Recent Advances in Integrating OWL and Rules (Technical Communication)

Matthias Knorr[2], David Carral Martínez[1], Pascal Hitzler[1], Adila A. Krisnadhi[1], Frederick Maier[3], and Cong Wang[1]

[1] Kno.e.sis Center, Wright State University, U.S.A.
[2] CENTRIA, Universidade Nova de Lisboa, Portugal
[3] Aston Business School, Aston University, UK

As part of the quest for a *unifying logic* for the Semantic Web Technology Stack,[4] a central issue is finding suitable ways of integrating description logics based on the Web Ontology Language (OWL) with rule-based approaches based on logic programming. Such integration is difficult since naive approaches typically result in the violation of one or more desirable design principles. For example, while both OWL 2 DL and RIF Core (a dialect of the Rule Interchange Format RIF) are decidable, their naive union is not, unless carefully chosen syntactic restrictions are applied.

We report on recent advances and ongoing work by the authors in integrating OWL and rules. We take an OWL-centric perspective, which means that we take OWL 2 DL as a starting point and pursue the question of how features of rule-based formalisms can be added without jeopardizing decidability. We also report on incorporating the closed world assumption and on reasoning algorithms. This paper essentially serves as an entry point to the original papers, to which we will refer throughout, where detailed expositions of the results can be found.

1 Rule-Extensions of OWL

In [4], Grosof et al. describe a fragment of the description logic $\mathcal{SHOIN}$ (a.k.a. OWL 1 DL) which, if syntactically transferred to first-order predicate logic (FOL) in a straightforward way, results in a set of function-free Horn clauses, i.e. a Datalog program under FOL semantics. This naive approach has been subsequently lifted to OWL 2 DL and given rise to the OWL 2 RL fragment [10]. This work does not, however, address the problem of identifying the rules of Datalog (under FOL semantics) expressible in OWL and its variants, and indeed recent results, including the work on description logic rules by Krötzsch et al. [7], show that OWL 2 RL can be improved significantly in this respect.

To formulate the recent findings, we first note that a directed graph G_r can be constructed from any given binary Datalog rule r, i.e. a rule containing only unary and binary predicates. The nodes of G_r are the variables occurring in the rule body of r, and there is exactly one directed edge between two variables x and y if there is at least one binary atom of the form $P(x, y)$ appearing in the

[4] http://www.w3.org/2007/03/layerCake.png

body of r. The following results then hold,[5] where z is the variable in the first argument of the head atom.

- If G_r is a tree with root z, then r can be expressed in $\mathcal{SROEL}$ (OWL 2 EL) [7].
- If G_r, with any edges inverted, is a tree with root z, then r can be expressed in $\mathcal{SROIEL}$ [7].
- If G_r, with edges considered undirected, does not contain four nodes which are path-connected by mutually disjoint paths in such a way that they constitute a 4-clique, then r can be expressed in $\mathcal{SROIEL}(\sqcap)$, i.e., in $\mathcal{SROIEL}$ extended by role conjunction [1].

The results above are based on the idea of retaining decidability by syntactically restricting the rules which are allowed to be used together with a DL knowledge base. A complementary line of work is based on the idea of weakening the semantics of rules in a suitable way. This was first voiced in the notion of DL-safe rules [13], which are rules in which the variables can bind only to known individuals, i.e. to constants present in the knowledge base, resulting in so-called *DL-safe SWRL*. This approach was then generalized in [9] in such a way that only *some* variables in rules—called *DL-safe variables*—were restricted this way. In [8], Krötzsch et al. ported this concept to description logics, resulting in a new syntactic construct called *nominal schemas*.

Nominal schemas can be understood as variable nominals. Syntactically, this new construct $\{x\}$ resembles a nominal, save that x is a variable rather than an individual, and it can only bind to individuals appearing in the knowledge base such that each occurrence of the nominal schema within one axiom is bound to the same individual. Semantically, this is realized by extending the interpretation with a first-order variable assignment binding variables to domain elements named by individuals in the knowledge base [8].

