Finite Controllability of Conjunctive Query Answering with Existential Rules: Two Steps Forward

Giovanni Amendola, Nicola Leone and **Marco Manna**

University of Calabria

amendola@mat.unical.it, leone@mat.unical.it, manna@mat.unical.it

Abstract

Reasoning with existential rules typically consists of checking whether a Boolean conjunctive query is satisfied by all models of a first-order sentence having the form of a conjunction of Datalog rules extended with existential quantifiers in rule-heads. To guarantee decidability, five basic decidable classes –linear, weakly-acyclic, guarded, sticky, and shy– have been singled out, together with several generalizations and combinations. For all basic classes, except shy, the important property of finite controllability has been proved, ensuring that a query is satisfied by all models of the sentence if, and only if, it is satisfied by all of its finite models. This paper takes two steps forward: (i) devise a general technique to facilitate the process of (dis)proving finite controllability of an arbitrary class of existential rules; and (ii) specialize the technique to complete the picture for the five mentioned classes, by showing that also shy is finitely controllable.

1 Introduction

The problem of answering a Boolean query q against a logical theory consisting of an extensional database D and an ontology Σ is attracting the attention of various fields of Computer Science, from Artificial Intelligence [Baget *et al.*, 2011; Calvanese *et al.*, 2013; Gottlob *et al.*, 2014; Amendola *et al.*, 2018] to Database Theory [Bienvenu *et al.*, 2014; Alviano and Pieris, 2015] and Logic [Pérez-Urbina *et al.*, 2010]. This problem, known as *ontology-based query answering* (OBQA), is usually stated as $D \cup \Sigma \models_a q$, and it is equivalent to checking whether q is satisfied by all models of $D \cup \Sigma$ according to the "open-world" semantics of first-order logic (where $\models_a$ is the entailment under *arbitrary* models).

Both Description Logics [Baader *et al.*, 2002] and Datalog$^\pm$ [Calì *et al.*, 2009a] have been recognized as effective families of formal knowledge representation languages to specify Σ, while union of (Boolean) conjunctive queries, U(B)CQs for short, is the most common and studied formalism to express q. Unfortunately, for both these families, OBQA is generally undecidable [Johnson and Klug, 1984; Rosati, 2007; Calì *et al.*, 2013]. Hence, a number of syntactic

decidable classes of the above languages have been singled out. But decidability alone is not the only desideratum. For example, a good balance between computational complexity and expressiveness is of high importance too. But there is another property that is turning out to be as interesting as challenging to prove: *finite controllability* [Johnson and Klug, 1984]. A class $\mathcal{C}$ of ontologies is finitely controllable if, for each triple $\langle D, \Sigma, q \rangle$ with $\Sigma \in \mathcal{C}$, it holds that $D \cup \Sigma \models_a q$ if, and only if, $D \cup \Sigma \models_f q$, where $\models_f$ is the entailment under *finite* models only. Note that this is equivalent to $D \cup \Sigma \not\models_a q$ implies there is a finite model M of $D \cup \Sigma$ such that $M \not\models q$, as the "only if" direction is trivially true. And there are applications, both in Databases [Johnson and Klug, 1984; Rosati, 2006] and Knowledge Representation [Rosati, 2008; Ibáñez-García *et al.*, 2014], where reasoning over finite models is preferred.

In this paper we focus on the Datalog$^\pm$ family, and consider for Σ a set of *existential rules* of the form $\forall \mathbf{X} \forall \mathbf{Y}(\phi(\mathbf{X}, \mathbf{Y}) \rightarrow \exists \mathbf{Z} p(\mathbf{X}, \mathbf{Z}))$, where the *body* $\phi(\mathbf{X}, \mathbf{Y})$ is a conjunction of atoms, and the *head* $p(\mathbf{X}, \mathbf{Z})$ is a single atom. The main decidable classes rely on the following basic five syntactic properties: *weak-acyclicity* [Fagin *et al.*, 2005], *linearity* [Calì *et al.*, 2009b], *guardedness* [Calì *et al.*, 2013], *stickiness* [Calì *et al.*, 2010], and *shyness* [Leone *et al.*, 2012], which underlie the basic classes called weakly-acyclic, linear, guarded, sticky, and shy, respectively. Several variants and combinations of these classes have been defined and studied too [Baget *et al.*, 2010; Krötzsch and Rudolph, 2011; Calì *et al.*, 2012; Gottlob *et al.*, 2013; 2018], as well as semantic properties subsuming the syntactic ones [Baget *et al.*, 2011; Leone *et al.*, 2012].

