Restricted Chase (Non)Termination for Existential Rules with Disjunctions

David Carral, Irina Dragoste, Markus Krötzsch

Abstract

The *restricted chase* is a sound and complete algorithm for conjunctive query answering over ontologies of disjunctive existential rules. We develop *acyclicity* conditions to ensure its termination. Our criteria cannot always detect termination (the problem is undecidable), and we develop the first *cyclicity* criteria to show *non*-termination of the restricted chase. Experiments on real-world ontologies show that our acyclicity notions improve significantly over known criteria.

1 Introduction

Answering conjunctive queries (CQs) over ontologies is an important reasoning task with many applications in knowledge representation and data management. A flurry of research efforts have significantly improved our understanding of this problem, and led to concrete solutions for many ontologies based either on description logics (DLs) [Calvanese *et al.*, 2007; Stefanoni *et al.*, 2014; Calvanese *et al.*, 2014; Bienvenu *et al.*, 2016] or on existential rules [Baget *et al.*, 2011; Cuenca Grau *et al.*, 2013; Calì *et al.*, 2013].

Nevertheless, the problem remains very challenging in theory and in practice. For the popular OWL Web Ontology Language (based on DLs), it is still unknown if the problem is decidable [Rudolph and Glimm, 2010]. For existential rules, which are syntactically closer to CQs, query answering is a basic reasoning task, but it is also undecidable. At the same time, existential rules are too weak to capture the non-deterministic features of ontology languages like OWL.

In this work, we study existential rules with disjunction, which generalise many expressive ontology languages. Our results apply to DLs (and OWL) and many rule languages.

Example 1. *The following rules capture basic part-whole relationships (meronomy) and disjunctive information.*

$$\mathit{Bicycle}(x) \to \exists v.\mathit{hasPart}(x, v) \wedge \mathit{Wheel}(v) \tag{1}$$

$$\mathit{Wheel}(x) \to \mathit{SpokeWheel}(x) \vee \mathit{DiscWheel}(x) \tag{2}$$

$$\mathit{SpokeWheel}(x) \to \exists w.\mathit{partOf}(x, w) \wedge \mathit{Bicycle}(w) \tag{3}$$

$$\mathit{hasPart}(x, y) \to \mathit{partOf}(y, x) \tag{4}$$

$$\mathit{partOf}(x, y) \to \mathit{hasPart}(y, x) \tag{5}$$

Similar modelling can be found, e.g., in many medical ontologies. This particular example can also be expressed in OWL.

The *oblivious chase* provides a sound and complete reasoning algorithm for this logic [Bourhis *et al.*, 2016], computing rule entailments in a bottom-up fashion and branching on disjunctive choices. Unfortunately, the chase often fails to terminate. In Example 1, each bicycle introduces a new wheel (1), which could be a spoke wheel (2), which may introduce a new bicycle (3) – an infinite chain of bikes and wheels. To avoid this, we define a *restricted chase* that tries to reuse existing elements before introducing new ones.

However, the restricted chase may still fail to terminate, and determining this is undecidable in general [Beeri and Vardi, 1981]. We therefore develop new *acyclicity* criteria that can ensure termination in the spirit of previous approaches for rules without disjunctions [Marnette, 2009; Krötzsch and Rudolph, 2011; Cuenca Grau *et al.*, 2013]. Existing notions can actually be applied to disjunctive rules by replacing $\vee$ with $\wedge$ for testing acyclicity. However, this only works for an oblivious chase, where non-termination is preserved if rules are logically strengthened. Our restricted chase is more challenging, since additional entailments can lead to termination. Indeed, the restricted chase terminates for Example 1 since rule (3) never needs to be applied to a wheel that was newly introduced by rule (1). Without the rules (4) and (5), this would not be true, and an infinite chase would be required. Only Carral *et al.* [2016] seem to have studied this setting so far, but their results are specific to the non-disjunctive DL Horn-$\mathcal{SRIQ}$ and do not apply to existential rules.

Acyclicity is sufficient but not necessary for termination. In experiments, we are often left with a significant amount of non-acyclic ontologies of which we cannot say anything. We therefore also develop a *cyclicity* notion to detect non-termination of the restricted chase. To the best of our knowledge, this is the first proposal for such a criterion. It allows us to show that a majority of practical ontologies is such that the chase will either terminate over all possible sets of input facts, or will necessarily be infinite over at least some inputs.

In summary, our main contributions are:

- we propose *restricted joint acyclicity* as a simple criterion for restricted chase termination on disjunctive rules,
- we design more general criteria by extending *model-faithful acyclicity* and *model-summarising acyclicity*,

- we characterise the complexity of query answering and of checking acyclicity for each of the new notions,
- we introduce the first criterion for checking non-termination of the restricted chase,
- we empirically evaluate our results on a large corpus of real-world ontologies.

Experiments suggest that our new notions can significantly improve over the state of the art, and that, moreover, a large subset of non-acyclic ontologies are indeed cyclic. This extended report includes an appendix with additional details on proofs omitted from the conference version.

2 Preliminaries

We consider a standard first-order signature based on mutually disjoint, countably infinite sets of *constants* $\mathbf{C}$, *function symbols* $\mathbf{F}$, *variables* $\mathbf{V}$, and *predicates* $\mathbf{P}$. Every function symbol or predicate f has an *arity* $ar(f) \geq 0$. *Terms* are built from $\mathbf{C}$, $\mathbf{V}$, and $\mathbf{F}$ as usual. We write lists of variables $\langle x_1, \dots, x_n \rangle$ as $\boldsymbol{x}$, and similarly for term lists $\boldsymbol{t}$. We treat such lists as sets when order is irrelevant. We will use function symbols only to represent auxiliary terms introduced during reasoning (via *skolemisation*; see below). An *atom* is a formula $p(\boldsymbol{t})$ with $p \in \mathbf{P}$ and $ar(p) = |\boldsymbol{t}|$. A term or formula is *ground* if it contains no variables. A *fact* is a ground atom. A *disjunctive existential rule*, or simply *rule*, is a formula

$$\forall \boldsymbol{x}, \boldsymbol{y}. (B[\boldsymbol{x}, \boldsymbol{y}] \to \textstyle\bigvee_{i=1}^{n} \exists \boldsymbol{v}_i . H_i[\boldsymbol{x}, \boldsymbol{v}_i]) \qquad (6)$$

where $n \geq 1$, and where B (the *body*) and H_i (the *heads*) are non-empty conjunctions of atoms that contain only variables from $\boldsymbol{x} \cup \boldsymbol{y}$ and $\boldsymbol{x} \cup \boldsymbol{v}_i$, respectively, and no constants or function symbols. We omit the universal quantifiers when writing rules. The variables $\boldsymbol{x}$ are called *frontier variables*. A rule is *deterministic* if $n = 1$ and *nondeterministic* otherwise; it is *generating* if it contains an existential variable and *non-generating* otherwise. When convenient, we treat conjunctions, such as B and H_i, as sets of atoms.

We consider finite *sets of rules* $\mathcal{R}$, where we assume without loss of generality that each variable in $\mathcal{R}$ occurs only in the scope of a single quantifier within a single rule (‡). An *instance* $\mathcal{I}$ is a finite set of function-free facts. A *program* is a pair $\langle \mathcal{R}, \mathcal{I} \rangle$ with $\mathcal{R}$ a rule set and $\mathcal{I}$ an instance.

A *Boolean conjunctive query* (*BCQ*) is a formula $\exists \boldsymbol{v}.Q[\boldsymbol{v}]$, where Q is a conjunction of function-free, constant-free atoms using only variables from $\boldsymbol{v}$. A program $\langle \mathcal{R}, \mathcal{I} \rangle$ *entails* a BCQ $\exists \boldsymbol{v}.Q[\boldsymbol{v}]$ if $\bigwedge \mathcal{R} \wedge \bigwedge \mathcal{I} \models \exists \boldsymbol{v}.Q[\boldsymbol{v}]$ under standard first-order semantics. It is well known that arbitrary conjunctive query answering can be reduced to BCQ entailment.

A *(ground) substitution* σ is a partial function from variables to (ground) terms. We denote finite substitutions as $[x_1/t_1, \dots, x_n/t_n]$ or $[\boldsymbol{x}/\boldsymbol{t}]$, and we set $\sigma(x) = x$ if σ is undefined for x. Given a term or formula F, we write $F\sigma$ for the expression obtained by concurrently replacing all unbound occurrences of variables x in F by $\sigma(x)$.

The *skolemisation* $sk(\rho)$ of a rule ρ as in (6) is the formula $\forall \boldsymbol{x}, \boldsymbol{y}. (B[\boldsymbol{x}, \boldsymbol{y}] \to \bigvee_{i=1}^{n} sk(H_i)[\boldsymbol{x}])$ where $sk(H_i)$ is obtained from H_i by replacing each variable $v \in \boldsymbol{v}_i$ by the term $f_v(\boldsymbol{x})$, where f_v is a fresh *skolem function symbol* specific to v (which, by (‡), occurs in only one quantifier).

The Restricted Chase We consider a *restricted* version of the *disjunctive chase* [Bourhis *et al.*, 2016], where rules are only applicable if their heads are not satisfied by existing facts. Moreover, we impose an order of rule applications that defers the use of generating rules.