A DL extended with nominal schemas not only completely covers DL-safe SWRL, it also makes it possible to completely express any Datalog program under the Herbrand semantics—without any restriction on arities of predicates or on forms of rules [5, 8]. Furthermore, $\mathcal{SROIQ}$ extended with nominal schemas, called $\mathcal{SROIQV}$, is of the same computational complexity as $\mathcal{SROIQ}$ [8].

So far, only monotonic rules are considered, despite the fact that the closed world assumption is often requested in order to be able to model defaults, exceptions, and integrity constraints. Following the spirit of description logics of minimal knowledge and negation as failure (MKNF) [3], two modal operators **K** and **A** are added to $\mathcal{SROIQV}$, yielding a more expressive yet still uniform formalism [5]. The two operators allow the inspection of the knowledge base, i.e. **K** represents minimal knowledge, while **A** is interpreted as autoepistemic assumption and corresponds to $\neg$**not**, where **not** is identical with default negation in non-monotonic rules. As is common in MKNF semantics, a set of interpretations is used instead of one interpretation, and the non-monotonic semantics

[5] Some of these statements can be improved, as detailed in the indicated papers.

is defined based on a preference relation among such sets, minimizing derivable knowledge.

This language trivially covers $\mathcal{SROIQV}$ (hence $\mathcal{SROIQ}$, the tractable OWL 2 profiles, and arbitrary Datalog rules as pointed out above), and $\mathcal{ALCK_{NF}}$ [3]. Thus, default reasoning, epistemic queries, closure of roles and concepts, and integrity constraints are available in the language. It also covers Hybrid MKNF [12], a tight integration of DLs and non-monotonic rules based on MKNF logics. Indeed, it is the first approach that covers the two distinct MKNF-based formalisms, [3] and [12]. Moreover, a decidable fragment of the full language is identified in [5], which contains most of the covered languages.

2 Algorithms for Reasoning with Nominal Schemas

There is a naive way of algorithmizing reasoning with nominal schemas, which we call *full grounding*: Replace each axiom with all *grounded* axioms, where nominal schemas are replaced by nominals, in all possible combinations which respect variable bindings. This yields a semantically equivalent knowledge base without nominal schemas, and a traditional reasoning algorithm can then be used. While this approach permits reasoning with nominal schemas, it is problematic in the sense that it is combinatorially explosive in cases involving axioms having many nominal schemas [2]. We have therefore started to investigate alternative approaches which ground nominal schemas in a dynamic fashion, thus reducing the overhead of full grounding. We briefly report on the preliminary findings of this ongoing work.

An alternative approach for dealing with nominal schemas is to extend standard tableau algorithms with grounding rules in such a way that grounding can be delayed until *required* [6]. In the following situations, grounding a nominal schema $\{x\}$ occurring within a concept C in the label of a tableau node is required before any standard tableau rules can be applied: (i) when $C = D \sqcup E$ and $\{x\}$ occurs in both D and E; (ii) when $C = \exists R.D$ and $\{x\}$ occurs in the top level of D, e.g., $\exists R.(\{x\} \sqcap E)$ needs grounding whereas $\exists R.\exists S.(A \sqcap \{x\})$ does not. On the other hand, when C is a conjunction or a universal restriction, grounding can be safely delayed. Other details such as when grounding should be performed, which variables should be grounded first, or to which individual names they are grounded are left for an actual implementation to specify. Work is still in progress to extend this idea to other expressive constructors, such as qualified number restrictions.

In addition to intelligent grounding as in the tableau approach just described, can we avoid grounding from the beginning? In pursuit of this idea, we have also started to investigate the resolution calculus for algorithmization, where grounding is handled on the fly via unification. Previous work on general resolution for DLs [11] was unable to deal with role chains, as it introduces further complications with termination. We solved this problem by using ordered chaining rules such that the inferred clauses will not be longer than the premises. Nominal schemas add yet another complication to the termination issue as normal forms of globally limited size are no longer readily available. We successfully addressed this by using a lifting lemma to show that resolution on nominal schema axioms takes fewer resolution steps than performing resolution on fully grounded knowledge bases. Details can be found in [14].