The above basic classes are pairwise uncomparable, except for linear which is strictly contained in guarded and shy, as reported in Figure 1. Interestingly, both weakly-acyclic and shy strictly contain datalog —the well-known class of existential-free rules of the form $\forall \mathbf{X} \forall \mathbf{Y}(\phi(\mathbf{X}, \mathbf{Y}) \rightarrow p(\mathbf{X}))$. Moreover, sticky strictly contains joinless —the class collecting sets of rules where each body contains no repeated variable. The latter, introduced by [Gogacz and Marcinkowski, 2017] to prove that sticky is finitely controllable, is also exploited in this paper. Finally, both linear and joinless strictly contain inclusion-dependencies —the well-known class of relational database dependencies collecting sets of rules with one single body atom and no repeated variable.
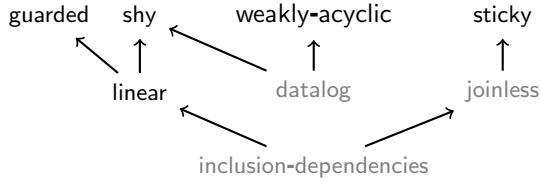
Figure 1: Taxonomy of the basic Datalog$^\pm$ classes.

Under arbitrary models, OBQA can be reduced to answering q over a *universal* model U that can be homomorphically embedded into every other arbitrary (finite or infinite) model of $D \cup \Sigma$. Hence, $D \cup \Sigma \models_a q$ if, and only if, $U \models q$. A way to compute such a U is to employ the *chase* procedure [Beeri and Vardi, 1984]. Intuitively, starting from D, the chase "repairs" violations of rules by repeatedly adding new atoms — introducing fresh values, called *nulls*, whenever required by an existential variable— until all rules are satisfied. In the classical setting, the chase is sound and complete. But when finite model reasoning is required, then the chase is generally uncomplete, unless ontologies are finitely controllable.

Finite controllability of weakly-acyclic comes for free as every ontology here admits a finite universal model [Fagin *et al.*, 2005]. Conversely, the proof of this property for the subsequent three classes has been a very different matter. Complex, yet intriguing, constructions have been devised for linear [Rosati, 2006; Bárány *et al.*, 2014], guarded [Bárány *et al.*, 2014], and more recently for sticky [Gogacz and Marcinkowski, 2017]. To complete the picture, we address the same problem for shy, while taking two steps forward:[1]

▶ We devise a general technique, called *canonical rewriting*, to facilitate the process of (dis)proving finite controllability of any class of existential rules. Indeed, via this technique, we can immediately (re)confirm that linear is finitely controllable since inclusion-dependencies is. Additionally, we can prove that sticky-join [Calì *et al.*, 2012] —generalizing both sticky and linear— is finitely controllable since sticky is.

▶ We specialize the canonical rewriting to complete the picture for the aforementioned five basic classes, and show that also shy is finitely controllable since joinless is.

2 Basics: Ontology-Based Query Answering

Basics. Let **C**, **N** and **V** denote pairwise disjoint discrete sets of *constants*, *nulls* and *variables*, respectively. A *term* t is an element of $\mathbf{T} = \mathbf{C} \cup \mathbf{N} \cup \mathbf{V}$. An *atom* α is a labeled tuple $p(t_1, ..., t_m)$, where p is a predicate symbol, m is the *arity* of both p anf α, and $t_1, ..., t_m$ are terms. An atom is *simple* if it contains no repeated term, and it is *propositional* if it has arity 0. Given two sets A and B of atoms, a homomorphism from A to B is a mapping $h : \mathbf{T} \to \mathbf{T}$ such that $c \in \mathbf{C}$ implies $h(c) = c$, and $h(A) \subseteq B$ with $h(A)$ obtained from A by replacing each term t by $h(t)$. An *instance* I is a discrete set of atoms where each term is either a constant or a null.

Syntax. A *database* D is a finite null-free instance. An *(existential) rule* ρ is a logical implication of the form $\forall \mathbf{X} \forall \mathbf{Y} (\phi(\mathbf{X}, \mathbf{Y}) \to \exists \mathbf{Z} \, p(\mathbf{X}, \mathbf{Z}))$, with $\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z} \subseteq \mathbf{V}$, whose body (resp., head) $body(\rho) = \phi(\mathbf{X}, \mathbf{Y})$ (resp.,

---

$head(\rho) = \{p(\mathbf{X}, \mathbf{Z})\})$ is a conjunction (or set) of atoms, possibly with constants. The set $\mathbf{X}$ is called the *frontier* of ρ. An *ontology* Σ is a set of rules. A *union of Boolean conjunctive query*, UBCQ for short, q is a first-order expression of the form $\exists \mathbf{Y}_1 \psi_1(\mathbf{Y}_1) \vee ... \vee \exists \mathbf{Y}_k \psi_k(\mathbf{Y}_k)$, where each $\psi_j(\mathbf{Y}_j)$ is a conjunction of atoms. Constants may occur also in q.

