Chapter 5
Scalable Reasoning with Tractable Fuzzy Ontology Languages

Giorgos Stoilos
National and Technical University of Athens, Greece

Jeff Z. Pan
University of Aberdeen, UK

Giorgos Stamou
National and Technical University of Athens, Greece

ABSTRACT
The last couple of years it is widely acknowledged that uncertainty and fuzzy extensions to ontology languages, like description logics (DLs) and OWL, could play a significant role in the improvement of many Semantic Web (SW) applications like matching, merging and ranking. Unfortunately, existing fuzzy reasoners focus on very expressive fuzzy ontology languages, like OWL, and are thus not able to handle the scale of data that the Web provides. For those reasons much research effort has been focused on providing fuzzy extensions and algorithms for tractable ontology languages. In this chapter, the authors present some recent results about reasoning and fuzzy query answering over tractable/polynomial fuzzy ontology languages namely Fuzzy DL-Lite and Fuzzy EL+. Fuzzy DL-Lite provides scalable algorithms for very expressive (extended) conjunctive queries, while Fuzzy EL+ provides polynomial algorithms for knowledge classification. For the Fuzzy DL-Lite case the authors will also report on an implementation in the ONTOSEARCH2 system and preliminary, but encouraging, benchmarking results.

INTRODUCTION
Nowadays, many applications and domains use some form of knowledge representation language and exploit their inference mechanisms in order to improve their capabilities and simulate intelligent human behavior. Many such examples exist, like knowledge-based multimedia analysis (Neumann & Möller,
Scalable Reasoning with Tractable Fuzzy Ontology Languages

2006; Simou et al., 2008a), bioinformatics (Dameron et al., 2004) and databases (Calvanese et al., 1998) and more. Nevertheless the most prominent example is undoubtedly the World Wide Web aiming for intelligently managing the vast amount of information that lays on the Web. Among several proposals for structuring knowledge in such applications, Description Logic based ontologies seem to be an approach that has gained considerable attention. Description Logics (DLs) (Baader et al., 2002) is a modern knowledge representation formalism that is a fragment of First-Order Logic, enjoying well-defined model-theoretic semantics, decidability and practically efficient reasoning systems. Most importantly expressive DLs form the logical underpinnings of the W3C standard language for representing ontologies in the Semantic Web, namely OWL (Bechhofer et al., 2004; Patel-Schneider et al., 2004). Although several successful OWL DL reasoning systems have been developed, like FaCT++ and Pellet, even very basic and inexpressive DLs come with come with (at least) \text{ExpTime} computational complexity. Thus, their ability to scale in large application like the once found on the Web is still an open issue. For those reasons the last years great research effort has been focusing in identifying fragments/clusters of the owl DL language for which it is known that reasoning is scalable and efficient. This research has led to the development of several languages, but the two most interesting and predominant ones are EL+ (Baader et al.) and DL-Lite (Calvanese et al., 2005; Calavanese et al., 2007). The interesting thing is that these languages will most likely form the logical underpinnings of the OWL 2 EL and OWL 2 QL recommendations which consist of profiles/fragments of the upcoming extension of OWL, OWL 2.

Although DLs are relatively quite expressive they feature limitations mainly with what can be said about imperfect (uncertain, vague/fuzzy or imprecise) knowledge. Such types of knowledge appears in many domains but also in several Semantic Web tasks, like in the representation of trust, in knowledge fusion, assessing the similarity between resources and many more. For those reasons fuzzy ontologies are envisioned to be useful in the Web (Stoilos et al., 2006) and fuzzy Description Logics (f-DLs) (Hölldobler et al., 2005; Straccia, 2001; Tresp & Molito, 1998) have been proposed as formalisms capable of capturing and reasoning with such knowledge. Research in f-DLs was mainly focused on providing reasoning support for very expressive fuzzy DLs, like reasoning with the f-DL \text{f}_{KD}^{SHIN} (Stoilos et al., 2007; Stoiilos et al. 2005b), reasoning with \text{f}_{KD}^{SHI} (Li et al., 2006), supporting reasoning in f-DLs that allow for general concept inclusion axioms (Li et al., 2006; Stoiilos et al., 2006), fuzzy extensions of the OWL language (Stoiilos et al., 2005a) supporting expressve datatypes (Wang et al., 2008) or adding more expressive fuzzy features, like comparison expressions (Kang et al., 2006; Lu et al., 2008) and concept modifiers (Hölldobler et al., 2006; Wang et al., 2006). Interestingly, there also exist two f-DL reasoners, FiRE\textsuperscript{5} (Stoiilos et al., 2007), which supports \text{f}_{KD}^{SHIN} and the fuzzyDL\textsuperscript{6} (Straccia, 2008), which supports \text{f}_{KD}^{SHI(D)} and \text{f}_{-SHI(D)}. Unfortunately, like their crisp counterparts, fuzzy-SHIN and fuzzy-SHIf(D) come with (at least) \text{ExpTime} computational complexity. Additionally, the practical behavior of implementations of such logics would also have to deal with the degrees thus adding more to the practical complexity.

Following current research developments in crisp DLs, there is an effort on developing lightweight fuzzy ontology languages. In particular, today there exist two such languages, namely fuzzy DL-Lite (Pan et al., 2008; Straccia, 2006) and fuzzy EL+ (Stoiilos et al., 2008). Like their crisp counterparts, fuzzy DL-Lite is specifically tailored for data intensive applications, offering for efficient instance retrieval services by utilizing database technologies, while fuzzy EL+ is especially tailored for applications that require the managements of large concept hierarchies/taxonomies offering for efficient classification services. Even more interestingly, in the fuzzy case fuzzy DL-Lite allows for far more expressive and flexible queries that utilize the power of the fuzzy component. For example, one can issue a query of the form: