On the Succinctness of Query Rewriting for Datalog$^{\pm}$

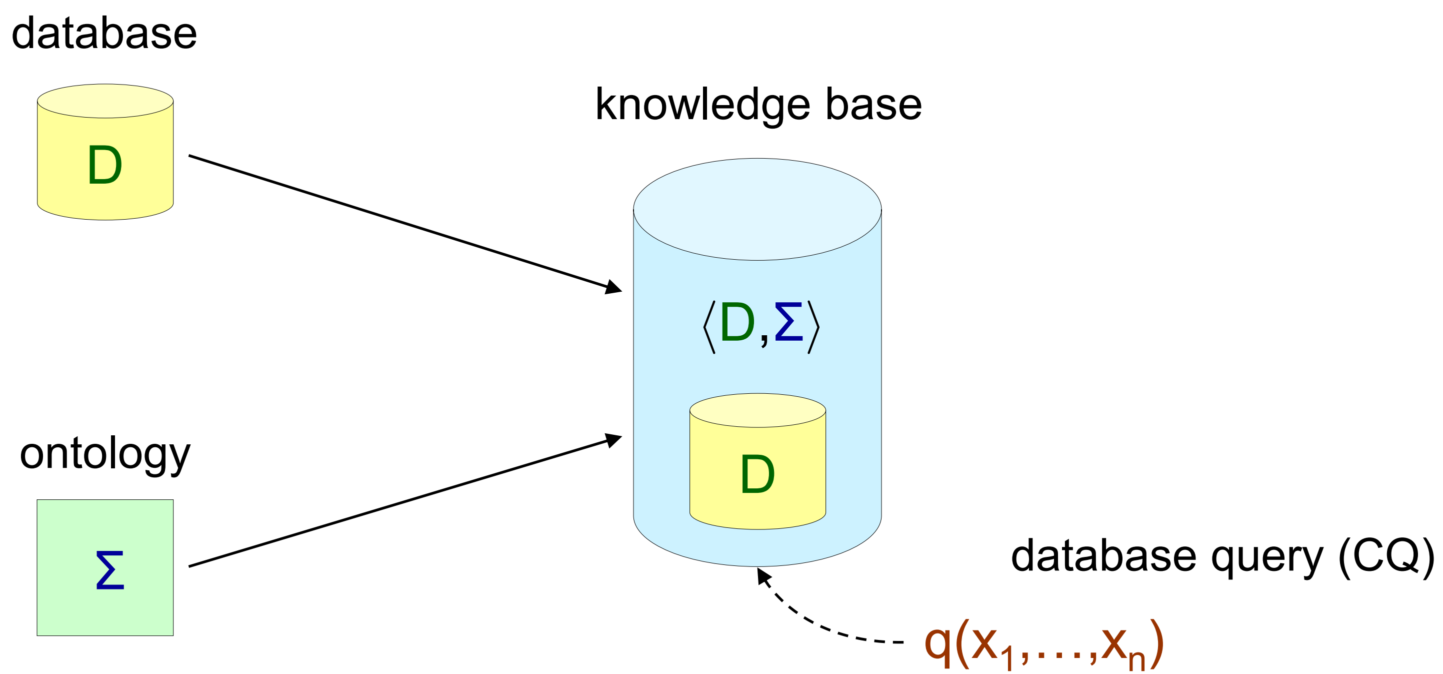
Shqiponja Ahmetaj[1] **Andreas Pieris**[2]

[1]Institute of Logic and Computation, TU Wien, Austria

[2]School of Informatics, University of Edinburgh, UK

Foundational Challenges in Data and Knowledge Management, Vienna, March 23, 2018

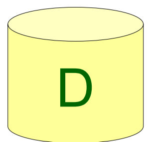
Ontology-Based Query Answering



$$\text{Certain-Answers}(q, D, \Sigma) \;=\; \{ (c_1,\ldots,c_n) \in \text{dom}(D)^n \;\mid\; D \wedge \Sigma \models q(c_1,\ldots,c_n) \}$$