Semantics. Consider a triple $\langle D, \Sigma, q \rangle$ as above. An instance I *satisfies* a rule $\rho \in \Sigma$, written $I \models \rho$, if whenever there is a homomorphism h from $body(\rho)$ to I, then there is a homomorphism $h' \supseteq h|_{\mathbf{X}}$ from $head(\rho)$ to I. Also, I *satisfies* Σ, written $I \models \Sigma$, if I satisfies each rule of Σ. The *(arbitrary) models* of $D \cup \Sigma$, written $amods(D, \Sigma)$, are the set $\{I : I \supseteq D \text{ and } I \models \Sigma\}$. A model U is *universal* if, for each $M \in amods(D, \Sigma)$, there is a homomorphism from U to M. An instance I *satisfies* q, written $I \models q$, if there is a homomorphism from some $\psi_j(\mathbf{Y}_j)$ to I. Also, q is *true* over $D \cup \Sigma$, written $D \cup \Sigma \models_a q$, if each model of $D \cup \Sigma$ satisfies q.

Finite controllability. A class $\mathcal{C}$ of ontologies is *finitely controllable* if, for each triple $\langle D, \Sigma, q \rangle$ with $\Sigma \in \mathcal{C}$, $D \cup \Sigma \not\models_a q$ implies there is $M \in fmods(D, \Sigma)$ s.t. $M \not\models q$, where $fmods(D, \Sigma)$ denotes the *finite models* of $D \cup \Sigma$. This is usually stated as $D \cup \Sigma \models_a q$ if, and only if, $D \cup \Sigma \models_f q$.

3 Step 1: Canonical Rewriting

We design a general technique (see Figure 2) to facilitate the process of (dis)proving finite controllability of an arbitrary class of existential rules. Specifically, any triple $\langle D, \Sigma, q \rangle$ is rewritten in the semantically equivalent (canonical) triple $\langle D^c, \Sigma^c, q^c \rangle$ that has the advantage of being structurally simpler than the original one. Indeed, D^c is a propositional database, while Σ^c and q^c are constant-free structures containing simple atoms only. Rather than giving boring formal details, we prefer to shed some light on our construction via a simple example. Let $D = \{p(a), p(b), r(a, b)\}$, and Σ consist of the following rules:

$$
\begin{array}{lrcl}
(\rho_0) & p(X) & \to & \exists Y \, r(Y, X) \\
(\rho_1) & r(X, Y) & \to & p(X) \\
(\rho_2) & r(X, X) & \to & s(X)
\end{array}
$$

We now construct $D^c = \{p_{[a]}, p_{[b]}, r_{[a,b]}\}$ as the propositional database obtained by encoding each atom of D in a different fresh predicate. Then, from Σ we build Σ^c as follows:

$$
\begin{array}{lrcl}
(\rho_{0,0}) & p_{[a]} & \to & \exists Y \, r_{[1,a]}(Y) \\
(\rho_{0,1}) & p_{[b]} & \to & \exists Y \, r_{[1,b]}(Y) \\
(\rho_{0,2}) & p_{[1]}(X) & \to & \exists Y \, r_{[1,2]}(Y, X) \\[4pt]
(\rho_{1,0}) & r_{[a,a]} & \to & p_{[a]} \\
(\rho_{1,1}) & r_{[a,b]} & \to & p_{[a]} \\
(\rho_{1,2}) & r_{[b,a]} & \to & p_{[b]} \\
(\rho_{1,3}) & r_{[b,b]} & \to & p_{[b]} \\
(\rho_{1,4}) & r_{[a,1]}(Y) & \to & p_{[a]} \\
(\rho_{1,5}) & r_{[b,1]}(Y) & \to & p_{[b]} \\
(\rho_{1,6}) & r_{[1,a]}(X) & \to & p_{[1]}(X) \\
(\rho_{1,7}) & r_{[1,b]}(X) & \to & p_{[1]}(X) \\
(\rho_{1,8}) & r_{[1,1]}(X) & \to & p_{[1]}(X) \\
(\rho_{1,9}) & r_{[1,2]}(X, Y) & \to & p_{[1]}(X) \\[4pt]
(\rho_{2,0}) & r_{[a,a]} & \to & s_{[a]} \\
(\rho_{2,1}) & r_{[b,b]} & \to & s_{[b]} \\
(\rho_{2,2}) & r_{[1,1]}(X) & \to & s_{[1]}(X)
\end{array}
$$

---

[1]This paper is an abridged version of [Amendola *et al.*, 2017].
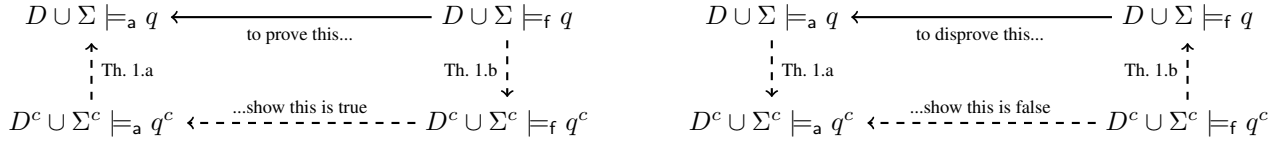
Figure 2: Application of the canonical rewriting.

Intuitively, new predicates encode tuples of terms consisting of constants (a and b) from $D \cup \Sigma$, and placeholders (1 and 2, where the latter is the maximum arity among predicates occurring in Σ) of arbitrary terms of $\mathbf{C} \cup \mathbf{N}$. Any rule $\rho_{i,j}$ of Σ^c is called *child* of ρ_i, while the latter is the *parent* of $\rho_{i,j}$. Finally, by following the same intuition, we apply the canonical rewriting to a set A of atoms over $\mathbf{C} \cup \mathbf{V}$ w.r.t. the constants of $D \cup \Sigma$. E.g., given $A = \{r(a, X), p(X)\}$, we obtain the following family of sets $A^c = \{\{r_{[a,a]}, p_{[a]}\}, \{r_{[a,b]}, p_{[b]}\}, \{r_{[a,1]}(X), p_{[1]}(X)\}\}$. Thus, given the UBCQ $q = \exists X\{r(a, X), p(X)\} \vee \exists Y\{r(Y, Y)\}$, its canonical rewriting will be $q^c = \{r_{[a,a]}, p_{[a]}\} \vee \{r_{[a,b]}, p_{[b]}\} \vee \exists X\{r_{[a,1]}(X), p_{[1]}(X)\} \vee \{r_{[a,a]}\} \vee \{r_{[b,b]}\} \vee \exists Y\{r_{[1,1]}(Y)\}$.

Accordingly, let us define the *unpacking function* $\mathfrak{R}$ from atoms over the set of canonical predicates to atoms over the set of original ones. The function generalizes in the obvious way to set of atoms, rules, and queries. For instance, if $A' = \{r_{[1,2]}(X, Y), r_{[1,a]}(\eta), r_{[1,1]}(\eta)\}$ with $\eta \in \mathbf{N}$, then $\mathfrak{R}(A') = \{r(X, Y), r(\eta, a), r(\eta, \eta)\}$. Informally, the unpacking function acts like an "inverse" operator of the canonical rewriting.

Consider now the model $M = D \cup \{p(\eta), r(\eta, a), r(\eta, b), r(\eta, \eta), s(\eta)\}$ of $D \cup \Sigma$. Also, let $M' = D^c \cup \{p_{[1]}(\eta), r_{[1,a]}(\eta), r_{[1,b]}(\eta), r_{[1,2]}(\eta, \eta)\}$ be a model of $D^c \cup \Sigma^c$. Note that, whenever there is a homomorphism h from the body of some rule $\rho_{i,j}$ to M', then h also maps the body of the parent ρ_i to M. Indeed, this is true as $\mathfrak{R}(M') \subseteq M$. However, although M' is a model of $D^c \cup \Sigma^c$, $\mathfrak{R}(M')$ is not a model of $D \cup \Sigma$, as $\alpha' = r_{[1,2]}(\eta, \eta)$ does not trigger any rule of Σ^c, while $\mathfrak{R}(\alpha')$ triggers ρ_2. In fact, this happens because: there is no bijection between the placeholders $\{1, 2\}$ and $\{\eta\}$, and also M' contains no atom of the form $r_{[1,1]}(\eta)$, where a bijection between $\{1\}$ and $\{\eta\}$ exists. To overcome such a mismatch, we focus on a suitable subset of $amods(D^c \cup \Sigma^c)$.

Definition 1. *A model N' of $D^c \cup \Sigma^c$ is* canonical *if: (i) it is constant-free; and (ii) for each non-simple atoms α' of N', there is a simple atom β' of N' such that $\mathfrak{R}(\alpha') = \mathfrak{R}(\beta')$.*

From any M' it is possible to build a canonical model N' that can be homomorphically mapped to M': (1) set N' to M'; (2) replace each constant c of N' by a fresh null η_c; (3) let E be the set of nulls occurring in atoms of N' that violate condition (ii); (4) for each null η of E, let $pad(\eta) = \{\eta_1, ..., \eta_k\}$, where k is the maximum number of times that η occurs in an atom of N'; (5) for each η of E, replace each atom α' of N' containing η with new atoms obtained from α' by arranging the elements of $pad(\eta)$ in the positions where η occurs, while avoiding repetitions; and (6) remove all atoms of N' violating condition (ii). Hence, in our running example, $N' = D^c \cup \{p_{[1]}(\eta_1), p_{[1]}(\eta_2), r_{[1,a]}(\eta_1), r_{[1,a]}(\eta_2), r_{[1,b]}(\eta_1), r_{[1,b]}(\eta_2), r_{[1,2]}(\eta_1, \eta_2), r_{[1,2]}(\eta_2, \eta_1)\}$. Now, from

the above intuitions, we collect some relevant properties.

