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\(^3\) 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\(^2\) [Calì et al., 2009], an emerging family of ontology languages complementing the expressive power of Description Logics (DLs) [Baader et al., 2003]. Indeed, datalog\(^2\) 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 \(D \cup Σ\) 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 \(D \cup Σ\) only. The resulting language, called datalog\(^3\)\(^*\), 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 \(ρ_1 = \text{good}(X) → ∃Y \text{hasPrice}(X, Y)\) and \(ρ_2 = \text{good}(X), \text{hasPrice}(X, Y) → \text{auctionable}(X)\), where \(Y\) is a closed variable. Given \(D_0 = \{\text{good}(\text{ferrari250})\}, Σ_0 = \{ρ_1, ρ_2\}, \) and the queries \(q_1 = ∃X ∃Y \text{hasPrice}(X, Y)\) and \(q_2 = ∃X ∃Y \text{hasPrice}(X, Y), \text{auctionable}(X)\), Clearly, \(q_1\) is entailed by \(D_0 \cup Σ_0\). But \(ρ_2\) is not since \(M = D_0 \cup \{\text{hasPrice}(\text{ferrari250}, 10)\}\) is a possible model of \(D_0 \cup Σ_0\). Indeed, 10 is not a reference price for \(\text{ferrari250}\) 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\(^3\)\(^*\) 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\(^3\) 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\(^3\)\(^*\).

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

- For each basic datalog\(^3\) class \(C\), we consider a “naive” and a “refined” extension, denoted by \(CH\) and \(CH^+\), respectively. In naive extension, the syntactic conditions underlying \(C\) 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 \(CH\) preserves the same data and combined complexity of each basic datalog\(^2\) class \(C\). Likewise, this holds with shy\(^+\) and w-acyclic\(^+\) w.r.t. their standard counterparts. Differently, guarded\(^+\) and sticky\(^+\) exhibit an increase in data complexity, while only linear\(^+\) has an increase in both data and combined complexity. (Section 4.)
We prove that datalog$_{\text{wH}}$ is (resp., CH and CH$^+$ are) strictly more expressive than datalog$^3$ (resp., each basic class $C$). In particular, going back to our running example, there is no datalog$^3$ ontology that, independently from the database at hand, behaves as $\Sigma_0$ w.r.t. both $q_1$ and $q_2$. Also, each CH$^+$ is even strictly more expressive than datalog$_{\text{wH}}$ (Section 5.1.).

We show that the well-known Description Logic $\mathcal{ELH}$ [Brandt, 2004b] is captured by linearH$^+$, even if we only focus on linearH$^+$ 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, linearH$^+$ keeps a lower computational complexity, compared to other datalog$^3$ classes that can express this DL, namely guarded and its extensions. (Section 5.2.)

