Combining Existential Rules and Description Logics

Antoine Amarilli$^{1,2}$, Michael Benedikt$^2$

$^1$ Télécom ParisTech, Paris, France

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

July 28, 2015
Open-world query answering:

We are given:

- Relational instance $I$ (ground facts)
- Logical constraints $\Sigma$
- Boolean conjunctive query $q$
Open-world query answering:

- We are given:
  - Relational instance $I$ (ground facts)
  - Logical constraints $\Sigma$
  - Boolean conjunctive query $q$

- We ask:
  - Consider all possible completions $J \supseteq I$
  - Restrict to those that satisfy the constraints $\Sigma$
  - Is $q$ certain among them?
Open-world query answering (QA)

Open-world query answering: – query entailment or containment

- We are given:
  - Relational instance $I$ (ground facts) – A-Box
  - Logical constraints $\Sigma$ – T-Box
  - Boolean conjunctive query $q$

- We ask:
  - Consider all possible completions $J \supseteq I$
  - Restrict to those that satisfy the constraints $\Sigma$
  - Is $q$ certain among them?
Decidable constraint languages for QA

<table>
<thead>
<tr>
<th>Rich description logics (DLs)</th>
<th>Frontier-guarded existential rules</th>
</tr>
</thead>
</table>

Emp ⊑ CEO ⊔ (Mgr : Emp) ∨ pwv Acpt (p; w; v) ∨ Trip (p; f; v)

Arity-two only
Arbitrary arity
Rich (disjunction, etc.)
Poor (conjunction and implication)

Functionality asserts Funct (Mgr)

QA is decidable for either language
Decidable constraint languages for QA

<table>
<thead>
<tr>
<th>Rich description logics (DLs)</th>
<th>Frontier-guarded existential rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{Emp} \sqsubseteq \text{CEO} \sqcup (\exists \text{Mgr}^- \cdot \text{Emp})$</td>
<td>$\forall p w v \text{Acpt}(p, w, v) \rightarrow \exists f \text{Trip}(p, f, v)$</td>
</tr>
</tbody>
</table>
### Decidable constraint languages for QA

<table>
<thead>
<tr>
<th>Rich description logics (DLs)</th>
<th>Frontier-guarded existential rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emp $\sqsubseteq$ CEO $\sqcup$ (∃Mgr$^-$ .Emp)</td>
<td>∀pwv Acpt($p$, $w$, $v$) $\rightarrow$ ∃f Trip($p$, $f$, $v$)</td>
</tr>
<tr>
<td>Arity-two only</td>
<td>Arbitrary arity</td>
</tr>
</tbody>
</table>

Funct(Mgr)
### Decidable constraint languages for QA

<table>
<thead>
<tr>
<th>Rich description logics (DLs)</th>
<th>Frontier-guarded existential rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emp $\sqsubseteq$ CEO $\sqcup$ ($\exists$ Mgr$^\neg$.Emp)</td>
<td>$\forall pwv$ Acpt($p$, $w$, $v$) $\rightarrow$ $\exists f$ Trip($p$, $f$, $v$)</td>
</tr>
<tr>
<td>Arity-two only</td>
<td>Arbitary arity</td>
</tr>
<tr>
<td>Rich (disjunction, etc.)</td>
<td>Poor (conjunction and implication)</td>
</tr>
</tbody>
</table>
## Decidable constraint languages for QA

<table>
<thead>
<tr>
<th>Rich description logics (DLs)</th>
<th>Frontier-guarded existential rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emp $\sqsubseteq$ CEO $\sqcup (\exists$ Mgr$^-$.$\text{Emp}$)</td>
<td>$\forall p w v \text{ Acpt}(p, w, v) \rightarrow \exists f \text{ Trip}(p, f, v)$</td>
</tr>
<tr>
<td>Arity-two only $\mathcal{T}$</td>
<td>Arbitrary arity $\mathcal{A}$</td>
</tr>
<tr>
<td>Rich (disjunction, etc.)</td>
<td>Poor (conjunction and implication)</td>
</tr>
<tr>
<td>Functionality asserts</td>
<td>n/a</td>
</tr>
<tr>
<td>$\text{Funct}(\text{Mgr}^-)$</td>
<td></td>
</tr>
</tbody>
</table>
Decidable constraint languages for QA