Ontology-Mediated Queries

database

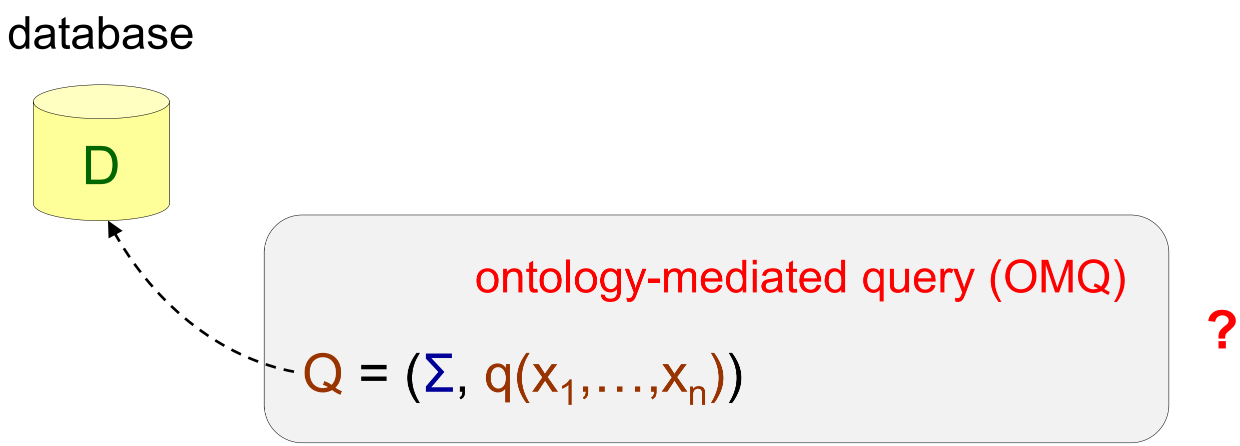


ontology-mediated query (OMQ)

$Q = (\Sigma, q(x_1,\dots,x_n))$

$Q(D) \quad = \quad \text{Certain-Answers}(q, D, \Sigma)$

Scalable OMQ Evaluation



Exploit standard RDBMSs - efficient technology for answering queries

Query Rewriting

$$Q = (\Sigma, q(x_1,\ldots,x_n))$$

$$\Big\downarrow \text{rewrite}$$

$$Q_{rew}(x_1,\ldots,x_n)$$

a query that can be executed by exploiting existing database technology

for every database D : $Q(D) = Q_{rew}(D)$

Query Rewriting: An Example

$$\{\ \forall x\ (Person(x) \rightarrow \exists y\ HasFather(x,y) \wedge Person(y))\ \}$$

$$\exists x\ (Person(x) \wedge HasFather(John,x))$$

$$Q = (\Sigma,\ q)$$

rewrite

$$Q_{rew} = \exists x\ (Person(x) \wedge HasFather(John,x)) \ \vee\ Person(John)$$

Query Rewritability

$(\mathbf{L}, \mathbf{CQ})$

an ontology language

(fragment of first-order logic)

the class of conjunctive queries

Definition: An OMQ language O is QL-Rewritable if every Q $\in$ O is QL-Rewritable

First-order (FO), $\exists$FO^{+}, Non-recursive Datalog (NDL), UCQ or Datalog

Query Rewritability: The Main Questions

1. **Can we isolate meaningful OMQ languages that are QL-Rewritable?**

2. **What is the price of QL-rewriting?**

...have been extensively studied for DL- and rule-based OMQ languages

Tuple-Generating Dependencies (TGDs)

(a.k.a. existential rules or Datalog$^{\pm}$ rules)

$$\forall \mathbf{x} \forall \mathbf{y} \, (\varphi(\mathbf{x},\mathbf{y}) \rightarrow \exists \mathbf{z} \, \psi(\mathbf{x},\mathbf{z}))$$

(**TGD**,**CQ**)

The Guarded Family

Weakly-Guarded

one body-atom contains all the harmful $\forall$-variables

$R(\mathbf{w}), \varphi(\mathbf{x},\mathbf{y}) \rightarrow \exists \mathbf{z}\, \psi(\mathbf{x},\mathbf{z})$ - $\mathbf{w} \subseteq \{\mathbf{x},\mathbf{y}\}$

[Calì, Gottlob & Kifer, KR 2008, J. Artif. Intell. Res. 2013]

↑

Guarded

one body-atom contains all the $\forall$-variables

$R(\mathbf{x},\mathbf{y}), \varphi(\mathbf{x},\mathbf{y}) \rightarrow \exists \mathbf{z}\, \psi(\mathbf{x},\mathbf{z})$

[Calì, Gottlob & Kifer, KR 2008, J. Artif. Intell. Res. 2013]

↑

Linear

one body-atom

$R(\mathbf{x},\mathbf{y}) \rightarrow \exists \mathbf{z}\, \psi(\mathbf{x},\mathbf{z})$

[Calì, Gottlob & Lukasiewicz, PODS 2009, J. Web Sem. 2012]