## 2 Existential Rules with Closed Variables

**Basics.** Let $C$ (constants or individuals) and $V$ (variables) be pairwise disjoint discrete sets of terms. A variable $(x, y, \ldots)$ is either standard $(X, Y, \ldots)$ or closed-(world) $(\bar{X}, \bar{Y}, \ldots)$. We denote by $V$, and $V$, the set of standard and closed variables, respectively. An atom $\alpha$ is a labeled tuple $p(t)$, where $p = \text{pred}(\alpha)$ is a predicate symbol, $t = t_1, \ldots, t_m$ is a tuple of terms, $m = |p|$ is the arity of $p$ or $\alpha$, and $\alpha[i] = t_i$. Given a finite domain $\Delta \subseteq C$ of “known” individuals, a $\Delta$-substitution is any map $\mu : V \rightarrow C$ such that $\bar{X} \in V$, implies $\mu(\bar{X}) \subseteq \Delta$. 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 datalog$^3$ rule $\rho$ is a logical implication of the form $\exists X \forall Y (\phi(X, Y) \rightarrow \exists Z \psi(X, Z))$ — with $X \cup Y \subseteq V$ and $Z \subseteq V$, whose body (resp., head) $b(\rho) = \phi(X, Y)$ (resp., $h(\rho) = \psi(X, Z)$) is a conjunction (or set of) atoms, possibly with constants. As usual, the head is nonempty. Universal and existential variables are respectively denoted by $\forall V$ and $\exists V$. The set $X$ is known as the frontier of $\rho$. If no closed variable is in $\rho$, then it is also a datalog$^2$ rule; and if even $\forall V(\rho) = \emptyset$, then it is also a datalog rule. A datalog$^3$ ontology $\Sigma$ is any finite set of datalog$^3$ rules. We denote by $\mathcal{R}(\Sigma)$ the set of predicates occurring in $\Sigma$. A position $p(\rho)$ is defined as a predicate of $\mathcal{R}(\Sigma)$ and its $i$-th attribute. Let $\text{pos}(p) = \{p[1], \ldots, p[|p|]\}$. A $C^0$ (hybrid conjunctive query) query is an expression of the form $q(\{X\}) = \exists Y \phi(X, Y)$, where $\phi$ is as above. In case $q$ contains no closed variable, it is also a $C$ (conjunctive) query. For a “structure” $\sigma$ over atoms (set, rule, query, ...), if $\bar{X}$ occurs in $\sigma$, then $X$ does not occur in $\sigma$. Also, $\text{atoms}(\sigma)$, $\text{terms}(\sigma)$, $\text{vars}(\sigma)$ and $\text{std}(\sigma)$ respectively denote the set of atoms in $\sigma$, the set of terms in $\text{atoms}(\sigma)$, the set of variables in $\text{atoms}(\sigma)$, and the structure built from $\sigma$ by replacing each $X$ with $X$. 

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

## 3 Decidability

Hereafter, $\text{QEVAL}$ refers to the following decision problem: Given a database $D$, a datalog$^3$ ontology $\Sigma$, a $C^0$ query $q(X)$ with $|X| = n$, and a tuple $t \in C^n$, decide whether $t \in \text{ans}(q, D, \Sigma)$ holds. In this section, we first introduce the five basic datalog$^2$ classes ensuring decidability of $\text{QEVAL}$, as well as some of their generalizations that we need in our technical analysis: j-acyclic [Krotzsch and Rudolph, 2011], w-sticky [Cali et al., 2012b], and w-guarded [Cali 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 datalog$^2$ Classes

Fix a datalog$^2$ ontology $\Sigma$. We assume that different rules of $\Sigma$ 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. $\text{pred}(\alpha) = p$ 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) $X$ occurs in any $\forall V(\rho)$ attacked by $X$ (i.e., $y$ occurs in $b(\rho)$ only if it is invaded by $X$) and $X$ occurs in $h(\rho)$ at position $p[i]$. A universal variable is protected if it is attacked by no variable.

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

**Acylicity.** The labeled graph of $\Sigma$ is $G(\Sigma) = \langle N, A \rangle$, where: (1) $N = \cup_{p \in \mathcal{R}(\Sigma)} \text{pos}(p)$; (2) $\{p[i], r[j], \forall\} \in A$ if there are $\rho \in \Sigma$ and $X \in \forall V(\rho)$ 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 $\rho \in \Sigma$, $X \in \forall V(\rho)$ also occurring in $h(\rho)$, and $Y \in \forall V(\rho)$ 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 $\Sigma$ is $G_\exists(\Sigma) = \langle N, A \rangle$, where $N = \cup_{p \in \mathcal{R}(\Sigma)} \text{pos}(p)$ and $(X, Y) \in A$ if the rule $\rho$ where $Y$ occurs contains a universal variable attacked by $X$ and occurring in $h(\rho)$. $\Sigma$ 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.** $\Sigma$ belongs to guarded if $\rho \in \Sigma$ implies that there is $\alpha \in b(\rho)$ s.t. $\forall V(\rho) = \text{vars}(\alpha)$. Also, $\Sigma$ belongs to w-guarded if, for each $\rho \in \Sigma$, there is an atom of $b(\rho)$ containing all the attacked variables of $\rho$.