Proposition 1. *Let A be a set of atoms over $\mathbf{C} \cup \mathbf{V}$ w.r.t. the constants of $D \cup \Sigma$, M' be a model of $D^c \cup \Sigma^c$, and N' be a canonical model of $D^c \cup \Sigma^c$. It holds that:*

a. *If, for some $A' \in A^c$, $M' \models A'$, then $\mathfrak{R}(M') \models A$;*

b. *If $\mathfrak{R}(N') \models A$, then $N' \models A'$, for some $A' \in A^c$;*

c. *There is $N'_0 \in camods(D^c, \Sigma^c)$ that can be homomorphically mapped to M'. Also, if M' is finite, then N'_0 is.*

d. *$\mathfrak{R}(N')$ is a model of $D \cup \Sigma$;*

e. *$amods(D, \Sigma) \subseteq \mathfrak{R}(amods(D^c, \Sigma^c))$.*

With these properties in hand we are able to provide a formal argument for the implications depicted in Figure 2.

Theorem 1. *It holds that:*

a. *$D \cup \Sigma \models_{\mathsf{a}} q$ if, and only if, $D^c \cup \Sigma^c \models_{\mathsf{a}} q^c$; and*

b. *$D \cup \Sigma \models_{\mathsf{f}} q$ if, and only if, $D^c \cup \Sigma^c \models_{\mathsf{f}} q^c$.*

Proof sketch. We now describe the proof sketch for finite model reasoning (b). We start with the "only-if" direction.

$$D^c \cup \Sigma^c \not\models_{\mathsf{f}} q^c \qquad \text{(by contradiction)}$$
$$\Rightarrow \exists N' \in cfmods(D^c, \Sigma^c) \text{ s.t. } N' \not\models q^c \qquad \text{(by Prop. 1.c)}$$
$$\Rightarrow \forall q' \in q^c \text{ it holds that } N' \not\models q' \qquad \text{(by definition)}$$
$$\Rightarrow \mathfrak{R}(N') \not\models q \qquad \text{(by Prop. 1.b)}$$
$$\Rightarrow D \cup \Sigma \not\models_{\mathsf{f}} q \qquad \text{(by Prop. 1.d)}$$

Now, we sketch the proof for the "if" direction.

$$D \cup \Sigma \not\models_{\mathsf{f}} q \qquad \text{(by contradiction)}$$
$$\Rightarrow \exists M \in fmods(D, \Sigma) \text{ s.t. } M \not\models q \qquad \text{(by definition)}$$
$$\Rightarrow \exists M' \in fmods(D^c, \Sigma^c) \text{ s.t. } \mathfrak{R}(M') = M \qquad \text{(by Prop. 1.e)}$$
$$\Rightarrow \forall q' \in q^c \text{ it holds that } M' \not\models q' \qquad \text{(by Prop. 1.a)}$$
$$\Rightarrow D^c \cup \Sigma^c \not\models_{\mathsf{f}} q^c \qquad \text{(by definition)}$$

The implication chains hold also for arbitrary model reasoning (a), by replacing $\models_{\mathsf{f}}$ by $\models_{\mathsf{a}}$, and $cfmods$ by $camods$. □

By exploiting our tool, we can reprove that linear is finitely controllable. In fact, the canonical rewriting of a linear ontology belongs to inclusion-dependencies. Moreover, since the canonical rewriting of a sticky-join [Calì *et al.*, 2012] ontology belongs to sticky, we prove (for the first time) that:

Theorem 2. sticky-join *is finitely controllable.*

Note that the definitions of linear and sticky-join in [Gogacz and Marcinkowski, 2017] are not standard (actually stricter): there, in linear repeated variables are admitted only in rule heads, and the difference between sticky and sticky-join "can only be seen if repeated variables in the heads of the rules are allowed". Hence, the finite controllability of sticky-join was unknown before our work. One can verify that the proof of their Lemma 4 breaks when moving to the linear (hence sticky-join) ontology $\Sigma = \{p(X, X) \rightarrow r(X); r(X) \rightarrow \exists Y \, r(Y)\}$ paired with the database $D = \{p(c, c)\}$.

