Open-World Query Answering
Under Number Restrictions

Antoine Amarilli$^{1,2}$

$^1$ Télécom ParisTech, Paris, France

$^2$ University of Oxford, Oxford, United Kingdom

January 17, 2014
Open-World Query Answering

- Instance $I$

\[ \{ R(a, b), T(b) \} \]
Open-World Query Answering

- **Instance** \( I \)
  \( \{ R(a, b), T(b) \} \)

- **Constraints** \( \Theta \) of a fragment \( F \)
  \( \forall xy R(x, y) \Rightarrow S(y) \)
  (here: fragments of **first-order logic** with no constants)
Open-World Query Answering

- **Instance** $I$  
  \{R(a, b), T(b)\}

- **Constraints** $\Theta$ of a fragment $F$  
  $\forall xy R(x, y) \Rightarrow S(y)$
  (here: fragments of first-order logic with no constants)

- **Query** $q$ of a class $Q$  
  $\exists x S(x) \land T(x)$
  (here: UCQ or CQ: (union of) Boolean conjunctive queries)
Open-World Query Answering

- Instance $I$  \( \{ R(a, b), T(b) \} \)
- Constraints $\Theta$ of a fragment $F$  \( \forall xy R(x, y) \Rightarrow S(y) \)
  (here: fragments of first-order logic with no constants)
- Query $q$ of a class $Q$  \( \exists x S(x) \land T(x) \)
  (here: UCQ or CQ: (union of) Boolean conjunctive queries)

\[ \Rightarrow QA_{\text{unr}}(F, Q) : \text{does } q \text{ hold in every } J \supseteq I \text{ satisfying } \Theta? \]
(written $I, \Theta \models_{\text{unr}} q$)
Open-World Query Answering

- **Instance** $I$ 
  \[ \{ R(a, b), T(b) \} \]
- **Constraints** $\Theta$ of a fragment $F$ 
  \[ \forall xy \ R(x, y) \Rightarrow S(y) \]
  (here: fragments of **first-order logic** with no constants)
- **Query** $q$ of a class $Q$ 
  \[ \exists x \ S(x) \land T(x) \]
  (here: **UCQ** or **CQ**: (union of) Boolean conjunctive queries)

$\Rightarrow$ **QA$_{unr}$**$(F, Q)$: does $q$ hold in every $J \supseteq I$ satisfying $\Theta$? 
(written $I, \Theta \models_{unr} q$)

$\Rightarrow$ **QA$_{fin}$**$(F, Q)$: does $q$ hold in every finite $J \supseteq I$ satisfying $\Theta$? 
(written $I, \Theta \models_{fin} q$)
Open-World Query Answering

- **Instance** $I$  
  \[ \{ R(a, b), T(b) \} \]

- **Constraints** $\Theta$ of a fragment $F$  
  \[ \forall xy \, R(x, y) \Rightarrow S(y) \]  
  (here: fragments of first-order logic with no constants)

- **Query** $q$ of a class $Q$  
  \[ \exists x \, S(x) \land T(x) \]  
  (here: UCQ or CQ: (union of) Boolean conjunctive queries)

$\Rightarrow$  

**QA*unr*(F, Q):** does $q$ hold in every $J \supseteq I$ satisfying $\Theta$?  
(written $I, \Theta \models_{\text{unr}} q$)

$\Rightarrow$  

**QA*fin*(F, Q):** does $q$ hold in every finite $J \supseteq I$ satisfying $\Theta$?  
(written $I, \Theta \models_{\text{fin}} q$)

$\Rightarrow$  

Equivalently: is there a (finite) model of $I \land \Theta \land \neg q$?
Dependencies DEP

\[ \tau : \forall x (\phi(x) \Rightarrow \exists y A(x, y)) \]
Dependencies DEP

$$\tau : \forall x(\phi(x) \Rightarrow \exists y A(x, y))$$

- **Tuple-Generating Dependencies** TGD: \(A\) is a **regular** atom.
- **Inclusion Dependencies** ID:
  - \(\Rightarrow \phi\) is an **atom**, no repeated variables.
- **Unary Inclusion Dependencies** UID:
  - \(\Rightarrow\) **Only one exported variable** (occurring in \(\phi\) and \(A\)).
  - **Example:** \(\forall e b, \text{Boss}(e, b) \Rightarrow \exists b' \text{Boss}(b, b')\).
  - **Written** \(\text{Boss}^2 \subset \text{Boss}^1\).
Dependencies DEP

\[ \tau : \forall x(\phi(x) \Rightarrow \exists y A(x, y)) \]

- **Tuple-Generating Dependencies** TGD: \( A \) is a regular atom.
  - **Inclusion Dependencies** ID:
    \[ \Rightarrow \phi \text{ is an atom, no repeated variables.} \]
  - **Unary Inclusion Dependencies** UID:
    \[ \Rightarrow \text{Only one exported variable (occurring in } \phi \text{ and } A). \]
    \[ \Rightarrow \text{Example: } \forall e, b, \text{ Boss}(e, b) \Rightarrow \exists b' \text{ Boss}(b, b'). \]
    \[ \Rightarrow \text{Written } \text{Boss}^2 \subseteq \text{Boss}^1. \]

- **Equality-Generating Dependencies** EGD: \( A \) is an equality.
  - **Functional Dependencies** FD:
    \[ \Rightarrow \forall xy \ (S(x) \wedge S(y) \wedge \bigwedge_{i \in L} x_i = y_i) \Rightarrow x_r = y_r. \]
  - **Unary Functional Dependencies:** \( |L| = 1. \)
    \[ \Rightarrow \text{Example: } \forall e, e', b, b', \text{ Boss}(e, b), \text{ Boss}(e', b'), e = e' \Rightarrow b = b'. \]
    \[ \Rightarrow \text{Written } \text{Boss}^1 \rightarrow \text{Boss}^2. \]
  - **Key Dependencies:** \( \bigwedge_{r \in \text{Pos}(R)} R^K \rightarrow R^r \text{ for some } K \subseteq \text{Pos}(R). \)
  - **Unary Key Dependencies:** \( |K| = 1. \)
Logics

- **Guarded Fragment GF:**
  - Contains regular atoms and equality atoms.
  - Closed under Boolean connectives $\land$, $\lor$, $\neg$, etc.
  - Quantification: given an atom $A(x, y)$ and formula $\phi(x, y)$ with free variables exactly as indicated:
    - $\forall x (A \Rightarrow \phi)$.
    - $\exists x (A \land \phi)$. 

Two-Variable Guarded Fragment $\text{GF}^2$:
- Only two distinct variables.
- Only unary and binary predicates of the signature ($\leq 2$).

Two-Variable Guarded Fragment with Counting $\text{GC}^2$:
- Quantifiers $\exists c. y; A(x, y)$ and $\forall c. y; A(x, y)$ with $A$ a binary atom and $c \in \mathbb{N}$.
- Example: $\forall e \exists b; \text{Boss}(e, b)$.
Logics

- **Guarded Fragment GF:**
  - Contains regular atoms and equality atoms.
  - Closed under Boolean connectives $\land$, $\lor$, $\neg$, etc.
  - Quantification: given an atom $A(x, y)$ and formula $\phi(x, y)$ with free variables exactly as indicated:
    - $\forall x (A \Rightarrow \phi)$.
    - $\exists x (A \land \phi)$.

- **Two-Variable Guarded Fragment GF$^2$:**
  - Only two distinct variables.
  - Only unary and binary predicates of the signature $(\sigma_{\leq 2})$. 
Logics

- **Guarded Fragment GF:**
  - Contains regular atoms and equality atoms.
  - Closed under Boolean connectives $\wedge$, $\vee$, $\neg$, etc.
  - Quantification: given an atom $A(x, y)$ and formula $\phi(x, y)$ with free variables exactly as indicated:
    - $\forall x (A \Rightarrow \phi)$.
    - $\exists x (A \land \phi)$.

- **Two-Variable Guarded Fragment GF$^2$:**
  - Only two distinct variables.
  - Only unary and binary predicates of the signature $(\sigma_{\leq 2})$.

- **Two-Variable Guarded Fragment with Counting GC$^2$:**
  - Quantifiers $\exists^{\leq c} y$, $A(x, y)$ and $\exists^{\geq c} y$, $A(x, y)$ with $A$ a binary atom and $c \in \mathbb{N}$.
  - Example: $\forall e \exists^{\leq 1} b$, Boss($e, b$).
General Results

- **Negative results:**
  - $QA_\bullet(FO, CQ^-)$ is undecidable [Trakhtenbrot, 1963].
  - $QA_\bullet(TGD, CQ^-)$ is undecidable [Calì et al., 2013].
  - $QA_\bullet(UKD \cup BID, CQ)$ is undecidable [Calì et al., 2003].
General Results

* Negative results:
  - $\text{QA}_\bullet(\text{FO}, \text{CQ}^-)$ is undecidable [Trakhtenbrot, 1963].
  - $\text{QA}_\bullet(\text{TGD}, \text{CQ}^-)$ is undecidable [Calì et al., 2013].
  - $\text{QA}_\bullet(\text{UKD} \cup \text{BID}, \text{CQ})$ is undecidable [Calì et al., 2003].

* Positive results:
  - $\text{QA}_\bullet(\text{GF}, \text{UCQ})$ is in 2EXPTIME [Barany et al., 2010].
  - $\text{QA}_\bullet(\text{GC}^2, \text{CQ})$ is decidable [Pratt-Hartmann, 2009].
General Results

- **Negative results:**
  - $QA_\bullet(\text{FO}, CQ^-)$ is undecidable [Trakhtenbrot, 1963].
  - $QA_\bullet(\text{TGD}, CQ^-)$ is undecidable [Calì et al., 2013].
  - $QA_\bullet(\text{UKD} \cup \text{BID}, CQ)$ is undecidable [Calì et al., 2003].

- **Positive results:**
  - $QA_\bullet(\text{GF}, \text{UCQ})$ is in 2EXPTIME [Barany et al., 2010].
  - $QA_\bullet(\text{GC}^2, CQ)$ is decidable [Pratt-Hartmann, 2009].

⇒ Can we have both high-arity constraints and expressive low-arity constraints, including equality constraints?
Table of Contents

1 Introduction

2 Extending GC² Query Answering

3 Unrestricted Query Answering

4 Finite Query Answering

5 Conclusion
Result Statement

- **Frontier-One Dependencies FR1:**
  - Subset of TGD which includes UID.
  - One exported variable.
  - No repeated variable in the head.
Result Statement

- **Frontier-One Dependencies FR1:**
  - Subset of TGD which includes UID.
  - One exported variable.
  - No repeated variable in the head.

- **Reification** $\mathcal{R}$ of a structure $M$ from $\sigma$ to (extended) $\sigma_{\leq 2}$:
  - Add binary predicates $R_i$ for every $i \in \text{Pos}(R)$ and $R \in \sigma_{> 2}$.
  - Replace facts $R(a)$ of $> 2$-ary predicates by a fresh element $f$ and $R_i(f, a_i)$ for all $i \in \text{Pos}(R)$.
  - Example: $R(a, a, b)$ becomes $R_1(f, a), R_2(f, a), R_3(f, b)$. 
Result Statement

- **Frontier-One Dependencies** FR1:
  - Subset of TGD which includes UID.
  - One exported variable.
  - No repeated variable in the head.

- **Reification** $\mathcal{R}$ of a structure $M$ from $\sigma$ to (extended) $\sigma_{\leq 2}$:
  - Add binary predicates $R_i$ for every $i \in \text{Pos}(R)$ and $R \in \sigma_{> 2}$.
  - Replace facts $R(a)$ of $> 2$-ary predicates by a fresh element $f$ and $R_i(f, a_i)$ for all $i \in \text{Pos}(R)$.
  - Example: $R(a, a, b)$ becomes $R_1(f, a), R_2(f, a), R_3(f, b)$.

- **Frontier-One Acyclic Dependencies** FR1$^a$:
  - The Gaifman graph of the reification of the body is acyclic.
Result Statement

- **Frontier-One Dependencies FR1:**
  - $\Rightarrow$ Subset of TGD which includes UID.
  - $\Rightarrow$ One exported variable.
  - $\Rightarrow$ No repeated variable in the head.

- **Reification $R$ of a structure $M$ from $\sigma$ to (extended) $\sigma_{\leq 2}:$**
  - $\Rightarrow$ Add binary predicates $R_i$ for every $i \in \text{Pos}(R)$ and $R \in \sigma_{> 2}.$
  - $\Rightarrow$ Replace facts $R(a)$ of $> 2$-ary predicates by a fresh element $f$ and $R_i(f, a_i)$ for all $i \in \text{Pos}(R)$.
  - $\Rightarrow$ Example: $R(a, a, b)$ becomes $R_1(f, a), R_2(f, a), R_3(f, b).$

- **Frontier-One Acyclic Dependencies FR1$^a$:**
  - $\Rightarrow$ The Gaifman graph of the reification of the body is acyclic.

---

**Theorem**

$\text{QA}_\bullet((\text{UKD} \cup \text{GC}^2 \cup \text{FR1}^a, \text{CQ}) \text{ is decidable.}$
Proof Idea

- **Encode** constraints from UKD ∪ GC² ∪ FR1ᵃ to GC².
- Show that QA under the original constraints is equivalent to QA for the encoded constraints (and decide it as GC² QA):
  - ⇒ The reification of counterexample models should be counterexample models for the encoding (easy).
  - ⇒ Counterexample models should be decodable from counterexample models for the encoded constraints (harder).
Proof Idea

- **Encode** constraints from \( \text{UKD} \cup \text{GC}^2 \cup \text{FR1}^a \) to \( \text{GC}^2 \).
- Show that QA under the original constraints is equivalent to QA for the encoded constraints (and decide it as \( \text{GC}^2 \) QA):
  - The reification of counterexample models should be counterexample models for the encoding (easy).
  - Counterexample models should be decodable from counterexample models for the encoded constraints (harder).

- **Well-formedness** constraints \( \text{wf}(\sigma) \) of \( \text{GC}^2 \) for the encoding:
  - Elements are regular elements or \( R \)-facts for some \( R \in \sigma_{>2} \).
  - The \( R \)'s connect regular elements and \( R \)-fact elements.
  - Every fact element for \( R \) has exactly one of each \( R_i \).
  - The \( R \in \sigma_{\leq 2} \) connect regular elements.
Encoding

- Encoding a key $\phi \in \text{UKD}$ to $\mathcal{R}(\phi)$:
  - $\Rightarrow$ “$R^i$ is a key” encoded to $\forall x \exists y \leq 1, R_i(y, x)$.
  - $\Rightarrow \mathcal{R}(\Phi)$ is clearly a $\text{GC}^2$ constraint.
Encoding

- Encoding a key $\phi \in \text{UKD}$ to $\mathcal{R}(\phi)$:
  $\Rightarrow$ “$R^i$ is a key” encoded to $\forall x \exists y \leq 1 \; R_i(y, x)$.
  $\Rightarrow$ $\mathcal{R}(\Phi)$ is clearly a $\text{GC}^2$ constraint.

- Encoding a high-arity constraint $\delta \in \text{FR1}^a$ to $\mathcal{R}(\delta)$:
  $\Rightarrow$ Apply reification to the body and modify the head if $\in \sigma_{>2}$.
  $\Rightarrow$ Example:
    $\delta : \forall xyz, S(y, x) \land R(x, x, z) \Rightarrow \exists w w', R(x, w, w')$
    $\Rightarrow$ $\mathcal{R}(\delta) : \forall x ((\exists y, S(y, x)) \land (\exists f, R_1(f, x) \land R_2(f, x) \land (\exists z, R_3(f, z)))$
    $\Rightarrow \exists f, R_1(f, x))$.
  $\Rightarrow$ $\mathcal{R}(\Delta)$ expressible as a $\text{GF}^2$ constraint.
Encoding

- Encoding a key $\phi \in \text{UKD}$ to $\mathcal{R}(\phi)$:
  \[ \Rightarrow \text{“} R^i \text{ is a key” encoded to } \forall x \exists \leq 1 y, R_i(y, x). \]
  \[ \Rightarrow \mathcal{R}(\Phi) \text{ is clearly a } \text{GC}^2 \text{ constraint.} \]

- Encoding a high-arity constraint $\delta \in \text{FR1}^a$ to $\mathcal{R}(\delta)$:
  \[ \Rightarrow \text{Apply reification to the body and modify the head if } \in \sigma_{>2}. \]
  \[ \Rightarrow \text{Example:} \]
  \[ \delta : \forall xyz, S(y, x) \land R(x, x, z) \Rightarrow \exists ww', R(x, w, w') \]
  \[ \Rightarrow \mathcal{R}(\delta) : \forall x ((\exists y, S(y, x)) \land (\exists f, R_1(f, x) \land R_2(f, x) \land (\exists z, R_3(f, z))) \]
  \[ \Rightarrow \exists f, R_1(f, x). \]
  \[ \Rightarrow \mathcal{R}(\Delta) \text{ expressible as a } \text{GF}^2 \text{ constraint.} \]

- Encode the instance $I$ to $\mathcal{R}(I)$ straightforwardly.
- Encode the query $q \in \text{CQ}$ to $\mathcal{R}(q)$ straightforwardly.
- Leave the constraints $\Theta \subseteq \text{GC}^2$ unchanged.
Concluding the Proof

- Take an extension $J$ of $I$ satisfying $\Delta$, $\Theta$, $\Phi$ and violating $q$:
  
  $\Rightarrow R(J)$ is an extension of $R(I)$ satisfying $R(\Delta)$, $\Theta$, $R(\Phi)$ and $\text{wf}(\sigma)$ and violating $R(q)$. 
Concluding the Proof

- Take an extension $J$ of $I$ satisfying $\Delta$, $\Theta$, $\Phi$ and violating $q$:
  \[ \Rightarrow \mathcal{R}(J) \text{ is an extension of } \mathcal{R}(I) \text{ satisfying } \mathcal{R}(\Delta), \Theta, \mathcal{R}(\Phi) \text{ and } \text{wf}(\sigma) \text{ and violating } \mathcal{R}(q). \]

- Conversely, take an extension of $\mathcal{R}(I)$ satisfying $\mathcal{R}(\Delta)$, $\Theta$, $\mathcal{R}(\Phi)$ and $\text{wf}(\sigma)$ and violating $\mathcal{R}(q)$.
  \[ \Rightarrow \text{Need to argue that, w.l.o.g., there are no duplicate facts (} f \text{ and } f' \text{ representing } R(a, b, c)). \]

- Decode an extension of $I$ satisfying $\Delta$, $\Theta$, $\Phi$ and violating $q$. 

