SWRL-F - A Fuzzy Logic Extension of the Semantic Web Rule Language

Tomasz Wiktor Wlodarczyk[1], Martin O'Connor[2], Chunming Rong[1], Mark Musen[2],

[1] University of Stavanger, Norway; [2] Stanford University, USA
tomasz.w.wlodarczyk@uis.no

Abstract. Enhancing Semantic Web technologies with an ability to express uncertainty and imprecision is widely discussed topic. While SWRL can provide additional expressivity to OWL-based ontologies, it does not provide any way to handle uncertainty or imprecision. We introduce an extension of SWRL called SWRL-F that is based on SWRL rule language and uses SWRL's strong semantic foundation as its formal underpinning. We extend it with a SWRL-F ontology to enable fuzzy reasoning in the rule base. The resulting language provides small but powerful set of fuzzy operations that do not introduce inconsistencies in the host ontology.

Keywords: SWRL, SWRL-F, fuzzy logic, fuzzy rules, fuzzy, rule language, risk.

1 Introduction

Fuzzy Logic (FL) has provides a way to express imprecise information and helps in simplifying knowledge representation. For these reasons it is considered to be an important element in Semantic Web (SW) research. Despite the existing research work the problem of supplementing SW with FL remains without implemented, generic, publicly available, standards-based and widely used solution.

In this paper we present SWRL-F, a Fuzzy Logic extension of the Semantic Web Rule Language. It allows expressing imprecise information and helps in simplifying knowledge representation in SWRL. It consists of two parts. SWRL-F ontology that allows representing FL knowledge in the ontology and SWRL rule base, and execution engine that integrates with Protégé [1]. One of the areas where fuzzy logic found significant application are control systems. In this work we based on the control system approach that follows the scheme: collect crisp inputs, fuzzify inputs, perform fuzzy inference, defuzzify inputs, apply crisp outputs [2].

Related Work. Pan et al. [3] propose f-SWRL, a fuzzy extension to SWRL. It includes fuzzy assertions and fuzzy rules, however, does not describe any implementation. Moreover, that approach is criticized in Agarwal and Hitzler [4], who explain that syntax and semantics of f-SWRL actually offer no fuzziness in f-SWRL rules. Bobillo et al. [5] present a semantic fuzzy expert system for a fuzzy balanced scorecard. They use OWL ontology to represent knowledge about variables. They also provide and interface to FuzzyJess to execute fuzzy rules. Protege is used as a development platform; however, implementation focuses only on balanced scorecard and rules are not based on SWRL. A need for more generic approach is mentioned in conclusions. Stoilos et al. [6] discuss Fuzzy OWL and uncertainty representation with rules. They present a fuzzy reasoning engine that implements a reasoning algorithm for a fuzzy DL language fKD-SHIN. It handles most of OWL features. However, the implementation is proprietary and does not connect directly with any established Semantic Web technologies or tools like OWL, SWRL or Protege. For additional related work one can refer to [7].

Contributions. In SWRL-F we aim to provide a FL extension to SWRL, which is based on standard OWL DL and SWRL. SWRL-F ontology enables description of FL knowledge and its application in SWRL rules. We also implemented a test execution engine and development environment that is publically available[1].

Organization of the Paper. After the Introduction, in Section 2 we explain our design choices for SWRL-F in term of their influence on semantics of rules and logical soundness of ontology. In Section 3 we mention basic constructs of SWRL-F ontology. Further, in Section 4, we describe how to understand and construct fuzzy rules with SWRL-F. We conclude in Section 5.

2 Design Choices

Connection between FL and SW technologies based on DL is a non-trivial problem. We have made four main design choices that influence semantic of the rules and logical soundness of the ontology.

First, SWRL-F must be standard based. It includes anchoring in the well established fuzzy logic scheme. Our leading idea was to follow fuzzy control systems scheme: fuzzification, inference, defuzzification. Moreover, SWRL-F can be fully expressed using OWL and SWRL, by importing SWRL-F ontology that we created. This ontology is purely OWL-based and it is described in the Section 3.

Second, fuzzy inference in SWRL-F is limited to the rules only. This way we can avoid inconsistencies in the ontology. Ontology is used to describe fuzzy knowledge base, however, it can be interpreted in a limited, non-fuzzy way by a DL-reasoner. Until we connect fuzzy rule reasoner knowledge based on SWRL-F ontology has limited use, but it does not create any inconsistencies with standard SW technologies.

Third, fuzzy assertions in SWRL are represented as a standard object property defined in SWRL-F ontology, which has special meaning when interpreted by a fuzzy rule reasoner. It provides the most natural way of expression and can be interpreted (though not in a fuzzy way) by a non-fuzzy rule reasoner.

Fourth, we decided to reuse existing fuzzy rule engine namely FuzzyJess [8]. This allowed us to implement our solution faster and be sure that it will be stable and reasonably efficient. As FuzzyJess is a superset of Jess we could automatically provide compatibility with existing extensions and built-ins available for SWRL and SWRLJESSTab [9]. There is, though, one notable limitation of such approach: not all the OWL constructs can be represented, which follows the limitations as described in [10].