<table>
<thead>
<tr>
<th>Rich description logics (DLs)</th>
<th>Frontier-guarded existential rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emp $\subseteq$ CEO $\sqcup$ ($\exists$ Mgr$^-$ . Emp)</td>
<td>$\forall pwv$ Acpt$(p, w, v) \rightarrow \exists f$ Trip$(p, f, v)$</td>
</tr>
<tr>
<td>Arity-two only</td>
<td>Arbitrary arity</td>
</tr>
<tr>
<td>Rich (disjunction, etc.)</td>
<td>Poor (conjunction and implication)</td>
</tr>
<tr>
<td>Functionality asserts</td>
<td>n/a</td>
</tr>
<tr>
<td>Funct(Mgr$^-$)</td>
<td></td>
</tr>
</tbody>
</table>

→ QA is decidable for either language
Can we have the best of both worlds?

- QA is decidable for rich DLs (i.e., expressible in GC^2, guarded two-variable first-order logic with counting)
- QA is decidable for frontier-guarded existential rules
Can we have the best of both worlds?

- QA is decidable for rich DLs (i.e., expressible in $GC^2$, guarded two-variable first-order logic with counting)
- QA is decidable for frontier-guarded existential rules
- Is QA decidable for rich DLs + some classes of rules?
Our problem

Can we have the best of both worlds?

- QA is decidable for rich DLs (i.e., expressible in $GC^2$, guarded two-variable first-order logic with counting)
- QA is decidable for frontier-guarded existential rules
  → Is QA decidable for rich DLs + some classes of rules?

We show:
Our problem

Can we have the best of both worlds?

- QA is decidable for rich DLs (i.e., expressible in GC², guarded two-variable first-order logic with counting)
- QA is decidable for frontier-guarded existential rules

→ Is QA decidable for rich DLs + some classes of rules?

We show:

- QA is undecidable for rich DLs and frontier-guarded rules
- QA with rich DLs is decidable for some new rule classes
- Functional dependencies can be added under some conditions
Restricting the language

Theorem

QA is **undecidable** for rich DLs and frontier-guarded rules
Theorem

QA is *undecidable* for rich DLs and frontier-guarded rules

Problem: inclusion dependencies $+$ Funct $=$ ID/FD implication
Restricting the language

Theorem

QA is undecidable for rich DLs and frontier-guarded rules

Problem: inclusion dependencies + Funct = ID/FD implication

→ Frontier-one rules: \( \forall x y \phi(x, y) \rightarrow \exists z \psi(x, z) \)
Restricting the language

Theorem

*QA is undecidable for rich DLs and frontier-guarded rules*

**Problem:** inclusion dependencies + Funct = ID/FD implication

→ *Frontier-one rules:* $\forall x y \phi(x, y) \rightarrow \exists z \psi(x, z)$

Theorem

*QA is undecidable for rich DLs and frontier-one rules*
Restricting the language

**Theorem**

*QA is undecidable for rich DLs and frontier-guarded rules*

**Problem:** inclusion dependencies + Funct = ID/FD implication

→ **Frontier-one rules:** $\forall x y \phi(x, y) \rightarrow \exists z \psi(x, z)$

**Theorem**

*QA is undecidable for rich DLs and frontier-one rules*

**Problem:** cycles in rules + Funct = grid
Restricting the language

**Theorem**

*QA is undecidable for rich DLs and frontier-guarded rules*

**Problem:** inclusion dependencies + Funct = ID/FD implication

→ Frontier-one rules: \( \forall x y \; \phi(x, y) \rightarrow \exists z \; \psi(x, z) \)

**Theorem**

*QA is undecidable for rich DLs and frontier-one rules*

**Problem:** cycles in rules + Funct = grid

→ Non-looping rules: prohibit cycles

- \( R(x, y) \; S(y, z) \; T(z, x) \)
- \( R(x, y, z) \; S(x, y, z) \)
Non-looping frontier-one rules: no cycles in body and in head
Non-looping frontier-one rules: no cycles in body and in head
→ We can shred them to DL rules