Concluding the Proof

- Take an extension $J$ of $I$ satisfying $\Delta$, $\Theta$, $\Phi$ and violating $q$:
  \[ R(J) \text{ is an extension of } R(I) \text{ satisfying } R(\Delta), \Theta, R(\Phi) \text{ and } \text{wf}(\sigma) \text{ and violating } R(q). \]

- Conversely, take an extension of $R(I)$ satisfying $R(\Delta)$, $\Theta$, $R(\Phi)$ and $\text{wf}(\sigma)$ and violating $R(q)$.
  \[ \Rightarrow \text{ Need to argue that, w.l.o.g., there are no duplicate facts } \ (f \text{ and } f' \text{ representing } R(a, b, c)). \]
  \[ \Rightarrow \text{ Decode an extension of } I \text{ satisfying } \Delta, \Theta, \Phi \text{ and violating } q. \]
  \[ \Rightarrow \text{ Decide } QA_\bullet(UKD \cup GC^2 \cup FR1^a, CQ) \text{ from } QA_\bullet(GC^2, CQ). \]
# Table of Contents

1. Introduction
2. Extending GC$^2$ Query Answering
3. Unrestricted Query Answering
4. Finite Query Answering
5. Conclusion
The Chase and Separability

- **Universal model**: extension of $I$ satisfying $\Theta$ and violating every $q$ unless $I, \Theta \models_{\text{unr}} q$.
- **The chase $I^{\Theta}$**: infinite universal model for TGD and UCQ:
  - $\Rightarrow$ Whenever a TGD is violated, create the missing head fact.
  - $\Rightarrow$ Always use fresh existential witnesses.
The Chase and Separability

- **Universal model**: extension of $I$ satisfying $\Theta$ and violating every $q$ unless $I, \Theta \models_{\text{unr}} q$.
- **The chase $I^\Theta$**: infinite universal model for TGD and UCQ:
  - $\Rightarrow$ Whenever a TGD is violated, create the missing head fact.
  - $\Rightarrow$ Always use fresh existential witnesses.

- $\Phi \cup \Delta \subseteq \text{EGD} \cup \text{TGD}$ is **separable** if $I \models \Phi$ implies $I^\Delta \models \Phi$.
  $\Rightarrow$ $\text{QA}_{\text{unr}}(\text{EGD} \cup (\text{TGD} \cap \text{GF}), \text{UCQ})$ is **decidable** in this case:
  - Check if $I \models \Phi$
  - Decide $\text{QA}_{\text{unr}}(\text{TGD} \cap \text{GF}, \text{UCQ})$ problem ignoring EGDs.

  $\Rightarrow$ $\text{QA}_{\text{unr}}(\text{FD} \cup \text{FR}1^a, \text{UCQ})$ is **decidable** (always separable).
Result and Intuition

Theorem

\[ \text{QA}_{\text{unr}}(\text{FD} \cup \text{GC}^2 \cup \text{FR1}, \text{CQ}) \text{ is decidable.} \]
Theorem

$\text{QA}_{\text{unr}} (\text{FD} \cup \text{GC}^2 \cup \text{FR1}, \text{CQ})$ is decidable.

Idea: counterexample models $M$ for $\text{GC}^2 \cup \text{FR1}^a$ satisfy w.l.o.g.:

Unicity. There are no two facts $R(a)$ and $R(b)$ with $a_i = b_i$ for $R \in \sigma_{>2}$ unless both are in the instance $I$.

$\Rightarrow$ Any FD violation for $\sigma_{>2}$ must occur in $I$.

$\Rightarrow$ FDs can be checked on $I$ and ignored afterwards.
Result and Intuition

Theorem

\[ \text{QA}_{unr}(\text{FD} \cup \text{GC}^2 \cup \text{FR1}, \text{CQ}) \text{ is decidable.} \]

Idea: counterexample models \( M \) for \( \text{GC}^2 \cup \text{FR1}^a \) satisfy w.l.o.g.:

**Unicity.** There are no two facts \( R(a) \) and \( R(b) \) with \( a_i = b_i \) for \( R \in \sigma_{>2} \) unless both are in the instance \( I \).

\( \Rightarrow \) Any FD violation for \( \sigma_{>2} \) must occur in \( I \).
\( \Rightarrow \) FDs can be checked on \( I \) and ignored afterwards.

**Acyclicity.** The Gaifman graph of \( \mathcal{R}(M) \) is acyclic except for \( I \):

\( \Rightarrow \) \( \text{FR1} \setminus \text{FR1}^a \) dependencies can only match on \( I \).
\( \Rightarrow \) Convert \( \text{FR1} \) to \( \text{FR1}^a \) (enumerate matches).