4 Step 2: Finite Controllability of Shy

Differently from linear and sticky-join, the canonical rewriting of a shy ontology —although it is still a shy ontology— falls in none of the known finitely controllable classes. Hence, to prove the finite controllability of shy, we are going to specialize the canonical rewriting. But before, let us recall the key properties of this class. Fix a triple $\langle D, \Sigma, q \rangle$:

Proposition 2 ([Leone *et al.*, 2012]). *Assume* $\Sigma \in$ shy. *Given a universal model U of $D \cup \Sigma$, a rule ρ of Σ, and a homomorphism h from the atoms of ρ to U, it holds that:*

a. *if a variable X of ρ occurs in two different atoms of $body(\rho)$, then $h(X) \in \mathbf{C}$; and*

b. *if two frontier variables X, Y of ρ occur in two different atoms of $body(\rho)$ and $h(X) = h(Y)$, then $h(X) \in \mathbf{C}$.*

Consider now the canonical rewriting $\langle D^c, \Sigma^c, q^c \rangle$ of $\langle D, \Sigma, q \rangle$. Inspired by well-supported interpretations [Fages, 1991], we define a suitable subset of all finite models:

Definition 2. *A finite model M' of $D^c \cup \Sigma^c$ is* well-supported *if there is an ordering $\alpha_1, ..., \alpha_m$ of its atoms s.t., for each $j \in \{1, ..., m\}$, it holds that: (i) $\alpha_j \in D^c$, or (ii) there is $\rho \in \Sigma^c$ and a homomorphism from the atoms of ρ to $\{\alpha_1, ..., \alpha_j\}$ mapping $body(\rho)$ to $\{\alpha_1, ..., \alpha_{j-1}\}$ and $head(\rho)$ to $\{\alpha_j\}$.*

Subsequently, we define $\Sigma^c_\ominus$ as the set of all joinless rules of Σ^c. For instance, consider the ontology $\Sigma = \{\rho_0 = p(X) \to r(X, Y); \rho_1 = p(X) \to s(X, Y); \rho_2 = r(X, Y), s(X, Y) \to t(X)\}$ and the database $D = \{p(a)\}$. According to the canonical rewriting, $\Sigma^c = \{\rho_{0,0} = p_{[a]} \to r_{[a,1]}(Y); \rho_{0,1} = p_{[1]}(X) \to r_{[1,2]}(X, Y); \rho_{1,0} = p_{[a]} \to s_{[a,1]}(Y); \rho_{1,1} = p_{[1]}(X) \to s_{[1,2]}(X, Y); \rho_{2,0} = r_{[a,a]}, s_{[a,a]} \to t_{[a]}; \rho_{2,1} = r_{[a,1]}(Y), s_{[a,1]}(Y) \to t_{[a]}; \rho_{2,2} = r_{[1,a]}(X), s_{[1,a]}(X) \to t_{[1]}(X); \rho_{2,3} = r_{[1,2]}(X, Y), s_{[1,2]}(X, Y) \to t_{[1]}(X)\}$, and $D^c = \{p_{[a]}\}$. Hence, $\Sigma^c_\ominus = \{\rho_{0,0}, \rho_{0,1}, \rho_{1,0}, \rho_{1,1}, \rho_{2,0}\}$. Let $M' = \{p_{[a]}, r_{[a,1]}(\eta), s_{[a,1]}(\eta), t_{[1]}(\eta)\}$ be a finite model of $D^c \cup \Sigma^c_\ominus$ (and not of $D^c \cup \Sigma^c$). It can be checked that M' is not well-supported. Intuitively, there is no support for the atom $t_{[1]}(\eta)$. However, we can find a subset of M' that is a well-supported model of $D^c \cup \Sigma^c_\ominus$. In this case, $N' = M' \setminus \{t_{[1]}(\eta)\}$ is the right one, and $p_{[a]}, r_{[a,1]}(\eta), s_{[a,1]}(\eta)$ is an ordering satisfying Definition 2.

Note that N' is not a model of $D^c \cup \Sigma^c$, as $\rho_{2,1}$ is not satisfied. Now, starting from N' we can build a well-supported finite model of $D^c \cup \Sigma^c$ that can be homomorphically embedded into N'. To this end, we have developed a technique called *propagation ordering*. Intuitively, as $r_{[a,1]}(\eta)$ and $s_{[a,1]}(\eta)$ can be generated only by rules $\rho_{0,0}$ and $\rho_{1,0}$, respectively, there is no "real necessity" to consider the same null to satisfy both rules. Hence, by exploiting the given ordering of N', we rename and propagate some of the terms of N' to construct a well-supported finite model of $D^c \cup \Sigma^c$. More precisely, we define $\langle r_{[a,1]}(\eta) \rangle = r_{[a,1]}(\langle \eta, 2, 1 \rangle)$, as $r_{[a,1]}(\eta)$ is the second atom in the ordering given above, and η occurs in the first position of $r_{[a,1]}$. Similarly, we set $\langle s_{[a,1]}(\eta) \rangle = s_{[a,1]}(\langle \eta, 3, 1 \rangle)$. That is, $\langle \eta, 2, 1 \rangle$ and $\langle \eta, 3, 1 \rangle$ are distinct fresh nulls corresponding to η. In this example, we can stop here, as no other propagation appears. It is clear that the new instance $N'_0 = \{p_{[a]}, r_{[a,1]}(\langle \eta, 2, 1 \rangle), s_{[a,1]}(\langle \eta, 3, 1 \rangle)\}$ is a well-supported model of $D^c \cup \Sigma^c$. Note that here is where the shyness of Σ^c is crucial. Also, N'_0 can be homomorphically embedded into N' by mapping $\langle \eta, 2, 1 \rangle$ to η, and $\langle \eta, 3, 1 \rangle$ to η. These insights are the basis of the following properties.