Theorem

QA is *decidable* for non-looping frontier-one rules + rich DLs
Non-looping frontier-one rules: no cycles in body and in head

→ We can shred them to DL rules

Theorem

QA is decidable for non-looping frontier-one rules + rich DLs

Head-non-looping frontier-one rules: no cycles in head
Non-looping frontier-one rules: no cycles in body and in head
→ We can shred them to DL rules

**Theorem**

*QA is decidable for non-looping frontier-one rules + rich DLs*

Head-non-looping frontier-one rules: no cycles in head
→ We can treeify the rules, soundness by unravelling the models

**Theorem**

*QA is decidable for head-non-looping frontier-one rules + rich DLs*
Theorem

**QA is decidable for head-non-looping frontier-one rules + rich DLs**

We want to add Functional dependencies (FDs) on arbitrary predicates: `Talk[speaker; session] determines Talk[title]`.

FDs plus single-head frontier-one rules already undecidable!

Must impose the non-conflicting condition.

Theorem

Decidable QA for:

- Rich DL constraints
- Single-head frontier-one rules
- Non-conflicting FDs
Adding functional dependencies

We want to add:

- Functional dependencies (FDs) on arbitrary predicates:
  \[ \text{Talk}[speaker, session] \text{ determines } \text{Talk}[title] \]
Adding functional dependencies

Theorem

* QA is *decidable* for head-non-looping frontier-one rules + rich DLs*

- We want to *add*:
  - Functional dependencies (FDs) on arbitrary predicates:
    
    ```
    Talk[speaker, session] determines Talk[title]
    ```

- FDs plus single-head frontier-one rules already *undecidable*
  - Must impose the *non-conflicting condition*
Adding functional dependencies

Theorem

*QA is decidable for head-non-looping frontier-one rules + rich DLs*

We want to add:

- Functional dependencies (FDs) on arbitrary predicates:
  \[ \text{Talk}[\text{speaker, session}] \text{ determines } \text{Talk}[\text{title}] \]

- FDs plus single-head frontier-one rules already undecidable
  \[ \rightarrow \text{ Must impose the non-conflicting condition} \]

Theorem

*Decidable QA for:*

- Rich DL constraints
- Single-head frontier-one rules
- Non-conflicting FDs
Summary of results

Combining Existential Rules and Description Logics

Open-world query answering under:

- Rich DL constraints
- Existential rules

For which rule classes is QA decidable with rich DLs?

- Must restrict to frontier-one rules
- Must prohibit cycles in rule heads
- QA is decidable for head-non-looping frontier-one + rich DLs

- Can add non-conflicting FDs

Thanks for your attention!

More details: see poster 76
Summary of results
Combining Existential Rules and Description Logics

- Open-world query answering (QA) under:
  - Rich DL constraints
  - Existential rules

- For which rule classes is QA decidable with rich DLs?
Summary of results
Combining Existential Rules and Description Logics

- Open-world query answering (QA) under:
  - Rich DL constraints
  - Existential rules

- For which rule classes is QA decidable with rich DLs?
  - Must restrict to frontier-one rules
  - Must prohibit cycles in rule heads
Summary of results
Combining Existential Rules and Description Logics

- Open-world query answering (QA) under:
  - Rich DL constraints
  - Existential rules

- For which rule classes is QA decidable with rich DLs?
  - Must restrict to frontier-one rules
  - Must prohibit cycles in rule heads
  - QA is decidable for head-non-looping frontier-one + rich DLs
  - Can add non-conflicting FDs

Thanks for your attention!
More details: see poster 76
Summary of results
Combining Existential Rules and Description Logics

- Open-world query answering (QA) under:
  - Rich DL constraints
  - Existential rules

- For which rule classes is QA decidable with rich DLs?
  → Must restrict to frontier-one rules
  → Must prohibit cycles in rule heads
  → QA is decidable for head-non-looping frontier-one + rich DLs
  → Can add non-conflicting FDs

Thanks for your attention!
More details: see poster 76