The Guarded Family

Theorem: It holds that

1. (**Linear**,**CQ**) is UCQ-Rewritable

2. (**Guarded**,**CQ**) is not FO-Rewritable, but is Datalog-Rewritable

3. (**Weakly-Guarded**,**CQ**) is not Datalog-Rewritable

(**Guarded**,**CQ**) is not FO-Rewritable

$Q = (\{R(x,y), P(y) \rightarrow P(x)\},\ P(c_n))$

$D \supseteq \{P(c_1)\}$, and contains no other P-atom

Q_{rew} has to check for the existence of an R-path in D of unbounded length

$$c_n \xrightarrow{R} \#_{n-1} \xrightarrow{R} \#_{n-2} \xrightarrow{R} \cdots \xrightarrow{R} \#_2 \xrightarrow{R} c_1$$

compute the transitive closure of R - not possible via a first-order query

(**Weakly-Guarded**,**CQ**) is not Datalog-Rewritable

Evaluation of (**Weakly-Guarded**,**CQ**) queries is

EXPTIME-complete in data complexity

…in fact, (**Weakly-Guarded$^{\neg\text{stratified}}$**,**CQ**) = EXPTIME, even w/o an order

[Gottlob, Rudolph & Šimkus, PODS 2014]

The Guarded Family

Theorem: It holds that

1. (**Linear**,**CQ**) is UCQ-Rewritable

2. (**Guarded**,**CQ**) is not FO-Rewritable, but is Datalog-Rewritable

3. (**Weakly-Guarded**,**CQ**) is not Datalog-Rewritable

The Guarded Family

Theorem: It holds that

1. (**Linear**,**CQ**) is UCQ-Rewritable

2. (**Guarded**,**CQ**) is not FO-Rewritable, but is Datalog-Rewritable

3. (**Weakly-Guarded**,**CQ**) is not Datalog-Rewritable

(Linear,CQ) is UCQ-Rewritable

Via a resolution-based algorithm - XRewrite

ALGORITHM 1: The algorithm XRewrite

Input: a CQ q over a schema $\mathcal{R}$ and a set Σ of TGDs over $\mathcal{R}$
Output: the perfect rewriting of q w.r.t. Σ

$i := 0$;
$Q_{\text{REW}} := \{\langle q, \mathsf{r}, \mathsf{u}\rangle\}$;
repeat
  $Q_{\text{TEMP}} := Q_{\text{REW}}$;
  foreach $\langle q, x, \mathsf{u}\rangle \in Q_{\text{TEMP}}$, where $x \in \{\mathsf{r}, \mathsf{f}\}$ **do**
    foreach $\sigma \in \Sigma$ **do**
      /* rewriting step
      foreach $S \subseteq body(q)$ such that σ is $\boxed{\text{applicable}}$ to S **do**
        $i := i + 1$;
        $q' := \gamma_{S,\sigma^i}(q[S/body(\sigma^i)])$;
        if there is no $\langle q'', \mathsf{r}, \star\rangle \in Q_{\text{REW}}$ such that $q' \simeq q''$ **then**
          $Q_{\text{REW}} := Q_{\text{REW}} \cup \{\langle q', \mathsf{r}, \mathsf{u}\rangle\}$;
        end
      end
      /* factorization step
      foreach $S \subseteq body(q)$ which is $\boxed{\text{factorizable}}$ w.r.t. σ **do**
        $q' := \gamma_S(q)$;
        if there is no $\langle q'', \star, \star\rangle \in Q_{\text{REW}}$ such that $q' \simeq q''$ **then**
          $Q_{\text{REW}} := Q_{\text{REW}} \cup \{\langle q', \mathsf{f}, \mathsf{u}\rangle\}$;
        end
      end
    end
    /* query q is now explored
    $Q_{\text{REW}} := (Q_{\text{REW}} \setminus \{\langle q, x, \mathsf{u}\rangle\}) \cup \{\langle q, x, \mathsf{e}\rangle\}$;
  end
until $Q_{\text{TEMP}} = Q_{\text{REW}}$;
$Q_{\text{FIN}} := \{q \mid \langle q, \mathsf{r}, \mathsf{e}\rangle \in Q_{\text{REW}}\}$;
return Q_{FIN}

---

applicability condition for TGDs

apply useful reduction steps, but only useful ones

[Gottlob, Orsi & P., ICDE 2011, ACM Trans. Database Syst. 2014]

(**Linear**,**CQ**) is UCQ-Rewritable

Via a resolution-based algorithm - XRewrite

Given an OMQ $Q = (\Sigma, q)$ from (**Linear**,**CQ**)