Proposition 3. *It holds that:*

a. *The ontology $\Sigma^c_\ominus$ is finitely controllable [Gogacz and Marcinkowski, 2017];*

b. *For each $M' \in fmods(D^c \cup \Sigma^c)$ there exists a well-supported finite model N' of $D^c \cup \Sigma^c$ s.t. $N' \subseteq M'$;*

c. *Whenever Σ is shy, for each $M' \in wsfmods(D^c, \Sigma^c_\ominus)$, there is a model $N' \in wsfmods(D^c, \Sigma^c)$ that can be homomorphically embedded into M'.*

With these properties in hand, we can now provide a formal argument to show the main result of the section.

Theorem 3. *Under* shy *ontologies, it holds that:*

$$D^c \cup \Sigma^c \models_{\mathsf{a}} q^c \text{ if, and only if, } D^c \cup \Sigma^c \models_{\mathsf{f}} q^c.$$

Proof sketch. We prove the "if" direction.

$$
\begin{array}{lll}
& D^c \cup \Sigma^c \not\models_{\mathsf{a}} q^c & \textit{(by contradiction)} \\
\Rightarrow & \exists M'_1 \in amods(D^c, \Sigma^c) \text{ s.t. } M'_1 \not\models q^c & \textit{(by definition)} \\
\Rightarrow & M'_1 \in amods(D^c, \Sigma^c_\ominus) & \textit{(by } \Sigma^c_\ominus \subseteq \Sigma^c) \\
\Rightarrow & \exists M'_2 \in fmods(D^c, \Sigma^c_\ominus) \text{ s.t. } M'_2 \not\models q^c & \textit{(by Prop. 3.a)} \\
\Rightarrow & \exists M'_3 \in wsfmods(D^c, \Sigma^c_\ominus) \text{ s.t. } M'_3 \not\models q^c & \textit{(by Prop. 3.b)} \\
\Rightarrow & \exists M'_4 \in wsfmods(D^c, \Sigma^c) \text{ s.t. } M'_4 \not\models q^c & \textit{(by Prop. 3.c)} \\
\Rightarrow & M'_4 \in fmods(D^c, \Sigma^c) & \textit{(by definition)} \\
\Rightarrow & D^c \cup \Sigma^c \not\models_{\mathsf{f}} q^c & \textit{(by definition)}
\end{array}
$$

Therefore, we can conclude that the statement is true since the "only if" direction trivially holds. $\square$

Finally, the finite controllability of shy follows by combining Theorem 3 with Theorem 1.

Theorem 4. shy *is finitely controllable.*

5 Conclusion

To complete the related works started with the Introduction, we recall that finite controllability in OBQA was formalized for the first time by [Rosati, 2006] while he was working on a question left open two decades before by [Johnson and Klug, 1984] about containment of conjunctive queries in case of both arbitrary and finite databases. Basically, using our terminology, they proved that ontologies mixing both inclusion-dependencies and functional-dependencies are not finitely controllable, by leaving open the case in which ontologies contain inclusion-dependencies only. Rosati then answered positively this question. And a decade later, by demonstrating that shy is finitely controllable (Step 2), we complete an important picture around the basic decidable Datalog$^\pm$ classes. But we believe that the techniques developed here could have future further applications. For example, it would be interesting to extend the canonical rewriting (Step 1) by encoding in the predicates also a bounded number of nulls (apart from constants and placeholders of nulls). This requires more complex techniques, which however would apply to both glut-guarded [Krötzsch and Rudolph, 2011] (extending guarded and weakly-acyclic), and weakly-sticky-join [Calì *et al.*, 2012] (extending sticky-join and weakly-acyclic).

Acknowledgments

References

[Alviano and Pieris, 2015] Mario Alviano and Andreas Pieris. Default negation for non-guarded existential rules. In *Proc. of PODS*, pages 91–103, 2015.

[Amendola *et al.*, 2017] Giovanni Amendola, Nicola Leone, and Marco Manna. Finite model reasoning over existential rules. *TPLP*, 17(5-6):726–743, 2017.

[Amendola *et al.*, 2018] Giovanni Amendola, Nicola Leone, Marco Manna, and Pierfrancesco Veltri. Enhancing existential rules by closed-world variables. In *Proc. of IJCAI*, 2018.

[Baader *et al.*, 2002] Franz Baader, Ian Horrocks, and Ulrike Sattler. Description logics for the semantic web. *KI*, 16(4):57–59, 2002.

[Baget *et al.*, 2010] Jean-François Baget, Michel Leclère, and Marie-Laure Mugnier. Walking the decidability line for rules with existential variables. In *Proc. of KR*, 2010.