**Stickiness.** A variable $X$ of $\Sigma$ 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, $\Sigma$ is sticky if, for each $\rho \in \Sigma$, $X$ occurs multiple times in $b(\rho)$ implies $X$ is not marked. Also, $\Sigma$ 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.** $\Sigma$ 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 incomparable, except for: linear $\subseteq$ guarded $\subseteq$ w-guarded, linear $\subseteq$ shy, datalog $\subseteq$ shy, sticky $\subseteq$ w-sticky, and datalog $\subseteq$ w-acyclic $\subseteq$ j-acyclic.
3.2 Decidable Hybrid Extensions

Let $\mathbb{B} = \{\text{linear, w-acyclic, guarded, sticky, shy}\}$. For each $C \in \mathbb{B}$, we define the "naive" and "refined" hybrid (-world) extension of $C$, respectively denoted by $\mathcal{CH}$ and $\mathcal{CH}^+$.

For each $C \subseteq \mathbb{B}$, we say that $C$ is notated if $\mathcal{C} \subseteq \mathcal{C}$ holds.

For the decidability analysis, we reduce $\text{QEVAL}$ over $\mathcal{CH}$ to $\text{Q-EVAL}$ over $\mathcal{CH}^+$. This to 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 $(D, \Sigma, q)$

<table>
<thead>
<tr>
<th>$\Sigma'$</th>
<th>$\Sigma''$</th>
<th>$\Sigma'''$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${\rho(X^p) \rightarrow \Gamma(X^p), \rho(X^p) : p \in \mathcal{R}(\Sigma)}$</td>
<td>${\text{std}(\rho(\bar{q})), \Gamma(V') \rightarrow \text{std}(\rho(\bar{q})), \Gamma(V') : \rho \in \Sigma}$</td>
<td>$\bar{q} \rightarrow \text{std}(\bar{q}), \Gamma(V')$</td>
</tr>
</tbody>
</table>

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

E.g., from $q = \exists X \exists Y \exists Z \exists r(X, Y) \land \Sigma = \{r(X, Y) \rightarrow \exists Z r(X, Z)\}$, we get $q' = \exists X \exists Y \exists Z \exists r(X, Y, Z)$, $\Gamma(r(X, Y), r(X, Z), r(X, Z), r(X, Z))$ and $\Sigma''' = \{(X, Y) : c(X, Y) \rightarrow \exists Z r(X, Z)\}$. Let us now highlight the key properties of $A_1$.

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

**Proof Sketch.** Via $\downarrow$-rules, each $p(t) \in D$ gives rise to a twin atom $p(t)$, and its constants are collected under the predicate $c$. Via $\uparrow$-rules, each predicate $p$ is renamed in $\rho$, each variable $V$ is replaced by $V$, and the atom containing $V$ is paired with the atom $c(V)$. This way, known individuals can be separated from anonymous ones, and $c$-atoms can mimic the semantics of closed variables.

The next result follows immediately.

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

4 Computational Complexity

We now study the combined and data complexity of $\text{QEVAL}$ 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, we have:

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

Each entry $C_1 \rightarrow C_2$ reads as follows: Algorithm $x$ defines a reduction $A_x$ from $\text{QEVAL}$ over $C_1$ to $\text{QEVAL}$ over $C_2$, 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 $\forall$ means that the result admits alternative proofs. Each entry $C_1 \subseteq C_2$ comes from Proposition 2. Each entry $A_1$ means that the upper bound is explicitly given by Algorithm $x$. The rest of the section is the proof.

Table 1: Computational Complexity of $\text{QEVAL}$, where LB and UB stand for lower and upper bound, respectively.

