ISWC 2008

The 7th International Semantic Web Conference

Fernando Bobillo
Paulo C. G. Costa
Claudia d'Amato
Nicola Fanizzi
Kathryn B. Laskey
Kenneth J. Laskey
Thomas Lukasiewicz
Trevor Martin
Matthias Nickles
Michael Pool
Pavel Smrz

Uncertainty Reasoning for the Semantic Web

October 26, 2008

The 7th International Semantic Web Conference
October 26 – 30, 2008
Congress Center, Karlsruhe, Germany

ISWC 2008

Platinum Sponsors

Ontoprise

Gold Sponsors

BBN
eyeworkers
Microsoft
NeOn
SAP Research
Vulcan

Silver Sponsors

ACTIVE
ADUNA
Saltlux
SUPER
X-Media
Yahoo

The 7th International Semantic Web Conference
October 26 – 30, 2008
Congress Center, Karlsruhe, Germany

ISWC 2008

Organizing Committee

General Chair
Tim Finin (University of Maryland, Baltimore County)

Local Chair
Rudi Studer (Universität Karlsruhe (TH), FZI Forschungszentrum Informatik)

Local Organizing Committee
Anne Eberhardt (Universität Karlsruhe)
Holger Lewen (Universität Karlsruhe)
York Sure (SAP Research Karlsruhe)

Program Chairs
Amit Sheth (Wright State University)
Steffen Staab (Universität Koblenz Landau)

Semantic Web in Use Chairs
Mike Dean (BBN)
Massimo Paolucci (DoCoMo Euro-labs)

Semantic Web Challenge Chairs
Jim Hendler (RPI, USA)
Peter Mika (Yahoo, ES)

Workshop chairs
Melliyal Annamalai (Oracle, USA)
Daniel Olmedilla (Leibniz Universität Hannover, DE)

Tutorial Chairs
Lalana Kagal (MIT)
David Martin (SRI)

Poster and Demos Chairs
Chris Bizer (Freie Universität Berlin)
Anupam Joshi (UMBC)

Doctoral Consortium Chairs
Diana Maynard (Sheffield)

Sponsor Chairs
John Domingue (The Open University)
Benjamin Grosof (Vulcan Inc.)

Metadata Chairs
Richard Cyganiak (DERI/Freie Universität Berlin)
Knud Möller (DERI)

Publicity Chair
Li Ding (RPI)

Proceedings Chair
Krishnaprasad Thirunarayan (Wright State University)

Fellowship Chair
Joel Sachs (UMBC)

The 4th Workshop on Uncertainty Reasoning for the Semantic Web

The Uncertainty Reasoning Workshop is an exciting opportunity for collaboration and cross-fertilization between the uncertainty reasoning community and the Semantic Web community.

Effective methods for reasoning under uncertainty are vital for realizing many aspects of the Semantic Web vision, but the ability of current-generation web technology to handle uncertainty is extremely limited. Recently, there has been a groundswell of demand for uncertainty reasoning technology among Semantic Web researchers and developers.

This surge of interest creates a unique opening to bring together two communities with a clear commonality of interest but little history of interaction. By capitalizing on this opportunity, URSW could spark dramatic progress toward realizing the Semantic Web vision.

Audience

The intended audience for this workshop includes the following:

Researchers in uncertainty reasoning technologies with interest in Semantic Web and Web-related technologies.

- Semantic web developers and researchers.
- People in the knowledge representation community with interest in the Semantic Web.
- Ontology researchers and ontological engineers.
- Web services researchers and developers with interest in the Semantic Web.
- Developers of tools designed to support semantic web implementation, e.g., Jena developers, Protégé and Protégé-OWL developers.

Topic List

We intend to have an open discussion on any topic relevant to the general subject of uncertainty in the Semantic Web (including fuzzy theory, probability theory, and other approaches). Therefore, the following list should be just an initial guide.

