Enhancing Existential Rules by Closed-World Variables

Giovanni Amendola, Nicola Leone, Marco Manna and **Pierfrancesco Veltri**
University of Calabria, Italy
amendola@mat.unical.it, leone@mat.unical.it, manna@mat.unical.it, veltri@mat.unical.it

Abstract

Existential rules generalize Datalog with existential quantification in the head. Natively, Datalog is interpreted under a closed-world semantics, while existential rules typically employ the open-world assumption. The interpretation domain in the latter case is enlarged by infinitely many "anonymous" individuals. Then, in any rule, each variable ranges over all individuals, even if not needed or required. In this paper, we enhance existential rules by closed-world variables to consciously reason on the properties of "known" (non-anonymous) and arbitrary individuals in different ways. Accordingly, we uniformly generalize the basic classes of existential rules that ensure decidability of ontology-based query answering. For them, after observing that decidability is preserved, we prove that a strict increase in expressiveness is gained, and in most cases the computational complexity is not altered.

1 Introduction

Existential rules, also known as TGDs or datalog$^{\exists}$ rules, are a fascinating research topic deeply studied not only in artificial intelligence [Baget *et al.*, 2011; Amendola *et al.*, 2017] but also in database theory [Bourhis *et al.*, 2016; Alviano and Pieris, 2015] and logic [Bárány *et al.*, 2014]. They are at the core of Datalog$^{\pm}$ [Calì *et al.*, 2009], an emerging family of ontology languages complementing the expressive power of Description Logics (DLs) [Baader *et al.*, 2003]. Indeed, datalog$^{\exists}$ generalizes the well-known language Datalog [Ceri *et al.*, 1989] with existential quantification in the head. Natively, Datalog is interpreted under a closed-world semantics, while existential rules typically employ the open-world assumption. For example, in classical *query answering* [Ortiz, 2013] —where a query q is evaluated over a logical theory consisting of a database D paired with an ontology Σ— the presence of existential quantifiers in Σ requires an interpretation domain of $D \cup \Sigma$ that extends the closed domain of D with infinitely many extra "anonymous" individuals. Then, each variable of Σ does range over all individuals.

To consciously reason on the properties of "known" (non-anonymous) and arbitrary individuals in different ways, we complement standard variables with *closed(-world) variables* that range over the individuals of $D \cup \Sigma$ only. The resulting language, called datalog$^{\exists,\mathrm{H}}$, offers novel modeling capabilities, as it allows to specify properties at both data and conceptual level in a uniform way. Consider, for example, a scenario in which one has to model that "every good has a price" and "a good is auctionable if some reference price can be associated to it". Such desiderata are expressible via the rules $\rho_1 = good(X) \rightarrow \exists Y\ hasPrice(X, Y)$ and $\rho_2 = good(X),\ hasPrice(X, \widehat{\mathrm{Y}}) \rightarrow auctionable(X)$, where $\widehat{\mathrm{Y}}$ is a closed variable. Given $D_0 = \{good(ferrari250)\}$, $\Sigma_0 = \{\rho_1, \rho_2\}$, and the queries $q_1 = \exists X\ \exists Y\ hasPrice(X, Y)$ and $q_2 = \exists X\ \exists Y\ hasPrice(X, Y),\ auctionable(X)$. Clearly, q_1 is entailed by $D_0 \cup \Sigma_0$. But q_2 is not since $M = D_0 \cup \{hasPrice(ferrari250, 10)\}$ is a possible model of $D_0 \cup \Sigma_0$. Indeed, 10 is not a reference price for $ferrari250$ but simply one of the infinitely many anonymous individuals not in D_0. Therefore rule ρ_2 is satisfied in M. Of course, a first natural question now is to wonder whether Σ_0 can be expressed via some equivalent datalog$^{\exists}$ ontology.

Existential rules, besides offering good modeling capability, are extremely challenging from a computational viewpoint, as they make query answering undecidable in the general case [Beeri and Vardi, 1984]. To remedy this fact, several syntactic conditions have been proposed in the literature, with some giving rise to the five *basic* decidable datalog$^{\exists}$ classes: linear [Calì *et al.*, 2012a], weakly-acyclic [Fagin *et al.*, 2005], guarded [Calì *et al.*, 2013], sticky [Calì *et al.*, 2012b], and shy [Leone *et al.*, 2012]. The second natural question now is to wonder whether these conditions can be generalized to preserve decidability of query answering also for datalog$^{\exists,\mathrm{H}}$.

Along the paper we give answers to the above questions, starting right here by summarizing the main contributions:

► For each basic datalog$^{\exists}$ class $\mathcal{C}$, we consider a "naive" and a "refined" extension, denoted by $\mathcal{C}\mathsf{H}$ and $\mathcal{C}\mathsf{H}^{+}$, respectively. In naive extension, the syntactic conditions underlying $\mathcal{C}$ treat closed variables as standard ones. In the refined one, the syntactic conditions are enforced over standard variables only. Decidability can be easily established. (Section 3.)

► We show that $\mathcal{C}\mathsf{H}$ preserves the same data and combined complexity of each basic datalog$^{\exists}$ class $\mathcal{C}$. Likewise, this holds with shyH^{+} and w-acyclicH^{+} w.r.t. their standard counterparts. Differently, guardedH^{+} and stickyH^{+} exhibit an increase in data complexity, while only linearH^{+} has an increase in both data and combined complexity. (Section 4.)

► We prove that $\mathsf{datalog}^{\exists,\mathsf{H}}$ is (resp., $\mathcal{C}\mathsf{H}$ and $\mathcal{C}\mathsf{H}^{+}$ are) strictly more expressive than $\mathsf{datalog}^{\exists}$ (resp., each basic class $\mathcal{C}$). In particular, going back to our running example, there is no $\mathsf{datalog}^{\exists}$ ontology that, independently from the database at hand, behaves as Σ_0 w.r.t. both q_1 and q_2. Also, each $\mathcal{C}\mathsf{H}^{+}$ is even strictly more expressive than datalog. (Section 5.1.)

► We show that the well-known Description Logic $\mathcal{ELH}$ [Brandt, 2004b] is captured by $\mathsf{linear}\mathsf{H}^{+}$, even if we only focus on $\mathsf{linear}\mathsf{H}^{+}$ ontologies of arity at most two and with at most two atoms in the body (where the combined complexity of query answering drops to NP as for $\mathcal{ELH}$). Interestingly, $\mathsf{linear}\mathsf{H}^{+}$ keeps a lower computational complexity, compared to other $\mathsf{datalog}^{\exists}$ classes that can express this DL, namely guarded and its extensions. (Section 5.2.)

2 Existential Rules with Closed Variables

Basics. Let $\mathbf{C}$ (*constants* or *individuals*) and $\mathbf{V}$ (*variables*) be pairwise disjoint discrete sets of *terms*. A variable $(x, y, ...)$ is either *standard* $(X, Y, ...)$ or *closed(-world)* $(\widehat{\mathrm{X}}, \widehat{\mathrm{Y}}, ...)$. We denote by $\mathbf{V}_s$ and $\mathbf{V}_c$, the set of standard and closed variables, respectively. An *atom* α is a labeled tuple $p(\mathbf{t})$, where $p = pred(\alpha)$ is a predicate symbol, $\mathbf{t} = t_1, ..., t_m$ is a tuple of terms, $m = |p|$ is the arity of p or α, and $\alpha[i] = t_i$. Given a finite domain $\Delta \subset \mathbf{C}$ of "known" individuals, a Δ-*substitution* is any map $\mu : \mathbf{V} \to \mathbf{C}$ such that $\widehat{\mathrm{X}} \in \mathbf{V}_c$ implies $\mu(\widehat{\mathrm{X}}) \in \Delta$. For a set A of atoms, $\mu(A)$ is obtained from A by replacing each variable x by $\mu(x)$. A *database* (resp., *instance*) is any variable-free finite (resp., possibly infinite) set of atoms.