\( \Rightarrow \) Reduce \( \text{QA}_{unr}(\text{FD} \cup \text{GC}^2 \cup \text{FR1}, \text{CQ}) \) to \( \text{QA}_{unr}(\text{GC}^2 \cup \text{FR1}^a, \text{CQ}) \).
Unraveling the Counterexample Model

Unravelling $M$ to a suitable $M'$ (with mapping $\pi'$):

- Add dummy binary facts covering and connecting all elements.
- Decompose the facts in bags:
  - one bag per fact of $\sigma_{>2}$,
  - one bag per guarded pair $\{a, b\}$ with all unary and binary facts.
Unraveling the Counterexample Model

**Unravelling** \( M \) to a suitable \( M' \) (with mapping \( \pi' \)):

- Add dummy binary facts **covering** and **connecting** all elements.
- Decompose the facts in **bags**:
  - one bag per fact of \( \sigma_{>2} \),
  - one bag per guarded pair \( \{a, b\} \) with all unary and binary facts.
- Build \( M' \) as a tree of bags by the following **inductive process**:
  - The root bag of \( M' \) is \( I \).
  - The children of \( t \in M' \) are, for every \( a \in \text{dom}(t) \):
    - For every \( \sigma_{\leq 2} \)-bag \( t' \) of \( M \) containing \( \pi'(a) \):
      An **isomorphic copy** of \( t' \) in \( M' \), with \( a \) and a fresh element.
    - For every \( R^i \in \text{Pos}(\sigma_{>2}) \) such that \( \pi'(a) \) occurs at \( R^i \) in \( M \), if \( a \) does not occur at \( R^i \) in \( M' \):
      A \( \sigma_{>2} \)-bag \( \{R(b)\} \) with \( b \) fresh except \( b_i = a \).
  - Do not consider in a bag the **previous element** used to reach it.
Example

\[
\begin{align*}
I &= \{ N(a, b) \} \\
M &= I \cup \{ R(b, c), \\
& \quad R(c, d), R(d, b), \\
& \quad S(b, b, d) \}
\end{align*}
\]
Example

\[ l = \{ N(a, b) \} \]
\[ M = l \cup \{ R(b, c), R(c, d), R(d, b), S(b, b, d) \} \]
Example

\[ I = \{ N(a, b) \} \]
\[ M = I \cup \{ R(b, c), R(c, d), R(d, b), S(b, b, d) \} \]
Example

\[ I = \{ N(a, b) \} \]
\[ M = I \cup \{ R(b, c), R(c, d), R(d, b), S(b, b, d) \} \]
Example

\[ I = \{ N(a, b) \} \]
\[ M = I \cup \{ R(b, c), R(c, d), R(d, b), S(b, b, d) \} \]
Example

\[ I = \{ N(a, b) \} \]
\[ M = I \cup \{ R(b, c), R(c, d), R(d, b), S(b, b, d) \} \]
Properties of the Construction

- Preserves the base instance $I$. 
Properties of the Construction

- Preserves the base instance $I$.
- Maps back to the original model by the homomorphism $\pi'$.
  - Ensures that the query is still false.
Properties of the Construction

- Preserves the base instance \( I \).
- Maps back to the original model by the homomorphism \( \pi' \).
  \[\Rightarrow\] Ensures that the query is still false.
- Isomorphism between 1-neighborhoods for \( \sigma_{\leq 2} \) following \( \pi' \).
  \[\Rightarrow\] Ensures that GF\(^2\) constraints are preserved (guarded bisimilar).
  \[\Rightarrow\] Ensures that number restrictions are preserved.
Properties of the Construction

- Preserves the base instance $I$.
- Maps back to the original model by the homomorphism $\pi'$.
  - $\Rightarrow$ Ensures that the query is still false.
- Isomorphism between 1-neighborhoods for $\sigma_{\leq 2}$ following $\pi'$.
  - $\Rightarrow$ Ensures that GF\(^2\) constraints are preserved (guarded bisimilar).
  - $\Rightarrow$ Ensures that number restrictions are preserved.
- The mapping $\pi'$ is surjective for guarded pairs.
  - $\Rightarrow$ Necessary for guarded bisimulation.
Properties of the Construction

- Preserves the base instance $I$.
- Maps back to the original model by the homomorphism $\pi'$.
  \[\Rightarrow\quad\text{Ensures that the query is still false.}\]
- Isomorphism between $1$-neighborhoods for $\sigma_{\leq 2}$ following $\pi'$.
  \[\Rightarrow\quad\text{Ensures that $GF^2$ constraints are preserved (guarded bisimilar).}\]
  \[\Rightarrow\quad\text{Ensures that number restrictions are preserved.}\]
- The mapping $\pi'$ is surjective for guarded pairs.
  \[\Rightarrow\quad\text{Necessary for guarded bisimulation.}\]
- Elements still occur at the same positions of $\text{Pos}(\sigma_{>2})$.
  \[\Rightarrow\quad\text{Ensures that FR1^a constraints are preserved.}\]
Properties of the Construction

- Preserves the base instance $I$.
- Maps back to the original model by the homomorphism $\pi'$.
  - $\Rightarrow$ Ensures that the query is still false.
- Isomorphism between 1-neighborhoods for $\sigma_{\leq 2}$ following $\pi'$.
  - $\Rightarrow$ Ensures that $GF^2$ constraints are preserved (guarded bisimilar).
  - $\Rightarrow$ Ensures that number restrictions are preserved.
- The mapping $\pi'$ is surjective for guarded pairs.
  - $\Rightarrow$ Necessary for guarded bisimulation.
- Elements still occur at the same positions of $\text{Pos}(\sigma_{>2})$:
  - $\Rightarrow$ Ensures that $FR1^a$ constraints are preserved.
- They do so at most once (except in the instance):
  - $\Rightarrow$ Ensures Unicity.
Properties of the Construction

- Preserves the base instance $I$.
- Maps back to the original model by the homomorphism $\pi'$.
  $\Rightarrow$ Ensures that the query is still false.
- Isomorphism between $1$-neighborhoods for $\sigma_{\leq 2}$ following $\pi'$.
  $\Rightarrow$ Ensures that $\text{GF}^2$ constraints are preserved (guarded bisimilar).
  $\Rightarrow$ Ensures that number restrictions are preserved.
- The mapping $\pi'$ is surjective for guarded pairs.
  $\Rightarrow$ Necessary for guarded bisimulation.
- Elements still occur at the same positions of $\text{Pos}(\sigma_{>2})$:
  $\Rightarrow$ Ensures that $\text{FR}1^a$ constraints are preserved.
- They do so at most once (except in the instance):
  $\Rightarrow$ Ensures Unicity.
- The model is a tree of bags.
  $\Rightarrow$ Ensures Acyclicity (and bounded treewidth).
# Table of Contents