- Syntax and semantics for extensions to Semantic Web languages to enable representation of uncertainty.
- Logical formalisms to support uncertainty in Semantic Web languages.
- Probability theory as a means of assessing the likelihood that terms in different ontologies refer to the same or similar concepts.
- Architectures for applying plausible reasoning to the problem of ontology mapping.
- Using fuzzy approaches to deal with imprecise concepts within ontologies.
- The concept of a probabilistic ontology and its relevance to the Semantic Web.
- Best practices for representing uncertain, incomplete, ambiguous, or controversial information in the Semantic Web.
- The role of uncertainty as it relates to Web services.
- Interface protocols with support for uncertainty as a means to improve interoperability among Web services.
- Uncertainty reasoning techniques applied to trust issues in the Semantic Web.
- Existing implementations of uncertainty reasoning tools in the context of the Semantic Web.
- Issues and techniques for integrating tools for representing and reasoning with uncertainty.
- The future of uncertainty reasoning for the Semantic Web.

Program Committee

Ameen Abu-Hanna - Universiteit van Amsterdam, the Netherlands.

Fernando Bobillo - University of Granada, Spain.

Silvia Calegari - University of Milano-Bicocca, Italy.

Paulo C. G. Costa - George Mason University, USA.

Fabio G. Cozman - Universidade de Sao Paulo, Brazil.

Claudia d'Amato - University of Bari, Italy.

Ernesto Damiani - University of Milan, Italy.

Nicola Fanizzi - University of Bari, Italy.

Francis Fung - Eduworks, Inc., USA.

Kathryn B. Laskey - George Mason University, USA.

Kenneth J. Laskey - MITRE Corporation, USA. Member of the W3C Advisory Board.

Thomas Lukasiewicz - Oxford University, UK.

Anders L. Madsen - Hugin Expert A/S, Denmark.

M. Scott Marshall - Adaptive Information Disclosure, Universiteit van Amsterdam, The Netherlands.

Trevor Martin - University of Bristol, UK.

Matthias Nickles - University of Bath, UK.

Yung Peng - University of Maryland, Baltimore County, USA.

Michael Pool - Convera, Inc., USA.

Livia Predoiu - Universität Mannheim, Germany.

Dave Robertson - University of Edinburgh, UK.

Daniel Sánchez - University of Granada, Spain.

Elie Sanchez - Université de La Méditerranée Aix-Marseille II , France.

Oreste Signore - Istituto di Scienza e Tecnologie dell' Informazione "A. Faedo", Italy. Manager of the W3C Office in Italy

Nematollaah Shiri - Concordia University, Canada.

Sergej Sizov - University of Koblenz-Landau, Germany.

Pavel Smrz - Brno University of Technology, Czech Republic.

Umberto Straccia - Istituto di Scienza e Tecnologie dell' Informazione "A. Faedo", Italy.

Heiner Stuckenschmidt - Universität Mannheim, Germany.

Masami Takikawa - Cleverset, Inc., USA.

Peter Vojtas - Charles University, Czech Republic.

Technical Papers

Position Papers

Technical Papers

Deciding Fuzzy Description Logics by Type Elimination*

Uwe Keller[1] and Stijn Heymans[2]

[1] Semantic Technology Institute (STI) Innsbruck, University of Innsbruck, Austria. eMail: uwe.keller@sti2.at
[2] Knowledge-based Systems Group, Institute of Information Systems, Vienna University of Technology, Austria. eMail: heymans@kr.tuwien.ac.at

Abstract. We present a novel procedure $\mathbf{FixIt}(\mathbb{ALC})$ for deciding knowledge base satisfiability in the Fuzzy Description Logic (FDL) $\mathbb{ALC}$. $\mathbf{FixIt}(\mathbb{ALC})$ does not search for tree-structured models as in tableau-based proof procedures, but embodies a fixpoint-computation of canonical models that are not necessarily tree-structured. Conceptually, the procedure is based on a type-elimination process. Soundness, completeness and termination are proven. To the best of our knowledge it is the first *fixpoint-based* decision procedure for FDLs, hence introducing a new class of inference procedures into FDL reasoning.

