Path-Based Query Computation by Automated Deduction

Peter Baumgartner
AG 2

www.mpi-sb.mpg.de/~baumgart/
Background

Deduction Systems

- First-Order
- Classical and Knowledge Representation Logics
- Refutational and Model Computation
- Implementations

Applications

- Diagnosis
- Deductive Databases
- Computational Linguistics

This Talk

- Computing paths through database schema
- Express task in logic programming framework
- Use “KRHyper” Deduction System
- Connection to AG 5: complements XXL search engine?
KRHyper

- Disjunctive logic programs
- Stratified default negation
- Model computation
- Perfect model semantics
- Serious implementation

\[
\begin{align*}
A & \leftarrow \quad (1) \\
B \lor C & \leftarrow A \quad (2) \\
A \lor D & \leftarrow C \quad (3) \\
\text{false} & \leftarrow A, B \quad (4)
\end{align*}
\]

\[
E \leftarrow C, \text{not } D
\]

\[
\{\} \not\models (1) \quad \{A\} \not\models (2) \quad \{A, B\} \not\models (4) \quad \{A, C\} \models (1)-(4)
\]

- Variant for predicate logic
- Extensions: minimal models, abduction, \textbf{default negation}
Description Logics (DL)

- Old-fashioned, problematic graphical notation

- ALC and successors:

  \[
  \text{PUBLICATION} \sqsupseteq \text{BOOK} \sqsubseteq \text{ARTICLE} \\
  \text{PUBLICATION} \sqsubseteq \exists \text{author}.\text{PERSON}
  \]

- Corresponds to decidable fragment of first-order logic
- Standard services: consistency, classify, retrieve, ...
- *The* KR formalism for the “Semantic Web”
More DL Language Features

Class definitions:
\[ \text{PUBLICATION} \sqsubseteq \exists \text{author}.\text{PERSON} \]
\[ \text{PUBLICATION} \sqsubseteq \forall \text{author}.\text{PERSON} \]

Subroles:
\[ \text{author} \sqsubseteq \text{creator} \]

Inverse/trans. roles:
\[ \text{author}^- / \text{colleague}^+ \]

QVR:
\[ \text{MANYAUTHORS} \sqsubseteq (\geq 10 \text{author}^-).\text{PUBLICATION} \]

Concrete domains:
\[ \text{Int, String, ...} \]

Nominals:
\[ \{ \text{JAR}, \text{JLC} \} \]

Cycles:
\[ \text{LISP} \eta \{ \text{NIL} \} \]
\[ (= 1 \text{CAR}).\text{Int} \]
\[ (= 1 \text{CDR}).\text{LISP} \]

Assertions:
\[ \text{publication(TACP)} \]
\[ \text{author(TACP,Knuth)} \]

DL language alone and services offered by systems do not suffice to solve task below!
Application: XML Schema Reasoning

Context: Semistructured Data, Schema Integration

Task: Determine path based database queries

Solution: XPath (?) vs. Query generated from schema
Application: XML Schema Reasoning

Generate query:

Steps:  
1. Schema $\rightarrow$ DL $\rightarrow$ If-Then Rules  
2. Start and End $\rightarrow$ Assertions/If-Then Rules  
3. Path based query $\rightarrow$ Computed model
Application: XML Schema Reasoning

(1) Schema → DL → If-Then:

RESEARCHER ⊆ ∃ publications.

(ARTICLE ∪ PUBLICATION ∪ MONOGRAPH)

ARTICLE ∩ MONOGRAPH ⊆ ⊥

PUBLICATION ⊇ BOOK ∪ ARTICLE
Application: XML Schema Reasoning

(2) Start and End → Assertions/If-Then rules

RESEARCHER(a)  

\[ \bot \leftarrow \neg \exists X \; \text{BOOK}(X) \]  

(Start)  
(End)

Solution (almost) trivial!
Conclusions

• Approach taken:
  - rather inexpressive DL
  - transformation to logic programming
  - model computation
  - purely declarative

• Did not succeed with “standard” Description Logic reasoners,
  although may be used complementary

• Further issues:
  - cycles in schema graph
  - numbers
  - first-order level reasoning useful?

• Presented first ideas only ...