1. The height of XRewrite(Q) is at most $|q|$
2. The size of XRewrite(Q) is at most $\#\mathrm{pred}(\Sigma)^{|q|} \cdot (\mathrm{arity}(\Sigma) \cdot |q|)^{\mathrm{arity}(\Sigma) \cdot |q|}$

worst-case optimal

[Gottlob, Orsi & P., ICDE 2011, ACM Trans. Database Syst. 2014]

Lower Bound for (**Linear**,**CQ**)

$\Sigma \;=\; \{R_i(x) \rightarrow P_i(x)\}_{i \in \{1,\ldots,n\}}$ $\qquad\qquad$ $q \;=\; \exists x\,(P_1(x) \wedge \ldots \wedge P_n(x))$

$$\begin{array}{ccc} & \exists x\, P_1(x) \wedge \ldots \wedge P_n(x) & \\ \nearrow & & \nwarrow \\ P_1(X) \vee R_1(X) & & P_n(X) \vee R_n(X) \end{array}$$

$\Rightarrow$ we need to consider 2^n disjuncts

Target More Succinct Query Languages

Theorem: For (**Linear**,**CQ**) there is

- No $\exists FO^+$/NDL-rewriting of polynomial size
- No FO-rewriting of polynomial size (unless the PH collapses)

Proof: Via succinctness of monotone Boolean circuits

NOTE: The above proof exploits databases with a single domain element

[Gottlob, Kikot, Kontchakov, Podolskii, Schwentick & Zakharyaschev, Artif. Intell. 2014]

Two Domain Elements

$$Q = (\Sigma, q(x_1, \ldots, x_n))$$

$$\Big\downarrow \text{rewrite in polynomial time}$$

$$Q_{rew}(x_1, \ldots, x_n)$$

for every database D_{01} : $Q(D_{01}) = Q_{rew}(D_{01})$

$\supseteq$ { Zero(0), One(1) }

Polynomial Rewritings

… assuming two domain elements

Theorem: A (**Linear**,**CQ**) query can be rewritten in polynomial time as:

- An $\exists$FO$^+$/NDL query for bounded arity predicates
- An FO query for arbitrary signatures

Proof:

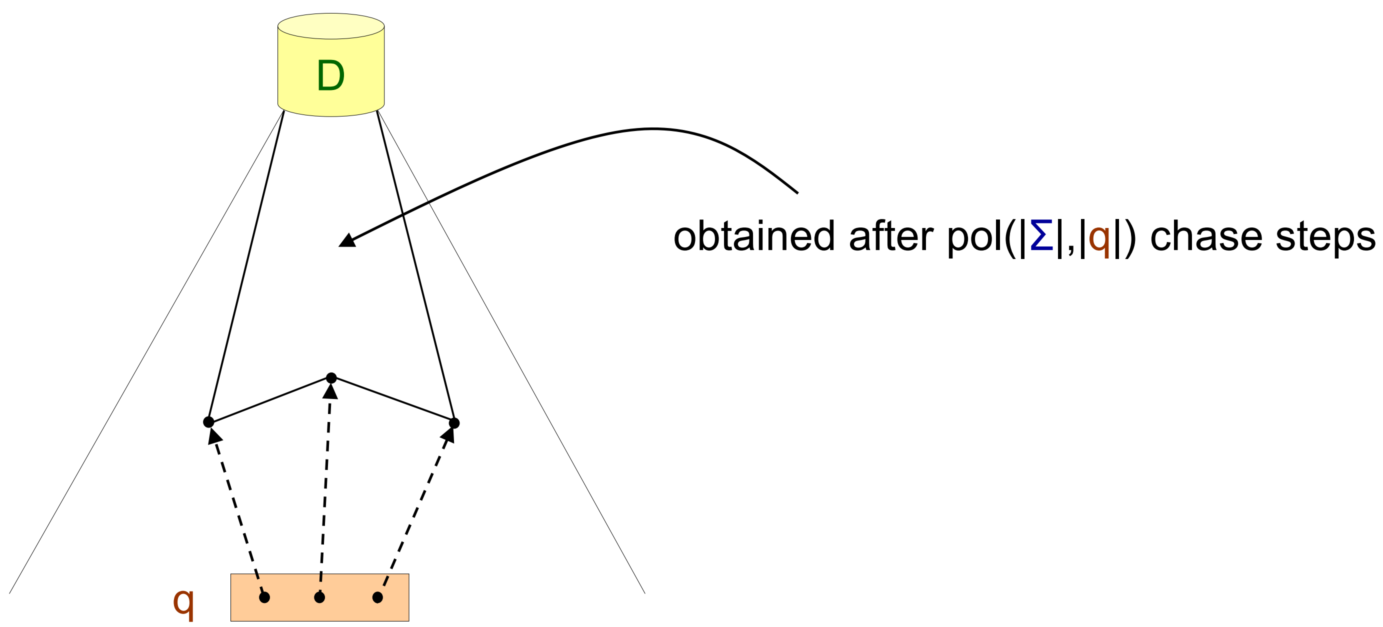
- Bounded arity signatures - via the polynomial witness property
- Arbitrary signatures - via proof generators