1 Introduction

Description Logics (DLs) [1] are a popular family of formally well-founded and decidable knowledge representation languages. DLs have a wide range of applications, e.g., they form the basis for Semantic Web (SW) ontology languages used such as OWL [7]. Fuzzy Description Logics (FDLs) [13] extend DLs to represent *vague* concepts and relations, and as such are very well suited to cover for representing and reasoning with uncertainty, a requirement that naturally arises in many practical applications of knowledge-based systems, in particular the SW.

So far, reasoning in Fuzzy DLs is mainly based on tableau-methods (e.g. [13,12,4,11,16,3]). Further, [14] demonstrates how to use inference procedures for classical DLs to perform reasoning in (some) FDLs. Still, reasoning in FDLs is at least as hard as reasoning in classical (crisp) DLs. Even in DLs of modest expressivity (e.g. $\mathbb{ALC}$ [13,14,12] the fuzzy variant of the DL $\mathcal{ALC}$ [10]) the worst-case complexity of reasoning is significant even in restricted cases [13]. Therefore, it is clear that there can not be a *single* inference method that works well on *all* problems.

Consequently, our goal is to enrich the range of available methods for reasoning with FDLs with a fundamentally different approach. In practical applications of DLs (and hence FDLs) a particularly important feature for representing domain models is the support of so-called *general terminologies* (see e.g. [12]), i.e., the possibility to capture (potentially recursive) interdependencies between complex concepts in a domain

* This work has been partially funded by the European Commission under the LarKC project (FP7 - 215535). Stijn Heymans is supported by the Austrian Science Fund (FWF) under projects P20305 and P20840.

model. However, besides the tableau-based methods for DLs (e.g [12,4,16,3]) there are at present no other FDL inference methods which can deal with general terminologies. We want to provide an alternative to tableau-based methods that can deal with general terminologies.

The main contributions of the paper are as follows:

- We present a novel procedure $\mathbf{FixIt}(\mathbb{ALC})$ (cf. Section 3) for deciding knowledge base (KB) satisfiability in the FDL $\mathbb{ALC}$ (cf. Section 2).
- We formally prove soundness, completeness and termination of the algorithm (cf. Section 3).
- $\mathbf{FixIt}(\mathbb{ALC})$ generalizes a type-elimination-based decision procedure [8] for the (classical) modal logic $\mathbf{K}$ (i.e. $\mathcal{KBDD}$ [6]) to the FDL $\mathbb{ALC}$. Additionally we integrate (fuzzy) ABoxes and general TBoxes which are not dealt with in $\mathcal{KBDD}$.
- To the best of our knowledge it is the first *fixpoint-based* decision procedure that has been proposed for FDL introducing a *new class of inference procedures* into FDL reasoning.
- Besides the tableau-based methods in [12,4,16,3], it is the only approach to integrate general terminologies in FDL reasoning and the first non-tableau-based one that we are aware of. General terminologies are handled in a fundamentally different way than in standard tableau-based method such as [12,4].

Our method is interesting especially regarding the last aspect since the handling of general terminologies in standard tableau-based methods (e.g. [12,4]) is a *major* source of non-determinism and thus computational inefficiency. In our case no non-deterministic choice is introduced by terminologies.

2 Preliminaries

We introduce $\mathbb{ALC}$ [13], the fuzzy variant of the Description Logic $\mathcal{ALC}$ [10] (the latter can be seen as a syntactic variant of the multi-modal logic $\mathbf{K}_{(\mathbf{m})}$ [9]). $\mathbb{ALC}$ provides the starting point for more expressive FDLs [15] that have been proposed to fuzzify major fragments of OWL [7].