<table>
<thead>
<tr>
<th>Class $C$</th>
<th>Data complexity</th>
<th>(LB)</th>
<th>(UB)</th>
<th>Combined complexity</th>
<th>(LB)</th>
<th>(UB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>linear</td>
<td>in $\mathcal{AC}_0$</td>
<td>$C \rightarrow$ linear</td>
<td>PSPACE</td>
<td>linear $\subset C$</td>
<td>$A_1$</td>
<td>$A_2$</td>
</tr>
<tr>
<td>linear</td>
<td>PTIME</td>
<td>datalog $\downarrow C$</td>
<td>$C \rightarrow$ shy</td>
<td>EXPTIME</td>
<td>datalog $\downarrow C$</td>
<td>$C \rightarrow$ shy $\lor A_1$</td>
</tr>
<tr>
<td>$w$-acyclic</td>
<td>PTIME</td>
<td>$w$-acyclic $\subset C$</td>
<td>$C \rightarrow$ j-acyclic</td>
<td>2EXPTIME</td>
<td>$w$-acyclic $\subset C$</td>
<td>$C \rightarrow$ j-acyclic</td>
</tr>
<tr>
<td>$w$-acyclic</td>
<td>PTIME</td>
<td>$w$-acyclic $\subset C$</td>
<td>$C \rightarrow$ j-acyclic</td>
<td>2EXPTIME</td>
<td>$w$-acyclic $\subset C$</td>
<td>$C \rightarrow$ j-acyclic</td>
</tr>
<tr>
<td>guarded</td>
<td>PTIME</td>
<td>$\text{guarded} \subset C$</td>
<td>$C \rightarrow$ guarded</td>
<td>2EXPTIME</td>
<td>$\text{guarded} \subset C$</td>
<td>$C \rightarrow$ guarded</td>
</tr>
<tr>
<td>guarded</td>
<td>EXPTIME</td>
<td>$\text{w-guarded} \subset C$</td>
<td>$C \rightarrow$ w-guarded</td>
<td>2EXPTIME</td>
<td>$\text{w-guarded} \subset C$</td>
<td>$C \rightarrow$ w-guarded</td>
</tr>
<tr>
<td>sticky</td>
<td>in $\mathcal{AC}_0$</td>
<td>$C \rightarrow$ sticky</td>
<td>EXPTIME</td>
<td>sticky $\subset C$</td>
<td>$A_1$</td>
<td>$A_2$</td>
</tr>
<tr>
<td>sticky</td>
<td>PTIME</td>
<td>datalog $\downarrow C$</td>
<td>$C \rightarrow$ w-sticky</td>
<td>EXPTIME</td>
<td>sticky $\subset C$</td>
<td>$A_1$</td>
</tr>
<tr>
<td>sticky</td>
<td>PTIME</td>
<td>datalog $\downarrow C$</td>
<td>$C \rightarrow$ w-sticky</td>
<td>EXPTIME</td>
<td>sticky $\subset C$</td>
<td>$A_1$</td>
</tr>
<tr>
<td>sticky</td>
<td>PTIME</td>
<td>datalog $\downarrow C$</td>
<td>$C \rightarrow$ w-sticky</td>
<td>EXPTIME</td>
<td>sticky $\subset C$</td>
<td>$A_1$</td>
</tr>
</tbody>
</table>

1678
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 $(D, \Sigma, q)$

\[
\begin{align*}
\Sigma' & \leftarrow \{ p(X^0) \rightarrow \Gamma(X^0), p_i(X^0), p_{c\ell}(X^0) : p \in \mathcal{R}(\Sigma) \}; \\
\Sigma'' & \leftarrow \{(\text{std}(b(\rho^o)) \rightarrow h(\rho^o), h(\rho), \Gamma(C^o)) : p \in \Sigma, \omega \in \Omega_p \}; \\
q' & \leftarrow \text{std}(q), \Gamma(V^o);
\end{align*}
\]

return $(D, \Sigma' \cup \Sigma'', q').$

Legend. $c(p)$ is the tuple $c, ..., c$ of symbols having length $|p|;$
$\Omega_p$ collects all maps of the form $\omega : \text{vars}(\rho) \rightarrow \{c, o\}$ such that $\omega(x) = o$ if $x \in \mathcal{E}(\rho),$ and $\omega(x) = c$ if $x \in \mathcal{U}(\rho)$ \cap $V_c$ (symbols $c$ and $o$ stand for closed and open, respectively).

$\rho^o$ denotes the rule obtained from $\rho$ by replacing each atom of the form $p(X)$ with $p_{c\ell}(X) | X$; and the rest is as in $A_1.$

Basically, $A_2$ avoids $\Gamma(V^o)$ in rule bodies by encoding in predicates those positions where only known individuals may occur. E.g., from $q = \exists X \exists Y r(X, Y)$ and $\Sigma = \{\rho\}$, where $\rho = r(X, Y) \rightarrow \exists Z r(Y, Z),$ we get $\Omega_{\rho} = \{\omega_1, \omega_2\}$ such that $\omega_1(Y) = \omega_2(Y) = c,$ $\omega_1(Z) = \omega_2(Z) = o,$ $\omega_1(X) = c,$ $\omega_2(X) = o.$ Hence, $\Sigma' \leftarrow \{r(X_1, X_2) \rightarrow c(X_2), c(X_2), r(X_1, X_2), r_{c\ell}(X_1, X_2)\}$ $\Sigma'' \leftarrow \{r_{c\ell}(X_1, X_2) \rightarrow \exists Z r_{c\ell}(Y, Z), r(Y, Z), r_{c\ell}(X_1, X_2)\}$ $q' \leftarrow \exists X \exists Y r(Y, X), c(Y)$. By considering any universal model $U$ of $\Sigma \cup \Sigma' \cup \Sigma''$ — i.e., a representative model of any other [Cali et al., 2013]— subscripts guarantee that whenever there is a substitution $\mu$ that maps both the body and the head of a $\frac{3}{2}$-rule $\rho^o$ to $U$, then $\mu(X) \notin \text{terms}(D)$ iff $\omega(x) = c$. Then,

Lemma 2. $A_2$ ensures $\text{ans}(q, \xi, \Sigma) = \text{ans}(q, D, \Sigma' \cup \Sigma'').$

In particular, it behaves as follows: $\text{CH} \rightarrow \text{CH}$ for each $C \in \mathcal{B}.$

Although exponential (each rule $\rho$ admits $2^{\mathcal{U}(\rho)} \cdot |\mathcal{V}|$ 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 $(D, \Sigma, q)$ where $\Sigma$ is in normal form $\Delta \rightarrow \text{terms}(D, \Sigma) \cap C; \ / \ \ / \ \ \ / \ \ \ /$ $k \leftarrow \{\text{vars}(\mu)\} : \max_{\rho \in \mathcal{R}(\Sigma)} |\rho|; \ / \ \ / \ \ / \ \ /$ $I \leftarrow \{a_1, ..., a_k\} \subseteq C$ such that $\Delta \cap I = \emptyset; \ / \ \ / \ \ / \ \ /$ guess a $\Delta$-substitution $\mu : \text{vars}(\mu) \rightarrow \Delta \cup I.$ $Q \leftarrow \mu(\text{atoms}(q)); \ / \ \ / \ \ / \ \ /$ for each $a \in I$ do $\ / \ \ / \ \ / \ \ / \ \ /$ $\text{guess } \alpha = \{p(t, a) : p \in \mathcal{R}(\Sigma), t \in (\Delta \cup I)\}; \ / \ \ / \ \ / \ \ /$ $\text{for each } a \in Q$ universally do $\ / \ \ / \ \ / \ \ /$ $\text{if } a \in D \text{ then accept else }$ $\text{guess } \rho \in \xi$ and a $\Delta$-substitution $\mu : \text{vars}(\mu) \rightarrow \{\Delta \cup I\}$ if $\mu$ is not compatible with $\alpha$ then reject else $Q \leftarrow \mu(b(\rho))$ and goto step $\dagger$

Legend. $\Sigma$ is in normal form if, for each $p \in \Sigma, |h(\rho)| = 1,$ $\mathcal{E}(\rho) \leq 1,$ and $\mathcal{E}(\rho) = 1$ implies the existential variable is in the last position; $\mu$ is not compatible with $\alpha$ if one of the following occurs: $\mu(h(\rho)) \neq \alpha; \text{ or } X \in \mathcal{E}(\rho),$ $\mu(X) \notin I_p,$ and $\alpha \neq \alpha_{\mu(X)}; \text{ or } \mu$ 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 $\Sigma$ is sticky $H^+$ or shy $H^+$, then $A_3$ is correct and it runs in $\text{ExpTime}$. If $\Sigma \in \text{linear}H$, then $A_3$ runs in $\text{PSPACE}$.

Proof Sketch. $A_3$ proves the query $q$ by exploring a “small” (at most exponential) portion of some universal model of $\Sigma$. 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 $\mu(\text{atoms}(q))$ of $q$.