Polynomial Witness Property (PWP)

Definition: (**L**,**CQ**) enjoys the PWP if: there exists a polynomial pol(·) such that

for every Q = (Σ, q(**x**)) ∈ (**L**,**CQ**), database D, and **t** ∈ dom(D)$^{|\mathbf{x}|}$

t ∈ Q(D) ⇒ q(**t**) can be entailed after pol(|Σ|,|q|) chase steps

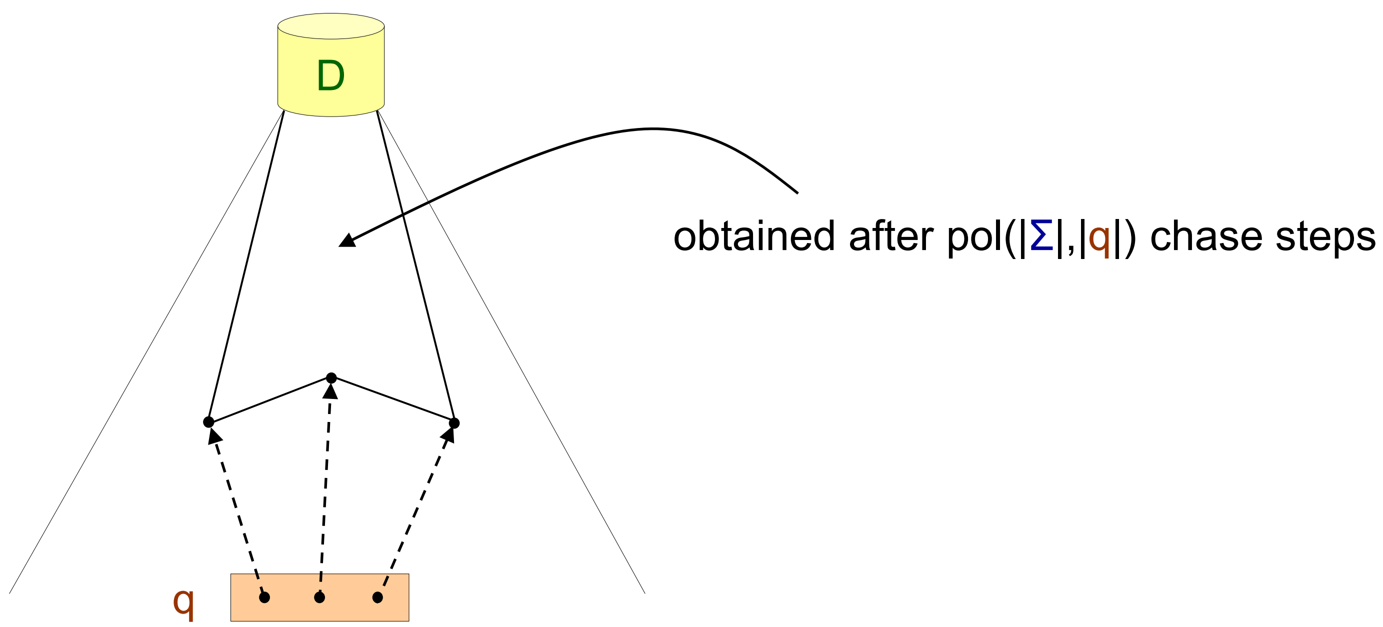


[Gottlob & Schwentick, KR 2012] + [Gottlob, Kikot, Kontchakov, Podolskii, Schwentick & Zakharyaschev, Artif. Intell. 2014]

Polynomial Witness Property (PWP)

Theorem: PWP $\Rightarrow$ $\exists$FO$^+$/NDL-rewritings constructible in polynomial time,

focusing on databases with at least two constants



[Gottlob & Schwentick, KR 2012] + [Gottlob, Kikot, Kontchakov, Podolskii, Schwentick & Zakharyaschev, Artif. Intell. 2014]