Definition 1. *Consider a rule ρ of form (6), a ground substitution σ defined exactly on the variables $\boldsymbol{x} \cup \boldsymbol{y}$ from ρ, and a set of facts $\mathcal{F}$. Then $\langle \rho, \sigma \rangle$ is* applicable *to $\mathcal{F}$ if (a) $\mathcal{F} \models B\sigma$, and (b) $\mathcal{F} \not\models \bigvee_{i=1}^{n} \exists \boldsymbol{v}_i . H_i \sigma$. In this case, the* result of applying *$\langle \rho, \sigma \rangle$* to *$\mathcal{F}$ is the set $\{\mathcal{F} \cup sk(H_1)\sigma, \dots, \mathcal{F} \cup sk(H_n)\sigma\}$, consisting of all sets of facts obtained from $\mathcal{F}$ by adding the skolemised, σ-instantiated atoms of some head of ρ.*

Note that condition (a) is the same as $B\sigma \subseteq \mathcal{F}$, while (b) states that there is no extension $\hat{\sigma}$ of σ to the variables $\boldsymbol{v}_i$ such that $H_i\hat{\sigma} \subseteq \mathcal{F}$, for any $i \in \{1, \dots, n\}$. The chase is the result of a possibly infinite process of recursive rule applications:

Definition 2. *A* chase tree *of $\langle \mathcal{R}, \mathcal{I} \rangle$ is a (possibly infinite) tree; each node is labelled by a set of facts, such that:*

(1) the root is labelled with $\mathcal{I}$,

(2) if a node labelled $\mathcal{F}$ has n children labelled $\mathcal{F}_1, \dots, \mathcal{F}_n$, then there is a rule $\rho \in \mathcal{R}$ and substitution σ such that $\{\mathcal{F}_1, \dots, \mathcal{F}_n\}$ is the result of applying $\langle \rho, \sigma \rangle$ to $\mathcal{F}$,

(3) if a node α is labelled with $\mathcal{F}$ and $\langle \rho, \sigma \rangle$ is applicable to $\mathcal{F}$, then each path starting from α contains a node in which $\langle \rho, \sigma \rangle$ is no longer applicable due Definition 1 (b),

(4) generating rules are only applied in a node where no non-generating rule of $\mathcal{R}$ is applicable.

The result of a restricted chase is the (possibly infinite) set of all (possibly infinite) sets of facts that are obtained as the union of all set of facts along some path.

Condition (3) ensures fair, exhaustive application, while (4) improves the rule application strategy to reduce the amount of applicable generating rules. Bourhis *et al.* [2016] omit (4) and Definition 1 (b), but restricted disjunctive chase algorithms were considered before, e.g., by Deutsch and Tannen [2002].

Example 2. *Consider the rules in Example 1, where we use first letters to abbreviate predicates from now on, and the instance $\{B(c)\}$. We obtain a finite chase tree with two leaves $\mathcal{F}_1 = \{B(c), hP(c, f_v(c)), W(f_v(c)), pO(f_v(c), c), SW(f_v(c))\}$ and $\mathcal{F}_2 = \{B(c), hP(c, f_v(c)), W(f_v(c)), pO(f_v(c), c), DW(f_v(c))\}$. In particular, rule (3) is not applicable to $\mathcal{F}_1$, since the facts $pO(f_v(c), c)$ and $B(c)$ already satisfy the head of this rule for the substitution $[x/c]$.*

In general, the chase tree and even its result is not unique, since the order of rule applications may matter, but we get the following consequence of well-known results:

Fact 1. *A program $\langle \mathcal{R}, \mathcal{I} \rangle$ entails a BCQ $\exists \boldsymbol{v}.Q$ iff $\mathcal{F} \models \exists \boldsymbol{v}.Q$ holds for all sets of facts $\mathcal{F}$ in the result of an (arbitrary) restricted chase.*

If the chase terminates, the chase tree is finite, and the result is the set of all (finite) leaf labels. In this case, Fact 1 leads to a decision procedure for BCQ entailment. Unfortunately, chase termination is undecidable even for deterministic rules [Beeri and Vardi, 1981]. We therefore study sound but incomplete tests for restricted chase termination.

3 Restricted Joint Acyclicity

We first consider a limited but easy-to-check condition to ensure chase termination. As noted in the introduction, we could apply existing criteria for the deterministic skolem chase, but the result is often unsatisfactory:

Example 3. *On the skolemisation of the rules in Example 1, the oblivious chase may produce an infinite set of facts* $\{hP(c, f_v(c)),\ W(f_v(c)),\ pO(f_v(c), c),\ SW(f_v(c)),\ pO(f_v(c), f_w(f_v(c))),\ B(f_w(f_v(c))),\ hP(f_w(f_v(c)), f_v(c)),\ hP(f_w(f_v(c)), f_v(f_w(f_v(c)))),\ W(f_v(f_w(f_v(c)))), \ldots\}$. *Definition 1 (b) does not reduce the derivable facts, since the presence of one skolem term does not allow us to omit the introduction of another skolem term.*

To address such cases, we extend the notion of *joint acyclicity* (JA), proposed by Krötzsch and Rudolph [2011] for deterministic rules. JA proceeds in two steps: (1) for each existential variable v, we compute a set Ω_v of predicate *positions*, to which values generated for v might propagate; (2) we build a dependency graph to show if the value generated for a variable v may participate in the generation of a new value for a variable w. A rule set is JA if this graph has no cycles. To adapt this idea to the restricted chase, we treat $\vee$ as $\wedge$ for (1) and add a new blocking criterion for (2). For the next definition, recall assumption (‡) from Section 2.

Definition 3. *A* position *in a predicate p is a pair $\langle p, i\rangle$ with $i \in \{1, \ldots, ar(p)\}$. A term t* occurs at position *$\langle p, i\rangle$ in a formula φ if φ contains an atom $p(\boldsymbol{s})$ with $s_i = t$. Given a rule ρ and variable z, a* body position (head position) *of z is a position in the body (in a head) of ρ at which z occurs.*

For a rule set $\mathcal{R}$ and an existential variable v of $\mathcal{R}$, a set Ω_v of positions is defined recursively: (i) Ω_v contains all head positions of v in a rule of $\mathcal{R}$; (ii) for every universally quantified variable x in $\mathcal{R}$, if Ω_v contains every body position of x, then Ω_v also contains every head position of x.

Example 4. *For the rules of Example 1, we obtain* $\Omega_v = \{\langle hP, 2\rangle, \langle W, 1\rangle, \langle SW, 1\rangle, \langle DW, 1\rangle, \langle pO, 1\rangle\}$ *and* $\Omega_w = \{\langle pO, 2\rangle, \langle B, 1\rangle, \langle hP, 1\rangle\}$.

For a set of facts $\mathcal{F}$ and set of rules $\mathcal{R}$, let $\mathcal{R}_{\mathsf{dng}}(\mathcal{F})$ be the set of facts obtained from $\mathcal{F}$ by exhaustive application of all deterministic, non-generating rules of $\mathcal{R}$. For an existentially quantified variable v, let H_v denote the (unique) head conjunction v occurs in, and let B_v be the body of the corresponding rule.

Definition 4. *The* restricted dependency graph *of a rule set $\mathcal{R}$ has the existentially quantified variables of $\mathcal{R}$ as its nodes, and an edge $v \to w$ if w occurs in a rule ρ_w of form* (6) *with a frontier variable $x \in \boldsymbol{x}$ such that*

(a) all body positions of x occur in Ω_v, and

(b) for the set $\mathcal{F} = (B_w \cup H_v[v/x] \cup B_v)\sigma$, where σ replaces all variables z by distinct constants c_z, we have $\mathcal{R}_{\mathsf{dng}}(\mathcal{F}) \not\models (\exists \boldsymbol{w}.H_w)\sigma$.

$\mathcal{R}$ is restricted jointly acyclic *(RJA) if its restricted dependency graph has no cycles.*

Example 5. *We consider Example 1 and rename variables by subscripting rule numbers. The existential dependency graph has nodes v and w. For the potential edge $v \to w$, condition (a) of Definition 4 is satisfied for rule* (3) *and variable x_3, since $\langle SW, 1\rangle \in \Omega_v$ (see Example 4). However, the set $\mathcal{F}$ of condition (b) is $\{SW(c_3), B(c_1), hP(c_1, c_3), W(c_3)\}$ and the application of deterministic, non-generating rules* (4) *and* (5) *yields $\mathcal{R}_{\mathsf{dng}}(\mathcal{F}) = \mathcal{F} \cup \{pO(c_3, c_1)\}$, which satisfies rule* (3)*. Hence, there is no edge $v \to w$. A similar argument rules out $w \to v$. For $v \to v$ and $w \to w$, condition (a) is not satisfied. The graph therefore has no edges and the rule set is RJA.*

RJA rules lead to a finite chase tree. The following property is the essence of this claim.

Lemma 2. *If the restricted chase tree of $\langle \mathcal{R}, \mathcal{I}\rangle$ contains a term $f_w(\boldsymbol{t})$ where $t_i = f_v(\boldsymbol{s})$ for some $i \in \{1, \ldots, |\boldsymbol{t}|\}$, then there is an edge $v \to w$ in the restricted dependency graph.*

Proof. We denote rules etc. as in Definition 4, and assume that $f_w(\boldsymbol{t})$ was derived by applying $\langle \rho_w, \theta\rangle$. The sets Ω_v overestimate the possible positions of f_v-terms, hence the applicability of ρ_w to $f_v(\boldsymbol{s})$ implies Definition 4 (a). Moreover, let $\mathcal{F}_w$ be the set of facts as it was when ρ_w was applied in the chase tree. There is a homomorphism $h : \mathcal{F} \to \mathcal{F}_w$ with $h(B_w\sigma) = B_w\theta$ and $h(B_v\sigma)$ the premise of the application of ρ_v that produced $f_v(\boldsymbol{s})$. By our chase strategy, $\mathcal{R}_{\mathsf{dng}}(\mathcal{F}_w) = \mathcal{F}_w$. Hence, if $\mathcal{R}_{\mathsf{dng}}(\mathcal{F}) \models H_w\sigma$, then $\mathcal{F}_w \models h(H_w\sigma) = H_w\theta$, such that $\langle \rho_w, \theta\rangle$ would not be applicable. Since it is, we get $\mathcal{R}_{\mathsf{dng}}(\mathcal{F}) \not\models H_w\sigma$ as claimed. □

Theorem 3. *Deciding BCQ entailment for programs $\langle \mathcal{R}, \mathcal{I}\rangle$ where $\mathcal{R}$ is RJA is co*N2EXPTIME*-complete, even if the arity of predicates is bounded.*

Proof sketch. Membership follows since BCQ non-entailment can be shown by finding a model of $\langle \mathcal{R}, \mathcal{I}\rangle$ that does not satisfy the query. For this we may non-deterministically guess a branch of the chase tree. The maximal nesting depth of function terms in the chase tree is bounded by the number of existentially quantified variables in $\mathcal{R}$, since a greater depth can only be achieved by repeating a function symbol, which would make the restricted dependency graph cyclic by Lemma 2. The maximal number of terms of linear depth is doubly exponential, so there are double exponentially many possible ground facts overall. A set of facts of this size can be computed in 2EXPTIME.