Syntax. Concept expressions are constructed from a signature $\Sigma = (\mathbf{C}, \mathbf{R}, \mathbf{I})$ with concept names $\mathbf{C}$, role names $\mathbf{R}$, and individual names $\mathbf{I}$. The set of concept expressions $\mathcal{C}(\Sigma)$ over Σ is defined as the smallest set of expressions that contains $\mathbf{C}$, $\top$ and is closed under the application of the concept constructors $C \sqcap D$ (intersection), $C \sqcup D$ (union), $\neg C$ (complement), and $\forall R.C$ (universal role restriction) for $R \in \mathbf{R}$ and $C, D \in \mathcal{C}(\Sigma)$. We allow expressions $\exists R.C$ for $C \in \mathcal{C}(\Sigma), R \in \mathbf{R}$ and $\bot$ and treat them as shortcuts for $\neg\forall R.\neg C$ and $\neg\top$ respectively. A TBox axiom (or general concept inclusion axiom (GCI)) is an expression of the form $C \sqsubseteq D$ s.t. $C, D \in \mathcal{C}(\Sigma)$. A terminology (or TBox) $\mathcal{T}$ is a finite set of TBox axioms. Syntactically, the vagueness of descriptions becomes explicit only when describing specific instances and their interrelations: a (fuzzy) ABox axiom is either a $\langle i : C \bowtie d\rangle$ or a $\langle R(i, i') \geq d\rangle$ s.t. $i, i' \in \mathbf{I}$, $d \in [0, 1]$, and $\bowtie \in \{\leq, \geq, =\}$. An ABox $\mathcal{A}$ is a finite set of ABox axioms. Finally, a knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. Let $Ind_{\mathcal{A}} \subseteq \mathbf{I}$ denote the individual names that occur in $\mathcal{A}$. We denote the set of all concept expressions that occur as subexpressions in $\mathcal{K}$ by $\mathsf{sub}(\mathcal{K})$.

Semantics. Semantically, vagueness is reflected in the use of fuzzy sets and relations when interpreting concepts and roles: an interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ consists of a non-empty set $\Delta^\mathcal{I}$ called the domain, and a function $\cdot^\mathcal{I}$ which maps each concept name $C \in \mathbf{C}$ to a fuzzy set $C^\mathcal{I} : \Delta^\mathcal{I} \to [0,1]$, each role name $R \in \mathbf{R}$ to a fuzzy relation $R^\mathcal{I} : \Delta^\mathcal{I} \times \Delta^\mathcal{I} \to [0,1]$ and each individual name $i \in \mathbf{I}$ to an element $i^\mathcal{I} \in \Delta^\mathcal{I}$. The interpretation function $\cdot^\mathcal{I}$ is extended to arbitrary concept expressions $C \in \mathcal{C}(\Sigma)$ as follows:

- $(C \sqcap D)^\mathcal{I}(o) = min(C^\mathcal{I}(o), D^\mathcal{I}(o))$
- $(C \sqcup D)^\mathcal{I}(o) = max(C^\mathcal{I}(o), D^\mathcal{I}(o))$
- $(\neg C)^\mathcal{I}(o) = 1 - C^\mathcal{I}(o)$
- $(\forall R.C)^\mathcal{I}(o) = inf_{o' \in \Delta^\mathcal{I}}\{max(1 - R^\mathcal{I}(o,o'), C^\mathcal{I}(o'))\}$
- $\top^\mathcal{I}(o) = 1$

for all $o \in \Delta^\mathcal{I}, C, D \in \mathcal{C}(\Sigma)$, $R \in \mathbf{R}$.

An interpretation $\mathcal{I}$ satisfies a TBox axiom $\alpha = C \sqsubseteq D$ iff. for all $o \in \Delta^\mathcal{I}$ it holds that $C^\mathcal{I}(o) \leq D^\mathcal{I}(o)$, i.e. C is a fuzzy subset of D. $\mathcal{I}$ satisfies an ABox axiom $\alpha = \langle i : C \bowtie d \rangle$ iff. $C^\mathcal{I}(i^\mathcal{I}) \bowtie d$. $\mathcal{I}$ satisfies an ABox axiom $\alpha = \langle R(i,i') \geq d \rangle$ iff. $R^\mathcal{I}(i^\mathcal{I}, i'^\mathcal{I}) \geq d$. In all these cases, we write $\mathcal{I} \models \alpha$. $\mathcal{I}$ satisfies a TBox $\mathcal{T}$ (or is a model of $\mathcal{T}$) iff. $\mathcal{I} \models \alpha$ for all $\alpha \in \mathcal{T}$. $\mathcal{I}$ satisfies an ABox $\mathcal{A}$ (or is a model of $\mathcal{A}$) iff. $\mathcal{I} \models \alpha$ for all $\alpha \in \mathcal{A}$. Finally, $\mathcal{I}$ satisfies a knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ (or is a model of $\mathcal{K}$) iff. $\mathcal{I} \models \mathcal{T}$ and $\mathcal{I} \models \mathcal{A}$.

