Complexities of Nominal Schemas
Extended Abstract

Markus Krötzsch and Sebastian Rudolph

Technische Universität Dresden, Germany

Abstract. In this extended abstract, we review our recent work *"Nominal Schemas in Description Logics: Complexities Clarified"* [6], to be presented at KR 2014.

The fruitful integration of reasoning on both schema and instance level poses a continued challenge to knowledge representation and reasoning. While description logics (DLs) excel at the former task, rule-based formalisms are often more adequate for the latter. An established and highly productive strand of research therefore continues to investigate ways of reconciling both paradigms.

A practical breakthrough in this area was the discovery of *DL-safe rules*, which ensure decidability of reasoning by restricting the applicability of rules to a finite set of elements that are denoted by an individual name [7]. As of today, DL-safe rules are the most widely used DL-rule extension, supported by several mainstream reasoners [3,8].

More recently, *nominal schemas* have been proposed as an even tighter integration of "DL-safe" instance reasoning with DL schema reasoning [5]. A *nominal* is a DL concept expression $\{a\}$ that represents a singleton set containing only (the individual denoted by) a. Nominal schemas replace a by a variable x that ranges over all individual names, so that it might represent arbitrary nominals $\{a\}$, where all occurrences of $\{x\}$ in one axiom represent the same nominal. For example,

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

represents the set of all individuals whose father (x) and mother (y) are married to each other, where the parents must be represented by individual names. No standard DL can express this in such a concise way. The interplay with other DL features also makes nominal schemas more expressive than the combination of DLs and DL-safe rules.

Nominal schemas have thus caused significant research interest, and several reasoning algorithms that exploit this succinct representation have been proposed [4,10,9,1]. Most recently, it was demonstrated that such algorithms can even outperform other systems for reasoning with DL-safe rules [9].

Surprisingly and in sharp contrast to these successes, many basic questions about the expressivity and complexity of nominal schemas have remained unanswered until recently. A naive reasoning approach is based on *grounding*, i.e., replacing nominal schemas by nominals in all possible ways, which leads to complexity upper bounds one exponential above the underlying DL. The only tight complexity result so far is that the N2ExpTime combined complexity of reasoning in the DL $\mathcal{SROIQ}$ is not affected by nominal schemas—a result that reveals almost nothing about the computational or expressive impact of nominal schemas in general [5]. Beyond this singular result, it is only known that nominal schemas can simulate Datalog rules of any arity using $\exists$, $\sqcap$, and the universal role U [2].

In our KR 2014 paper *"Nominal Schemas in Description Logics: Complexities Clarified"* [6], we give a comprehensive account of the reasoning complexities of a wide range of DLs, considering both combined complexities (w.r.t. the size of the given knowledge base) and data complexities (w.r.t. the size of the ABox only). Figure 1 summarizes our results for combined complexities for DLs with nominal schemas (right; marked by the letter $\mathcal{V}$) in comparison with known complexities of DLs with nominals (left). It turns out that $\mathcal{SROIQ}$ is an exception, while most other DLs experience exponential complexity increases due to nominal schemas.

The effects on the data complexity are even more striking. The data complexity of standard DLs is either in P (for $\mathcal{EL}$ and Horn-DLs, which restrict the use of $\sqcup$ and $\neg$) or in NP. In contrast, the data complexities for all nominal-schema DLs in Fig. 1 are only one exponential below their combined complexity, i.e., ExpTime or NExpTime for most cases.

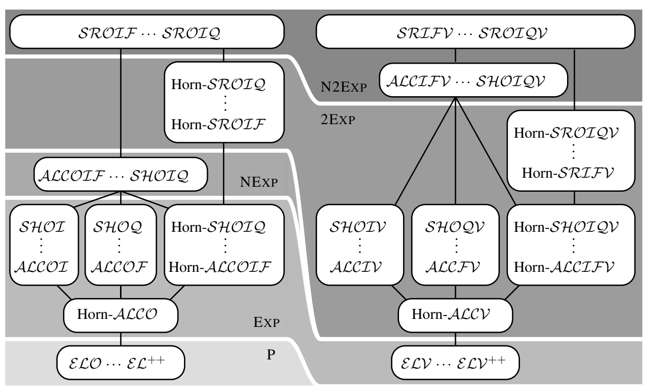

Fig. 1. Combined complexities for DLs with nominals compared to DLs with nominal schemas