Syntax. A $\mathsf{datalog}^{\exists,\mathsf{H}}$ *rule* ρ is a logical implication of the form $\forall\mathbf{X}\forall\mathbf{Y}\,(\phi(\mathbf{X},\mathbf{Y}) \to \exists\mathbf{Z}\,\psi(\mathbf{X},\mathbf{Z}))$ —with $\mathbf{X}\cup\mathbf{Y} \subseteq \mathbf{V}$ and $\mathbf{Z} \subseteq \mathbf{V}_s$— whose body (resp., head) $b(\rho) = \phi(\mathbf{X},\mathbf{Y})$ (resp., $h(\rho) = \psi(\mathbf{X},\mathbf{Z})$) is a conjunction (or set) of atoms, possibly with constants. As usual, the head is nonempty. Universal and existential variables are respectively denoted by $\mathbf{UV}(\rho)$ and $\mathbf{EV}(\rho)$. The set $\mathbf{X}$ is known as the *frontier* of ρ. If no closed variable is in ρ, then it is also a $\mathsf{datalog}^{\exists}$ rule; and if even $\mathbf{EV}(\rho) = \emptyset$, then it is also a datalog rule. A $\mathsf{datalog}^{\exists,\mathsf{H}}$ ontology Σ is any finite set of $\mathsf{datalog}^{\exists,\mathsf{H}}$ rules. We denote by $\mathcal{R}(\Sigma)$ the set of predicates occurring in Σ. A *position* $p[i]$ is defined as a predicate p of $\mathcal{R}(\Sigma)$ and its i-th attribute. Let $pos(p) = \{p[1], ..., p[|p|]\}$. A $\mathbb{C}^{\mathsf{H}}$ (hybrid conjunctive) *query* is an expression of the form $q(\mathbf{X}) = \exists\mathbf{Y}\ \phi(\mathbf{X},\mathbf{Y})$, where ϕ is as above. In case q contains no closed variable, it is also a $\mathbb{C}$ (conjunctive) query. For a "structure" ς over atoms (set, rule, query, ...), if $\widehat{\mathrm{X}}$ occurs in ς, then X does not occur in ς. Also, $atoms(\varsigma)$, $terms(\varsigma)$, $vars(\varsigma)$ and $std(\varsigma)$ respectively denote the set of atoms in ς, the set of terms in $atoms(\varsigma)$, the set of variables in $atoms(\varsigma)$, and the structure built from ς by replacing each $\widehat{\mathrm{X}}$ with X.

Semantics. Consider a triple $\langle D, \Sigma, q\rangle$ as above, and let $\Delta = terms(D,\Sigma) \cap \mathbf{C}$. A *model* of $D \cup \Sigma$ is any instance $M \supseteq D$ such that, for each $\rho \in \Sigma$ and each Δ-substitution μ, $\mu(b(\rho)) \subseteq M$ implies $\mu'(h(\rho)) \subseteq M$ for some Δ-substitution $\mu' \supseteq \mu|_{\mathbf{X}}$. The *answer* to q over M is the set $ans(q, M)$ of $|\mathbf{X}|$-tuples $\mathbf{t}$ for which there is a Δ-substitution μ such that $\mu(\phi(\mathbf{t},\mathbf{Y})) \subseteq M$. The set of all models is denoted by $mods(D,\Sigma)$. The (certain) *answer* to q is the set $ans(q, D, \Sigma) = \bigcap_{M\in mods(D,\Sigma)} ans(q, M)$.

3 Decidability

Hereafter, QEVAL refers to the following decision problem: *Given a database D, a* $\mathsf{datalog}^{\exists,\mathsf{H}}$ *ontology Σ, a $\mathbb{C}^{\mathsf{H}}$ query $q(\mathbf{X})$ with $|\mathbf{X}| = n$, and a tuple $\mathbf{t} \in \mathbf{C}^n$, decide whether $\mathbf{t} \in ans(q, D, \Sigma)$ holds.* In this section, we first introduce the five basic $\mathsf{datalog}^{\exists}$ classes ensuring decidability of QEVAL, as well as some of their generalizations that we need in our technical analysis: j-acyclic [Krötzsch and Rudolph, 2011], w-sticky [Calì *et al.*, 2012b], and w-guarded [Calì *et al.*, 2013]. Then, we define hybrid(-world) extensions of the basic classes, and show that decidability is preserved.

3.1 Overview of Some Decidable $\mathsf{datalog}^{\exists}$ Classes

Fix a $\mathsf{datalog}^{\exists}$ ontology Σ. We assume that different rules of Σ share no variable. A term t occurs in a set A of atoms at position $p[i]$ if there is $\alpha \in A$ s.t. $pred(\alpha) = p$ and $\alpha[i] = t$. Position $p[i]$ is *invaded* by an existential variable X if there is $\rho \in \Sigma$ s.t.: (1) X occurs in $h(\rho)$ at position $p[i]$; or (2) some $y \in \mathbf{UV}(\rho)$ *attacked* by X (i.e., y occurs in $b(\rho)$ only at positions invaded by X) occurs in $h(\rho)$ at position $p[i]$. A universal variable is *protected* if it is attacked by no variable.

Linearity. Ontology Σ belongs to linear if, for each $\rho \in \Sigma$, $b(\rho)$ contains at most one body atom.

Acyclicity. The labeled graph of Σ is $G(\Sigma) = \langle N, A\rangle$, where: (1) $N = \cup_{p\in\mathcal{R}(\Sigma)} pos(p)$; (2) $(p[i], r[j], \forall) \in A$ if there are $\rho \in \Sigma$ and $X \in \mathbf{UV}(\rho)$ s.t. X occurs both in $b(\rho)$ at position $p[i]$ and in $h(\rho)$ at position $r[j]$; and (3) $(p[i], r[j], \exists) \in A$ if there are $\rho \in \Sigma$, $X \in \mathbf{UV}(\rho)$ also occurring in $h(\rho)$, and $Y \in \mathbf{EV}(\rho)$ s.t. both X occurs in $b(\rho)$ at position $p[i]$ and Y occurs in $h(\rho)$ at position $r[j]$. The existential graph of Σ is $G_{\exists}(\Sigma) = \langle N, A\rangle$, where $N = \cup_{\rho\in\Sigma}\mathbf{EV}(\rho)$ and $(X, Y) \in A$ if the rule ρ where Y occurs contains a universal variable attacked by X and occurring in $h(\rho)$. Σ belongs to weakly-acyclic (resp., j-acyclic) if $G(\Sigma)$ (resp., $G_{\exists}(\Sigma)$) has no cycle through an $\exists$-arc (resp., is acyclic).

Guardedness. Σ belongs to guarded if $\rho \in \Sigma$ implies that there is $\alpha \in b(\rho)$ s.t. $\mathbf{UV}(\rho) = vars(\alpha)$. Also, Σ belongs to w-guarded if, for each $\rho \in \Sigma$, there is an atom of $b(\rho)$ containing all the attacked variables of ρ.