Witnesses and Linearity

$O = \{\ \}$ $\qquad$ $q = \exists z \exists o\ \mathrm{Number}(o,\ldots,o,z,o)$

$$\{\mathrm{Number}(x_1,\ldots,x_{n-i},z,\underbrace{o,\ldots,o}_{i-1},z,o) \rightarrow \mathrm{Number}(x_1,\ldots,x_{n-i},o,\underbrace{z,\ldots,z}_{i-1},z,o)\}_{i \in \{1,\ldots,n\}}$$

$0 \quad \mathrm{Number}(\mathbf{0,\ldots,0,0},0,1)$

$\downarrow$

$1 \quad \mathrm{Number}(\mathbf{0,\ldots,0,1},0,1)$

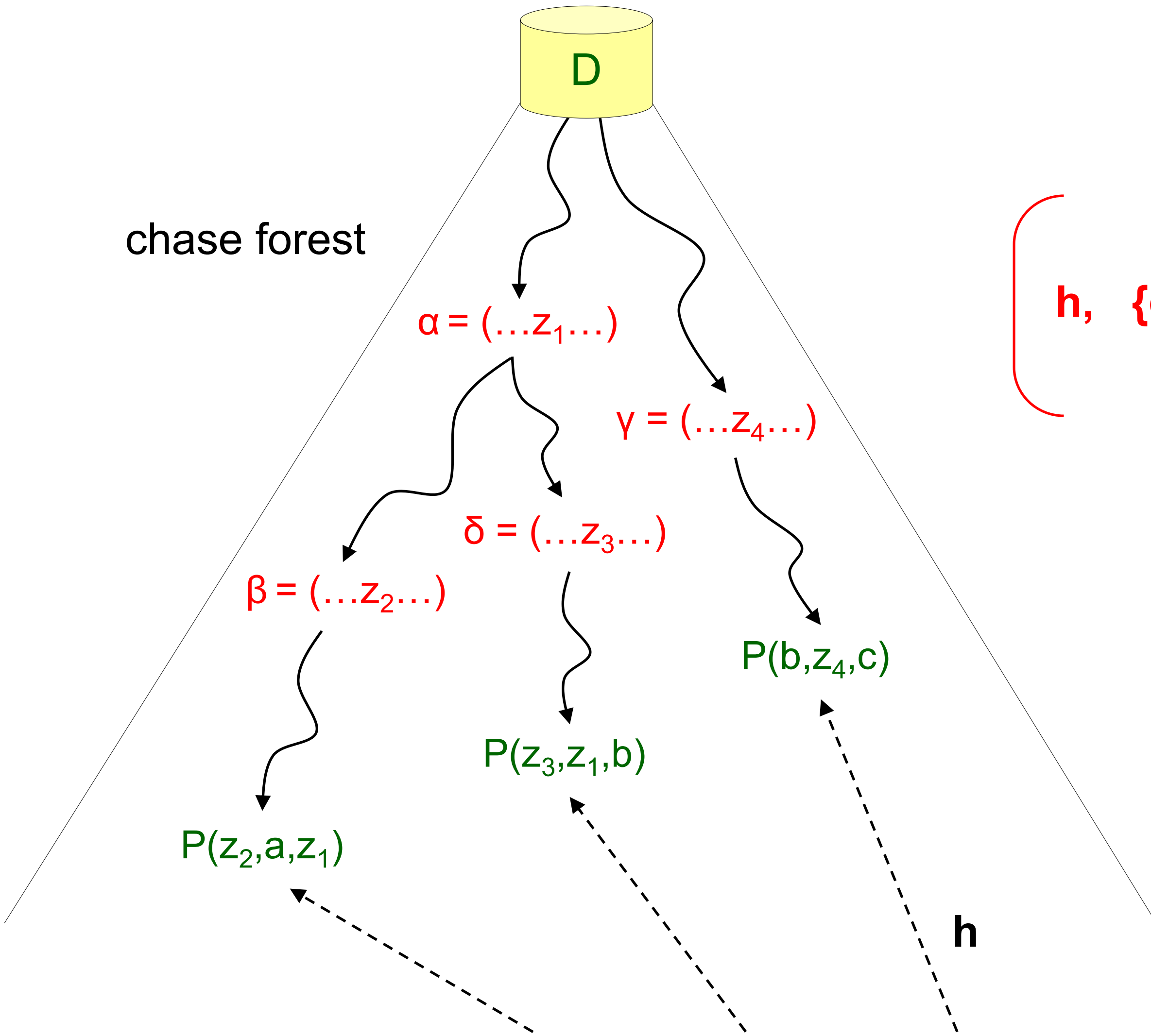
$\downarrow$

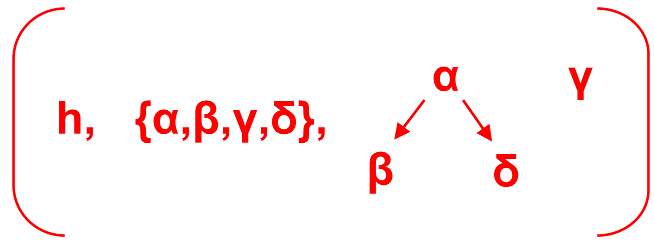
$2 \quad \mathrm{Number}(\mathbf{0,\ldots,1,0},0,1)$