Reasoning in $\mathbb{ALC}$. Given a fuzzy KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, fuzzy ABox axioms or GCIs α and concept expressions $C, D \in \mathcal{C}(\Sigma)$, we can analyze particular semantic characteristics and interdependencies: We say that $\mathcal{K}$ is *satisfiable* (or consistent) iff. there is a model $\mathcal{I}$ for $\mathcal{K}$. $\mathcal{K}$ *entails* α (denoted as $\mathcal{K} \models \alpha$) iff. all models $\mathcal{I}$ of $\mathcal{K}$ satisfy α. Concept C is subsumed by concept D (wrt. a KB $\mathcal{K}$) iff. $\mathcal{K} \models C \sqsubseteq D$. Two concepts C and D are called *equivalent* (wrt. a KB $\mathcal{K}$) iff. for any model $\mathcal{I}$ of $\mathcal{K}$ it holds that $C^\mathcal{I}(o) = D^\mathcal{I}(o)$ for all $o \in \Delta^\mathcal{I}$. Two concepts C and D are called *disjoint* (wrt. a KB $\mathcal{K}$) iff. for any model $\mathcal{I}$ of $\mathcal{K}$ it holds that there does not exists an $o \in \Delta^\mathcal{I}$ such that $C^\mathcal{I}(o) > 0$ and $D^\mathcal{I}(o) > 0$. A concept C is called satisfiable (wrt. a KB $\mathcal{K}$) iff. there exists a model $\mathcal{I}$ of $\mathcal{T}$ such that $C^\mathcal{I}(o) > 0$ for some $o \in \Delta^\mathcal{I}$. Further, one might want to compute the truth value bounds for a given ABox assertion α wrt. $\mathcal{K}$ to determine the possibility interval that is enforced for α by the background knowledge in $\mathcal{K}$: The *greatest lower bound* of α wrt. $\mathcal{K}$ is defined as $glb(\alpha, \mathcal{K}) := sup\{d \in [0,1] \mid \mathcal{K} \models \langle \alpha \geq d \rangle\}$ and the *least upper bound* of α wrt. $\mathcal{K}$ is defined as $lub(\alpha, \mathcal{K}) := inf\{d \in [0,1] \mid \mathcal{K} \models \langle \alpha \leq d \rangle\}$ (where $sup\, \emptyset = inf\, \emptyset = 0$). Computing $glb(\alpha, \mathcal{K})$ and $lub(\alpha, \mathcal{K})$ is usually called the *best truth value bounds* (BTVB) problem.

One of the most fundamental reasoning problems is to determine whether a given fuzzy KB $\mathcal{K}$ is satisfiable. A lot of other reasoning tasks (e.g., checking for concept satisfiability wrt. a TBox, entailment of fuzzy ABox assertions, or the BTVB problem) can be reduced to KB satisfiability checking (cnf. [13]) and therefore solved by a respective decision procedure. For this reason, we consider KB satisfiability as the reasoning problem to be solved.

3 Fixpoint-based Decision Procedure

We present a decision procedure for KB satisfiability in $\mathbb{ALC}$ which does not rely on systematic search in the first place (as e.g. tableau-based methods), but instead constructs a canonical interpretation by means of a fixpoint construction. The so-constructed (canonical) interpretation (if non-empty) satisfies the TBox of a KB and allows to derive a model for the given knowledge base $\mathcal{K}$ iff. $\mathcal{K}$ is satisfiable. In contrast to tableau-based procedures a canonical interpretation is in general *not* tree-shaped. Further, it can be shown that the number of iterations required to reach a fixpoint is *linear* in the modal depth of $\mathcal{K}$.