Stickiness. A variable X of Σ is *marked* if (1) there is $\rho \in \Sigma$ s.t. X occurs in $b(\rho)$ but not in $h(\rho)$; or (2) there are $\rho, \rho' \in \Sigma$ s.t. a marked variable occurs in $b(\rho)$ at some position $p[i]$ and X occurs in $h(\rho')$ at position $p[i]$ too. Then, Σ is sticky if, for each $\rho \in \Sigma$, X occurs multiple times in $b(\rho)$ implies X is not marked. Also, Σ belongs to w-sticky if, for each $\rho \in \Sigma$, X occurs multiple times in $b(\rho)$ implies X is not marked or X occurs in some position never involved in cycles going through an $\exists$-arc of $G(\Sigma)$.

Shyness. Σ belongs to shy if, for each $\rho \in \Sigma$: (1) X occurs in two different atoms of $b(\rho)$ implies X is protected; and (2) if X and Y occur both in $h(\rho)$ and in two different atoms of $b(\rho)$, then X and Y are not attacked by the same variable.

Proposition 1. *The considered classes are pairwise uncomparable, except for:* linear $\subset$ guarded $\subset$ w-guarded, linear $\subset$ shy, datalog $\subset$ shy, sticky $\subset$ w-sticky, *and* datalog $\subset$ w-acyclic $\subset$ j-acyclic.

Class $\mathcal{C}$	*Data complexity*	(LB)	(UB)	*Combined complexity*	(LB)	(UB)
linearH	in AC_0		$\mathcal{C} \xrightarrow{2}$ linear	PSPACE	linear $\subset \mathcal{C}$	A_3
linearH$^+$	PTIME	datalog $\xrightarrow{4} \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ shy	EXPTIME	datalog $\xrightarrow{4} \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ shy $\vee A_3$
w-acyclicH	PTIME	w-acyclic $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1\vee2}$ j-acyclic	2EXPTIME	w-acyclic $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ j-acyclic
w-acyclicH$^+$	PTIME	w-acyclic $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ j-acyclic	2EXPTIME	w-acyclic $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ j-acyclic
guardedH	PTIME	guarded $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1\vee2}$ guarded	2EXPTIME	guarded $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ guarded
guardedH$^+$	EXPTIME	w-guarded $\xrightarrow{4} \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ w-guarded	2EXPTIME	guarded $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ w-guarded
stickyH	in AC_0		$\mathcal{C} \xrightarrow{2}$ sticky	EXPTIME	sticky $\subset \mathcal{C}$	A_3
stickyH$^+$	PTIME	datalog $\xrightarrow{4} \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ w-sticky	EXPTIME	sticky $\subset \mathcal{C}$	A_3
shyH	PTIME	shy $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1\vee2}$ shy	EXPTIME	shy $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ shy $\vee A_3$
shyH$^+$	PTIME	shy $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ shy	EXPTIME	shy $\subset \mathcal{C}$	$\mathcal{C} \xrightarrow{1}$ shy $\vee A_3$

Table 1: Computational Complexity of QEVAL, where LB and UB stand for lower and upper bound, respectively.

3.2 Decidable Hybrid Extensions

Let $\mathbb{B} = \{$linear, w-acyclic, guarded, sticky, shy$\}$. For each $\mathcal{C} \in \mathbb{B}$, we define the "naive" and "refined" *hybrid(-world)* extension of $\mathcal{C}$, respectively denoted by $\mathcal{C}$H and $\mathcal{C}$H$^+$. Formally, for each $\Sigma \in$ datalog$^{\exists,\text{H}}$, $\Sigma \in \mathcal{C}$H if $std(\Sigma) \in \mathcal{C}$, while $\Sigma \in \mathcal{C}$H$^+$ if $thin(\Sigma) \in \mathcal{C}$, where $thin(\Sigma)$ is obtained from Σ by replacing each closed variable by some constant and then eliminating every atom containing only constants. For example, $g(X, \widehat{\text{Y}}), s(\widehat{\text{Y}}) \rightarrow r(X)$ belongs to linearH$^+$ but not to linearH since $g(X, c) \rightarrow r(X)$ belongs to linear but the rule $g(X, Y), s(Y) \rightarrow r(X)$ does not.

Proposition 2. *For each $\mathcal{C}, \mathcal{C}' \in \mathbb{B}$, $\mathcal{C} \subset \mathcal{C}$H $\subset \mathcal{C}$H$^+$ holds, as well as $\mathcal{C} \subset \mathcal{C}'$ implies both $\mathcal{C}$H $\subset \mathcal{C}'$H and $\mathcal{C}$H$^+ \subset \mathcal{C}'$H$^+$.*

For the decidability analysis, we reduce QEVAL over datalog$^{\exists,\text{H}}$ to QEVAL over datalog$^{\exists}$. To this end, we devise the following algorithm, whose key principle is reminiscent of analogous methods from the literature [Motik *et al.*, 2005]:

Algorithm 1. Reduction A_1 from a hybrid triple $\langle D, \Sigma, q \rangle$

$\dagger$ $\Sigma' \leftarrow \{p(\mathbf{X}^p) \rightarrow \Gamma(\mathbf{X}^p), \bar{p}(\mathbf{X}^p) \ : \ p \in \mathcal{R}(\Sigma)\}$;
$\ddagger$ $\Sigma'' \leftarrow \{std(b(\bar{\rho})), \Gamma(\mathbf{V}^\rho) \rightarrow std(h(\bar{\rho})), \Gamma(\mathbf{C}^\rho) \ : \ \rho \in \Sigma\}$;
$q' \leftarrow std(\bar{q}), \Gamma(\mathbf{V}^q)$;
return $\langle D, \Sigma' \cup \Sigma'', q' \rangle$;

Legend. $\bar{q}$ (resp., $\bar{\rho}$) is obtained from q (resp., ρ) by replacing each predicate p with $\bar{p}$; $\Gamma(t_1, ..., t_n) = \mathsf{c}(t_1), ..., \mathsf{c}(t_n)$, for any $n > 0$; $\mathbf{V}^\diamond = \{V : \widehat{\text{V}} \in vars(\diamond)\}$; $\mathbf{C}^\rho = \mathbf{C} \cap terms(h(\rho))$; $\mathbf{X}^p = X_1, ..., X_{|p|}$; and $\{\mathsf{c}, \bar{p}\} \cap \mathcal{R}(\Sigma) = \emptyset$.

E.g., from $q = \exists X \, \exists \widehat{\text{Y}} \, r(X, \widehat{\text{Y}})$ and $\Sigma = \{r(\widehat{\text{X}}, Y) \rightarrow \exists Z \, r(Y, Z)\}$, we get $q' = \exists X \, \exists Y \, \bar{r}(X, Y), \mathsf{c}(Y)$, and $\Sigma' = \{r(X_1, X_2) \rightarrow \mathsf{c}(X_1), \mathsf{c}(X_2), \ \bar{r}(X_1, X_2)\}$ and $\Sigma'' = \{\bar{r}(X, Y), \ \mathsf{c}(X) \rightarrow \exists Z \, \bar{r}(Y, Z)\}$. Let us now highlight the key properties of A_1.

Lemma 1. *A_1 ensures $ans(q, D, \Sigma) = ans(q', D, \Sigma' \cup \Sigma'')$, and it behaves as follows:* (1) guardedH $\rightarrow$ guarded*;* (2) guardedH$^+ \rightarrow$ w-guarded*;* (3) stickyH$^+ \rightarrow$ w-sticky*;* (4) w-acyclicH$^+ \rightarrow$ jointly-acyclic*; and* (5) shyH$^+ \rightarrow$ shy.

