mathcal{J}, \nabla(\mathcal{S}))$. $\square$

We denote the signature of $\mathcal{S}$ with $\mathbb{S}$, and define for every instance $\mathcal{J}$ the restriction $\mathcal{J}_{|\mathbb{S}}$ of $\mathcal{J}$ to the signature $\mathbb{S}$.

Observation 49. *For every instance $\mathcal{J}$, rule set $\mathcal{S}$, rule $\rho \in \mathcal{S}$ and homomorphism h*

- *If $\langle \rho_{\exists}, h \rangle$ is a trigger such that some atom in $h(\text{body}(\rho_{\exists}))$ is mapped to $Ch_1(\mathcal{J}, \nabla_{\text{init}}(\mathcal{S})) \setminus \mathcal{J}$, then there is some trigger $\langle \rho_{\text{init}}, h' \rangle$ for $\mathcal{J}$ such that $\text{output}(\langle \rho_{\text{init}}, h' \rangle) = h(\text{body}(\rho_{\exists}))$.*
- *If $\langle \rho_{\text{DL}}, h \rangle$ is a trigger such that some atom in $h(\text{body}(\rho_{\text{DL}}))$ is mapped to $Ch_1(\mathcal{J}, \nabla_{\exists}(\mathcal{S})) \setminus \mathcal{J}$, then there is some trigger $\langle \rho_{\exists}, h' \rangle$ for $\mathcal{J}$ such that $\text{output}(\langle \rho_{\exists}, h' \rangle) = h(\text{body}(\rho_{\text{DL}}))$.*

PROOF. Notice that all the atoms in the body of these rules feature a term that is an existential variable in the head of the previous rule. Thus, this variable has to be generated by a rule application. Since for all $\rho \in \mathcal{S}$, ρ_{init} 's head is $\rho_{\exists}$'s body, and the same for $\rho_{\exists}$ and ρ_{DL} , we get the result. $\square$

Lemma 50. *For every rule set $\mathcal{S}$ and instance $\mathcal{J}$, let $\mathcal{J}' = Ch_1(Ch_1(\mathcal{J}, \nabla_{\exists}(\mathcal{S})), \nabla_{\text{DL}}(\mathcal{S}))_{|\mathbb{S}}$. For all i , there is a homomorphism π_i^∇ from $Ch_{3i}(\mathcal{J}, \nabla(\mathcal{S}))$ to*

$$\mathcal{R}_i = Ch_1(Ch_1(Ch_i(\mathcal{J}', \mathcal{S}), \nabla_{\text{init}}(\mathcal{S})), \nabla_{\exists}(\mathcal{S})) \cup Ch_1((\mathcal{J} \setminus \mathcal{J}_{|\mathbb{S}}), \nabla_{\exists}(\mathcal{S}))$$

PROOF. We construct π_i^∇ by induction on i . For the base case, $Ch_0(\mathcal{J}, \nabla(\mathcal{S})) = \mathcal{J}$ is included in $\mathcal{R}_0$, as $\mathcal{J}_{|\mathbb{S}} \subseteq \mathcal{J}'$, so π_0^∇ is the identity. We then focus on the inductive case, by assuming that π_i^∇ is constructed. Triggers generating atoms in $Ch_{3i+3}(\mathcal{J}, \nabla(\mathcal{S})) \setminus Ch_{3i}(\mathcal{J}, \nabla(\mathcal{S}))$ can be applied at step $3i + 1$, $3i + 2$ or $3i + 3$, and use a rule in $\nabla_{\text{init}}(\mathcal{S})$, $\nabla_{\exists}(\mathcal{S})$ or $\nabla_{\text{DL}}(\mathcal{S})$. Let $\tau = \langle r, h \rangle$ be such a trigger. To construct π_{i+1}^∇ , we show in these nine cases how $\pi_i^\nabla(\text{output}(\tau))$ can be mapped to $\mathcal{R}_{i+1}$.

