Nominal Schemas in Description Logics: Complexities Clarified

Markus Krötzsch　　Sebastian Rudolph

Technische Universität Dresden

DL 2014

Nominal Schemas

- Nominals: concept expressions $\{a\}$ for individual a
- Nominal schema: concept expressions $\{x\}$ for variable x

Example

Individuals whose parents are married (with each other):

$$\exists\texttt{hasFather}.\{x\} \sqcap \exists\texttt{hasMother}.(\{y\} \sqcap \exists\texttt{married}.\{x\})$$

- Denoted by letter $\mathcal{V}$, as in $\mathcal{ELV}$, $\mathcal{ALCIV}$, $\mathcal{SROIQV}$, ...
- Several recent studies and implementations

TECHNISCHE UNIVERSITÄT DRESDEN

Open Questions

How complex is reasoning with nominal schemas?

How expressive are DLs with nominal schemas?

Semantics

- Semantics defined via grounding:
  replace variables by individuals in all possible ways

Example

Grounding with two individuals a and b:

$$\exists\texttt{hasFather}.\{a\} \sqcap \exists\texttt{hasMother}.(\{a\} \sqcap \exists\texttt{married}.\{a\})$$
$$\exists\texttt{hasFather}.\{a\} \sqcap \exists\texttt{hasMother}.(\{b\} \sqcap \exists\texttt{married}.\{a\})$$
$$\exists\texttt{hasFather}.\{b\} \sqcap \exists\texttt{hasMother}.(\{a\} \sqcap \exists\texttt{married}.\{b\})$$
$$\exists\texttt{hasFather}.\{b\} \sqcap \exists\texttt{hasMother}.(\{b\} \sqcap \exists\texttt{married}.\{b\})$$

- Always based on finite set of constants
  $\leadsto$ Option 1: finite signature
  $\leadsto$ Option 2: restrict to constants in knowledge base

TECHNISCHE UNIVERSITÄT DRESDEN

Expressivity of Nominal Schemas (1)

- Nominal schemas capture nominals:

  Replace $\{a\}$ by O_a and add axioms

$$O_a(a)$$
$$\top \sqsubseteq \exists \mathtt{somenom}.\{x\}$$
$$O_a \sqcap \exists \mathtt{somenom}.(\{x\} \sqcap O_a) \sqsubseteq \{x\}$$

Expressivity of Nominal Schemas (2)

- Nominal schemas capture Datalog rules:

  Replace $B_1 \wedge \ldots \wedge B_\ell \rightarrow H$
  by $\mathrm{enc}(B_1) \sqcap \ldots \sqcap \mathrm{enc}(B_\ell) \sqsubseteq \mathrm{enc}(H)$ where

  $$\mathrm{enc}(p(t_1, \ldots, t_n)) = \exists \mathtt{atom}.(A_p \sqcap \exists \mathtt{arg}_1.\{t_1\} \sqcap \ldots \sqcap \exists \mathtt{arg}_\mathtt{n}.\{t_n\})$$

TECHNISCHE UNIVERSITÄT DRESDEN

Expressivity of Nominal Schemas (2)

- Nominal schemas capture Datalog rules:

  Replace $B_1 \wedge \ldots \wedge B_\ell \rightarrow H$
  by $\text{enc}(B_1) \sqcap \ldots \sqcap \text{enc}(B_\ell) \sqsubseteq \text{enc}(H)$ where

  $$\text{enc}(p(t_1, \ldots, t_n)) = \exists \texttt{atom}.(A_p \sqcap \exists \texttt{arg}_1.\{t_1\} \sqcap \ldots \sqcap \exists \texttt{arg}_n.\{t_n\})$$

- Nominal schemas capture DL-safe rules:

  Encode rules as for Datalog and add axioms like

  $$\exists \texttt{atom}.(A_r \sqcap \exists \texttt{arg}_1.\{x\} \sqcap \exists \texttt{arg}_2.\{y\}) \sqsubseteq \exists \texttt{aux}.(\{x\} \sqcap \exists r.\{y\})$$

Complexity of Reasoning with Nominal Schemas

What we know so far:

- $\mathcal{SROIQV}$ is N2ExpTime-complete, like $\mathcal{SROIQ}$ [WWW 2011]
- Datalog encoding

  $\leadsto$ ExpTime lower bound for DLs above $\mathcal{ELV}$
- Grounding: exponential overall, polynomial in data

  $\leadsto$ upper bound for $\mathcal{LV}$ exponentially higher than $\mathcal{LO}$
  $\leadsto$ upper bound data complexity of $\mathcal{LV}$: comb. complexity of $\mathcal{LO}$

TBox Internalisation

TBox-to-ABox Internalisation

A TBox is replaced by a small set of "template axioms" with nominal schemas, and the original TBox is expressed with ABox assertions.

Idea:

- represent concept A by $\exists\texttt{type}.\{c_A\}$
  $\leadsto$ TBox axioms contains only roles and individuals
- use variables for all individuals
- use ABox axioms to force bindings of variables

TBox Internalisation: Results

- Axioms of the same structure require just one TBox template
- DLs $\mathcal{L}$ with finite number of normal form axioms
  $\leadsto$ fixed number of templates
  $\leadsto$ data complexity $\mathcal{LV}$ = combined complexity $\mathcal{L}$

Lower bounds

- The data complexity of $\mathcal{ELV}$ is P-hard.
- The data complexity of Horn-$\mathcal{ALCV}$ is ExpTime-hard.
- The data complexity of $\mathcal{ALCIFV}$ is (co)NExpTime-hard.

TBox Internalisation: Results

- Axioms of the same structure require just one TBox template
- DLs $\mathcal{L}$ with finite number of normal form axioms
  $\leadsto$ fixed number of templates
  $\leadsto$ data complexity $\mathcal{LV}$ = combined complexity $\mathcal{L}$

Lower bounds

- The data complexity of $\mathcal{ELV}$ is P-hard.
- The data complexity of Horn-$\mathcal{ALCV}$ is ExpTime-hard.
- The data complexity of $\mathcal{ALCIFV}$ is (co)NExpTime-hard.

Does not work for (Horn-)$\mathcal{SROIQ}$: needs unbounded set of roles

Upper bound

- The data complexity of (Horn-)$\mathcal{SROIQ}$ is in ExpTime.

TECHNISCHE UNIVERSITÄT DRESDEN

GCI Iterators

GCI Iterators

Templates of GCIs are instantiated by replacing placeholder concepts by concepts from an exponential list of "indexed" concept names.

Example:

$$A[\mathtt{i}] \sqsubseteq \exists r.A[\mathtt{i}+1] \qquad [\mathtt{i}{=}1,\ldots,s]$$

is short for

$$\begin{aligned} A_1 &\sqsubseteq \exists r.A_2 \\ A_2 &\sqsubseteq \exists r.A_3 \\ &\ldots \\ A_{s-1} &\sqsubseteq \exists r.A_s \end{aligned}$$

TECHNISCHE UNIVERSITÄT DRESDEN

GCI Iterators: Results

- Nominal schemas can encode GCI iterators succinctly
  $\leadsto$ can represent exponential knowledge bases in polynomial time
  $\leadsto$ combined complexity $\mathcal{LV}$ = exp combined complexity $\mathcal{L}$

Lower bounds

- The combined complexity of $\mathcal{ALCIFV}$ is (co)N2ExpTime-hard.
- The combined complexity of Horn-$\mathcal{ALCV}$ is 2ExpTime-hard.

GCI Iterators: Results

- Nominal schemas can encode GCI iterators succinctly
  $\leadsto$ can represent exponential knowledge bases in polynomial time
  $\leadsto$ combined complexity $\mathcal{LV}$ = exp combined complexity $\mathcal{L}$

Lower bounds

- The combined complexity of $\mathcal{ALCIFV}$ is (co)N2EXPTIME-hard.
- The combined complexity of Horn-$\mathcal{ALCV}$ is 2EXPTIME-hard.

Does not work for (Horn-)$\mathcal{SROIQ}$: needs unbounded set of roles

Upper bounds

- The combined complexity of $\mathcal{SROIQV}$ is in N2EXPTIME.
- The combined complexity of Horn-$\mathcal{SROIQV}$ is in 2EXPTIME.