Proof Sketch. Via $\dagger$-rules, each $p(\mathbf{t}) \in D$ gives rise to a twin atom $\bar{p}(\mathbf{t})$, and its constants are collected under the predicate c. Via $\ddagger$-rules, each predicate p is renamed in $\bar{p}$, each variable $\widehat{\text{V}}$ is replaced by V, and the atom containing V is paired with the atom $\mathsf{c}(V)$. This way, known individuals can be separated from anonymous ones, and c-atoms can mimic the semantics of closed variables. Consider now the range of the reduction; due to space limits, we only consider cases (3) and (4). In case (3), let $\Sigma \in$ stickyH$^+$. Each rules $\rho^\dagger$ cannot violate stickiness as no repeated variable appears in $b(\rho^\dagger)$. Now, let X be a variable occurring multiple times in $std(b(\bar{\rho})), \Gamma(\mathbf{V}^\rho)$. We distinguish two cases: (i) X was a standard variable in $b(\rho)$. Then, it also occurred multiple times in $b(\rho)$, Hence, by definition of stickyH$^+$, X was not marked in Σ; and by A_1 it appears multiple times in $std(b(\bar{\rho}))$ only. Hence, X is also not marked in $\Sigma' \cup \Sigma''$. (ii) X was a closed variable in $b(\rho)$. Then, it appears both in $std(b(\bar{\rho}))$ and in $\Gamma(\mathbf{V}^\rho)$. But position c[1] is never involved in cycles going through an $\exists$-arc of $G(\Sigma' \cup \Sigma'')$. Hence, $\Sigma' \cup \Sigma'' \in$ w-sticky. In case (4), let $\Sigma \in$ w-acyclicH$^+$. Note that, given an existential variable X appearing in $\Sigma' \cup \Sigma''$ (and so in Σ), for each i, $p[i]$ and c[1] are not invaded by X. Assume that there is a loop in $G_\exists(\Sigma' \cup \Sigma'')$. Hence, there is a $\ddagger$-rule ρ' s.t. $X \in \mathbf{EV}(\rho')$, and $Y \in \mathbf{UV}(\rho')$ is attacked by X. Now, Y cannot appear in c. Hence, it is a standard variable in $b(\rho)$. Then, Y is attacked by X in Σ. Thus, $\Sigma \notin$ w-acyclicH$^+$. An induction on the length of the cycle concludes the proof. $\square$

The next result follows immediately.

Theorem 3. *Let $\mathcal{C} \in \mathbb{B}$. Then,* QEVAL *for $\mathbb{C}^\text{H}$ queries over $\mathcal{C}$H and $\mathcal{C}$H$^+$ ontologies is decidable.*

4 Computational Complexity

We now study the *combined* and *data* complexity of QEVAL over our hybrid extensions. The former is calculated by considering everything as input, while the latter by considering fixed both the query and the ontology. From our analysis,

Theorem 4. *All results in Table 1 do hold.*

Each entry "$\mathcal{C}_1 \xrightarrow{x} \mathcal{C}_2$" reads as follows: Algorithm x defines a reduction A_x from QEVAL over $\mathcal{C}_1$ to QEVAL over $\mathcal{C}_2$, according to Lemma x possibly combined with Propositions 1 and 2. In particular, if $x \in \{1, 4\}$, then A_x works in polynomial-time. Symbol $\vee$ means that the result admits alternative proofs. Each entry "$\mathcal{C}_1 \subset \mathcal{C}_2$" comes from Proposition 2. Each entry "A_x" means that the upper bound is explicitly given by Algorithm x. The rest of the section is the proof

of Theorem 4. To complete data complexity upper bounds of all naive extensions, consider the following algorithm:

Algorithm 2. Reduction A_2 from a hybrid triple $\langle D, \Sigma, q\rangle$

$\dagger\ \Sigma' \leftarrow \{p(\mathbf{X}^p) \rightarrow \Gamma(\mathbf{X}^p), \bar{p}(\mathbf{X}^p), p_{[c^{|p|}]}(\mathbf{X}^p)\ :\ p \in \mathcal{R}(\Sigma)\}$;
$\ddagger\ \Sigma'' \leftarrow \{std(b(\rho^\omega) \rightarrow h(\rho^\omega), h(\bar{\rho}), \Gamma(\mathbf{C}^\rho))\ :\ \rho \in \Sigma, \omega \in \Omega_\rho\}$;
$q' \leftarrow std(\bar{q}), \Gamma(\mathbf{V}^q)$;
return $\langle D, \Sigma' \cup \Sigma'', q'\rangle$;

Legend. $c^{|p|}$ is the tuple $c, ..., c$ of symbols having length $|p|$; Ω_ρ collects all maps of the form $\omega : vars(\rho) \rightarrow \{c, o\}$ such that $\omega(x) = o$ if $x \in \mathbf{EV}(\rho)$, and $\omega(x) = c$ if $x \in \mathbf{UV}(\rho) \cap \mathbf{V}_c$ (symbols c and o stand for closed and open, respectively); ρ^ω denotes the rule obtained from ρ by replacing each atom of the form $p(\mathbf{X})$ with $p_{[\omega(\mathbf{X})]}(\mathbf{X})$; and the rest is as in A_1.

Basically, A_2 avoids $\Gamma(\mathbf{V}^\rho)$ in rule bodies by encoding in predicates those positions where only known individuals may occur. E.g., from $q = \exists X\, \exists \widehat{\mathrm{Y}}\, r(X, \widehat{\mathrm{Y}})$ and $\Sigma = \{\rho\}$, where $\rho = r(\widehat{\mathrm{X}}, Y) \rightarrow \exists Z\, r(Y, Z)$, we get $\Omega_\rho = \{\omega_1, \omega_2\}$ s.t.: $\omega_1(\widehat{\mathrm{Y}}) = \omega_2(\widehat{\mathrm{Y}}) = c$, $\omega_1(Z) = \omega_2(Z) = o$, $\omega_1(Y) = c$, $\omega_2(Y) = o$. Hence,

$\Sigma' \leftarrow \{r(X_1, X_2) \rightarrow \mathsf{c}(X_1), \mathsf{c}(X_2), \bar{r}(X_1, X_2), r_{[c,c]}(X_1, X_2)\}$
$\Sigma'' \leftarrow \{r_{[c,c]}(X, Y) \rightarrow \exists Z\, r_{[c,o]}(Y, Z), \bar{r}(Y, Z);$
$\quad r_{[c,o]}(X, Y) \rightarrow \exists Z\, r_{[o,o]}(Y, Z), \bar{r}(Y, Z)\}$
$q' \leftarrow \exists X\, \exists Y\ \bar{r}(X, Y), \mathsf{c}(Y)$

By considering any *universal model* U of $D \cup \Sigma' \cup \Sigma''$ — i.e., a representative model of any other [Calì *et al.*, 2013]— subscripts guarantee that whenever there is a substitution μ that maps both the body and the head of a $\ddagger$-rule ρ^ω to U, then $\mu(X) \in terms(D)$ iff $\omega(X) = c$. Then,

Lemma 2. *A_2 ensures $ans(q, D, \Sigma) = ans(q', D, \Sigma' \cup \Sigma'')$. In particular, it behaves as follows: $\mathcal{C}\mathsf{H} \rightarrow \mathcal{C}$ for each $\mathcal{C} \in \mathbb{B}$.*