Hardness is established by modifying the construction of a 2EXPTIME Turing machine given for deterministic, weakly acyclic rules by Calì *et al.* [2010]. The construction yields a grid of doubly exponential size, using predicates of arity ≤ 3. Using disjunction in rules, it is not hard to simulate a nondeterministic Turing machine in the same way. □

Theorem 4. *Deciding if $\mathcal{R}$ is RJA is* EXPTIME*-complete, co*NP*-complete if the arity of predicates is bounded, and* P*-complete if the number of variables per rule is bounded.*

Proof. Sets Ω_v can be computed in polynomial time, and there are only polynomially many possible edges and body variables x as in Definition 4 to be considered. However, Definition 4 (b) corresponds to the EXPTIME-complete Datalog reasoning task of checking non-entailment of a set of facts [Dantsin *et al.*, 2001]. The task becomes coNP-complete for predicates of bounded arity: hardness follows from hardness of conjunctive query entailment (rule bodies are CQs); membership follows since there are only polynomially many ground facts over this signature, hence the derivation of any such fact can be represented as a polynomial directed acyclic graph with (true) facts as nodes and edges connecting premises with conclusions, such that fact entailment can be checked by guessing this graph and verifying each rule application. The task becomes P-complete for bounded numbers of variables, since grounding (instantiation of rules with constants) polynomially reduces fact entailment to propositional Horn logic entailment. In all cases, detecting cycles in the (polynomial) dependency graph is possible in P. □

The complexity of Theorem 4 is dominated by the reasoning for condition (b) in Definition 4. In practice, this reasoning task is usually fairly simple, since only a very small number of facts is given, and most rules can be ignored.

4 Restricted Model-Faithful Acyclicity

While RJA is fairly easy to check, it is not sufficient to capture all realistic cases. We therefore develop more general, though also more complex criteria.

Example 6. *We extend Example 1 with the following rules:*

$$\mathit{SpokeWheel}(x_6) \to \exists u.\mathit{hasPart}(x_6, u) \wedge \mathit{Spoke}(u) \qquad (7)$$

$$\mathit{Spoke}(x_7) \to \exists z.\mathit{partOf}(x_7, z) \wedge \mathit{Bicycle}(z) \qquad (8)$$

$$\mathit{hasPart}(x_8, y_8) \wedge \mathit{hasPart}(y_8, z_8) \to \mathit{hasPart}(x_8, z_8) \qquad (9)$$

The resulting rule set still leads to a finite restricted chase for all instances, but it is not RJA. Indeed, the restricted dependency graph contains a cycle $v \to u \to z \to v$. For example, when considering $z \to v$, the set $\mathcal{F}$ in Definition 4 (b) is $\{B(c_1), pO(c_7, c_1), S(c_7)\}$, which cannot entail $W(c_1)$.

For cases as in Example 6, we extend the notion of model-faithful acyclicity (MFA) [Cuenca Grau *et al.*, 2013]. To determine if a set of deterministic rules $\mathcal{R}$ is MFA, one computes the chase on $\langle sk(\mathcal{R}), \mathcal{I}^\star_\mathcal{R}\rangle$, where $\mathcal{I}^\star_\mathcal{R}$ is the *critical instance*, which contains all possible ground facts based on predicates of $\mathcal{R}$ and the single constant symbol $\star$. $\mathcal{R}$ is MFA if this chase terminates without introducing a *cyclic term* $f(\boldsymbol{t})$, which is such that f occurs in the terms $\boldsymbol{t}$. Alternatively, a cyclic term must appear after at most doubly exponentially many steps. Deciding MFA indeed is 2EXPTIME-complete.

MFA uses the fact that the chase terminates on every instance if it terminates on the critical instance. This is not true for the restricted chase, as no rule is applicable in the presence of the critical instance. We therefore consider a relaxed condition of applicability, which, in the spirit of Definition 4 (b), determines a rule's applicability from a smaller set of facts not including the whole critical instance. The body of the rule (instance) that is to be applied can always be assumed as given. Further facts can be obtained from this body's skolem terms, since each skolem function is introduced by one specific rule:

Definition 5. *For a rule set $\mathcal{R}$ and a ground term $t = f_v(\boldsymbol{s})$ using skolem functions from $sk(\mathcal{R})$, the set $\mathcal{F}_t$ contains all ground facts involved in the derivation of facts containing t:*

(1) Let $B[\boldsymbol{x}, \boldsymbol{y}] \to \bigvee_i \exists \boldsymbol{v}_i.H_i[\boldsymbol{x}, \boldsymbol{v}_i]$ be the unique rule that contains v in head disjunct H_k, and consider the substitution $\theta = [\boldsymbol{x}/\boldsymbol{s}, \boldsymbol{y}/\boldsymbol{c}]$, where $\boldsymbol{c}$ is a list of fresh constant symbols not used elsewhere. Then $B\theta \cup sk(H_k)\theta \subseteq \mathcal{F}_t$.

(2) For every functional term $s_j \in \boldsymbol{s}$, we have $\mathcal{F}_{s_j} \subseteq \mathcal{F}_t$.

Example 7. *For the rules from Examples 6 and 1, and term $t = f_u(f_v(b))$, we have $\mathcal{F}_t = \{SW(f_v(b)), hP(f_v(b), t), S(t)\} \cup \mathcal{F}_{f_v(b)}$ with $\mathcal{F}_{f_v(b)} = \{B(b), hP(b, f_v(b)), W(f_v(b))\}$.*

The next example illustrates another difficulty: even if we only take a rule's body into account to check its applicability, we might get much fewer derivations on the critical instance than on other instances.

Example 8. *Consider the rule $\rho : p(x, y) \to \exists v.p(v, v) \vee \exists w.p(y, w)$. On the critical instance, one could try to apply ρ with substitution $[x/\star, y/\star]$. But already the instantiated body $p(\star, \star)$ prevents the application of ρ, since $p(\star, \star) \models \exists v.p(v, v)$. The restricted chase terminates immediately. Yet, on the instance $\{p(a, b)\}$, one can compute an infinite set of facts $\{p(a, b), p(b, f_w(b)), p(f_w(b), f_w(f_w(b))), \ldots\}$, so the restricted chase is not finite in general.*

To handle this issue, we rename distinct occurrences of $\star$.

Definition 6. *Consider a rule $\rho : B[\boldsymbol{x}, \boldsymbol{y}] \to \bigvee_i \exists \boldsymbol{v}_i.H_i[\boldsymbol{x}, \boldsymbol{v}_i]$, and a ground substitution σ defined exactly on $\boldsymbol{x} \cup \boldsymbol{y}$. Let σ' be such that, for all $x \in \mathbf{V}$, $\sigma'(x)$ is $\sigma(x)$ with each occurrence of a constant renamed so that no constant occurs more than once in the image of σ'. The set $\mathcal{B}_{\rho,\sigma}$ is the union of $B\sigma'$ and each of the sets $\mathcal{F}_t$ for which there is a skolem term t in $B\sigma'$. We say that $\langle \rho, \sigma\rangle$ is* blocked *if $\mathcal{R}_{\mathsf{dng}}(\mathcal{B}_{\rho,\sigma}) \models \bigvee_i \exists \boldsymbol{v}_i.H_i\sigma'$.*

Example 9. *Consider the rules of Example 6, and especially rule ρ of (8) under substitution $[x_7/t]$ with $t = f_u(f_v(b))$. Intuitively speaking, t represents a spoke that was introduced as part of wheel $f_v(b)$, which in turn is part of bicycle b. We want to show that ρ does not need to be applied to introduce another bicycle $f_z(t)$. We do not need to rename any constants here, so $\mathcal{B}_{\rho,\sigma} = \{S(t)\} \cup \mathcal{F}_t$ with $\mathcal{F}_t$ as in Example 7. While $\mathcal{B}_{\rho,\sigma}$ does not satisfy the head of ρ yet, we get $\mathcal{R}_{\mathsf{dng}}(\mathcal{B}_{\rho,\sigma}) = \mathcal{B}_{\rho,\sigma} \cup \{hP(b, t), pO(t, f_v(b)), pO(f_v(b), b), pO(t, b)\}$. Therefore $\mathcal{R}_{\mathsf{dng}}(\mathcal{B}_{\rho,\sigma}) \models \exists z.pO(t, z) \wedge B(z)$ and $\langle \rho, \sigma\rangle$ is blocked as expected.*

Our adaptation of MFA conducts a (deterministic) chase on the critical instance, but applies rules only if not blocked.