---

[1] http://protege.cim3.net/cgi-bin/wiki.pl?SWRLF

3 SWRL-F ontology

In order to express necessary fuzzy knowledge, namely fuzzy: sets, terms, variables and values, we have created SWRL-F ontology. Due to limited space, we present here only a few key elements. Representation follows Manchester syntax [11].

```
Class: FuzzyVariable
Class: FuzzyTerm
Class: FuzzyValue
Class: FuzzySet
ObjectProperty: hasFuzzySet
  Domain: FuzzyTerm, FuzzyValue
  Range: FuzzySet
ObjectProperty: hasFuzzyTerm
  Domain: FuzzyVariable
  Range: FuzzyTerm
ObjectProperty: hasFuzzyValue
  Domain: FuzzyVariable
  Range: FuzzyValue
ObjectProperty: hasFuzzyVariable
  Domain: FuzzyValue
  Range: FuzzyVariable
```

4 SWRL-F Rules

Having FuzzyValues and FuzzyTerms described one can construct rules in SWRL-F. To do so we use modified SWRLJessTab. SWRL-F rules are normal SWRL rules that make use of fuzzymatch object property from SWRL-F ontology. If executed using standard rule engine like Jess this property acts as any other object property. However, if run using modified version of SWRLJessTab together with FuzzyJ and FuzzyJess packages, fuzzymatch property allows constructing fuzzy rules.

```
ObjectProperty: fuzzymatch
  Domain: FuzzyValue
  Range: FuzzyTerm
```

Let us analyze a generic example:

```
FuzzyValue (?v1) ∧ fuzzymatch(?v1, someFuzzyTerm) ∧
FuzzyValue(?v2) → fuzzymatch(?v2, otherFuzzyTerm)
```

The fuzzymatch property is used to calculate degree of membership of FuzzyValue ?v1 in the someFuzzyTerm. FuzzyValues and FuzzyTerms are related by FuzzyVariables. Second use of fuzzymatch allows to bind the value of otherFuzzyTerm to the ?v2 FuzzyValue, basing on the calculated degree of membership.

Many rules can assign new values to the same FuzzyValue. In contrast with standard SWRL where such assertions would not carry any additional semantics, in SWRL-F the values that each rule assigns are then grouped together and collectively defuzified into one final crisp result. Apart from simplifying management and creation of rules, this allows to create rules in a more natural way.

5 Conclusions

In this paper we presented SWRL-F. It is an extension to SWRL that allows constructing fuzzy rules using lexical variables described it OWL-based ontology. Its general design is based on fuzzy control system approach and together with proper construction of SWRL-F ontology it allows to avoid conflicts between FL and DL in the ontology. SWRL-F can be used to extend any SW application with FL capabilities basing on Protege editor and modified SWRLJessTab.

SWRL-F does not introduce any inconsistencies into a DL-based ontology due to limiting fuzzy inference to rules basing on SWRL-F ontology construction. However, it has the some limitations with regards to OWL representation as explained in [10].

SWRL-F allows easier knowledge management by moving numerical values from rules to ontology. This results in simpler rules and removes hard-coding of those numerical values in rules.

References

[1] "The Protégé Ontology Editor and Knowledge Acquisition System" Available: http://protege.stanford.edu/.

[2] T.J. Ross, "Fuzzy Control Systems," *Fuzzy Logic with Engineering Applications*, Wiley-Blackwell, 2004.

[3] J.Z. Pan, G. Stamou, V. Tzouvaras, and I. Horrocks, "f-SWRL: A Fuzzy Extension of SWRL," *Artificial Neural Networks: Formal Models and Their Applications - ICANN 2005*, 2005, pp. 829-834.

[4] S. Agarwal and P. Hitzler, "Modeling Fuzzy Rules with Description Logics."

[5] F. Bobillo, M. Delgado, J. Gómez-Romero, and E. López, "A semantic fuzzy expert system for a fuzzy balanced scorecard," *Expert Syst. Appl.*, vol. 36, 2009, pp. 423-433.

[6] G. Stoilos, N. Simou, G. Stamou, and S. Kollias, "Uncertainty and the Semantic Web," *IEEE Intelligent Systems*, vol. 21, 2006, pp. 84-87.

[7] T. Lukasiewicz and U. Straccia, "Managing uncertainty and vagueness in description logics for the Semantic Web," *Web Semant.*, vol. 6, 2008, pp. 291-308.

[8] B. Orchard, "Controlling with fuzzy rules," *Jess in Action: Java Rule-Based Systems*, Manning Publications, 2003.

[9] "ProtegeWiki: SWRLJess Tab" Available: http://protege.cim3.net/cgi-bin/wiki.pl?SWRLJessTab.

[10] M. O'connor, H. Knublauch, S. Tu, B. Grosof, M. Dean, W. Grosso, and M. Musen, "Supporting Rule System Interoperability on the Semantic Web with SWRL," *The Semantic Web – ISWC 2005*, 2005, pp. 974-986.

[11] "OWL 2 Web Ontology Language Manchester Syntax" Available: http://www.w3.org/TR/owl2-manchester-syntax/.