Although exponential (each rule ρ admits $2^{|\mathbf{UV}(\rho) \cap \mathbf{V}_s|}$ different maps), when combined with Lemma 2, reduction A_2 gives us the desired bounds. To complete with upper bounds, we design the following algorithm:

Algorithm 3. Alternating decision procedure A_3

Input: Hybrid-world triple $\langle D, \Sigma, q\rangle$ where Σ is in normal form
$\Delta \leftarrow terms(D, \Sigma) \cap \mathbf{C}$; / * known * /
$k \leftarrow (1 + |vars(q)|) \cdot \max_{p \in \mathcal{R}(\Sigma)} |p|$;
$I \leftarrow \{a_1, ..., a_k\} \subset \mathbf{C}$ such that $\Delta \cap I = \emptyset$; / * anonymous * /
guess a Δ-substitution $\mu : vars(q) \rightarrow \Delta \cup I$
$Q \leftarrow \mu(atoms(q))$ and $I_q \leftarrow terms(Q) \cap I$
for each $a \in I_q$ **do** / * guess atom α_a introducing each a * /
  guess $\alpha_a \in \{p(\mathbf{t}, a) : p \in \mathcal{R}(\Sigma), \mathbf{t} \in (\Delta \cup I)^{|p|-1}\}$
$\dagger$ **for each** $\alpha \in Q$ **universally do** / * prove each atom α * /
  if $\alpha \in D$ **then** *accept* **else**
    guess $\rho \in \Sigma$ and a Δ-substitution $\mu : vars(\rho) \rightarrow \{\Delta \cup I\}$
    if μ is not compatible with α **then** *reject* **else**
      $Q \leftarrow \mu(b(\rho))$ and **goto** step $\dagger$

Legend. Σ is in normal form if, for each $\rho \in \Sigma$, $|h(\rho)| = 1$, $|\mathbf{EV}(\rho)| \leq 1$, and $|\mathbf{EV}(\rho)| = 1$ implies the existential variable is in the last position; μ is not compatible with α if one of the following occurs: $\mu(h(\rho)) \neq \alpha$; or $X \in \mathbf{EV}(\rho)$, $\mu(X) \in I_q$, and $\alpha \neq \alpha_{\mu(X)}$; or μ maps some non-frontier variable into I_q.

It is a resolution-based algorithm, generally working in alternating polynomial space, hence in exponential time.

Lemma 3. *If Σ is* stickyH$^+$ *or* shyH$^+$*, then A_3 is correct and it runs in* ExpTime*. If $\Sigma \in$* linearH*, then A_3 runs in* PSpace*.*

Proof Sketch. A_3 proves the query q by exploring a "small" (at most exponential) portion of some universal model of $D \cup \Sigma$. In case of linear rules, the algorithm works in nondeterministic polynomial space as step $\dagger$ is universal only once, namely at the very beginning when Q contains the image $\mu(atoms(q))$ of q. □

We close the section by providing missing lower bounds:

Algorithm 4. Reduction A_4 from a standard triple $\langle D, \Sigma, q\rangle$

$\mathbf{V}_p \leftarrow protectedVars(\Sigma)$;
$\Sigma' \leftarrow \{cls(\rho, \mathbf{V}_p)\ :\ \rho \in \Sigma\}$;
return $\langle D, \Sigma', q\rangle$;

Legend. $\mathbf{V}_p$ collects all protected standard universal variables of Σ and $cls(\rho, \mathbf{V}_p)$ replaces each variable $X \in \mathbf{V}_p$ by the closed one $\widehat{\mathrm{X}}$.

Lemma 4. *A_3 ensures $ans(q, D, \Sigma) = ans(q, D, \Sigma')$. In particular, it behaves as follows:* 1) datalog $\rightarrow \mathcal{C}\mathsf{H}^+$ *for each $\mathcal{C} \in \mathbb{B}$; and* 2) w-guarded $\rightarrow$ guardedH$^+$.

Proof Sketch. Equality of certain answers follows by the fact that protected variables implicitly behave as closed ones.

(1) Let $\Sigma \in$ datalog. Then, each variable appearing in Σ is protected. Hence, each rule in Σ' has closed variables only. Thus, Σ' belongs to each refined extension, as the syntactic conditions are enforced over standard variables only.

(2) Let $\Sigma \in$ w-guarded. Let $\rho \in \Sigma$, and ρ' be the corresponding rule in Σ'. Then, by definition of w-guarded, there is an atom in ρ that covers all the non-protected universal variables appearing in $b(\rho)$. Hence, the corresponding atom in ρ' covers all the standard universal variables appearing in $b(\rho')$, as each protected variable is replaced by a closed one. Thus, $\Sigma' \in$ guardedH$^+$. □

5 Expressive Power

We now investigate the expressiveness of datalog$^{\exists,\mathrm{H}}$. After showing that there are simple hybrid ontologies that cannot be expressed by any datalog$^\exists$ one under model equivalence, we consider the classical notion of *program expressive power* [Arenas *et al.*, 2014], also known as *query inseparability*, which relies on answer equivalence and turns out to be more appropriate for OBQA purposes. However, also in this case we can show that datalog$^{\exists,\mathrm{H}}$ is strictly more expressive than datalog$^\exists$. In particular, for both extensions of each basic datalog$^\exists$ class, we prove a strict increase in expressiveness. We close the section by showing that linearH$^+$ is strictly more expressive than the Description Logic $\mathcal{ELH}$ [Brandt, 2004b].

5.1 datalog$^{\exists,\mathrm{H}}$ versus datalog$^\exists$

Two ontologies Σ_1 and Σ_2 are *model-equivalent* (ME), shortly $\Sigma_1 \equiv \Sigma_2$, if $mods(D, \Sigma_1) = mods(D, \Sigma_2)$, for each database D. Accordingly, a class $\mathcal{C}_2$ of ontologies is strictly more expressive (under ME) than $\mathcal{C}_1$, denoted by $\mathcal{C}_1 < \mathcal{C}_2$, if $(M1)$ for each $\Sigma_1 \in \mathcal{C}_1$ there is $\Sigma_2 \in \mathcal{C}_2$ s.t. $\Sigma_1 \equiv \Sigma_2$, and $(M2)$ for some $\Sigma_2 \in \mathcal{C}_2$ there is no $\Sigma_1 \in \mathcal{C}_1$ s.t. $\Sigma_1 \equiv \Sigma_2$.

Theorem 5. *It holds that:* (i) $\mathsf{datalog}^{\exists} < \mathsf{datalog}^{\exists,\mathsf{H}}$*, and* (ii) *both* $\mathcal{C} < \mathcal{C}\mathsf{H}$ *and* $\mathcal{C} < \mathcal{C}\mathsf{H}^{+}$*, for each* $\mathcal{C} \in \mathbb{B}$*.*