Step $3i + 1$ and $r = \rho_{\text{init}}$: The body of these rules is over $\mathbb{S}$, meaning that $\langle r, \pi_i^\nabla \circ h \rangle$ is a trigger over $Ch_i(\mathcal{J}', \mathcal{S})$. Thus, $\pi_i^\nabla(\text{output}(\langle r, h \rangle))$ maps into $\text{output}(\langle r, \pi_i^\nabla \circ h \rangle) \subseteq Ch_1(Ch_i(\mathcal{J}', \mathcal{S}), \nabla_{\text{init}}(\mathcal{S})) \subseteq \mathcal{R}_i$.

Step $3i + 1$ and $r = \rho_{\exists}$: The body of these rules is over $\mathbb{S}_A$, so it can only be mapped to atoms in $\mathcal{J} \setminus \mathcal{J}_{|\mathbb{S}}$ or in $Ch_1(Ch_i(\mathcal{J}', \mathcal{S}), \nabla_{\text{init}}(\mathcal{S}))$. If $\pi_i^\nabla(h(\text{body}(\rho_{\exists})))$ is included in $\mathcal{J}$, then

$\pi_i^\nabla(\text{output}(\langle \rho_\exists, h \rangle))$ can be mapped to $Ch_1((\mathcal{J} \setminus \mathcal{J}_\mathbb{S}), \nabla_\exists(\mathcal{S}))$. Otherwise, by Observation 49, there is some trigger $\langle \rho_{\text{init}}, h' \rangle$ such that $\pi_i^\nabla(h(\text{body}(\rho_\exists))) = \text{output}(\langle \rho_{\text{init}}, h' \rangle)$. Thus, $\pi_i^\nabla(\text{output}(\langle \rho_\exists, h \rangle))$ can be mapped into $\text{output}(\langle \rho_\exists, \pi_i^\nabla \circ h \rangle) \subseteq \mathcal{R}_i$.

Step $3i + 1$ and $r = \rho_{\text{DL}}$: If $\pi_i^\nabla(h(\text{body}(\rho_{\text{DL}}))) \subseteq \mathcal{J} \setminus \mathcal{J}_\mathbb{S}$, then $\pi_i^\nabla(\text{output}(\langle \rho_{\text{DL}}, h \rangle))$ can be mapped into $\text{output}(\langle \rho_{\text{DL}}, \pi_i^\nabla \circ h \rangle) \subseteq \mathcal{J}' \subseteq \mathcal{R}_i$, as rules in $\nabla_{\text{DL}}(\mathcal{S})$ only produce atoms over $\mathbb{S}$. Otherwise, by applying Observation 49 twice, there are some triggers $\langle \rho_\exists, h' \rangle$ over $Ch_1(Ch_i(\mathcal{J}', \mathcal{S}), \nabla_{\text{init}}(\mathcal{S}))$ and $\langle \rho_{\text{init}}, h'' \rangle$ over $Ch_i(\mathcal{J}', \mathcal{S})$ such that $\text{output}(\langle \rho_{\text{init}}, h'' \rangle) = h'(\text{body}(\rho_\exists))$ and $\text{output}(\langle \rho_\exists, h' \rangle) = \pi_i^\nabla(h(\text{body}(\rho_{\text{DL}})))$. Thus, the trigger $\langle \rho, h'' \rangle$ is a trigger over $Ch_i(\mathcal{J}', \mathcal{S})$ such that $\text{output}(\langle \rho, h'' \rangle)$ and $\text{output}(\langle \rho_{\text{DL}}, \pi_i^\nabla \circ h \rangle)$ are isomorphic. Thus, we can extend this isomorphism to map the set $\pi_i^\nabla(\text{output}(\langle \rho_{\text{DL}}, h \rangle))$ into $\text{output}(\langle \rho, h'' \rangle) \subseteq Ch_{i+1}(\mathcal{J}', \mathcal{S}) \subseteq \mathcal{R}_{i+1}$.