To obtain these results, we identify general modeling techniques that use nominal schemas to express complex schema information very succinctly. Two fundamental techniques provide the basis for most of our hardness proofs:

TBox-to-ABox Internalization A TBox is replaced by a small set of "template axioms" with nominal schemas, and the original TBox is expressed with ABox assertions. The underlying transformation is captured by the following definition.

Definition 1. *Consider a DL $\mathcal{L}$ with $\mathcal{EL} \subseteq \mathcal{L} \subseteq \mathcal{SROIQ}$ and an $\mathcal{L}$ TBox axiom $\alpha = C \sqsubseteq D$. The* template for α*, denoted* $\mathsf{tmpl}(\alpha)$*, is defined as follows. Let $\sigma_1, \ldots, \sigma_n$ be a list of all individual names and concept names in α. Let A_α be a fresh concept name, and let* $\mathtt{gci}$*,* $\mathtt{type}$*, and* $\mathtt{symb}_i$ *($1 \leq i \leq n$) be fresh role names. Then* $\mathsf{tmpl}(\alpha)$ *is the $\mathcal{LV}$ axiom*

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

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

The template instance for α*, denoted* $\mathsf{tins}(\alpha)$*, is the following set of ABox assertions:*

$$\{A_\alpha(c_\alpha), \mathtt{symb}_1(c_\alpha, c_{\sigma_1}), \ldots, \mathtt{symb}_n(c_\alpha, c_{\sigma_n})\}$$

where c_α and $c_{\sigma_1}, \ldots, c_{\sigma_n}$ are fresh individual names.

It turns out that this transformation preserves entailments of ground atoms under some additional assumption about the knowledge base (referred to as *unboundedness*) that is not to hard to impose. TBox-to-ABox internalization explains why the data complexity of most DLs with nominal schemas agrees with the combined complexity of their underlying standard DL. $\mathcal{SROIQV}$ is a noteworthy exception where the internalization is not possible.

GCI Iterators Templates of TBox axioms (general concept inclusions, short GCIs) are instantiated by replacing placeholder concepts by concepts from an exponentially long list of "indexed" concept names.

Definition 2. *Consider a DL signature $\langle \mathsf{N}_I, \mathsf{N}_C, \mathsf{N}_R \rangle$. A* GCI iterator *over this signature is an expression $C \sqsubseteq D$ $[\mathtt{i} = \mathtt{1}, \ldots, n]$ where $n \geq 1$ and $C \sqsubseteq D$ is a general concept inclusion over $\langle \mathsf{N}_I, \mathsf{N}_C \cup \{A[1], \ldots, A[n+1], A[\mathtt{i}], A[\mathtt{i}+\mathtt{1}] \mid A \in \mathsf{N}_C\}, \mathsf{N}_R \rangle$. Note that $\mathtt{i}$ is a literal part of the syntax, not a placeholder for a specific number. The additional concept names $A[\ldots]$ are assumed to be distinct from all concepts in N_C. The* expansion *of a GCI iterator is the set of GCIs over $\langle \mathsf{N}_I, \mathsf{N}_C \cup \{A[1], \ldots, A[n+1] \mid A \in \mathsf{N}_C\}, \mathsf{N}_R \rangle$ obtained by replacing, for each $i \in \{1, \ldots, n\}$, all concepts $A[\mathtt{i}]$ by $A[i]$, and all concepts $A[\mathtt{i}+\mathtt{1}]$ by $A[i+1]$.*

For a DL $\mathcal{L}$, we let $\mathcal{L}_{\mathrm{GI}}$ be $\mathcal{L}$ extended by GCI iterators as axioms. The semantics of an $\mathcal{L}_{\mathrm{GI}}$ knowledge base KB *is given by the translation into $\mathcal{L}$ through replacing all GCI iterators by their expansions, denoted* $\mathsf{expand}(\mathrm{KB})$.