$\vdots$

$2^n \quad \mathrm{Number}(\mathbf{1,\ldots,1,1},0,1)$

Proof Generator

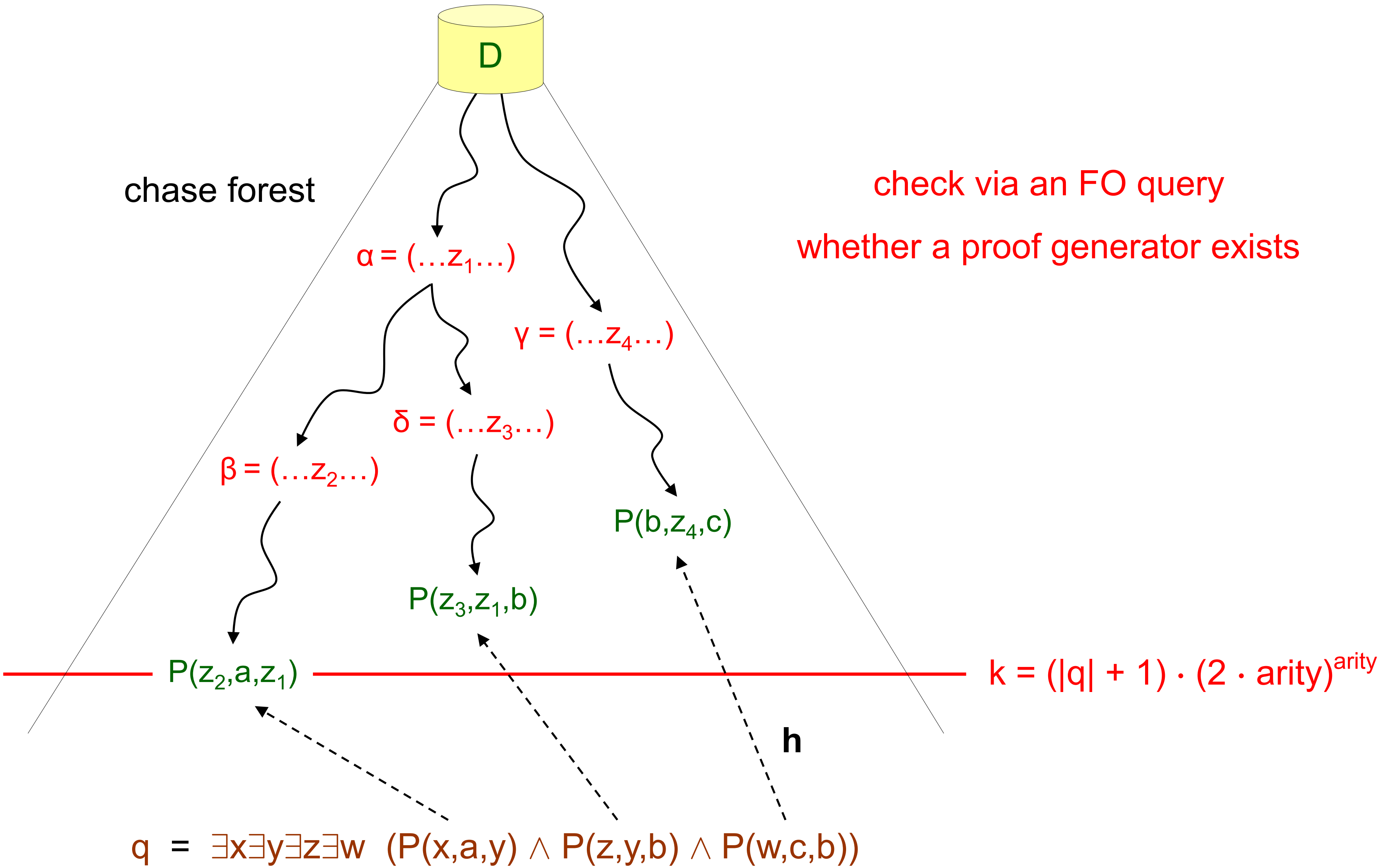




$q \;=\; \exists x \exists y \exists z \exists w \;\; (P(x,a,y) \wedge P(z,y,b) \wedge P(w,c,b))$