TECHNISCHE UNIVERSITÄT DRESDEN

Nominal Schema Semantics Revisited

Grounding semantics depends on chosen set of constants:

Entailment of axioms with nominal schemas

$$\{a\} \sqsubseteq \{b\} \models \{a\} \sqsubseteq \{x\}$$

holds if there are only two constants *a* and *b* (but not otherwise)

Can we use a fixed, infinite set of constants?

TECHNISCHE UNIVERSITÄT DRESDEN

Nominal Schema Semantics Revisited

Grounding semantics depends on chosen set of constants:

Entailment of axioms with nominal schemas

$$\{a\} \sqsubseteq \{b\} \models \{a\} \sqsubseteq \{x\}$$

holds if there are only two constants *a* and *b* (but not otherwise)

Can we use a fixed, infinite set of constants?

- Grounding no longer works, but
- we get the same entailments (under some mild assumptions)
- and the same complexities

Conclusions

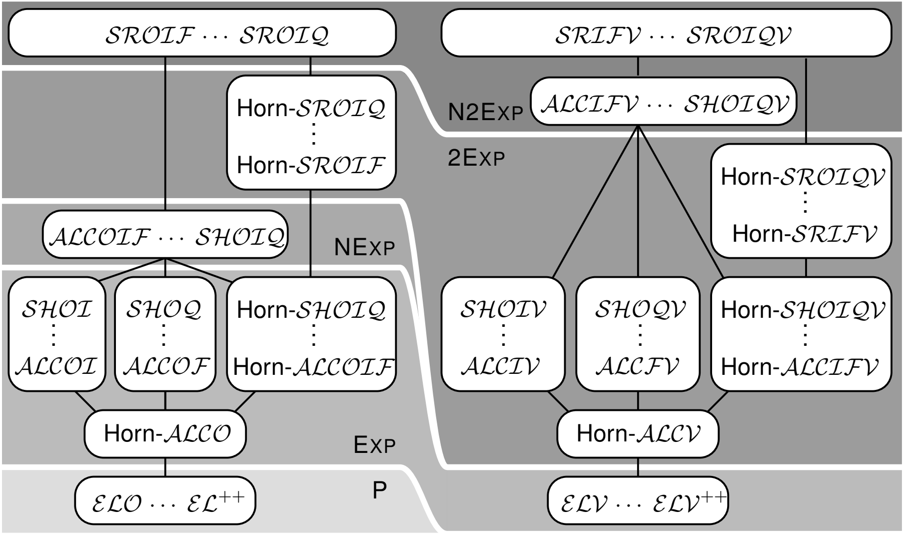
Complexity

- Nominal schemas usually increase complexity exponentially
- Exception 1: $\mathcal{SROIQ}$ and Horn-$\mathcal{SROIQ}$ (no increase)
- Exception 2: $\mathcal{ELV}$ data complexity is still polynomial

Expressiveness

- Nominal schemas subsume nominals, Datalog, and DL-safe rules
- Strictly more expressive than DL with DL-safe rules
- DLs with nominal schemas do not admit a normal form

Combined Complexities Picture



TBox Internalisation: Details

- Template for $C \sqsubseteq D$ with n concepts/individuals:

$$\exists\mathtt{gci}.(A_{C,D} \sqcap \exists\mathtt{symb}_1.\{x_1\} \sqcap \ldots \sqcap \mathtt{symb}_n.\{x_n\}) \sqcap C' \sqsubseteq D'$$

  where C' and D' are obtained from C and D by replacing:
  - each concept name σ_i by $\exists\mathtt{type}.\{x_i\}$;
  - each individual name σ_j by x_j

- Additional axiom $\top \sqsubseteq \exists\mathtt{gci}.\{x\}$

- Template instance:

$$\{A_{C,D}(d), \mathtt{symb}_1(d, c_1), \ldots, \mathtt{symb}_n(d, c_n)\}$$

  ($c_1, \ldots, c_n$ represent the original n concepts/individuals)