Preprocessing. Without loss of generality, we can restrict ourselves to *normalized* knowledge bases, i.e. knowledge bases which contain only fuzzy ABox assertions of the form $\langle \alpha \geq d \rangle$ [12]. Further, we can assume that all axioms in $\mathcal{K}$ are in box normal form (BNF) [6] (i.e. the only negative concept subexpressions are of the form $\neg \forall R.C$ or negated atomic concept names $\neg C$).

3.1 Basic Notions and Intuition

Types. Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ denote a normalized $\mathbb{ALC}$ knowledge base in BNF. Let $\mathsf{sub}(\mathcal{K})$ denote the set of all concept expressions that occur as subexpressions somewhere in an axiom in $\mathcal{K}$. The *closure* of a knowledge base $\mathsf{cl}(\mathcal{K})$ is defined as the smallest set of concept expressions such that for all $C \in \mathsf{sub}(\mathcal{K})$, if C is not of the form $\neg D$, then $\{C, \neg C\} \subseteq \mathsf{cl}(\mathcal{K})$. Further, let $\mathsf{PossDeg}(\mathcal{K})$ denote the set of all relevant possibility degrees that can be derived from $\mathcal{K}$, i.e. $\mathsf{PossDeg}(\mathcal{K}) = \{0, 0.5, 1\} \cup \{d | \langle \alpha \geq d \rangle \in \mathcal{A}\} \cup \{1 - d | \langle \alpha \geq d \rangle \in \mathcal{A}\}$. It has been shown in [13,14] that if $\mathcal{K}$ is satisfiable, then there is as well a model of $\mathcal{K}$ which assigns possibility degrees in $\mathsf{PossDeg}(\mathcal{K})$ only. Hence, for our purposes we do not need to consider arbitrary possibility degrees $d \in [0, 1]$, but only the *finite* set $\mathsf{PossDeg}(\mathcal{K})$ that can be derived from $\mathcal{K}$.

The closure $\mathsf{cl}(\mathcal{K})$ and the relevant possibility degrees $\mathsf{PossDeg}(\mathcal{K})$ together give us the basic vocabulary to describe individuals and their (fuzzy) properties in interpretations for $\mathcal{K}$. More specifically, the notion of a *type* allows to represent individuals of an interpretation in a syntactic way: a *fuzzy $\mathcal{K}$-type* τ is a maximal subset of $\mathsf{cl}(\mathcal{K}) \times \mathsf{PossDeg}(\mathcal{K})$ such that:

1. if $\langle C, d \rangle \in \tau$ and $\langle C, d' \rangle \in \tau$ then $d = d'$
2. if $C = \neg C'$ then $\langle C, d \rangle \in \tau$ iff. $\langle C', 1 - d \rangle \in \tau$
3. if $C = C' \sqcap C''$ then $\langle C, d \rangle \in \tau$ iff. $\langle C', d' \rangle \in \tau$ and $\langle C'', d'' \rangle \in \tau$ and $d = min(d', d'')$
4. if $C = C' \sqcup C''$ then $\langle C, d \rangle \in \tau$ iff. $\langle C', d' \rangle \in \tau$ and $\langle C'', d'' \rangle \in \tau$ and $d = max(d', d'')$
5. for all $C \sqsubseteq C' \in \mathcal{T}$: if $\langle C, d \rangle \in \tau$ and $\langle C', d' \rangle \in \tau$ then $d \leq d'$
6. if $C = \top$ then $\langle C, 1 \rangle \in \tau$

Since $\mathsf{cl}(\mathcal{K})$ and $\mathsf{PossDeg}(\mathcal{K})$ are both finite sets, there are at most $2^{|\mathsf{cl}(D)| \cdot |\mathsf{PossDeg}(\mathcal{K})|}$

different $\mathcal{K}$-types. Each type τ can be seen as an individual and syntactically represents *all* (fuzzy) properties that can be observed about that individual: $\langle C, d\rangle \in \tau$ represents the statement that the respective individual τ belongs to concept C with the possibility degree d. Hence, the set of all $\mathcal{K}$-types (or simply types) provides enough vocabulary to let us describe all kinds of interpretations for $\mathcal{K}$ simply be fixing how to interconnect individuals (and therefore types).