1. Introduction
2. Extending GC² Query Answering
3. Unrestricted Query Answering
4. Finite Query Answering
5. Conclusion
Finite Controllability

- **Finite controllability (FC):** finite and unrestricted QA coincide.
Finite Controllability

- Finite controllability (FC): finite and unrestricted QA coincide.
- Holds for GF but fails with number restrictions:
  - Consider \( \Theta : R^2 \rightarrow R^1 \), \( R^2 \subseteq R^1 \), and \( I = \{ A(a), R(a, b) \} \).
  - Universal infinite chase model \( A(a), R(a, b), R(b, c), \ldots \).
  - Finite model has to loop back, on \( a \) because of the FD:
    \( A(a), R(a, b), R(b, c), \ldots, R(y, z), R(z, a) \).
  - For \( q : R(x, y) \land A(y) \), we have \( I, \Theta \models_{\text{fin}} q \) but \( I, \Theta \not\models_{\text{unr}} q \).
Finite Controllability

- **Finite controllability** (FC): finite and unrestricted QA coincide.

- Holds for GF but fails with **number restrictions**:
  - Consider $\Theta : R^2 \rightarrow R^1, R^2 \subseteq R^1$, and $I = \{ A(a), R(a, b) \}$.
  - Universal infinite chase model $A(a), R(a, b), R(b, c), \ldots$.
  - Finite model has to loop back, on $a$ because of the FD: $A(a), R(a, b), R(b, c), \ldots, R(y, z), R(z, a)$.
  - $\Rightarrow$ For $q : R(x, y) \wedge A(y)$, we have $I, \Theta \models_{\text{fin}} q$ but $I, \Theta \not\models_{\text{unr}} q$.

- **Separability** not useful for finite QA (the chase is infinite):
  - Separability not closed under finite implication [Rosati, 2006].
  - $\Rightarrow$ $QA_{\text{fin}}(KD \cup ID, CQ)$ undecidable even assuming separability.
Decidable Finite QA

- \( QA_{\text{fin}}(\mathbf{GC}^2, \mathbf{CQ}) \) not FC but \textit{decidable} [Pratt-Hartmann, 2009].

  \( \Rightarrow \) Only for \textit{arity-two}.
Decidable Finite QA

- $\text{QA}_{\text{fin}}(\text{GC}^2, \text{CQ})$ not FC but **decidable** [Pratt-Hartmann, 2009].
  $\Rightarrow$ Only for arity-two.

- Enforce **chase termination** to get a finite universal model.
  $\Rightarrow$ Too restrictive.
Decidable Finite QA

- $\text{QA}_{\text{fin}}(\text{GC}^2, \text{CQ})$ not FC but **decidable** [Pratt-Hartmann, 2009].
  - ⇒ Only for **arity-two**.

- Enforce **chase termination** to get a finite universal model.
  - ⇒ Too **restrictive**.

- Restrict the language to **enforce FC**:
  - ⇒ $\text{KD} \cup \text{ID}$ under a **foreign key** condition is FC [Rosati, 2011].
  - ⇒ Also **restrictive**.
Result Statement

- We focus on unary IDs and (general) FDs, arbitrary arity.
- The implication problem for UIDs and FDs is decidable: PTIME finite closure construction [Cosmadakis et al., 1990].
- We show that FC holds up to finite closure:
Result Statement

- We focus on **unary** IDs and (general) FDs, arbitrary arity.
- The **implication problem** for UIDs and FDs is decidable: PTIME **finite closure** construction [Cosmadakis et al., 1990].
- We show that FC holds **up to finite closure**:

**Theorem**

*For every* \( \Phi \cup \Delta \subseteq \text{FD} \cup \text{UID} \) *with finite closure* \( \Phi^* \cup \Delta^* \), *for* \( q \in \text{UCQ} \) *and* \( I \) *an instance s.t.* \( I \models \Phi^* \), *we have* \( I, \Phi \cup \Delta \models_{\text{fin}} q \) *iff* \( I, \Delta^* \models_{\text{unr}} q \).
Result Statement

- We focus on unary IDs and (general) FDs, arbitrary arity.
- The implication problem for UIDs and FDs is decidable: PTIME finite closure construction [Cosmadakis et al., 1990].
- We show that FC holds up to finite closure:

**Theorem**

*For every $\Phi \cup \Delta \subseteq FD \cup UID$ with finite closure $\Phi^* \cup \Delta^*$, for $q \in UCQ$ and $I$ an instance s.t. $I \models \Phi^*$, we have $I, \Phi \cup \Delta \models_{\text{fin}} q$ iff $I, \Delta^* \models_{\text{unr}} q$.***

$\Rightarrow$ QA$_{\text{unr}} (FD \cup UID, UCQ)$ is in NP [Johnson and Klug, 1984] so QA$_{\text{fin}} (FD \cup UID, UCQ)$ is in NP.
Finite Chase

- The **chase** is a universal model but it is infinite.
- The **finite chase** [Rosati, 2011]: for all \( k \), there is a finite universal model for queries of size \( \leq k \).
- **Reuse** similar elements as nulls when chasing.

\[
N(a, b) \\
R(b, c) \\
R(c, d) \\
R(d, e) \\
R(e, f) \\
R(f, g) \\
R(g, h) \\
R(h, e)
\]

\( R^2 \subseteq R^1 \)
Acyclic Queries

- Reuses must not make new queries true relative to the chase.
- We focus on Berge-acyclic constant-free queries of size $\leq k$.
  - The graph $G$ of $q$ has its atoms as vertices.
  - Two atoms are connected if they share one variable.
  - We require $G$ to be acyclic (including self-loops).
- We will eliminate cycles later to take care of cyclic queries.
Acyclic Queries

- Reuses must not make new queries true relative to the chase.
- We focus on Berge-acyclic constant-free queries of size \( \leq k \).
  - The graph \( G \) of \( q \) has its atoms as vertices.
  - Two atoms are connected if they share one variable.
  - We require \( G \) to be acyclic (including self-loops).
- We will eliminate cycles later to take care of cyclic queries.