Proof. Consider the ontology $\Sigma = \{p(\widehat{\mathrm{X}}) \rightarrow r(\widehat{\mathrm{X}})\}$. We proceed by contradiction. Assume Σ admits a model-equivalent $\mathsf{datalog}^{\exists}$ ontology Σ'. Let $D_\emptyset = \emptyset$. According to Section 2, $M_1 = \{p(1)\}$ is a model of $D_\emptyset \cup \Sigma$ as the interpretation domain of the closed variables is empty. Hence, M_1 is a model of $D_\emptyset \cup \Sigma'$. Let $D_1 = \{p(1)\}$. In this case, $M_1 = \{p(1)\}$ is not a model of $D_1 \cup \Sigma$ as $r(1)$ is required. Thus M_1 is not a model of $D_1 \cup \Sigma'$. But this is not possible for classical first-order theories. In fact, $M_1 \supseteq D_1$. Hence, if M_1 is not a model of $D_1 \cup \Sigma'$ the only reason is that there exists some rule $\rho \in \Sigma'$ that is not satisfied. But since $M_1 \supseteq D_\emptyset$ also holds, this means that M_1 cannot be a model of $D_\emptyset \cup \Sigma'$ as the same rule ρ would be unsatisfied. Hence, (i) follows since $\mathsf{datalog}^{\exists} \subset \mathsf{datalog}^{\exists,\mathsf{H}}$, while (ii) holds since, for each $\mathcal{C} \in \mathbb{B}$, $\mathcal{C} \subset \mathcal{C}\mathsf{H} \subset \mathcal{C}\mathsf{H}^{+}$ by Proposition 2, and $\Sigma \in \mathcal{C}\mathsf{H}$. □

We now consider a smoother notion of expressiveness. Two ontologies Σ_1 and Σ_2 are *answer-equivalent* (AE), shortly $\Sigma_1 \cong \Sigma_2$, if $ans(q, D, \Sigma_1) = ans(q, D, \Sigma_2)$, for each database D and for each query q. Hence, if two ontologies are model-equivalent, then they are also answer-equivalent (i.e., $\Sigma_1 \equiv \Sigma_2$ implies $\Sigma_1 \cong \Sigma_2$, for each Σ_1 and Σ_2). Similarly, a class $\mathcal{C}_2$ of ontologies is strictly more expressive (under AE) than $\mathcal{C}_1$, denoted by $\mathcal{C}_1 \prec \mathcal{C}_2$, if $(A1)$ for each $\Sigma_1 \in \mathcal{C}_1$ there is $\Sigma_2 \in \mathcal{C}_2$ s.t. $\Sigma_1 \cong \Sigma_2$, and $(A2)$ for some $\Sigma_2 \in \mathcal{C}_2$ there is no $\Sigma_1 \in \mathcal{C}_1$ s.t. $\Sigma_1 \cong \Sigma_2$. Note that, if $\mathcal{C}_1 < \mathcal{C}_2$ (resp., $\mathcal{C}_1 \prec \mathcal{C}_2$), then condition $(A1)$ (resp., $(M2)$) is guaranteed.

By Lemma 4, ontology $\Sigma = \{p(\widehat{\mathrm{X}}) \rightarrow r(\widehat{\mathrm{X}})\}$ in the proof of Theorem 5 admits an answer-equivalent $\mathsf{datalog}^{\exists}$ ontology. Indeed, it is the output of reduction A_3 when it takes the ontology $std(\Sigma) = \{p(X) \rightarrow r(X)\}$ as input. Hence, to prove the next result, we need a stronger argument.

Theorem 6. *It holds that:* (i) $\mathsf{datalog}^{\exists} \prec \mathsf{datalog}^{\exists,\mathsf{H}}$*, and* (ii) $\mathcal{C} \prec \mathcal{C}\mathsf{H}$*,* $\mathcal{C} \prec \mathcal{C}\mathsf{H}^{+}$*, and also* $\mathsf{datalog} \prec \mathcal{C}\mathsf{H}^{+}$*, for each basic class* $\mathcal{C} \in \mathbb{B}$*.*

Proof. Consider $\Sigma_h = \{p(\widehat{\mathrm{X}}) \rightarrow r(\widehat{\mathrm{X}})\} \cup \{\rightarrow \exists Y\, p(Y)\}$. We proceed by contradiction. Assume Σ_h admits an answer-equivalent $\mathsf{datalog}^{\exists}$ ontology Σ'_h. Let $q_1 = \exists X\, p(X), r(X)$ and $D_c = \{p(c)\}$, for each constant $c \in \mathbf{C}$. According to Section 2, $ans(q_1, D_c, \Sigma_h) = \{\langle\rangle\}$, and therefore also $ans(q_1, D_c, \Sigma'_h) = \{\langle\rangle\}$. Since Σ' is a standard first-order theory, this means that $D_c \cup \Sigma'_h \models q_1$, and therefore that $\Sigma'_h \models \{p(c) \rightarrow q_1\}$, or equivalently $mods(\Sigma'_h) \subseteq mods(\{p(c) \rightarrow q_1\})$. Hence, we have that $mods(\Sigma'_h) \subseteq \bigcap_{c \in \mathbf{C}} mods(\{p(c) \rightarrow q_1\})$. But the common models are exactly those of $\phi_1 = \exists Y p(Y) \rightarrow q_1$. Therefore, $mods(\Sigma'_h) \subseteq mods(\phi_1)$. Let $q_2 = \exists X p(X)$ and $D_\emptyset = \emptyset$. Clearly, $ans(q_2, D_\emptyset, \Sigma_h) = \{\langle\rangle\}$, and therefore also $ans(q_2, D_\emptyset, \Sigma'_h) = \langle\rangle$. But this means that $\Sigma'_h \models q_2$, or equivalently $mods(\Sigma'_h) \subseteq mods(q_2)$. By combining the above results, we have $mods(\Sigma'_h) \subseteq mods(\phi_1) \cap mods(q_2)$. But the common models are exactly those of q_1. This means that $mods(\Sigma'_h) \subseteq mods(q_1)$, from which we get $\Sigma'_h \models q_1$, implying that $ans(q_1, D_\emptyset, \Sigma'_h) = \{\langle\rangle\}$. But this is not possible since $ans(q_1, D_\emptyset, \Sigma_h) = \emptyset$. Hence, (i) follows since $\mathsf{datalog}^{\exists} \subset \mathsf{datalog}^{\exists,\mathsf{H}}$, while (ii) holds since, for each $\mathcal{C} \in \mathbb{B}$, $\mathcal{C} \subset \mathcal{C}\mathsf{H} \subset \mathcal{C}\mathsf{H}^{+}$ by Proposition 2, and $\Sigma_h \in \mathcal{C}\mathsf{H}$, and since $\mathsf{datalog} \prec \mathcal{C}\mathsf{H}^{+}$ holds by Lemma 4. □

5.2 $\mathsf{linearH}^{+}$ versus $\mathcal{ELH}$

We now show that $\mathsf{linearH}^{+}$ is strictly more expressive than $\mathcal{ELH}$ [Brandt, 2004a; 2004b], even if we focus on $\mathsf{linearH}^{+}$ ontologies with *bounded-rules* (namely, both arities and number af atoms of each rule are bounded by some integer constant), in which case the combined complexity of QEVAL drops to NP as the complexity of QEVAL for $\mathbb{C}$ queries over $\mathcal{ELH}$. (Note that $\mathcal{ELH}$ is not expressible in $\mathsf{linearH}$.) In particular, we provide a polynomial time transformation that maps $\mathcal{ELH}$ ontologies into answer-equivalent $\mathsf{linearH}^{+}$ ones. This also shows that $\mathcal{ELH}$ is no more succinct than $\mathsf{linearH}^{+}$.

In DLs, rules are called *inclusions*, which in $\mathcal{ELH}$ are of the form: $C \sqsubseteq D$; $C \sqcap D \sqsubseteq E$; $R \sqsubseteq S$; $C \sqsubseteq \exists R.D$; $\exists R.D \sqsubseteq C$; where C, D, E are concepts, and R, S are roles. According to the semantics of DLs, they are model-equivalent (hence answer-equivalent) to the following existential rules [Baader *et al.*, 2003], respectively: (i) $C(X) \rightarrow D(X)$; (ii) $C(X), D(X) \rightarrow E(X)$; (iii) $R(X,Y) \rightarrow S(X,Y)$; (iv) $C(X) \rightarrow \exists Y\, R(X,Y), D(Y)$ (v) $R(X,Y), D(Y) \rightarrow C(X)$. Only rules of the form (i), (iii), and (iv) are linear.