Canonical Model. It turns out that it is possible to connect types in a fixed (or canonical) way, such that the interconnection defined is consistent with *almost* all properties specified syntactically in the type. The interconnections can be derived from the types themselves:

For a set of types T we can define for each role R a *canonical accessibility relation* $\Delta_R : T \times T \to \mathsf{PossDeg}(\mathcal{K})$ that "maximally" interconnects types $\tau, \tau' \in T$ with possibility degree $d \in \mathsf{PossDeg}(\mathcal{K})$: Let $\delta(d, d') := 1$ if $d \leq d'$ and $\delta(d, d') := 1 - d$ if $d > d'$. Then, we can define Δ_R by

$$\Delta_R(\tau, \tau') := min\{\delta(d, d') | \langle \forall R.C, d\rangle \in \tau, \langle C, d'\rangle \in \tau'\}$$

if $\forall R.C \in \mathsf{cl}(\mathcal{K})$ for some $C \in \mathbf{C}$, and $\Delta_R(\tau, \tau') := 1$ otherwise.

This way, we can construct a canonical interpretation $\mathcal{I}_T$ for any given set of types T using the canonical interconnection of types by Δ_R as follows: $\mathcal{I}_T = (T, \cdot^{\mathcal{I}_T})$ with (i) for any (atomic) concept name C in $\mathcal{K}$ and any $\tau \in T$ we set $C^{\mathcal{I}_T}(\tau) = d$ if $\langle C, d\rangle \in \tau$, and (ii) $R^{\mathcal{I}_T}(\tau, \tau') = \Delta_R(\tau, \tau')$ for any role R in $\mathcal{K}$ and any $\tau, \tau' \in T$. Please note, that by our definition of $\mathcal{K}$-types, $\mathcal{I}_T$ is well-defined for any concept name or role name. However, our definition deliberately leaves open the interpretation of individuals. We therefore define in fact a class of canonical interpretations, each of which fixes a specific way of how to interpret the individuals in a KB $\mathcal{K}$.

The canonical interconnection in $\mathcal{I}_T$ is chosen in such a way that all assignments of possibility degrees to concepts of the form $C = \forall R.C \in \tau$ are lower bounds for the possibility degrees that are in fact assigned by a canonical interpretation $\mathcal{I}_T$. Hence, such a canonical interpretation is *almost* immediately a (canonical) model for the terminology $\mathcal{T}$, i.e. it satisfies that

$$C^{\mathcal{I}_T}(\tau) = d \text{ iff. } \langle C, d\rangle \in \tau \qquad (*)$$

for *almost* all $C \in \mathsf{cl}(\mathcal{K})$ it holds and therefore $\mathcal{I}_T \models C \sqsubseteq C'$ for all $C \sqsubseteq C' \in \mathcal{T}$ by clause (5) in our definition of $\mathcal{K}$-types. That $(*)$ is satisfied by $\mathcal{I}_T$ is straightforward for the cases of concept names C, or complex concepts of the form $C = C' \sqcap C''$, $C = C' \sqcup C''$, $C = \neg C'$ and the $C^{\mathcal{I}_T}(\tau) \geq d$ case for $C = \forall R.C$ by our definition of types and the definition of Δ_R. The only cases where $(*)$ can be violated by $\mathcal{I}_T$ is for types τ containing universally role restricted concepts $\forall R.C$ that are assigned a possibility degree which is *too small* (wrt. the R-successor types τ' in $\mathcal{I}_T$) to properly reflect the semantics of $\forall R.C$ in $\mathbb{ALC}$, i.e. to coincide with the *greatest* lower bound of the set

$$\{max(1 - R^{\mathcal{I}_T}(\tau, \tau'), C^{\mathcal{I}_T}(\tau')) \,|\, \tau' \in T\}$$

Types τ in which the possibility degree assigned d to $\forall R.C$ is too small to be consistent with the semantics of $\mathbb{ALC}$ are called *bad types*. Bad types $\tau \in T$ can be detected

easily, since they satisfy that there exist $R \in \mathbf{R}, C \in \mathcal{C}(\Sigma), d \in \mathsf{PossDeg}(\mathcal{K})$ s.t. $\langle \forall R.C, d\rangle \in \tau$ and for all $\tau' \in T$: if $\langle C, d'\rangle \in \tau'$ then $max(1 - \Delta_R(\tau, \tau'), d') > d$.