Step $3i + 2$ and $r = \rho_{\text{init}}$: Some atom of $h(\text{body}(\rho_{\text{init}}))$ has to be mapped into a newly created atom at step $3i + 1$ by a rule in $\nabla_{\text{DL}}(\mathcal{S})$. For all such trigger $\langle \rho'_{\text{DL}}, h' \rangle$, we can map $\pi_i^\nabla(\text{output}(\langle \rho'_{\text{DL}}, h' \rangle))$ into $Ch_{i+1}(\mathcal{J}', \mathcal{S})$ by the previous case. Thus, by Observation 49, we can map $\pi_i^\nabla(h(\text{body}(\rho_{\text{init}})))$ entirely into $Ch_{i+1}(\mathcal{J}', \mathcal{S})$, so that $\langle \rho_{\text{init}}, \pi_i^\nabla \circ h \rangle$ is a trigger for it, and thus $\pi_i^\nabla(\text{output}(\langle \rho_{\text{init}}, h \rangle))$ can be mapped into $\text{output}(\langle \rho_{\text{init}}, \pi_i^\nabla \circ h \rangle) \subseteq \mathcal{R}_{i+1}$.

Step $3i + 2$ and $r \in \{\rho_\exists, \rho_{\text{DL}}\}$: These cases are treated exactly as step $3i + 1$.

Step $3i + 3$ and $r \in \{\rho_{\text{init}}, \rho_{\text{DL}}\}$: These cases are treated exactly as step $3i + 2$.

Step $3i + 3$ and $r = \rho_\exists$: Some atom of $h(\text{body}(\rho_\exists))$ has to be mapped into a newly created atom at step $3i+2$ by a rule in $\nabla_{\text{init}}(\mathcal{S})$. For all such trigger $\langle \rho'_{\text{init}}, h' \rangle$, we can map $\pi_i^\nabla(\text{output}(\langle \rho'_{\text{init}}, h' \rangle))$ into $Ch_1(Ch_{i+1}(\mathcal{J}', \mathcal{S}), \nabla_{\text{init}}(\mathcal{S}))$ by the previous cases. Thus, by Observation 49, we can map $\pi_i^\nabla(h(\text{body}(\rho_\exists)))$ entirely into $Ch_1(Ch_{i+1}(\mathcal{J}', \mathcal{S}), \nabla_{\text{init}}(\mathcal{S}))$, so that $\langle \rho_\exists, \pi_i^\nabla \circ h \rangle$ is a trigger for it, and thus $\pi_i^\nabla(\text{output}(\langle \rho_\exists, h \rangle))$ can be mapped into $\text{output}(\langle \rho_\exists, \pi_i^\nabla \circ h \rangle) \subseteq \mathcal{R}_{i+1}$. $\square$

Lemma 24 (restated). *For every rule signature $\mathbb{S}$, rule set $\mathcal{S}$ and instance $\mathcal{J}$ over $\mathbb{S}$, we have that $Ch(\mathcal{J}, \mathcal{S})$ is homomorphically equivalent to $Ch(\mathcal{J}, \nabla(\mathcal{S}))$ when restricted to signature of $\mathcal{S}$.*

PROOF. We get this result as a corollary of Lemma 48 and Lemma 50, by noticing that

$$(Ch_1(Ch_1(Ch_i(\mathcal{J}', \mathcal{S}), \nabla_{\text{init}}(\mathcal{S})), \nabla_\exists(\mathcal{S})) \cup (\mathcal{J} \setminus \mathcal{J}_\mathbb{S}))|_\mathbb{S} = Ch_i(\mathcal{J}', \mathcal{S})$$

and $\mathcal{J}' = \mathcal{J}$ when $\mathcal{J}$ is over $\mathbb{S}$. $\square$

B Proof of Lemma 25

Lemma 25 (restated). *Given a rule set $\mathcal{S}$ we have that $\nabla(\mathcal{S})$ is forward-existential and predicate-unique. Moreover, if $\mathcal{S}$ is UCQ-rewritable then $\nabla(\mathcal{S})$ is UCQ-rewritable as well.*

