Faster OWL Using Split Programs

Pascal Hitzler* and **Denny Vrandečić†**

AIFB, Universität Karlsruhe, 76128 Karlsruhe, Germany, {hitzler, denny}@aifb.uni-karlsruhe.de

1 The OWL scalability problem

Knowledge representation and reasoning on the Semantic Web is done by means of ontologies. While the quest for suitable ontology languages is still ongoing, OWL [5] has been established as a core standard. It comes in three flavours, as OWL Full, OWL DL and OWL Lite, where OWL Full contains OWL DL, which in turn contains OWL Lite. The latter two coincide semantically with certain description logics and can thus be considered fragments of first-order predicate logic.

OWL ontologies can be understood to consist of two parts, one intensional, the other extensional. In description logics terminology, the intensional part consists of a TBox and an RBox, and contains knowledge about concepts (called *classes*) and complex relations between them (called *roles*). The extensional part consists of an ABox, and contains knowledge about entities and how they relate to the classes and roles from the intensional part. The Semantic Web envisions a distributed knowledge source, built from OWL ontologies and intertwining the knowledge like the Web interconnects websites today.

With an estimated 25 million active websites today and correspondingly more webpages, it is apparent that reasoning on the Semantic Web will have to deal with very large ABoxes. Complexity of ABox reasoning — also called *data complexity* — measures complexity in terms of ABox size only, while considering the intensional part of the ontology to be of constant size. For the different OWL variants, data complexity is at least NP-hard, which indicates that it will not scale well in general. Therefore, methods are sought to cope with large ABoxes in an approximate manner. The idea is to use quick heuristic reasoning when time constraints are more important than the correctness of the answers. A typical use case is online question answering, where it is more important to give rough answers quickly than to have precise responses at the cost of long delays.

2 From OWL to datalog

The approach which we propose is based on the fact that data complexity is polynomial for non-disjunctive datalog. We utilise recent research results about the transformation of OWL DL ontologies into disjunctive datalog, and perform heuristic approximate reasoning by transforming the disjunctive database into a non-disjunctive one.

The transformation is based on the fact that OWL DL is a subset of first-order logic. OWL axioms can thus be translated directly into logical formulas and transformed into clausal form using any of the standard algorithms. The resulting clauses can be represented as disjunctive datalog rules which do not contain negation.

Note, however, that due to possible skolemization steps in the clausal form translation, the resulting datalog rules may contain function symbols. In general, datalog with function symbols is undecidable, but since we obtain the datalog program by a translation from OWL DL, which is decidable, inferencing over the resulting program must be decidable. Standard datalog engines, however, do in general not terminate in the presence of function symbols. To cope whith this problem, a sophisticated method has been presented in [2; 3] which allows to get rid of the function symbols without loosing ABox consequences. As a result, we obtain a function- and negation-free disjunctive datalog program, which can be dealt with using standard techniques.

There is one other catch: The approach presented in [2; 3] does not allow to deal with nominals, i.e. it supports only $\mathcal{SHIQ}(\mathbf{D})$ instead of $\mathcal{SHOIN}(\mathbf{D})$ (the latter is the description logic coinciding with OWL DL). We remark that to date — and to the best of our knowledge — no reasoning algorithms for $\mathcal{SHOIN}(\mathbf{D})$ have been implemented. We will return to a possible treatment of nominals in our approach later.

A full presentation of the translation with correctness proofs is technically involved and lengthy, and space restrictions forbid to go into further detail; we refer the interested reader to [2; 3].

3 Approximate SLD-Resolution

Having obtained datalog rules of the form

$$H_1 \vee \cdots \vee H_m \leftarrow A_1, \ldots, A_k,$$

ABox reasoning is still NP-hard. For our approximate reasoning approach, we utilize the fact that when all rules are non-disjunctive, i.e. when $m = 1$, then standard resolution methods can be used which render the reasoning to be polynomial with regards to the number of facts. Hence, we use a modified notion of *split programs* [6]. Given the above rule, the *derived split rules* are defined as:

$$H_1 \leftarrow A_1, \ldots, A_k \quad \ldots \quad H_m \leftarrow A_1, \ldots, A_k.$$

For a given disjunctive program P, its *split program P'* is defined as the collection of all split rules derived from rules in P. Polynomial ABox reasoning can now be performed using the split program and classic resolution techniques, e.g. SLD-resolution as used in standard Prolog systems [4]. The combined reasoning method, which we call *approximate SLD-resolution*, is obviously complete but unsound, and hence it is necessary to pursue the question of exactly what notion of entailment underlies the approximate reasoning technique we propose. Space restrictions forbid us to go into detail, so it shall suffice to say that approximate SLD-resolution boils down to brave reasoning with well-supported models, where the latter notion is a straightforward adaptation of the notion of well-supported model from [1] to the disjunctive case.