Definition 7. *For a rule set $\mathcal{R}$, $\mathsf{RMFA}(\mathcal{R})$ is the least set of facts for which $\mathcal{I}^\star_\mathcal{R} \subseteq \mathsf{RMFA}(\mathcal{R})$ and, whenever $\rho : B \to \bigvee_i \exists \boldsymbol{v}_i.H_i$ is a rule in $\mathcal{R}$, and σ is such that $B\sigma \subseteq \mathsf{RMFA}(\mathcal{R})$ and $\langle \rho, \sigma\rangle$ is not blocked, then $sk(H_1)\sigma \cup \ldots \cup sk(H_n)\sigma \subseteq \mathsf{RMFA}(\mathcal{R})$. $\mathcal{R}$ is* restricted model-faithfully acyclic *(RMFA) if $\mathsf{RMFA}(\mathcal{R})$ does not contain a cyclic term.*

Example 10. *The rules of Examples 1 and 6 together are RMFA, as one can easily check along the lines of Example 9.*

Theorem 5. *Deciding if $\mathcal{R}$ is RMFA is 2EXPTIME-complete even if the arity of predicates or the number of variables per rule is bounded. It is EXPTIME-complete if each rule contains at most one frontier variable.*

Proof sketch. Membership in 2ExpTime follows as in Theorem 3 by bounding the possible ground skolem terms. Similarly, rules with one frontier variable lead to unary skolem functions, which can form only exponentially many terms.

2ExpTime-Hardness can be shown as for the case of MFA, where it was done by reduction from the 2ExpTime-hard problem of BCQ entailment checking for weakly acyclic (WA) rules [Cuenca Grau *et al.*, 2013]. The hardness proof for WA in turn is based on a direct Turing machine construction using predicates of bounded arity and rules with a bounded number of variables [Calì *et al.*, 2010]. One may verify that no rule application is ever blocked in this particular construction, so that we can use the same proof for our purposes. ExpTime-hardness can be obtained using a known proof for the case of MFA, which exhibit the same complexity for small frontiers [Cuenca Grau *et al.*, 2013]. In particular, their construction in Lemma 59 works even when using blocking. □

The main correctness property of RMFA is as follows.

Lemma 6. *Let $\pi^\star$ be the function that maps a term t to the term obtained from t by replacing all constants with $\star$. Then, for every term t that occurs in any restricted chase of $\langle\mathcal{R},\mathcal{I}\rangle$ for any instance $\mathcal{I}$, the term $\pi^\star(t)$ occurs in $\mathsf{RMFA}(\mathcal{R})$.*

Proof sketch. We can show that $\pi^\star$ is a homomorphism from any leaf of the chase tree into $\mathsf{RMFA}(\mathcal{R})$. This is done by induction following the rule applications of one branch of the restricted chase. The main observation is that, if $\langle\rho,\sigma\rangle$ is applicable to a set of facts $\mathcal{F}$ during the restricted chase, and we define $\sigma^\star(z) = \pi^\star(\sigma(z))$, for all z, then $\langle\rho,\sigma^\star\rangle$ is not blocked. This in turn is shown by finding a homomorphism from $\mathcal{R}_{\mathsf{dng}}(\mathcal{B}_{\rho,\sigma^\star})$ (Definition 6) into $\mathcal{F}$. □

Theorem 7. *Deciding BCQ entailment for programs $\langle\mathcal{R},\mathcal{I}\rangle$ where $\mathcal{R}$ is RMFA is co*N2ExpTime*-complete, even if the arity of predicates is bounded.*

Proof sketch. The proof is similar as for Theorem 3. By Lemma 6, any chase derives at most doubly exponentially many facts, since this is the maximal number of non-cyclic terms, and $\pi^\star$ preserves non-cyclicity. Hardness follows by Theorem 3, as any RJA rule set can be shown to be RMFA (cf. the relation of JA and MFA [Cuenca Grau *et al.*, 2013]). □

Theorem 5 motivates the search for a simpler test that still extends RJA. We can achieve this by adapting *model-summarising acyclicity* (MSA) to our setting [Cuenca Grau *et al.*, 2013]. This criterion resembles MFA in that a chase on the critical instance is conducted to discover cycles. However, instead of using skolem terms, existential variables now are replaced by fresh constants, and cycles are tracked explicitly during the chase instead of using cyclic terms. Definition 6 can remain unchanged if we treat the fresh constants as nullary "skolem" function symbols in Definition 5 (the recursive case (2) for subterms becomes irrelevant).

Definition 8. *For a rule set $\mathcal{R}$, let S be a binary predicate not used in $\mathcal{R}$, and let θ be a substitution that maps each existentially quantified variable v in $\mathcal{R}$ to a unique fresh constant c_v. $\mathsf{RMSA}(\mathcal{R})$ is the least set of facts for which $\mathcal{I}^\star_\mathcal{R} \subseteq \mathsf{RMSA}(\mathcal{R})$ and, whenever $\rho : B[\boldsymbol{x},\boldsymbol{y}] \to \bigvee_{i=1}^n \exists\boldsymbol{v}_i.H_i[\boldsymbol{x},\boldsymbol{v}_i]$ is a rule in $\mathcal{R}$, and σ is such that $B\sigma \subseteq \mathsf{RMSA}(\mathcal{R})$ and $\langle\rho,\sigma\rangle$ is not blocked, then $H_1\theta\sigma \cup \ldots \cup H_n\theta\sigma \cup \{S(x\sigma, v\theta) \mid x \in \boldsymbol{x}, v \in \boldsymbol{v}_i, 1 \leq i \leq n\} \subseteq \mathsf{RMSA}(\mathcal{R})$.*

$\mathcal{R}$ is restricted model-summarising acyclic *(RMSA) if $\mathsf{RMSA}(\mathcal{R})$ does not contain a directed cycle of S-relations.*

BCQ answering remains as hard as for RJA and RMFA, but recognising RMSA is only as hard as for RJA. The proof is similar to the proofs of Theorems 4 and 5.

Theorem 8. *Deciding if $\mathcal{R}$ is RMSA is* ExpTime*-complete, and* P*-complete if the number of variables per rule is fixed.*

Example 11. *RMSA cannot capture Example 6, but it generalises RJA. Consider the set $\mathcal{R}$ of the rules* (1), (2), (7), *and*

$$\mathit{hasPart}(x,y) \wedge \mathit{BicycleChain}(y) \to \mathit{Bicycle}(x). \qquad (10)$$

$\mathcal{R}$ is RMSA since rule (10) *is never applicable to fresh constants (which do not have parts that are bicycle chains). However, $\mathcal{R}$ is not RJA since* (10) *leads to $\langle\mathit{Bicycle}, 1\rangle \in \Omega_v$.*

5 Proving Nontermination

Even if rules are not acyclic by any of our criteria, they might still have a finite chase. In this section, we introduce a complementary criterion that is sufficient (but not necessary) to show that the chase will be infinite. By combining this with our previous acyclicity notions, we hope to decide the question of chase termination for most practical ontologies.

Sufficient conditions for nontermination can also look for cycles, e.g., by detecting cyclic terms in a chase as done for MFA. The critical instance cannot be used here, since it overestimates what can really be derived repeatedly. However, a cyclic term $f_v(\ldots f_v(\boldsymbol{t})\ldots)$ might indicate nontermination if it was derived in a chase that started from nothing but the facts $B_v \cup \mathit{sk}(H_v)$, where we replace each variable z by a fresh constant c_z (B_v and H_v are the unique body and head for v as in Section 3). Indeed, if this happens, then each application of the rule of v creates a set of facts that eventually enables another application of the same rule that will produce a skolem term of increased nesting depth. This discussion suggests a criterion for showing nontermination of the skolem chase:

Definition 9. *Consider a set $\mathcal{R}$ of deterministic rules, and a rule $\rho : B[\boldsymbol{x},\boldsymbol{y}] \to \exists\boldsymbol{v}.H[\boldsymbol{x},\boldsymbol{v}]$ in $\mathcal{R}$. The set $\mathcal{I}_\rho$ is obtained from $B \cup \mathit{sk}(H)$ by replacing all occurrences of every variable z with a fresh constant c_z. Let $\mathcal{F}_\rho$ be the set of facts obtained by exhaustive application of rules from $\mathit{sk}(\mathcal{R})$ to $\mathcal{I}_\rho$, under the condition that no rule is applied to facts with a cyclic term. Then $\mathcal{R}$ has a ρ-*cycle *if there is a variable $v \in \boldsymbol{v}$ such that $\mathcal{F}_\rho$ contains a cyclic term $f_v(\ldots f_v(\boldsymbol{t})\ldots)$. $\mathcal{R}$ is* model-faithful cyclic *(MFC) if it has a ρ-cycle for a generating rule $\rho \in \mathcal{R}$.*

Excluding facts with cyclic terms in the computation of $\mathcal{F}_\rho$ is necessary to ensure termination, since otherwise that chase might be infinite although none of the cyclic terms have the form $f_v(\ldots f_v(\boldsymbol{t})\ldots)$ with $v \in \boldsymbol{v}$. With this restriction, however, MFC can be checked in 2ExpTime.

Example 12. *The rules in Example 1 are MFC if we replace $\vee$ by $\wedge$ to make them deterministic. Let ρ be rule* (1). *Then $\mathcal{I}_\rho = \{B(c), \mathit{hP}(c, f_v(c)), W(f_v(c))\}$ and $\mathcal{F}_\rho$ contains the facts $\mathit{SW}(f_v(c))$, $\mathit{pO}(f_v(c), f_w(f_v(c)))$, $B(f_w(f_v(c)))$, and $\mathit{hP}(f_w(f_v(c)), f_v(f_w(f_v(c))))$, which has a cyclic term.*

Using the argumentation given before Definition 9, we can establish the following result.