It should be clear from the definition of $\nabla(\mathcal{S})$ that it is forward-existential and predicate unique. We show now that it is UCQ-rewritable as well. Let $\nabla_\exists(\mathcal{S})$ be $\{\rho_\exists \mid \rho \in \mathcal{S}\}$ and $\nabla_{\text{DL}}(\mathcal{S})$ be $\{\rho_{\text{DL}} \mid \rho \in \mathcal{S}\}$. For $\rho \in \mathcal{S}$ define the rule ρ_{full} as:

$$\rho_{\text{full}} = \phi(\bar{x}, \bar{y}) \rightarrow \exists \bar{z}, w \ \psi(\bar{y}, \bar{z}) \ \wedge \ A_0^\rho(w) \wedge \bigwedge_{y \in \bar{y}} A_{y,w}^\rho(y, w) \ \wedge \ \bigwedge_{z \in \bar{z}} B_{y',z}^\rho(y', z)$$

and let $\mathcal{S}_{\text{full}}$ be the set $\{\rho_{\text{full}} \mid \rho \in \mathcal{S}\}$ and $\mathcal{S}' = \mathcal{S}_{\text{full}} \cup \nabla_\exists(\mathcal{S}) \cup \nabla_{\text{DL}}(\mathcal{S})$. It should be clear that for any instance $\mathcal{I}$:

$$Ch(\mathcal{I}, \mathcal{S}') \leftrightarrow Ch(\mathcal{I}, \nabla(\mathcal{S})).$$

Due to the above, the rewritings against $\mathcal{S}'$ and $\nabla(\mathcal{S})$ are equivalent – so $\mathcal{S}'$ is UCQ-rewritable iff $\nabla(\mathcal{S})$ is – therefore it is enough to show that $\mathcal{S}'$ is UCQ-rewritable. Now, observe the following:

$$Ch(\mathcal{I}, \mathcal{S}') \leftrightarrow Ch(Ch(\mathcal{I}, \nabla_\exists(\mathcal{S}) \cup \nabla_{\text{DL}}(\mathcal{S})), \mathcal{S}_{\text{full}}).$$

We argue it holds, as $\nabla_{\exists}(\mathcal{S}) \cup \nabla_{\text{DL}}(\mathcal{S})$ is non-recursive and applications of rules from $\mathcal{S}_{\text{full}}$ cannot trigger rules from $\nabla_{\exists}(\mathcal{S}) \cup \nabla_{\text{DL}}(\mathcal{S})$ — the heads of their rules are already contained in the respective heads of rules of $\mathcal{S}_{\text{full}}$.

Due to the above equivalence, we can argue the $\mathcal{S}'$ is UCQ-rewritable if $\mathcal{S}_{\text{full}}$ is. It holds as given a CQ q and its bdd constant c under $\mathcal{S}_{\text{full}}$ the bdd constant of q under $\mathcal{S}'$ is no greater than $c + 2$, as $\nabla_{\exists}(\mathcal{S}) \cup \nabla_{\text{DL}}(\mathcal{S})$ is not recursive, and all queries under $\nabla_{\exists}(\mathcal{S}) \cup \nabla_{\text{DL}}(\mathcal{S})$ have the bdd constant of at most 2.

Therefore, by showing $\mathcal{S}_{\text{full}}$ is UCQ-rewritable we would complete the proof. Let $\mathbb{S}'$ contain the fresh predicates A or B (with indices) appearing in $\mathcal{S}_{\text{full}}$. Note that predicates from $\mathbb{S}'$ are not used in $\mathcal{S}_{\text{full}}$ recursively and that when restricted to predicates from $\mathbb{S}$ the rule sets $\mathcal{S}_{\text{full}}$ and $\mathcal{S}$ are identical. Therefore the derivations of $\mathcal{S}_{\text{full}}$ and $\mathcal{S}$ are identical — from this we conclude that $\mathcal{S}_{\text{full}}$ is indeed UCQ-rewritable as $\mathcal{S}$ is UCQ-rewritable. $\square$

Received December 2024; revised February 2025; accepted March 2025