To obtain a $\mathsf{linearH}^{+}$ ontology answer-equivalent to an $\mathcal{ELH}$ one, a possible way is to "close" join variables in the body of non-linear rules, i.e., of the form (ii) and (v). This would preserve soundness, but not necessarily completeness. Hence, to guarantee answer equivalence, one should complement such (hybrid) rules with new linear ones that "bypass" propagations inhibited by closed variables. Formally,

Theorem 7. *Under answer-equivalence,* $\mathsf{linearH}^{+}$ *with bounded-rules is strictly more expressive than* $\mathcal{ELH}$*. In particular, for each* $\mathcal{ELH}$ *ontology, an equivalent* $\mathsf{linearH}^{+}$ *one of quadratic size can be constructed in polynomial time.*

Proof Sketch. Given an $\mathcal{ELH}$ ontology Σ in $\mathsf{datalog}^{\exists}$ form, we construct a $\mathsf{datalog}^{\exists,\mathsf{H}}$ ontology Σ' as follows: (0) Let $\Sigma' = \emptyset$; (1) Add to Σ' each rule of Σ of the form (i), (iii) or (iv); (2) For each rule of Σ of the form (ii) (resp., (v)), add to Σ' the hybrid rule $C(\widehat{\mathrm{X}}), D(\widehat{\mathrm{X}}) \rightarrow E(\widehat{\mathrm{X}})$ (resp., $R(X, \widehat{\mathrm{Y}}), D(\widehat{\mathrm{Y}}) \rightarrow C(X)$); (3) For each pair (B, A) of unary predicates (i.e., concepts) occurring in Σ, add to Σ' the standard "bypass" rule $B(X) \rightarrow A(X)$, provided that Σ logically entails the rule $B(X) \rightarrow A(X)$, namely whether $\Sigma \models B \sqsubseteq A$ (B is subsumed by A in Σ) in DLs terminology. By construction, Σ' is $\mathsf{linearH}^{+}$. Also, the addition of bypass rules makes Σ' answer-equivalent to Σ (they share the same universal models). This completes our reduction, which works in polynomial time, since it is known that also concept subsumption in $\mathcal{ELH}$ can be performed in polynomial time [Brandt, 2004a]. Regarding the size of Σ', it suffices to observe that $|\Sigma'| = |\Sigma|$ at the end of step (2), and also that the number of rules added at step (3) are at most quadratic in the number of concepts occurring in Σ. To conclude our proof, we consider the $\mathsf{linearH}^{+}$ ontology $\Sigma = \{p(\widehat{\mathrm{X}}),\, s(\widehat{\mathrm{Y}}) \rightarrow g(\widehat{\mathrm{X}}, \widehat{\mathrm{Y}})\}$. It is well-known that Σ cannot be expressed in $\mathcal{ELH}$, as it defines the so-called cross-product, namely $p \times s \sqsubseteq g$. □

6 Related Work

Interest in reconciling open- and closed-world semantics has a long history [Cadoli *et al.*, 1990]. Since then, different paradigms have been proposed: *epistemic and modal operators* [Donini *et al.*, 1992; Calvanese *et al.*, 2007], *hybrid knowledge bases* [Motik *et al.*, 2005; Rosati, 2005; 2006; Eiter *et al.*, 2008; Krötzsch *et al.*, 2008; Motik and Rosati, 2010; Knorr *et al.*, 2011; Libkin and Sirangelo, 2011; Bajraktari *et al.*, 2017], *closed predicates* [Seylan *et al.*, 2009; Lutz *et al.*, 2013; 2015; Ngo *et al.*, 2016], and *nominal schemas* [Krötzsch *et al.*, 2011; Krötzsch and Rudolph, 2014].

In case of monotonic Horn DLs, modal operator K behaves as closed variables. Indeed, axiom $\mathsf{K}C \sqsubseteq D$ is answer-equivalent to the rule $C(\widehat{\mathrm{x}}) \rightarrow D(\widehat{\mathrm{x}})$. Hybrid KBs typically combine DLs and rule-based formalisms by enforcing syntactic safety condition, while closed predicates are those whose extensions have to be interpreted as complete. So, they are less related to our framework, although we borrowed from them some useful tool, as said in Section 3. Nominal schemas, instead, represent the proposal which is closest to ours. Intuitively, a nominal variable $\{z\}$ is represents some arbitrary nominal (i.e., known individual). When occurring in the left-hand side of a concept inclusion, $\{z\}$ behaves as the closed variable $\widehat{\mathrm{z}}$. Indeed, axiom $\exists hasParent.\{z\} \sqcap \exists hasParent.\exists married.\{z\} \sqsubseteq C$ is model-equivalent to $hasParent(X, \widehat{\mathrm{z}})$, $hasParent(X, Y)$, $married(Y, \widehat{\mathrm{z}}) \rightarrow C(X)$. But in DLs, $\{z\}$ may also be existentially quantified to mimic disjunction among nominals.

Concerning expressiveness, different notions have been also considered in the literature. In [Gottlob *et al.*, 2014], Σ_1 and Σ_2 are gr-equivalent if $ans(q, D, \Sigma_1) = ans(q, D, \Sigma_2)$, for each database D and query $q \in \mathbb{G}$, where $\mathbb{G}$ collects ground queries, i.e., all variable-free atoms. Under this notion guarded $\prec_{gr}$ datalog, and $\mathsf{datalog}^{\exists,\mathsf{H}}$ is no more expressive than $\mathsf{datalog}^{\exists}$ (the latter obtained via a minor modification of Algorithm 1). Indeed, closed variables do not increase the so-called *query expressivity* [Rudolph and Thomazo, 2015], defined by fixing a special predicate goal as the only possible ground query. In [Gottlob *et al.*, 2018], (Σ_1, q_1) and (Σ_2, q_2) are re-equivalent if $ans(q_1, D, \Sigma_1) = ans(q_2, D, \Sigma_2)$, for each database D. Then, $\langle \mathsf{guarded}, \mathbb{C}\rangle \prec_{re} \langle \mathsf{datalog}, \mathbb{G}\rangle$ and also $\langle \mathsf{sticky}, \mathbb{C}\rangle \preceq_{re} \langle \emptyset, \mathbb{UC}\rangle$, where $\mathbb{UC}$ is the class of union of conjunctive queries. Differently from program expressive power, however, these notions are more suitable to compare ontology formalisms from a computational viewpoint rather than from a knowledge representation one. Indeed, re-equivalence coincides with the so-called query rewritability.