Theorem 9. *If $\mathcal{R}$ is MFC, then there is an instance $\mathcal{I}$ for which the chase of $\langle sk(\mathcal{R}), \mathcal{I}\rangle$ is infinite.*

Unfortunately, the approach of performing a chase on minimal sets of facts fails when the application of rules can be prevented by the presence of additional facts. We therefore perform an additional applicability check for the restricted chase. Recall that $\mathcal{I}^\star_\mathcal{R}$ denotes the critical instance for $\mathcal{R}$.

Definition 10. *Consider a set $\mathcal{R}$ of deterministic rules, a rule $\rho : B[\boldsymbol{x}, \boldsymbol{y}] \to \exists \boldsymbol{v}.H[\boldsymbol{x}, \boldsymbol{v}] \in \mathcal{R}$, and a ground substitution σ on $\boldsymbol{x} \cup \boldsymbol{y}$. Let $\sigma^\star$ be such that $\sigma^\star(z)$ is $\sigma(z)$ with all constants replaced by $\star$ for all $z \in \mathbf{V}$. Let $\mathcal{R}^\star$ be $\mathcal{R}$ with existential quantifiers omitted and existential variables replaced by $\star$, and let $\rho^\star \in \mathcal{R}^\star$ be the rule obtained from ρ.*

We define $\mathcal{I}_{\rho,\sigma,\mathcal{R}} = \mathcal{I}^\star_\mathcal{R} \cup B\sigma^\star \cup \bigcup_{t \in (\boldsymbol{x}\cup\boldsymbol{y})\sigma^\star} \mathcal{F}_t$. The set $\mathcal{U}_{\rho,\sigma,\mathcal{R}}$ is obtained by exhaustive application of rules from $\mathcal{R}^\star$ to $\mathcal{I}_{\rho,\sigma,\mathcal{R}}$, with the exception of the rule $\rho^\star$ under substitution $\sigma^\star$. Then $\langle \rho, \sigma\rangle$ is unblockable *for $\mathcal{R}$ if $\mathcal{U}_{\rho,\sigma,\mathcal{R}} \not\models \exists \boldsymbol{v}.H\sigma^\star$.*

Intuitively speaking, $\mathcal{U}_{\rho,\sigma,\mathcal{R}}$ represents a gross overestimation of what might be derivable in a situation where we would like to apply a rule ρ. Note that, for $\langle \rho, \sigma\rangle$ to be unblockable, the body $B\sigma$ must contain a functional term.

Lemma 10. *Consider the chase tree for a program $\langle \mathcal{R}, \mathcal{I}\rangle$, let $\mathcal{F}$ be the label of some node n in this tree, and let $\mathcal{R}_\wedge$ be $\mathcal{R}$ with $\vee$ replaced by $\wedge$. If $\rho : B[\boldsymbol{x}, \boldsymbol{z}] \to \exists \boldsymbol{v}.H[\boldsymbol{x}, \boldsymbol{v}]$ is a deterministic rule in $\mathcal{R}$, $\mathcal{F} \models B\sigma$, and $\langle \rho, \sigma\rangle$ is unblockable for $\mathcal{R}_\wedge$, then $\langle \rho, \sigma\rangle$ is applied in a node above or below n.*

Proof sketch. For a contradiction, suppose that the preconditions hold but $\langle \rho, \sigma\rangle$ is not applied. Since applications are fair, Definition 1 (b) is violated in all leaf nodes below n. For any such leaf n' labelled $\mathcal{F}'$, we map $\mathcal{F}'$ to $\mathcal{U}_{\rho,\sigma,\mathcal{R}}$ as follows: every term t in $B\sigma^\star$ (including subterms) is mapped to itself; all other terms are mapped to $\star$. One can show by induction over the chase of $\mathcal{F}'$ that this is a homomorphism. Since $\langle \rho, \sigma\rangle$ is not applicable to $\mathcal{F}'$, we find $\mathcal{U}_{\rho,\sigma,\mathcal{R}} \models \exists \boldsymbol{v}.H\sigma^\star$. □

Definition 11. *For a set $\mathcal{R}$ of rules and a rule $\rho \in \mathcal{R}$, we define $\mathcal{I}_\rho$ and $\mathcal{F}_\rho$ as in Definition 9, but with the additional restriction that the computation of $\mathcal{F}_\rho$ uses only applications of deterministic rules that are unblockable for $\mathcal{R}_\wedge$, obtained from $\mathcal{R}$ by replacing $\vee$ with $\wedge$.*

As before $\mathcal{R}$ has a restricted ρ-cycle *if there is a variable $v \in \boldsymbol{v}$ such that $\mathcal{F}_\rho$ contains a cyclic term $f_v(\ldots f_v(\boldsymbol{t}) \ldots)$. $\mathcal{R}$ is* restricted model-faithful cyclic *(RMFC) if it has a ρ-cycle for some generating rule $\rho \in \mathcal{R}$.*

The following result is obtained by combining the arguments for Theorem 9 with the insights from Lemma 10.

Theorem 11. *If $\mathcal{R}$ is RMFC, then there is an instance $\mathcal{I}$ for which the restricted chase of $\langle \mathcal{R}, \mathcal{I}\rangle$ is infinite.*

6 Evaluation

To evaluate the effectiveness of our criteria, we have used MOWLCorp, a large corpus of real-world OWL ontologies [Matentzoglu and Parsia, 2014; Matentzoglu *et al.*, 2013], which we transformed into rules. To this end, we first normalised ontologies by structural decomposition of complex axioms, and then rewrote axioms into first-order logic to obtain rules. We refer to Cuenca Grau *et al.* [2013] for details on this standard process; our normal forms are as in their Table 1 but with an added form $A_1 \sqcap \ldots \sqcap A_n \sqsubseteq B_1 \sqcup \ldots \sqcup B_m$.

We excluded ontologies with nominals (oneOf) and *at-most*-restrictions (maxCardinality) since they require equality reasoning. There are well-known techniques for this [Cuenca Grau *et al.*, 2013], but they are not our focus. We then considered all ontologies with up to 1,000 existential quantifiers after normalisation, leading to a set of 1,576 ontologies.

We have implemented tests for RMSA, RMFA, RMFC, MSA, and MFA using RDFox [Motik *et al.*, 2014] as a rule engine. The creation of new terms during the chase and our blocking conditions are implemented on top of RDFox. For MSA and MFA, we replaced $\vee$ by $\wedge$ in all ontologies. We treat $\top$ (universal class) and $\bot$ (empty class) as regular unary predicates, and we modify our tests to ensure that all elements are always in $\top$; likewise for the universal and the empty property. Special treatment of $\bot$ is not needed, as inconsistencies are expected and should be ignored when chasing from the critical instance. We have not implemented RJA since we found the more general RMSA to perform well.

	#∃	#	MSA	MFA	RMSA	RMFA	RMFC	open
determin.	1–4	443	293	293	314	314	127	2
	5–69	368	243	243	272	272	72	24
	70–1K	409	348	348	350	350	40	19
	1–1K	**1220**	**884**	**884**	**936**	**936**	**239**	**45**
non-det.	1–9	128	48	48	53	53	3	72
	10–59	110	19	19	39	40	5	65
	60–1K	118	23	23	30	30	20	68
	1–1K	**356**	**90**	**90**	**122**	**123**	**28**	**205**

Table 1: Experimental results

Table 1 shows our results for ontologies without (top) and with (bottom) disjunctions, grouped by their number of existential axioms (#∃). The column "#" gives the number of ontologies per group; "open" counts ontologies that are neither RMFA nor RMFC. We can see that RMSA performs better than MFA, while (R)MFA hardly improves over (R)MSA. Using MFA, chase termination remains open for 602 ontologies overall (38.2%). The combination of RMFA and RMFC reduces this number to 250 (15.8%). As expected, many ontologies are indeed cyclic, but there are also an additional 85 that are acyclic (14.1% of the formerly open ones). In the deterministic case, our notions perform rather well and allow us to characterize 96.3% ontologies as acyclic or cyclic.

7 Conclusion

To the best of our knowledge, this is the first systematic study of termination of the *restricted* chase on existential rules (with disjunctions) and the first ever approach to restricted chase *non*termination. We have shown our criteria to be theoretically and empirically more general than previous notions, deciding termination for 84.2% of the tested ontologies.

Our work motivates and enables further research on chase-based reasoning procedures for ontologies. Many tableau-based OWL reasoners already implement chase-like algorithms that could be a starting point. In an early test with a modified version of the reasoner HermiT [Motik *et al.*, 2009], we have already answered conjunctive queries over an acyclic ontology with tens of thousands of facts. We believe this is a highly promising direction in description logics ontologies and existential rules alike.

References

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

[Beeri and Vardi, 1981] Catriel Beeri and Moshe Y. Vardi. The implication problem for data dependencies. In Shimon Even and Oded Kariv, editors, *Proc. 8th Colloquium on Automata, Languages and Programming (ICALP'81)*, volume 115 of *LNCS*, pages 73–85. Springer, 1981.

[Bienvenu *et al.*, 2016] Meghyn Bienvenu, Peter Hansen, Carsten Lutz, and Frank Wolter. First order-rewritability and containment of conjunctive queries in Horn description logics. In Subbarao Kambhampati, editor, *Proc. 25th Int. Joint Conf. on Artificial Intelligence (IJCAI'16)*, pages 965–971. IJCAI/AAAI Press, 2016.

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

[Calì *et al.*, 2010] Andrea Calì, Georg Gottlob, and Andreas Pieris. Query answering under non-guarded rules in Datalog+/-. In Pascal Hitzler and Thomas Lukasiewicz, editors, *Proc. 4th Int. Conf. on Web Reasoning and Rule Systems (RR 2010)*, volume 6333 of *LNCS*, pages 1–17. Springer, 2010.