GCI iterators are a kind of generalized TBox axiom that can be used to encode exponentially large TBoxes polynomially using nominal schemas. This technique can be applied to TBoxes from known hardness proofs to boost complexities by one exponential. Both techniques provide concrete illustrations for the expressive power of nominal schemas and outline ways to obtain results for DLs that we did not consider.

After establishing these results, we revisit the formal semantics of nominal schemas. Normally, nominal schemas are considered to represent a finite set of nominals, based on individuals that either occur in the knowledge base or are part of some finite signature. This can lead to unintuitive effects, since entailments may become invalid when adding more individuals. We thus study the semantics obtained when using an infinite set of individual names instead. This makes it impossible to replace nominal schemas by nominals in all possible ways to decide entailment. Surprisingly, reasoning is still decidable with the same complexity results. Indeed, the consequences of both approaches turn out to agree under some mild assumptions.

Acknowledgement This work was supported by the DFG in project DIAMOND (Emmy Noether grant KR 4381/1-1).

References

1. Carral Martínez, D., Wang, C., Hitzler, P.: Towards an efficient algorithm to reason over description logics extended with nominal schemas. In: Faber, W., Lembo, D. (eds.) Proc. 7th Int. Conf. on Web Reasoning and Rule Systems (RR 2013). LNCS, vol. 7994, pp. 65–79. Springer (2013)
2. Knorr, M., Hitzler, P., Maier, F.: Reconciling OWL and non-monotonic rules for the Semantic Web. In: Raedt, L.D., Bessière, C., Dubois, D., Doherty, P., Frasconi, P., Heintz, F., Lucas, P.J.F. (eds.) Proc. 20th European Conf. on Artificial Intelligence (ECAI'12). Frontiers in Artificial Intelligence and Applications, vol. 242, pp. 474–479. IOS Press (2012)
3. Kolovski, V., Parsia, B., Sirin, E.: Extending the $\mathcal{SHOIQ}(D)$ tableaux with DL-safe rules: First results. In: Parsia, B., Sattler, U., Toman, D. (eds.) Proc. 19th Int. Workshop on Description Logics (DL'06). CEUR WS Proceedings, vol. 198. CEUR-WS.org (2006)
4. Krisnadhi, A., Hitzler, P.: A tableau algorithm for description logics with nominal schema. In: Krötzsch, M., Straccia, U. (eds.) Proc. 6th Int. Conf. on Web Reasoning and Rule Systems (RR 2012). LNCS, vol. 7497, pp. 234–237. Springer (2012)
5. Krötzsch, M., Maier, F., Krisnadhi, A.A., Hitzler, P.: A better uncle for OWL: Nominal schemas for integrating rules and ontologies. In: Proc. 20th Int. Conf. on World Wide Web (WWW'11). pp. 645–654. ACM (2011)
6. Krötzsch, M., Rudolph, S.: Nominal schemas in description logics: Complexities clarified. In: Proc. 14th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'14). AAAI Press (2014)
7. Motik, B., Sattler, U., Studer, R.: Query answering for OWL DL with rules. J. of Web Semantics 3(1), 41–60 (2005)
8. Motik, B., Shearer, R., Horrocks, I.: Hypertableau reasoning for description logics. J. of Artificial Intelligence Research 36, 165–228 (2009)
9. Steigmiller, A., Glimm, B., Liebig, T.: Nominal schema absorption. In: Rossi, F. (ed.) Proc. 23rd Int. Joint Conf. on Artificial Intelligence (IJCAI'13). pp. 1104–1110. AAAI Press/IJCAI (2013)
10. Wang, C., Hitzler, P.: A resolution procedure for description logics with nominal schemas. In: Takeda, H., Qu, Y., Mizoguchi, R., Kitamura, Y. (eds.) Proc. 2nd Joint Int. Conf. on Semantic Technology (JIST'12). LNCS, vol. 7774, pp. 1–16. Springer (2013)