Larry González, Alex Ivliev, Stephan Mennicke, Markus Krötzsch
Knowledge-Based Systems Group, TU Dresden

Efficient Dependency Analysis for Existential Rules

AMW 2023, Santiago - Chile, 24th May
Motivation

\[ \text{human}(x) \rightarrow \exists p. \text{child}(x, p) \rightarrow \text{human}(p) \]

\[ \text{parent}(x, y) \rightarrow \text{child}(y, x) \]

\[ \text{father}(x, y) \rightarrow \text{parent}(x, y) \]

father(bob, alice)

human(bob) human(alice)
Motivation

human(x) $\rightarrow \exists p.\ child(x, p)$ human(p)

parent(x, y) $\rightarrow$ child(y, x)

father(x, y) $\rightarrow$ parent(x, y)

father(bob, alice)

human(bob) human(alice)
Motivation

human(x) $\rightarrow \exists p. \text{ child}(x, p)$ human(p)
parent(x, y) $\rightarrow \text{ child}(y, x)$
father(x, y) $\rightarrow \text{ parent}(x, y)$
father(bob, alice)
human(bob) human(alice)
Motivation

\[
\text{human}(x) \rightarrow \exists p. \text{ child}(x, p) \text{ human}(p)
\]

\[
\text{parent}(x, y) \rightarrow \text{child}(y, x)
\]

\[
\text{father}(x, y) \rightarrow \text{parent}(x, y)
\]

\[
\text{father}(\text{bob}, \text{alice})
\]

\[
\text{human}(\text{bob}) \quad \text{human}(\text{alice})
\]
Motivation

human(x) → ∃p. child(x, p) human(p)
parent(x, y) → child(y, x)
father(x, y) → parent(x, y)
father(bob, alice)
human(bob) human(alice)
Reliances describe interactions between rules
Reliances describe interactions between rules

A rule \textbf{positively relies} on another rule if the application of the first enables the application of the second.

• Important termination criterion
Reliances

Reliances describe interactions between rules

A rule restrains another rule if applying the second before the first introduces a redundancy.

- Checking for core stratification
Computing Reliances is Hard

Global Optimizations
- Reduce number of considered rule pairs

Local Optimizations
- Reduce effort of computing reliance for a single pair

Efficient Dependency Analysis for Existential Rules
Larry González, Alex Ivliev, Stephan Mennicke, Markus Krötzsch
Computing Reliances is Hard

Global Optimizations
Reduce number of considered rule pairs
Computing Reliances is Hard

Global Optimizations
Reduce number of considered rule pairs

Local Optimizations
Reduce effort of computing reliance for a single pair
Global Optimizations

**Goal**: Reduce the number of considered rule pairs

- parent(x, y) → child(y, x)
- penguin(x) → bird(x)

Only consider compatible rules

- parent(x, y) → child(y, x)
- mother(x, y) → parent(x, y)
- author(z, t) → written(t, z)
- writer(q, w) → author(q, w)

Hash previous results

Efficient Dependency Analysis for Existential Rules
Larry González, Alex Ivliev, Stephan Mennicke, Markus Krötzsch
Global Optimizations

**Goal:** Reduce the number of considered rule pairs

- `parent(x, y)` $\rightarrow$ `child(y, x)`
- `penguin(x)` $\rightarrow$ `bird(x)`

Only consider compatible rules

Hash previous results

Examples:
- `parent(x, y)` $\rightarrow$ `child(y, x)`
- `mother(x, y)` $\rightarrow$ `parent(x, y)`
- `author(z, t)` $\rightarrow$ `written(t, z)`
- `writer(q, w)` $\rightarrow$ `author(q, w)`
Local Optimizations

**Goal:** Check whether there is a positive reliance
Local Optimizations

**Goal:** Check whether there is a positive reliance
Local Optimizations

**Goal:** Check whether there is a positive reliance
**Local Optimizations**

**Goal:** Check whether there is a positive reliance

**Problem:** Exponentially many possibilities
Local Optimizations

Efficient Dependency Analysis for Existential Rules
Larry González, Alex Ivliev, Stephan Mennicke, Markus Krötzsch
Local Optimizations

Efficient Dependency Analysis for Existential Rules
Larry González, Alex Ivliev, Stephan Mennicke, Markus Krötzsch
Local Optimizations

Efficient Dependency Analysis for Existential Rules
Larry González, Alex Ivliev, Stephan Mennicke, Markus Krötzsch

Slide 7 of 14
Local Optimizations
Local Optimizations

Efficient Dependency Analysis for Existential Rules
Larry González, Alex Ivliev, Stephan Mennicke, Markus Krötzsch
Local Optimizations

- Efficient Dependency Analysis for Existential Rules
  Larry González, Alex Ivliev, Stephan Mennicke, Markus Krötzsch
Experiments

Oxford Ontology Repository

• 201 rule sets of various sizes
  – 63 small (<1,000 rules)
  – 90 medium (<10,000 rules)
  – 49 large (>10,000 rules)

• Individual rules contain up to 31 atoms

Questions

• What is the impact of our optimizations?
• Can we speed up some applications?
• Is core-stratification a prevalent property among rule sets?
Impact of Optimizations

- Timeout (>60s)
- Slow (<60s)
- Fast (<1s)

None | Local | Global | All
--- | --- | --- | ---

Efficient Dependency Analysis for Existential Rules
Larry González, Alex Ivliev, Stephan Mennicke, Markus Krötzsch
Application – MFA

MFA: Expensive termination criterion

![Graph showing time in milliseconds vs. VLog and Ours]
**Proportion of Core Stratification**

**Core-Stratification:** Redundancy can be avoided

- Stratified: 75
- Not stratified: 125
Breaking News

New rule engine!

- Nemo\textsuperscript{1}
- [github.com/knowsys/nemo](https://github.com/knowsys/nemo)
- all-new Rust implementation
- fast and scalable in-memory data processing
- supports Existential Rules
- more features to come

\textsuperscript{1}To appear in ICLP 2023
Join us in Dresden!

* Open positions available immediately
* Ph.D. students & postdocs
* 100% English
* KR / Databases / symbolic AI

Contact:
Markus Krötsch markus.kroetsch@tu-dresden.de
Summary

What have we learned?

• Computing reliances is feasible in practice
• Performance of applications was improved
• Core stratification appears often in practice

More scalable computation and more natural results for rule-based reasoning

Future Work

• Use reliances to speed up reasoning ... in Nemo
• Improve the notion of core stratification