We close the section by providing missing lower bounds:

Algorithm 4. Reduction $A_4$ from a standard triple $(D, \Sigma, q)$

\[
\begin{align*}
\mathcal{V}_p & \leftarrow \text{protectedVars}(\Sigma); \\
\Sigma' & \leftarrow \{\text{cls}(\rho, \mathcal{V}_p) : \rho \in \Sigma\}; \\
\text{return } (D, \Sigma', q).
\end{align*}
\]

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

Lemma 4. $A_4$ ensures $\text{ans}(q, D, \Sigma) = \text{ans}(q, D, \Sigma').$ In particular, it behaves as follows: (1) $\text{datalog} \rightarrow \text{CH}^+ \text{ for each } C \in \mathcal{B}; \text{ and } 2\text{-guarded} \rightarrow \text{guarded} H^+.$

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

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

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

5 Expressive Power

We now investigate the expressiveness of datalog$^{3,18}$. After showing that there are simple hybrid ontologies that cannot be expressed by any datalog$^3$ 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$^{3,18}$ is strictly more expressive than datalog$^3$. In particular, for both extensions of each basic datalog$^3$ class, we prove a strict increase in expressiveness. We close the section by showing that linear$H^+$ is strictly more expressive than the Description Logic $\mathcal{ELH}$ [Brandt, 2004b].

5.1 datalog$^{3,18}$ versus datalog$^3$

Two ontologies $\Sigma_1$ and $\Sigma_2$ are model-equivalent (ME), shortly $\Sigma_1 \equiv \Sigma_2$, if $\text{mods}(D, \Sigma_1) = \text{mods}(D, \Sigma_2)$, for each database $D$. Accordingly, a class $C_2$ of ontologies is strictly more expressive (under ME) than $C_1$, denoted by $C_1 < C_2$, if $(M1)$ for each $\Sigma_1 \in C_1$ there is $\Sigma_2 \in C_2$ s.t. $\Sigma_1 \equiv \Sigma_2$, and $(M2)$ for some $\Sigma_2 \in C_2$ there is no $\Sigma_1 \in C_1$ s.t. $\Sigma_1 \equiv \Sigma_2$.
Theorem 5. It holds that: (i) $\text{datalog}^3 \prec \text{datalog}^{3,n}$, and (ii) both $C \prec CH$ and $C \prec CH^+$, for each $C \in \mathbb{B}$.

Proof. Consider the ontology $\Sigma = \{ p(\bar{x}) \rightarrow r(\bar{x}) \}$. We proceed by contradiction. Assume $\Sigma$ admits a model-equivalent datalog$^3$ ontology $\Sigma'$. Let $D_0 = \emptyset$. According to Section 2, $M_1 = \{p(1)\}$ is a model of $D_0 \cup \Sigma$ as the interpretation domain of the closed variables is empty. Hence, $M_1$ is a model of $D_0 \cup \Sigma'$. 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 $p \in \Sigma'$ that is not satisfied. But since $M_1 \supseteq D_1$, also, $M_1$ cannot be a model of $D_0 \cup \Sigma'$ as the same rule $p$ would be unsatisfied. Hence, (i) follows since datalog$^3 \prec \text{datalog}^{3,n}$, while (ii) holds since, for each $C \in \mathbb{B}$, $C \subseteq CH \subseteq CH^+$ by Proposition 2, and $\Sigma_h \notin CH$, and since datalog$^3 \prec CH^+$ holds by Lemma 4.

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

We now show that linearH$^+$ is strictly more expressive than $\mathcal{ELH}$ [Brandt, 2004a; 2004b], even if we focus on linearH$^+$ ontologies with bounded-rules (namely, both arities and number of 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 $C$ queries over $\mathcal{ELH}$. (Note that $\mathcal{ELH}$ is not expressible in linearH$^+$.) In particular, we provide a polynomial time transformation that maps $\mathcal{ELH}$ ontologies into answer-equivalent linearH$^+$ ones. This also shows that $\mathcal{ELH}$ is no more succinct than linearH$^+$.