[Gottlob, Manna & P., IJCAI 2015]

Proof Generator



check via an FO query

whether a proof generator exists

[Gottlob, Manna & P., IJCAI 2015]

Polynomial Rewritings

… assuming two domain elements

Theorem: A (**Linear**,**CQ**) query can be rewritten in polynomial time as:

- An $\exists FO^{+}$/NDL query for bounded arity predicates
- An FO query for arbitrary signatures

Proof:

- Bounded arity signatures - via the polynomial witness property
- Arbitrary signatures - via proof generators

The Guarded Family

Theorem: It holds that

1. (**Linear**,**CQ**) is UCQ-Rewritable

2. (**Guarded**,**CQ**) is not FO-Rewritable, but is Datalog-Rewritable

3. (**Weakly-Guarded**,**CQ**) is not Datalog-Rewritable

[Bárány, Benedikt & ten Cate, MFCS 2013] + [Gottlob, Rudolph & Šimkus, PODS 2014]

(**Guarded**,**CQ**) is Datalog-Rewritable

Via inference rules - inspired by DLs

$$\frac{\alpha \to \beta \wedge A}{\alpha \to A} \quad A \text{ has no existential variables}$$

$$\frac{\alpha \to \beta \quad \gamma_1 \wedge \gamma_2 \to \delta}{\alpha \wedge h(\gamma_1) \to \beta \wedge h(\delta)}$$

$\gamma_1 \wedge \gamma_2 \to \delta$ is a Datalog rule, h is a homomorphism from γ_2 to β with $\mathsf{vars}(h(\gamma_1)) \subseteq \mathsf{vars}(\alpha)$.

$$\frac{\alpha \to \beta}{g(\alpha) \to g(\beta)} \quad g : \mathsf{vars}(\alpha) \to \mathsf{vars}(\alpha)$$

[Gottlob, Rudolph & Šimkus, PODS 2014]

(**Guarded**,**CQ**) is Datalog-Rewritable

Via inference rules - inspired by DLs

Given an OMQ $Q = (\Sigma, q)$ from (**Guarded**,**CQ**),

the size of the Datalog rewriting is at most $2^\wedge\big(\#\text{pred}(\Sigma) \cdot \#\text{body-vars}(\Sigma)^{\text{arity}(\Sigma)}\big)$

worst-case optimal?

[Gottlob, Rudolph & Šimkus, PODS 2014]

Polynomial Rewritings

… assuming two domain elements

Theorem: A (**Guarded**, **Full CQ**) or (**Guarded**, **Acyclic CQ**) query over bounded arity predicates can be rewritten in polynomial time as a Datalog query

Proof: Via types

- Build all possible types
- Mark "bad" types
- From marked types to Datalog rules that capture ground consequences

Tuesday, March 27, ICDT4: Logic and Dependencies

[Ahmetaj, Ortiz & Šimkus, ICDT 2018]

Wrap Up

(**Linear**,**CQ**) $\xrightarrow{\text{EXPTIME}}$ UCQ

(**Linear**,**CQ**) $\overset{\text{PTIME}}{\not\longrightarrow}$ $\exists FO^+$, NDL, FO

(**Linear**,**CQ**) $\xrightarrow[\text{bounded arity, \{0,1\}}]{\text{PTIME}}$ $\exists FO^+$, NDL

(**Linear**,**CQ**) $\xrightarrow[\{0,1\}]{\text{PTIME}}$ FO

Wrap Up

(**Guarded**,**CQ**) $\not\longrightarrow$ FO

(**Guarded**,**CQ**) $\xrightarrow{\text{2EXPTIME}}$ Datalog

(**Guarded**,**CQ**) $\xrightarrow[\text{bounded arity}]{\text{EXPTIME}}$ Datalog

(**Guarded**,**FCQ/ACQ**) $\xrightarrow[\text{bounded arity, \{0,1\}}]{\text{PTIME}}$ Datalog

Open Problems

(**Linear**,**CQ**) $\xrightarrow[\text{bounded arity, \{0,1\}}]{\text{PTIME}}$ UCQ

(**Linear**,**CQ**) $\xrightarrow[\text{\{0,1\}}]{\text{PTIME}}$ NDL

(**Guarded**,**CQ**) $\xrightarrow[\text{bounded arity, \{0,1\}}]{\text{PTIME}}$ Datalog

Thank You!