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

[Calvanese *et al.*, 2007] Diego Calvanese, Guiseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. *J. of Automated Reasoning*, 39(3):385–429, 2007.

[Calvanese *et al.*, 2014] Diego Calvanese, Thomas Eiter, and Magdalena Ortiz. Answering regular path queries in expressive description logics via alternating tree-automata. *Inf. Comput.*, 237:12–55, 2014.

[Carral *et al.*, 2016] David Carral, Cristina Feier, and Pascal Hitzler. A practical acyclicity notion for query answering over Horn-$\mathcal{SRIQ}$ ontologies. In Paul T. Groth, Elena Simperl, Alasdair J. G. Gray, Marta Sabou, Markus Krötzsch, Freddy Lécué, Fabian Flöck, and Yolanda Gil, editors, *Proc. 15th Int. Semantic Web Conf. (ISWC'16)*, volume 9981 of *LNCS*, pages 70–85. Springer, 2016.

[Cuenca Grau *et al.*, 2013] Bernardo Cuenca Grau, Ian Horrocks, Markus Krötzsch, Clemens Kupke, Despoina Magka, Boris Motik, and Zhe Wang. Acyclicity notions for existential rules and their application to query answering in ontologies. *J. of Art. Int. Research*, 47:741–808, 2013.

[Dantsin *et al.*, 2001] Evgeny Dantsin, Thomas Eiter, Georg Gottlob, and Andrei Voronkov. Complexity and expressive power of logic programming. *ACM Computing Surveys*, 33(3):374–425, 2001.

[Deutsch and Tannen, 2002] Alin Deutsch and Val Tannen. Optimization properties for classes of conjunctive regular path queries. In Giorgio Ghelli and Gösta Grahne, editors, *Proc. 21st Int. Workshop on Database Programming Languages (DBPL 2001)*, volume 2397 of *LNCS*, pages 21–39. Springer, 2002.

[Krötzsch and Rudolph, 2011] Markus Krötzsch and Sebastian Rudolph. Extending decidable existential rules by joining acyclicity and guardedness. In Toby Walsh, editor, *Proc. 22nd Int. Joint Conf. on Artificial Intelligence (IJCAI'11)*, pages 963–968. AAAI Press/IJCAI, 2011.

[Marnette, 2009] Bruno Marnette. Generalized schema-mappings: from termination to tractability. In J. Paredaens and J. Su, editors, *Proc. 28th Symp. on Principles of Database Systems (PODS'09)*, pages 13–22. ACM, 2009.

[Matentzoglu and Parsia, 2014] Nicolas Matentzoglu and Bijan Parsia. The Manchester OWL Corpus (MOWLCorp), original serialisation, July 2014. `https://doi.org/10.5281/zenodo.10851`.

[Matentzoglu *et al.*, 2013] Nicolas Matentzoglu, Samantha Bail, and Bijan Parsia. A snapshot of the OWL Web. In Harith Alani, Lalana Kagal, Achille Fokoue, Paul T. Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha F. Noy, Chris Welty, and Krzysztof Janowicz, editors, *Proc. 12th Int. Semantic Web Conf. (ISWC'13)*, volume 8218 of *LNCS*, pages 331–346. Springer, 2013.

[Motik *et al.*, 2009] Boris Motik, Rob Shearer, and Ian Horrocks. Hypertableau reasoning for description logics. *J. of Artificial Intelligence Research*, 36:165–228, 2009.

[Motik *et al.*, 2014] Boris Motik, Yavor Nenov, Robert Piro, Ian Horrocks, and Dan Olteanu. Parallel materialisation of Datalog programs in centralised, main-memory RDF systems. In Carla E. Brodley and Peter Stone, editors, *Proc. 28th AAAI Conf. on Artificial Intelligence (AAAI'14)*, pages 129–137. AAAI Press, 2014.

[Rudolph and Glimm, 2010] Sebastian Rudolph and Birte Glimm. Nominals, inverses, counting, and conjunctive queries or: Why infinity is your friend! *J. of Artificial Intelligence Research*, 39:429–481, 2010.

[Stefanoni *et al.*, 2014] Giorgio Stefanoni, Boris Motik, Markus Krötzsch, and Sebastian Rudolph. The complexity of answering conjunctive and navigational queries over OWL 2 EL knowledge bases. *J. of Art. Int. Research*, 51:645–705, 2014.

A Proofs for Section 3

Theorem 3. *Deciding BCQ entailment for programs $\langle\mathcal{R},\mathcal{I}\rangle$ where $\mathcal{R}$ is RJA is co*N2EXPTIME*-complete, even if the arity of predicates is bounded.*

Proof. Towards membership, we determine the maximal number of ground (skolem) terms and corresponding facts that may occur in the chase. Let n be the number of skolem functions in $sk(\mathcal{R})$, and let m be the maximal arity of such functions. The maximal nesting depth of ground terms in the chase is n, since a greater depth can only be achieved by repeating a function symbol, which would make the restricted dependency graph cyclic by Lemma 2. Ground skolem terms then correspond to trees of depth at most n, fan-out at most m, and with leafs from the set $\mathbf{C}_{\mathcal{I}}$ of constants in $\mathcal{I}$. Such trees have at most m^n leafs, and at most $n \cdot m^n$ nodes in total. As each node is assigned a constant or function symbol, there are at most $T = (|\mathcal{I}_{\mathbf{C}}|+n)^{n\cdot m^n}$ trees, and hence ground skolem terms, overall.

Now if $\langle\mathcal{R},\mathcal{I}\rangle$ contains k different predicate symbols of arity at most ℓ, then the maximal number of ground facts based on T terms is $A = kT^\ell = k(|\mathcal{I}_{\mathbf{C}}| + n)^{\ell\cdot n\cdot m^n}$. A is therefore double exponential in (measures that depend linearly on) the size of $\langle\mathcal{R},\mathcal{I}\rangle$.

Membership then follows since BCQ non-entailment can be shown by finding a model of $\langle\mathcal{R},\mathcal{I}\rangle$ that does not satisfy the query. For this we may non-deterministically guess a branch of a chase tree which, by Definition 2, may at most have depth A.

For hardness, we reduce the word problem of double-exponentially time-bounded non-deterministic Turing machines (TMs) to BCQ non-entailment. Consider a N2EXPTIME Turing Machine (TM) M. We simulate the computation of M on an input string I by constructing a program $\langle\mathcal{R},\mathcal{I}\rangle$ such that $\langle\mathcal{R},\mathcal{I}\rangle$ does not entail some nullary predicate *Reject* if and only if M accepts I.

To address computation steps and tape cells, we recall a construction by Calì *et al.* to (deterministically) construct a chain of double exponentially many elements. Let $\mathcal{I} = \{r_0(0), r_0(1), \mathit{succ}_0(0,1), \mathit{min}_0(0), \mathit{max}_0(1)\}$. For each $i \in \{0,\ldots,n-1\}$, with n the length of the input I, we add the following rules:

$$
\begin{aligned}
r_i(x) \wedge r_i(y) &\to \exists z s_i(x,y,z)\\
s_i(x,y,z) &\to r_{i+1}(z)\\
s_i(x,y,z) \wedge s_i(x,y',z') \wedge \mathit{succ}_i(y,y') &\to \mathit{succ}_{i+1}(z,z')\\
s_i(x,y,z) \wedge s_i(x',y',z') \wedge \mathit{max}_i(y) \wedge &\\
\mathit{min}_i(y') \wedge \mathit{succ}_i(x,x') &\to \mathit{succ}_{i+1}(z,z')\\
\mathit{min}_i(x) \wedge s_i(x,x,y) &\to \mathit{min}_{i+1}(y)\\
\mathit{max}_i(x) \wedge s_i(x,x,y) &\to \mathit{max}_{i+1}(y)
\end{aligned}
$$

It can be shown, by induction on i, that in any path of any chase tree of $\langle\mathcal{R},\mathcal{I}\rangle$, the relation r_n contains 2^{2^n} elements, which are linearly ordered by succ_n.

The remaining TM simulation follows standard constructions (cf. [Dantsin *et al.*, 2001]), using elements of the r_n chain to refer to specific time points and tape cells when encoding a run of the TM. Non-deterministic transitions are captured using rules with disjunction. Assuming that the state of M at step s is captured with facts $\mathit{State}_q(s)$ for all states Q, we can complete the simulation by adding rules

$$\mathit{State}_q(x) \wedge \mathit{max}_n(x) \to \mathit{Reject} \tag{11}$$

for all non-accepting states q of M. We can assume without loss of generality that M runs for the maximum double-exponential number of steps on all rejecting runs, so that the query *Reject* is entailed if and only if there are no accepting runs.

As observed by Calì *et al.* [2010], the resulting rules are weakly acyclic (WA), and in particular this holds for the rules used to construct r_n. In consequence, the rules are also RJA, since RJA generalises joint acyclicity which in turn generalises WA. □

B Proofs for Section 4

Theorem 5. *Deciding if $\mathcal{R}$ is RMFA is* 2EXPTIME*-complete even if the arity of predicates or the number of variables per rule is bounded. It is* EXPTIME*-complete if each rule contains at most one frontier variable.*

Proof. Membership is shown as for Theorem 3, since we obtain the same upper bounds for the number of ground skolem terms that are not cyclic. The chase is deterministic in this case, leading to the claimed complexity. Note that the problem of deciding whether a rule and a substitution are blocked is in EXPTIME, since it only requires only reasoning with deterministic non-generating rules (corresponding to Datalog [Dantsin *et al.*, 2001]). Performing blocking checks therefore does not worsen the overall complexity of the RMFA check.