7 Future Work and Conclusion

In conclusion, closed variables represent a very natural, flexible and effective extension of standard existential rules. In the future, we would like to investigate whether our naive or refined extensions can express other ontology languages, as well as to close a question that has been left (partially) open in Theorems 5 and 6 concerning the expressivity of $\mathcal{C}\mathsf{H}$ vs. $\mathcal{C}\mathsf{H}^{+}$ by varying $\mathcal{C}$. Indeed, so far, what is known is a strict increase in expressivity in the two cases exhibiting a jump in data complexity from AC_0 to PTIME, namely when $\mathcal{C} \in \{\mathsf{linear}, \mathsf{sticky}\}$. Also, it would be reasonable to extend the computational analysis to other known classes. Interestingly, concerning the latter point, while moving to decidable "abstract" (i.e., not recognizable) classes of rules [Baget *et al.*, 2011], such as fes generalizing w-acyclic, we realized that there are ontologies in $\mathsf{fes}\mathsf{H}$ that are not mapped to fes via Algorithm 1; hence, a separate approach is needed here. Also, one could study the impact of stratified negation in rules and queries, for reasoning even on the anonymity of individuals. As for nominal schemas in DLs, existentially quantified closed variables can be certainly considered to mimic some form of disjunction. Finally, implementing closed variables in some existing $\mathsf{datalog}^{\exists}$ system as well as testing performances on real-world ontologies are also tasks in our agenda.

Acknowledgments

The paper has been partially supported by the MISE under project "S2BDW" (n. F/050389/01-03/X32) - "PON I&C 2014-20, and by Regione Calabria under project "DLV Large Scale" (CUP J28C17000220006) - POR Calabria 2014-20.

References

[Alviano and Pieris, 2015] Mario Alviano and Andreas Pieris. Default negation for non-guarded existential rules. In *Proc of PODS*, pages 79–90, 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.

[Arenas *et al.*, 2014] Marcelo Arenas, Georg Gottlob, and Andreas Pieris. Expressive languages for querying the semantic web. In *Proc. of PODS*, pages 14–26, 2014.

[Baader *et al.*, 2003] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The description logic handbook*. 2003.

[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.

[Bajraktari *et al.*, 2017] Labinot Bajraktari, Magdalena Ortiz, and Mantas Simkus. Clopen knowledge bases: Combining description logics and answer set programming. In *Proc. of DL*, 2017.

[Bárány *et al.*, 2014] Vince Bárány, Georg Gottlob, and Martin Otto. Querying the guarded fragment. *LMCS*, 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.

[Bourhis *et al.*, 2016] Pierre Bourhis, Marco Manna, Michael Morak, and Andreas Pieris. Guarded-based disjunctive tuple-generating dependencies. *ACM TODS*, 41(4):27:1–27:45, 2016.

[Brandt, 2004a] Sebastian Brandt. On subsumption and instance problem in ELH w.r.t. general tboxes. In *Proc. of DL*, 2004.

[Brandt, 2004b] Sebastian Brandt. Polynomial time reasoning in a description logic with existential restrictions, GCI axioms, and - what else? In *Proc. of ECAI*, pages 298–302, 2004.

[Cadoli *et al.*, 1990] Marco Cadoli, Francesco M. Donini, and Marco Schaerf. Closed world reasoning in hybrid systems. In *Proc. of ISMIS*, pages 474–481, 1990.

[Calì *et al.*, 2009] 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.*, 2012a] Andrea Calì, Georg Gottlob, and Thomas Lukasiewicz. A general datalog-based framework for tractable query answering over ontologies. *JWS*, 14:57–83, 2012.

[Calì *et al.*, 2012b] 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.*, 2007] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Eql-lite: Effective first-order query processing in description logics. In *Proc. of IJCAI*, pages 274–279, 2007.

[Ceri *et al.*, 1989] Stefano Ceri, Georg Gottlob, and Letizia Tanca. What you always wanted to know about datalog (and never dared to ask). *TKDE*, 1(1):146–166, 1989.

[Donini *et al.*, 1992] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Andrea Schaerf, and Werner Nutt. Adding epistemic operators to concept languages. In *Proc. of KR*, pages 342–353, 1992.

[Eiter *et al.*, 2008] Thomas Eiter, Giovambattista Ianni, Thomas Lukasiewicz, Roman Schindlauer, and Hans Tompits. Combining answer set programming with description logics for the semantic web. *AIJ*, 172(12-13):1495–1539, 2008.

[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.

[Gottlob *et al.*, 2014] Georg Gottlob, Sebastian Rudolph, and Mantas Simkus. Expressiveness of guarded existential rule languages. In *Proc. of PODS*, pages 27–38, 2014.

[Gottlob *et al.*, 2018] Georg Gottlob, Andreas Pieris, and Mantas Simkus. The impact of active domain predicates on guarded existential rules. *Fundam. Inform.*, 159(1-2):123–146, 2018.

[Knorr *et al.*, 2011] Matthias Knorr, José Júlio Alferes, and Pascal Hitzler. Local closed world reasoning with description logics under the well-founded semantics. *AIJ*, 175(9-10):1528–1554, 2011.

[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.

[Krötzsch and Rudolph, 2014] Markus Krötzsch and Sebastian Rudolph. Nominal schemas in description logics: complexities clarified. In *Proc. of KR*, pages 308–317, 2014.

[Krötzsch *et al.*, 2008] Markus Krötzsch, Sebastian Rudolph, and Pascal Hitzler. ELP: tractable rules for OWL 2. In *Proc. of ISWC*, pages 649–664, 2008.

[Krötzsch *et al.*, 2011] Markus Krötzsch, Frederick Maier, Adila Krisnadhi, and Pascal Hitzler. A better uncle for OWL: nominal schemas for integrating rules and ontologies. In *Proc. of WWW*, pages 645–654, 2011.

[Leone *et al.*, 2012] Nicola Leone, Marco Manna, Giorgio Terracina, and Pierfrancesco Veltri. Efficiently computable Datalog$^{\exists}$ programs. In *Proc. of KR*, pages 13–23, 2012.

[Libkin and Sirangelo, 2011] Leonid Libkin and Cristina Sirangelo. Data exchange and schema mappings in open and closed worlds. *JCSS*, 77(3):542–571, 2011.

[Lutz *et al.*, 2013] Carsten Lutz, Inanç Seylan, and Frank Wolter. Ontology-based data access with closed predicates is inherently intractable(sometimes). In *Proc. of IJCAI*, pages 1024–1030, 2013.

[Lutz *et al.*, 2015] Carsten Lutz, Inanç Seylan, and Frank Wolter. Ontology-mediated queries with closed predicates. In *Proc. of IJCAI*, pages 3120–3126, 2015.

[Motik and Rosati, 2010] Boris Motik and Riccardo Rosati. Reconciling description logics and rules. *J. ACM*, 57(5):30:1–30:62, 2010.

[Motik *et al.*, 2005] Boris Motik, Ulrike Sattler, and Rudi Studer. Query answering for OWL-DL with rules. *JWS*, 3(1):41–60, 2005.

[Ngo *et al.*, 2016] Nhung Ngo, Magdalena Ortiz, and Mantas Simkus. Closed predicates in description logics: Results on combined complexity. In *Proc. of KR*, pages 237–246, 2016.

[Ortiz, 2013] Magdalena Ortiz. Ontology based query answering: The story so far. In *Proc. of AMW*, 2013.

[Rosati, 2005] Riccardo Rosati. On the decidability and complexity of integrating ontologies and rules. *JWS*, 3(1):61–73, 2005.

[Rosati, 2006] Riccardo Rosati. Dl+log: Tight integration of description logics and disjunctive datalog. In *Proc. of KR*, pages 68–78, 2006.

[Rudolph and Thomazo, 2015] Sebastian Rudolph and Michaël Thomazo. Characterization of the expressivity of existential rule queries. In *Proc. of IJCAI*, pages 3193–3199, 2015.

[Seylan *et al.*, 2009] Inanç Seylan, Enrico Franconi, and Jos de Bruijn. Effective query rewriting with ontologies over dboxes. In *Proc. of IJCAI*, pages 923–925, 2009.