**Lemma**

*If an extension of \( I \) satisfying \( \Delta \) has a homomorphism to the quotient of the chase by the \( k \)-neighborhood equivalence relation then it is universal for constant-free Berge-acyclic CQs of size \( \leq k \).*
Finite Chase and FDs

- The **dangerous** positions of $R^i$ are the $R^i \in \text{Pos}(R) \setminus \{R^i\}$ such that the FD $R^i \rightarrow R^i$ holds.
- At non-dangerous positions, reusing elements cannot violate **unary** FDs.
- At dangerous positions, we cannot reuse elements!

\[
\begin{align*}
N(a, b) \\
R(b, c) \\
R(c, d) \\
R(d, e) \\
R(e, f) \\
R(f, g) \\
R(g, h) \\
R(h, e)
\end{align*}
\]

\[
\begin{align*}
R^2 & \subseteq R^1 \\
R^2 & \rightarrow R^1
\end{align*}
\]
Finite Chase and FDs and Closure

- **Finite closure** [Cosmadakis et al., 1990]:
  - Whenever $R^i \subseteq S^j$ holds then $\langle R^i \rangle \leq \langle S^j \rangle$.
  - Whenever $S^i \rightarrow S^j$ holds then $\langle S^i \rangle \leq \langle S^j \rangle$.
  - Inequality chains imply the reverse inequalities in the finite.
  - Add the reverse dependencies for such invertible cycles.
Finite Chase and FDs and Closure

- **Finite closure** [Cosmadakis et al., 1990]:
  - Whenever \( R^i \subseteq S^j \) holds then \( \langle R^i \rangle \leq \langle S^j \rangle \).
  - Whenever \( S^i \rightarrow S^j \) holds then \( \langle S^j \rangle \leq \langle S^i \rangle \).
  - Inequality chains imply the reverse inequalities in the finite.
  - Add the reverse dependencies for such invertible cycles.

\[ N(a, b) \]
\[ R(b, c) \]
\[ R(c, d) \]
\[ R(z, b) \]
\[ R(d, e) \]
\[ R(y, z) \]
\[ R(e, f) \]
\[ R(x, y) \]
\[ R(f, g) \]
\[ R(w, x) \]
\[ R(g, w) \]

\( R^2 \subseteq R^1 \)
\( R^2 \rightarrow R^1 \)
\( R^2 \subseteq R^1 \)
\( R^1 \rightarrow R^2 \)

\( \Rightarrow \) When we create a chain with no possibility to reuse, the reverse dependencies must hold.

\( \Rightarrow \) Intuitively: glue both chains together.
Locality Result

After chasing by \( k \) consecutive reversible UIDs, elements at positions connected by UIDs have the same \( k \)-neighborhood.
General Scheme

- Start with the \textit{instance} \( I \).
- \textbf{Chase} by the IDs.
- \textbf{Reuse} elements at non-dangerous positions.
- \textbf{Connect together} elements at dangerous positions.
  \( \Rightarrow \) Use the previous lemma to justify they can be \textit{paired}. 
General Scheme

- Start with the **instance** $I$.
- **Chase** by the IDs.
- **Reuse** elements at non-dangerous positions.
- **Connect together** elements at dangerous positions.
  - ⇒ Use the previous lemma to justify they can be **paired**.

- **Connect elements within an invertible cycle:**
  - ⇒ We say that $(R^i \subseteq S^j) \leftrightarrow (S^p \subseteq T^q)$ if $S^p \rightarrow S^j$.
  - ⇒ An **invertible path** is a cycle of $\leftrightarrow$.
  - ⇒ Chase by the ID of SCCs of $\leftrightarrow$ in **topological order**.
Higher-Arity FDs

- Non-dangerous positions defined w.r.t. unary FDs.
- The non-unary FDs are not considered in the finite closure.
- Reusing the same patterns may violate higher-arity FDs:
  - Must make many patterns out of limited reusable elements.
  - Ex: $R(x_1, a_1, b_1), R(x_2, a_2, b_2), R(x_3, a_1, b_2), R(x_4, a_2, b_1)$.
  - If $R^2 \rightarrow R^3$ then the non-dangerous positions have a unary key so higher-arity FDs are subsumed by UFDs.
Higher-Arity FDs

- Non-dangerous positions defined w.r.t. unary FDs.
- The non-unary FDs are not considered in the finite closure.
- Reusing the same patterns may violate higher-arity FDs:
  - Must make many patterns out of limited reusable elements.
  - Ex: \( R(x_1, a_1, b_1), R(x_2, a_2, b_2), R(x_3, a_1, b_2), R(x_4, a_2, b_1) \).
  - If \( R^2 \rightarrow R^3 \) then the non-dangerous positions have a unary key so higher-arity FDs are subsumed by UFDs.
  - We need to justify that we can make many patterns out of a limited number of elements to reuse.
    - Formally: from \( N \) elements, for any \( K \), make \( NK \) patterns (unless there is a unary key preventing this).
Dense Models

The possibility to find such patterns is a consequence of:

**Lemma**

For any FDs $\Phi$ over $R$, there exists $D \leq |R|$ such that either $R$ has a unary key, or there exists a finite model of $\Phi$ with $O(N)$ elements and $O(N^{D/(D-1)})$ facts.
Dense Models

The possibility to find such patterns is a consequence of:

**Lemma**

For any FDs $\Phi$ over $R$, there exists $D \leq |R|$ such that either $R$ has a unary key, or there exists a finite model of $\Phi$ with $O(N)$ elements and $O(N^{D/(D-1)})$ facts.