Rules with at most one frontier variable lead to unary skolem functions. In consequence, only exponentially many terms and atoms may occur in the chase of programs with rule sets satisfying such restriction. Indeed, the size of the frontier limits the arity of skolem functions, such that ground skolem terms are now trees of fan-out 1, i.e., words, with at most n function symbols or constants at each node. Therefore, deciding RMFA membership of a one frontier variable rule sets is in EXPTIME. As before, blocking checks do not worsen this complexity.

2EXPTIME-Hardness can be shown as for the case of MFA, where it was done by reduction from the 2EXPTIME-hard problem of BCQ entailment checking for weakly acyclic (WA) rules [Cuenca Grau *et al.*, 2013]. The hardness proof for WA in turn is based on a direct Turing machine construction using predicates of bounded arity and rules with a bounded number of variables [Calì *et al.*, 2010]. One may verify that no rule application is ever blocked in this particular construction and thus, we can use the same argument for our purposes. EXPTIME-hardness can be obtained using a known proof for the case of MFA, which sports the same complexity for small frontiers [Cuenca Grau *et al.*, 2013]. In particular, the construction in their Lemma 59 works even when using blocking because, as in the previous case, no rule application is ever blocked. □

Lemma 6. *Let $\pi^\star$ be the function that maps a term t to the term obtained from t by replacing all constants with $\star$. Then, for every term t that occurs in any restricted chase of $\langle \mathcal{R}, \mathcal{I}\rangle$ for any instance $\mathcal{I}$, the term $\pi^\star(t)$ occurs in* $\mathsf{RMFA}(\mathcal{R})$.

Proof. Consider any set of facts $\mathcal{F}$ that is the label of some node in the restricted chase, and which contains a term t. We show that $\pi^\star$ is a homomorphism from $\mathcal{F}$ to $\mathsf{RMFA}(\mathcal{R})$, which implies that $\pi^\star(t)$ occurs in $\mathsf{RMFA}(\mathcal{R})$.

We proceed by induction along the rule applications that have produced $\mathcal{F}$. The base case is clear since $\pi^\star(t)$ is a homomorphism from $\mathcal{I}$ to $\mathcal{I}^\star_{\mathcal{R}} \subseteq \mathsf{RMFA}(\mathcal{R})$. For the induction step, assume that $\mathcal{F}$ is the result of applying rule $\rho : B \to \bigvee_i \exists \boldsymbol{v}_i.H_i$ under substitution σ to a previous set of facts $\mathcal{F}'$, and in particular that $\mathcal{F} = \mathcal{F}' \cup \mathit{sk}(H_k)\sigma'$ for some index k. By the induction hypothesis, $\pi^\star$ is a homomorphism from $\mathcal{F}'$ to $\mathsf{RMFA}(\mathcal{R})$. Therefore $\pi^\star(B\sigma) \subseteq \mathsf{RMFA}(\mathcal{R})$.

Let $\sigma^\star$ be such that $\sigma^\star(x) = \pi^\star(\sigma(x))$ for all $x \in \mathbf{V}$. We show that $\langle \rho, \sigma^\star\rangle$ is not blocked. Indeed, consider the set $\mathcal{B}_{\rho,\sigma^\star}$ from Definition 6, and let σ' be the substitution from this definition, i.e., $\sigma'(x)$ is $\sigma^\star(x)$ with each occurrence of a constant renamed. Then $B\sigma$ can be obtained from $B\sigma'$ by replacing each of the renamed constants c that occurs at a certain position in $B\sigma'$ by the constant $\eta(c)$ that occurs at the same position in $B\sigma$. We can extend this mapping η to terms and atoms that contain constants from $B\sigma'$. We claim that η can be further extended to a homomorphism from $\mathcal{B}_{\rho,\sigma^\star}$ to $\mathcal{F}'$. Indeed, $\eta(B\sigma') = B\sigma \subseteq \mathcal{F}'$ by definition. This also implies that each skolem (sub)term $f_v(\boldsymbol{s})$ in $B\sigma'$ has a corresponding term $\eta(f_v(\boldsymbol{s})) = f_v(\eta(\boldsymbol{s}))$ that occurs in $\mathcal{F}'$. This can only be if the rule $\rho' : B' \to \bigvee_i \exists \boldsymbol{v}'_i.H'_i$ that introduces $f_v(\eta(\boldsymbol{s}))$ has been applied in $\mathcal{F}'$, which requires that $(B' \cup \mathit{sk}(H'_\ell))\theta \subseteq \mathcal{F}'$ for the head H'_ℓ that contains v and some substitution θ that maps the frontier $\boldsymbol{x}'$ of ρ' to $\eta(\boldsymbol{s})$. The set $\mathcal{F}_{f_v(\boldsymbol{s})}$ considered in Definition 6 and defined in Definition 5 may contain additional constants c_y introduced for non-frontier variables y; we extend η to map each such constant c_y to $y\theta$. This yields $\eta(\mathcal{F}_{f_v(\boldsymbol{s})}) = (B' \cup \mathit{sk}(H'_\ell))\theta \subseteq \mathcal{F}'$, and finishes our construction of the homomorphism $\eta : \mathcal{B}_{\rho,\sigma^\star} \to \mathcal{F}'$.

Now suppose for a contradiction that $\langle \rho, \sigma^\star\rangle$ is blocked, i.e., $\mathcal{R}_{\mathsf{dng}}(\mathcal{B}_{\rho,\sigma}) \models \bigvee_i \exists \boldsymbol{v}_i.H_i\sigma'$. Clearly, $\eta(\mathcal{B}_{\rho,\sigma^\star}) \subseteq \mathcal{F}'$ implies $\mathcal{R}_{\mathsf{dng}}(\eta(\mathcal{B}_{\rho,\sigma^\star})) \subseteq \mathcal{F}'$, so we obtain $\mathcal{F}' \models \eta(\bigvee_i \exists \boldsymbol{v}_i.H_i\sigma') = \bigvee_i \exists \boldsymbol{v}_i.H_i\sigma$. Hence, $\langle \rho, \sigma\rangle$ does not satisfy Definition 1 (b) for $\mathcal{F}'$, contradicting our assumptions.

We therefore find that $\langle \rho, \sigma^\star\rangle$ is not blocked. Combining this with our earlier observations that $\pi^\star(B\sigma) = B\sigma^\star \subseteq \mathsf{RMFA}(\mathcal{R})$, we can apply Definition 7 to obtain $\mathit{sk}(H_k)\sigma^\star = \pi^\star(\mathit{sk}(H_k)\sigma) \subseteq \mathsf{RMFA}(\mathcal{R})$. This completes the proof. □

Theorem 7. *Deciding BCQ entailment for programs $\langle \mathcal{R}, \mathcal{I}\rangle$ where $\mathcal{R}$ is RMFA is co*N2EXPTIME*-complete, even if the arity of predicates is bounded.*

Proof. Membership is shown as for Theorem 3. Indeed, by Lemma 6, cyclic terms are not derived in the chase, since $\pi^\star$ maps non-cyclic terms to non-cyclic terms. We can apply the same counting argument as for Theorem 3 to establish a doubly exponential upper bound for the number of facts that can be derived in any branch of the restricted chase.

Hardness follows by Theorem 3, as any RJA rule set can be shown to be RMFA. This proof is analogous to the proof for the relation of JA and MFA [Cuenca Grau *et al.*, 2013], so we do not repeat it here. □

Theorem 8. *Deciding if $\mathcal{R}$ is RMSA is* EXPTIME*-complete, and* P*-complete if the number of variables per rule is bounded.*

Proof. Membership follows since the number of derivable ground facts in *RMSA*$(\mathcal{R})$ is exponential (in the arity of predicate symbols). The number of constants used in computing *RMSA*$(\mathcal{R})$ is the number of existentially quantified variables in $\mathcal{R}$ plus one (for the constant $\star$), and therefore linear in the size of $\mathcal{R}$. Checking whether a rule is blocked likewise corresponds to a Datalog-like computation, leading to an overall EXPTIME upper bound.

When the number of variables per rule is bounded by a constant, then the grounding of the rules (obtained by uniformly replacing each variable by some constant) yields a polynomially large propositional Horn theory, for which reasoning is in P [Dantsin *et al.*, 2001]. The same applies to the blocking check.

For hardness, we reduce the EXPTIME-hard fact entailment problem of Datalog [Dantsin *et al.*, 2001] to RMSA checking. Let $\mathcal{R}$ be a set of Datalog rules (i.e., non-generating, deterministic rules), let $\mathcal{I}$ be an instance, and let $A(c)$ be a ground fact for which we want to check if $\mathcal{R} \cup \mathcal{I} \models A(c)$. Without loss of generality, we can assume that each rule body in $\mathcal{R}$ forms a connected structure. This can be achieved by introducing a fresh binary prdicate U, adding a fact $U(a, b)$ for all constants a and b in $\mathcal{I}$, and extending each rule body with an atom $U(x, y)$ for all pairs of variables x and y in this body. Clearly, this is a polynomial reduction.