In order to be able to deal with all of OWL DL, we need to add a preprocessing step to get rid of nominals. We can do this by *Language Weakening* as follows: For every occurrence of $\{o_1, \ldots, o_n\}$, where $n \in \mathbb{N}$ and the o_i are abstract or concrete individuals, replace $\{o_1, \ldots, o_n\}$ by some new concept name D, and add ABox assertions $D(o_1), \ldots, D(o_n)$ to the knowledge base. Note that the transformation just given does in general not yield a logically equivalent knowledge base, because some information is lost in the process.

Putting all the pieces together, the following steps describe our approximate ABox reasoning for OWL DL.

1. Apply Language Weakening as just mentioned in order to obtain a $\mathcal{SHIQ}(\mathbf{D})$ knowledge base.

2. Apply transformations as in Section 2 in order to obtain a negation-free disjunctive datalog program.

3. Use approximate SLD-resolution for query-answering.

The first two steps can be considered to be preprocessing steps for setting up the intensional part of the database. ABox reasoning is then done in the last step. From our discussions, we can conclude the following properties of approximate ABox reasoning for $\mathcal{SHIQ}(\mathbf{D})$.

- It is complete with respect to first-order predicate logic semantics.

- It is sound and complete with respect to brave reasoning with well-supported models.

- Data complexity of our approach is polynomial.

4 SCREECH OWL

We have implemented the proposed approach as SCREECH[1], based on KAON2[2]. It utilizes KAON2's sophisticated translation algorithms from OWL DL into datalog, and returns the

---

[1] http://logic.aifb.uni-karlsruhe.de/screech

[2] http://kaon2.semanticweb.org

corresponding split program which can be fed into any standard Prolog interpreter for ABox reasoning. As an additional feature, the transformed program allows to keep track of the number of disjunctions which are being ignored during the query answering process, thus giving an estimate about the accuracy of the answers.

5 Conclusions

In a nutshell, our proposed procedure approximates reasoning by disregarding non-Horn features of OWL DL ontologies. We argue that this is a reasonable approach to approximate reasoning with OWL DL in particular because many — if not most — of the currently existing ontologies use only a small number of language constructs outside of the Horn fragment of OWL DL. A survey in [7] substantiates this claim.

Our approach provides ABox reasoning with polynomial time complexity. It is complete, but it is also unsound with respect to first-order logic. However, the inference underlying our approach can be characterized using standard methods from the area of non-monotonic reasoning.

The checking whether a conjunctive query is a predicate logic consequence of a (negation-free) disjunctive logic program P amounts to checking whether the query is valid in *all* minimal models of P, i.e. corresponds to *cautious* reasoning with minimal models. Along this insight, we foresee the possibility to develop an algorithm which would first find a brave answer of a query, and then substantiate this answer by subsequent calculations. This and other refinements of our approach are in development.

References

[1] F. Fages. Consistency of Clark's completion and existence of stable models. *Journal of Methods of Logic in Computer Science*, 1:51–60, 1994.

[2] U. Hustadt, B. Motik, and U. Sattler. Reasoning in description logics with a concrete domain in the framework of resolution. In Ramon López de Mántaras and Lorenza Saitta, editors, *Proceedings of the 16th Eureopean Conference on Artificial Intelligence, ECAI'2004, Valencia, Spain, August 2004*, pages 353–357. IOS Press, 2004.

[3] U. Hustadt, B. Motik, and U. Sattler. Reducing $\mathcal{SHIQ}^-$ Description Logic to Disjunctive Datalog Programs. In *Proc. of the 9th Conference on Knowledge Representation and Reasoning (KR2004)*. AAAI Press, June 2004.

[4] J. W. Lloyd. *Foundations of Logic Programming*. Springer, Berlin, 1988.

[5] P. F. Patel-Schneider, P. Hayes, I. Horrocks, and F. van Harmelen. OWL Web Ontology Language; Semantics and Abstract Syntax, W3C Candidate Recommendation. http://www.w3.org/TR/owl-semantics/, November 2002.

[6] C. Sakama and K. Inoue. An alternative approach to the semantics of disjunctive logic programs and deductive databases. *Journal of Automated Reasoning*, 13:145–172, 1994.

[7] R. Volz. *Web Ontology Reasoning with Logic Databases*. PhD thesis, AIFB, University of Karlsruhe, 2004.