[Baget *et al.*, 2011] Jean-François Baget, Michel Leclère, Marie-Laure Mugnier, and Eric Salvat. On rules with existential variables: Walking the decidability line. *AIJ*, 175(9-10):1620–1654, 2011.

[Bárány *et al.*, 2014] Vince Bárány, Georg Gottlob, and Martin Otto. Querying the guarded fragment. *Log. Methods Comput. Sci.*, 10(2), 2014.

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

[Bienvenu *et al.*, 2014] Meghyn Bienvenu, Balder ten Cate, Carsten Lutz, and Frank Wolter. Ontology-based data access: A study through disjunctive datalog, csp, and MMSNP. *ACM TODS*, 39(4):33:1–33:44, 2014.

[Calì *et al.*, 2009a] Andrea Calì, Georg Gottlob, and Thomas Lukasiewicz. Datalog$^\pm$: a unified approach to ontologies and integrity constraints. In *Proc. of ICDT*, pages 14–30, 2009.

[Calì *et al.*, 2009b] Andrea Calì, Georg Gottlob, and Thomas Lukasiewicz. A general datalog-based framework for tractable query answering over ontologies. In *Proc. of PODS*, pages 77–86, 2009.

[Calì *et al.*, 2010] Andrea Calì, Georg Gottlob, and Andreas Pieris. Advanced processing for ontological queries. *PVLDB*, 3(1):554–565, 2010.

[Calì *et al.*, 2012] Andrea Calì, Georg Gottlob, and Andreas Pieris. Towards more expressive ontology languages: The query answering problem. *AIJ*, 193:87–128, 2012.

[Calì *et al.*, 2013] Andrea Calì, Georg Gottlob, and Michael Kifer. Taming the infinite chase: Query answering under expressive relational constraints. *JAIR*, 48:115–174, 2013.

[Calvanese *et al.*, 2013] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Data complexity of query answering in description logics. *AIJ*, 195:335–360, 2013.

[Fages, 1991] François Fages. A new fixpoint semantics for general logic programs compared with the well-founded and the stable model semantics. *New Generation Comput.*, 9(3/4):425–444, 1991.

[Fagin *et al.*, 2005] Ronald Fagin, Phokion G. Kolaitis, Renée J. Miller, and Lucian Popa. Data exchange: semantics and query answering. *TCS*, 336(1):89–124, 2005.

[Gogacz and Marcinkowski, 2017] Tomasz Gogacz and Jerzy Marcinkowski. Converging to the chase - A tool for finite controllability. *JCSS*, 83(1):180–206, 2017.

[Gottlob *et al.*, 2013] Georg Gottlob, Marco Manna, and Andreas Pieris. Combining decidability paradigms for existential rules. *TPLP*, 13(4-5):877–892, 2013.

[Gottlob *et al.*, 2014] Georg Gottlob, Stanislav Kikot, Roman Kontchakov, Vladimir V. Podolskii, Thomas Schwentick, and Michael Zakharyaschev. The price of query rewriting in ontology-based data access. *AIJ*, 213:42–59, 2014.

[Gottlob *et al.*, 2018] Georg Gottlob, Marco Manna, and Andreas Pieris. Finite model reasoning in hybrid classes of existential rules. In *Proc. of IJCAI*, 2018.

[Ibáñez-García *et al.*, 2014] Yazmin Angélica Ibáñez-García, Carsten Lutz, and Thomas Schneider. Finite model reasoning in horn description logics. In *Proc. of KR*, 2014.

[Johnson and Klug, 1984] David S. Johnson and Anthony C. Klug. Testing containment of conjunctive queries under functional and inclusion dependencies. *JCSS*, 28(1):167–189, 1984.

[Krötzsch and Rudolph, 2011] Markus Krötzsch and Sebastian Rudolph. Extending decidable existential rules by joining acyclicity and guardedness. In *Proc. of IJCAI*, pages 963–968, 2011.

[Leone *et al.*, 2012] Nicola Leone, Marco Manna, Giorgio Terracina, and Pierfrancesco Veltri. Efficiently computable Datalog$^\exists$ programs. In *Proc. of KR*, 2012.

[Pérez-Urbina *et al.*, 2010] Héctor Pérez-Urbina, Boris Motik, and Ian Horrocks. Tractable query answering and rewriting under description logic constraints. *J. Appl. Logic*, pages 186–209, 2010.

[Rosati, 2006] Riccardo Rosati. On the decidability and finite controllability of query processing in databases with incomplete information. In *Proc. of PODS*, pages 356–365, 2006.

[Rosati, 2007] Riccardo Rosati. The limits of querying ontologies. In *Proc. of ICDT*, pages 164–178, 2007.

[Rosati, 2008] Riccardo Rosati. Finite model reasoning in dl-lite. In *Proc. of ESWC*, pages 215–229, 2008.