We define a set $\mathcal{R}'$ of existential rules using a fresh binary predicate R and a list of variables $\boldsymbol{w}$ consisting of one variable w_a for every constant a in $\mathcal{I}$. $\mathcal{R}'$ then contains all rules of $\mathcal{R}$ and one additional rule

$$R(x, y) \wedge A(y) \to \exists \boldsymbol{w}.R(y, w_c) \wedge \bigwedge_{p(c_1,\ldots,c_n)\in\mathcal{I}} p(w_{c_1}, \ldots, w_{c_n}) \tag{12}$$

Hardness of RMSA checking follows from the fact that $\mathcal{R} \cup \mathcal{I} \models A(c)$ if and only if the set $\mathcal{R}'$ is not RMSA. Indeed, (12) applies to the critical instance once. This application creates a copy of $\mathcal{I}$ with new constants. If $A(c)$ can be derived from $\mathcal{I}$, then a similar derivation is possible from this copy of $\mathcal{I}$, leading to another application of rule (12). This process can be repeated so that the chase does indeed not terminate; in particular $\mathcal{R}'$ is not RMSA in this case. The assumption that rule bodies are connected is important to ensure that each copy of $\mathcal{I}$ is used in isolation when applying rules.

We can make an analogous argument to show that the RMSA check is P-hard when the number of variables per rule is bounded. However, (12) does not have a bounded number of variables (unless we also impose a bound on the number of constants). Therefore, we now use the fact that entailment in propositional Horn logic is already P-hard. Let $\mathcal{H}$ be a propositional Horn theory consisting (without loss of generality) of facts p and binary rules $p \wedge q \to r$. For each propositional letter p, we introduce a unary predicate T_p, and each Horn rule $p \wedge q \to r$ is rewritten to $T_p(x) \wedge T_q(x) \to T_r(x)$. Rule (12) is then replaced by a rule

$$R(x, y) \wedge T_q(y) \to \exists w.R(y, w) \wedge \bigwedge_{p \in \mathcal{H}} T_p(w)$$

Then $\mathcal{H} \models q$ if and only if the resulting set of rules is not RMSA. □

C Proofs for Section 5

Theorem 9. *If $\mathcal{R}$ is MFC, then there is an instance $\mathcal{I}$ for which the chase of $\langle sk(\mathcal{R}), \mathcal{I}\rangle$ is infinite.*

Proof. We can show the claim using the critical instance $\mathcal{I} = \mathcal{I}^\star_\mathcal{R}$. Assume that $\mathcal{R}$ is MFC and that $\rho : B \to \exists \boldsymbol{v}.H$ is a rule in $\mathcal{R}$ that satisfies the conditions of Definition 9. We show by induction that the chase of $\langle sk(\mathcal{R}), \mathcal{I}\rangle$ contains n distinct applications of ρ, for every n, and is therefore infinite.

The base case $n = 1$ follows since all rules are applicable at least once to $\mathcal{I} = \mathcal{I}^\star_\mathcal{R}$.

For the inductive step, assume that ρ was applied n times during the chase, with the nth application using substitution σ. Consider the set $\mathcal{I}_\rho$ from Definition 9, and let π be a function that maps each constant c_z in $\mathcal{I}_\rho$ for a variable z in ρ to $z\sigma$. We extend π to skolem terms and to ground facts as usual. Then π is a homomorphism from $\mathcal{I}_\rho$ to the set $B\sigma \cup sk(H)\sigma$, which in turn is a subset of the chase of $\langle sk(\mathcal{R}), \mathcal{I}\rangle$.

Since rule derivations are preserved under homomorphisms, we find that π extends to a homomorphism from $\mathcal{F}_\rho$ to the chase of $\langle sk(\mathcal{R}), \mathcal{I}\rangle$. Therefore, since $\mathcal{F}_\rho$ contains a cyclic term $f_v(\ldots f_v(\boldsymbol{t})\ldots)$ for some variable $v \in \boldsymbol{v}$, we find that $\pi(f_v(\ldots f_v(\boldsymbol{t})\ldots)) = f_v(\ldots f_v(\pi(\boldsymbol{t}))\ldots)$ occurs in the chase. Since $\mathcal{I}_\rho$ contains no cyclic terms, the computation of $\mathcal{F}_\rho$ involved an application of rule ρ (the only rule that may introduce an f_v-term) to facts derived from $\mathcal{I}_\rho$. This additional $(n + 1)$th application of ρ must also occur in the chase for $\pi(f_v(\ldots f_v(\boldsymbol{t})\ldots))$ to occur there. □

Lemma 10. *Consider a chase tree for a program $\langle\mathcal{R}, \mathcal{I}\rangle$, let $\mathcal{F}$ be the label of some node n in this tree, and let $\mathcal{R}_\wedge$ be $\mathcal{R}$ with $\vee$ replaced by $\wedge$. If $\rho : B \to \exists \boldsymbol{v}.H$ is a deterministic rule in $\mathcal{R}$, $\mathcal{F} \models B\sigma$, and $\langle\rho, \sigma\rangle$ is unblockable for $\mathcal{R}_\wedge$, then $\langle\rho, \sigma\rangle$ is applied in a node above or below n.*

Proof. For a contradiction, suppose that the preconditions hold but $\langle\rho, \sigma\rangle$ is not applied. Since applications are fair, Definition 1 (b) is violated in all leaf nodes below n, which are saturated under the application of non-generating rules (condition (4) of Definition 2). For any such leaf n' labelled $\mathcal{F}'$, we map $\mathcal{F}'$ to $\mathcal{U}_{\rho,\sigma,\mathcal{R}}$ using a function π. For every term or subterm t that occurs in $B\sigma^\star$, we define $\pi(t) = t$; for all other terms we define $\pi(t) = \star$. We extend π to facts and sets of facts as usual.

We show that π is a homomorphism from $\mathcal{F}'$ to $\mathcal{U}_{\rho,\sigma,\mathcal{R}}$. We proceed by induction over the chase step in which each fact was derived. The base case is the starting instance $\mathcal{I}$, for which we find that $\pi(\mathcal{I}) \subseteq \mathcal{I}^\star_\mathcal{R} \subseteq \mathcal{I}_{\rho,\sigma,\mathcal{R}} \subseteq \mathcal{U}_{\rho,\sigma,\mathcal{R}}$.

For the induction step, assume that π is a homomorphism from some intermediate set of facts $\mathcal{F}_1$ to $\mathcal{U}_{\rho,\sigma,\mathcal{R}}$, and that the next set of facts on the chase path is $\mathcal{F}_2$, obtained by applying the skolemised version of rule $\rho' : B' \to \bigvee_i \exists \boldsymbol{w}_i.H'_i$ under substitution σ'. That is $\mathcal{F}_2 = \mathcal{F}_1 \cup \{sk(H'_k\sigma')\}$ for some index k. We consider two cases.

(1) If $sk(H'_k)\sigma'$ introduces a new (ground) skolem term t that occurs as a (sub)term in $B\sigma^\star$, then $\mathcal{F}_t \subseteq \mathcal{U}_{\rho,\sigma,\mathcal{R}}$. We claim that $\pi(sk(H'_k)\sigma') \subseteq \mathcal{F}_t$. Indeed, for the frontier $\boldsymbol{x}'$ of ρ', t has the form $f(\boldsymbol{x}'\sigma')$, and all (sub)terms of $\boldsymbol{x}'\sigma'$ therefore also occur in $B\sigma^\star$. We therefore obtain $\pi(\boldsymbol{x}'\sigma') = \boldsymbol{x}'\sigma'$. In other words, π is the identity on all frontier variables of ρ', and therefore $\pi(sk(H'_k)\sigma') = sk(H'_k)\sigma'$. By the definitions, we get $sk(H'_k)\sigma' \subseteq \mathcal{F}_t$, so $\pi(sk(H'_k)\sigma') \subseteq \mathcal{F}_t \subseteq \mathcal{U}_{\rho,\sigma,\mathcal{R}}$ as required.

(2) If $sk(H'_k)\sigma'$ contains no new skolem term that occurs in $B\sigma$, then $\pi(sk(H'_k)\sigma')$ is contained in the facts that are derived by applying the rule $\rho'^\star \in \mathcal{R}^\star$ (obtained from ρ by replacing $\vee$ with $\wedge$, and existential variables with $\star$) to the facts $\pi(sk(B')\sigma')$. By the induction hypothesis, the latter are contained in $\mathcal{U}_{\rho,\sigma,\mathcal{R}}$, so that $\rho'^\star$ is indeed applied. This shows the claim and completes the induction.

Since $\langle\rho, \sigma\rangle$ is not applicable to $\mathcal{F}'$, we find that $\mathcal{F}' \models \exists \boldsymbol{v}.H\sigma$. Together with $\pi(\mathcal{F}') \subseteq \mathcal{U}_{\rho,\sigma,\mathcal{R}}$ we obtain $\mathcal{U}_{\rho,\sigma,\mathcal{R}} \models \pi(\exists \boldsymbol{v}.H\sigma) = \exists \boldsymbol{v}.H\sigma^\star$. Therefore, $\langle\rho, \sigma\rangle$ is not unblockable – contradiction. □

Theorem 11. *If $\mathcal{R}$ is RMFC, then there is an instance $\mathcal{I}$ for which the restricted chase of $\langle\mathcal{R}, \mathcal{I}\rangle$ is infinite.*

Proof. We can use the same inductive argument as in the proof of Theorem 9 to show that the restricted chase over the instance $\mathcal{I} = \mathcal{I}_\rho$ as in Definition 9 contains infinitely many applications of a rule ρ. In fact, since RMFC only considers deterministic rules for detecting cycles, *all* paths of the restricted chase must be infinite in this case. The fact that $\mathcal{I}_\rho$ contains skolem terms does not concern us, since we could simply replace them with fresh constants to get a corresponding infinite chase from an instance that has no function symbols.

The argument for the base case is the same as for the induction step, since we start in both cases with a set of facts that corresponds to a situation where ρ has just been applied. We can argue that another application of ρ is possible just as we did for the proof of Theorem 9, where we additionally note that, by Lemma 10, it is certain that every unblockable (deterministic) rule will eventually be applied. This ensures that the derivation of cyclic terms as required in the induction can indeed be found in the restricted chase. □