- First, **collapse** any UFD cycles of $R$.
- Then, consider the UFD “roots” $T$ of $R$ (there are $\geq 2$) such that $\forall t \in T, \exists s \in Pos(R), s \to t$, and **reduce** to the case:
  - the attributes of $R$ are the non-empty parts of $T$.
  - the roots that determine $X \in Pos(R)$ are exactly those of $X$.
  - the non-unary FDs are as **pessimistic** as possible.
- Finally, **construct** the desired model on this relation.
Expanding Cycles

- We need to **enlarge** cycles of the model, preserving constraints.
Expanding Cycles

- We need to **enlarge** cycles of the model, preserving constraints.
- Group $G$ generated by $X$ is **$k$-acyclic** if there is no word $w$ of length $\leq k$ of $X$ s.t. $w_1 \cdots w_n = e$ unless $w_i = w_i^{-1}$ for some $i$. 

\[
\text{Example: } M = \{R(a, a)g, M' = \{R((a, e), (a, g))R((a, g), (a, e))g, ...\}
\]
Expanding Cycles

- We need to *enlarge* cycles of the model, preserving constraints.
- Group $G$ generated by $X$ is *$k$-acyclic* if there is no word $w$ of length $\leq k$ of $X$ s.t. $w_1 \cdots w_n = e$ unless $w_i = w_{i+1}^{-1}$ for some $i$.
- Build the *product* of the model with a finite acyclic group:
  - Let $L(M) = \{ l^F_i \mid F \in M, 1 \leq i \leq |F| \}$.
  - Let $G$ be a $k$-acyclic group generated by $L(M)$.
  - For $F = R(a) \in M, g \in G$, create $R((a_1, g^{l^F_1} R), \ldots, (a_{|R|}, g^{l^F_{|R|}} R))$.
  - Ex: $M = \{ R(a, a) \}$, $M' = \{ R((a, e), (a, g)), R((a, g), (a, e)) \}$. 
Expanding Cycles

- We need to **enlarge** cycles of the model, preserving constraints.

- Group $G$ generated by $X$ is **$k$-acyclic** if there is no word $w$ of length $\leq k$ of $X$ s.t. $w_1 \cdots w_n = e$ unless $w_i = w_{i+1}^{-1}$ for some $i$.

- Build the **product** of the model with a finite acyclic group:
  - Let $L(M) = \{I_i^F \mid F \in M, 1 \leq i \leq |F|\}$.
  - Let $G$ be a $k$-acyclic group generated by $L(M)$.
  - For $F = R(a) \in M, g \in G$, create $R((a_1, gI_{1}^{F}), \ldots, (a_{|R|}, gI_{|R|}^{F}))$.
  - Ex: $M = \{R(a, a)\}$, $M' = \{R((a, e), (a, g)), R((a, g), (a, e))\}$.

- Properties:
  - $\Rightarrow$ Can be adjusted to preserve the instance as-is.
  - $\Rightarrow$ Preserves unary overlaps so preserves UIDs.
  - $\Rightarrow$ Homomorphism back to $M$ so no new queries are true.
  - $\Rightarrow$ Cycles in $M'$ of size $\leq k$ must take one edge **back-and-forth**.
  - $\Rightarrow$ This may violate FDs!
Expanding Cycles With FDs

- Our models have a **homomorphism** $h$ to $\mathcal{I}^\Theta/\equiv_k$.
- **Overlaps** are between facts with the same $h$-image.
- Adjust the product $M \times G$ with $L(\mathcal{I}^\Theta/\equiv_k)$ not $L(M)$:
  - If $F = R(a, b, c)$ and $F' = R(a, b, d)$ then $h(F) = h(F')$ and the FD $R^1 \rightarrow R^2$ cannot be violated.
  - Any cycles in $M \times G$ are mapped by the homomorphism $(x, g) \mapsto (h(x), g)$ to cycles in the “regular” product $\mathcal{I}^\Theta/\equiv_k \times G$.
  - In other words:
    - $M$ satisfies the right dependencies (including FDs),
    - $\mathcal{I}^\Theta/\equiv_k \times G$ satisfies the right queries,
    - $M \times G$ satisfies both.

- More work required to preserve the instance.
Table of Contents

1. Introduction
2. Extending GC\(^2\) Query Answering
3. Unrestricted Query Answering
4. Finite Query Answering
5. Conclusion
Summary

We have shown the decidability of:

- $\text{QA}_\bullet(\text{UKD} \cup \text{GC}^2 \cup \text{FR1}^a, \text{CQ})$
- $\text{QA}_{\text{unr}}(\text{FD} \cup \text{GC}^2 \cup \text{FR1}, \text{CQ})$
- $\text{QA}_{\text{fin}}(\text{FD} \cup \text{UID}, \text{UCQ})$
Summary

We have shown the decidability of:

- $QA_{\bullet}(\text{UKD} \cup GC^2 \cup \text{FR1}^a, \text{CQ})$
- $QA_{\text{unr}}(\text{FD} \cup GC^2 \cup \text{FR1}, \text{CQ})$
- $QA_{\text{fin}}(\text{FD} \cup \text{UID}, \text{UCQ})$

Further work:

- Derive upper and lower complexity bounds.
- For unrestricted QA:
  - Find a more homogeneous fragment than $GF^2 \cup \text{FR1}$.
  - Must limit the interaction with FD and number restrictions.
- For finite QA:
  - What about $\text{FD} \cup GC^2 \cup \text{FR1}$?
  - Can we generalize the proof beyond UIDs?
Summary

We have shown the decidability of:

- $\text{QA}_\bullet(\text{UKD} \cup \text{GC}^2 \cup \text{FR1}^a, \text{CQ})$
- $\text{QA}_{\text{unr}}(\text{FD} \cup \text{GC}^2 \cup \text{FR1}, \text{CQ})$
- $\text{QA}_{\text{fin}}(\text{FD} \cup \text{UID}, \text{UCQ})$

Further work:

- Derive upper and lower complexity bounds.
- For unrestricted QA:
  - Find a more homogeneous fragment than $\text{GF}^2 \cup \text{FR1}$.
  - Must limit the interaction with FD and number restrictions.
- For finite QA:
  - What about $\text{FD} \cup \text{GC}^2 \cup \text{FR1}$?
  - Can we generalize the proof beyond UIDs?

Thanks for your attention!