References

1. Carral Martínez, D., Hitzler, P.: Extending description logic rules. In: Simperl, E., Cimiano, P., Polleres, A., Corcho, Ó., Presutti, V. (eds.) The Semantic Web: Research and Applications - 9th Extended Semantic Web Conference, ESWC 2012, Heraklion, Crete, Greece, May 27-31, 2012. Proceedings. pp. 345–359. Lecture Notes in Computer Science, Springer (2012)
2. Carral Martínez, D., Krisnadhi, A., Maier, F., Sengupta, K., Hitzler, P.: Reconciling OWL and rules. Tech. rep., Kno.e.sis Center, Wright State University, Dayton, Ohio (2011), available from http://www.pascal-hitzler.de/
3. Donini, F.M., Nardi, D., Rosati, R.: Description logics of minimal knowledge and negation as failure. ACM Transactions on Computational Logic 3(2), 177–225 (2002)
4. Grosof, B., Horrocks, I., Volz, R., Decker, S.: Description logic programs: combining logic programs with description logic. In: Proceedings of the 12th International Conference on World Wide Web (WWW'03). pp. 48–57. ACM (2003)
5. Knorr, M., Hitzler, P., Maier, F.: Reconciling OWL and non-monotonic rules for the Semantic Web. In: Proceeding of the 20th European Conference on Artificial Intelligence, Montpellier, France, August 27-31, 2012 (2012), to appear
6. Krisnadhi, A., Hitzler, P.: A tableau algorithm for description logics with nominal schemas. In: Proceedings RR2012 (2012), to appear
7. Krötzsch, M., Rudolph, S., Hitzler, P.: Description logic rules. In: Ghallab, M., Spyropoulos, C.D., Fakotakis, N., Avouris, N.M. (eds.) Proceeding of the 18th European Conference on Artificial Intelligence, Patras, Greece, July 21-25, 2008. pp. 80–84. IOS Press, Amsterdam (2008)
8. Krötzsch, M., Maier, F., Krisnadhi, A.A., Hitzler, P.: A better uncle for OWL: Nominal schemas for integrating rules and ontologies. In: Proceedings of the 20th International Conference on World Wide Web (WWW'11). pp. 645–654. ACM (2011)
9. Krötzsch, M., Rudolph, S., Hitzler, P.: ELP: Tractable Rules for OWL 2. In: Sheth, A.P., et al. (eds.) Proceedings of the 7th International Semantic Web Conference, ISWC 2008, Karlsruhe, Germany, October 26-30, 2008. Lecture Notes in Computer Science, vol. 5318, pp. 649–664. Springer (2008)
10. Motik, B., Cuenca Grau, B., Horrocks, I., Wu, Z., Fokoue, A., Lutz, C. (eds.): OWL 2 Web Ontology Language: Profiles. W3C Recommendation (27 October 2009), available at http://www.w3.org/TR/owl2-profiles/
11. Motik, B.: Reasoning in Description Logics using Resolution and Deductive Databases. Ph.D. thesis, Univesität Karlsruhe (TH), Karlsruhe, Germany (2006)
12. Motik, B., Rosati, R.: Reconciling Description Logics and Rules. Journal of the ACM 57(5), 93–154 (2010)
13. Motik, B., Sattler, U., Studer, R.: Query answering for OWL-DL with rules. Journal of Web Semantics 3(1), 41–60 (2005)
14. Wang, C., Hitzler, P.: A tractable resolution procedure for $\mathcal{SROELV}_n(\sqcap_s, \times)$. Tech. rep., Kno.e.sis Center, Wright State University, Dayton, Ohio (2012), available from http://www.pascal-hitzler.de/