In DLs, rules are called inclusions, which in $\mathcal{ELH}$ are of the form: $C \subseteq D; C \cap D \subseteq E; R \subseteq S; C \subseteq \exists R.D; \exists R.D \subseteq C$; 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) $C(X) \rightarrow \exists Y R(X,Y), D(Y) \rightarrow R(X,Y), D(Y) \rightarrow C(X)$. Only rules of the form (i), (iii) and (iv) are linear.

To obtain a 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, linearH$^+$ with bounded-rules is strictly more expressive than $\mathcal{ELH}$. In particular, for each $\mathcal{ELH}$ ontology, an equivalent linearH$^+$ one of quadratic size can be constructed in polynomial time.

Proof Sketch. Given an $\mathcal{ELH}$ ontology $\Sigma$ in datalog$^3$ form, we construct a datalog$^{3,n}$ ontology $\Sigma'$ as follows: (0) Let $\Sigma' = \emptyset$. (1) Add to $\Sigma'$ each rule of $\Sigma$ of the form (i), (iii) or (iv); (2) For each rule of $\Sigma$ of the form (ii) resp., (v), add to $\Sigma'$ the hybrid rule $C(\bar{x}), D(\bar{x}) \rightarrow E(\bar{x})$ resp., $R(\bar{x}, \bar{y}) \rightarrow C(\bar{x})$; (3) For each pair $(B, A)$ of unary predicates (i.e., concepts) occurring in $\Sigma$, add to $\Sigma'$ the standard “bypass” rule $B(\bar{x}) \rightarrow A(\bar{x})$, provided that $\Sigma$ logically entails the rule $B(\bar{x}) \rightarrow A(\bar{x})$, namely whether $\Sigma \models B \subseteq A (B$ is subsumed by $A$ in $\Sigma$) in DLs terminology. By construction, $\Sigma'$ is linearH$^+$. Also, the addition of bypass rules makes $\Sigma'$ answer-equivalent to $\Sigma$ (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 $\Sigma'$, 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 $\Sigma$. To conclude our proof, we consider the linearH$^+$ ontology $\Sigma = \{p(\bar{x}), s(\bar{y}) \rightarrow g(\bar{x}, \bar{y})\}$. It is well-known that $\Sigma$ cannot be expressed in $\mathcal{ELH}$, as it defines the so-called cross-product, namely $p \times s \subseteq g$.
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; Bajraktar 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 $KC \subseteq D$ is answer-equivalent to the rule $C(\bar{x}) \rightarrow D(\bar{x})$. 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 $\bar{Z}$. Indeed, axiom $\exists \text{hasParent}.\{z\} \land \exists \text{married}.\{z\} \subseteq C$ is model-equivalent to $\exists \text{hasParent}(X, \bar{Z}), \exists \text{hasParent}(X, Y), \exists \text{married}(Y, \bar{Z}) \rightarrow C(X)$. But in DLs, $\{z\}$ may also be existentially quantified to mimic disjunction among nominals.

Concerning expressiveness, different notions have also been considered in the literature. In [Gottlob et al., 2014], $\Sigma_1$ and $\Sigma_2$ are $gr$-equivalent if $\text{ans}(q, D, \Sigma_1) = \text{ans}(q, D, \Sigma_2)$, for each database $D$ and query $q \in G$, where $G$ collects ground queries, i.e., all variable-free atoms. Under this notion guarded $\preceq_{gr}$ datalog, and datalog$^{\Sigma_1}$ is no more expressive than datalog$^{\Sigma_2}$ (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], $(\Sigma_1, q_1)$ and $(\Sigma_2, q_2)$ are re-equivalent if $\text{ans}(q_1, D, \Sigma_1) = \text{ans}(q_2, D, \Sigma_2)$, for each database $D$. Then, $(\text{guarded}, C) \preceq_{re} (\text{datalog}, G)$ and also $(\text{sticky}, C) \preceq_{re} (\emptyset, \forall C)$, where $\forall C$ 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 $CH$ vs. $CH^+$ by varying $C$. 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 $C \in \{\text{linear, sticky}\}$. 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 $\text{fes}$ generalizing $w$-acyclic, we realized that there are ontologies in $\text{fesH}$ that are not mapped to $\text{fes}$ 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 datalog$^+$ 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