This suggest the the following simple algorithm (which uses a *fuzzy type elimination* process as its core): in order to compute a maximal interpretation that satisfies all terminological axioms, we start off with the maximal set of types (i.e all $\mathcal{K}$-types) and iteratively fix all problems that prevent $(*)$ from being satisfied by removing bad types. This way, we must eventually reach a fixpoint after finitely many steps. If the resulting set of types is non-empty, we know that $(*)$ must hold (since all problems have been fixed) and therefore we can be certain that the corresponding canonical interpretation satisfies $\mathcal{T}$ (and covers all other possible models of $\mathcal{T}$ at the same time). Hence, we eventually need to check if all ABox axioms are satisfied by the canonical interpretation. If this is the case, we have found a model for $\mathcal{K}$, otherwise, we know that there can not be any interpretation that satisfies both $\mathcal{T}$ and $\mathcal{A}$ at the same time. In other words, $\mathcal{K}$ is not satisfiable.

Algorithm. The type elimination process sketched above can be formalized as shown in Fig. 1. Note that the emptiness test for the fixpoint T is covered implicitly: if the fixpoint T is empty, then the test in the if-statement fails trivially.

procedure satisfiable($\mathcal{K}$): boolean
T := $\{\tau | \tau$ is a $\mathcal{K}$-type $\}$;
repeat
  $T' := T$;
  $T := T' \setminus badtypes(T')$;
until $T = T'$;
if *there exists a total function* $\pi : Ind_{\mathcal{A}} \rightarrow T$ *s.t.* $\langle C, d'\rangle \in \pi(o)$ *and* $d \leq d'$ *for each* $\langle o : C \geq d\rangle \in \mathcal{A}$, *and* $\Delta_R(\pi(o), \pi(o')) \geq d$ *for each* $\langle R(o, o') \geq d\rangle \in \mathcal{A}$ **then**
  return true;
end
return false;

function $badtypes(T) : 2^T$
return $\{\tau \in T | \langle \forall R.C, d\rangle \in \tau$ and for all $\tau' \in T$: if $\langle C, d'\rangle \in \tau'$ then $max(1 - \Delta_R(\tau, \tau'), d') > d\}$;

Algorithm 1: The Type Elimination-based Decision Procedure $\mathbf{FixIt}(\mathbb{ALC})$

3.2 Soundness, Completeness and Termination

The termination, soundness, and completeness of our algorithm can be proven formally.

Theorem 1 (Termination). *For any* $\mathbb{ALC}$ *knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ *the algorithm* $\mathbf{FixIt}(\mathbb{ALC})$ *terminates after finitely many steps with either* true *or* false *as return value.*

Proof. The initialization step of the algorithm takes finitely many steps since the number of $\mathcal{K}$-types is finite. The repeat-loop must terminate after finitely many steps, since we start with a finite set of types T in the beginning: if we do not remove any type in an iteration (i.e. $badtypes(T') = \emptyset$) we have $T = T'$ at the end of the loop (i.e. reaching a fixpoint) and therefore terminate the loop. On the other hand, if $badtypes(T') \neq \emptyset$ in an iteration, at least one type is removed from T' and hence $T \subset T'$. This means, that the input set of types T for the next iteration is finite and strictly smaller. Clearly, the empty set is a fixpoint of $badtypes(\cdot)$ too, i.e. $badtypes(\emptyset) = \emptyset$. Hence, we can repeat the loop only finitely many times until we finally will reach a fixpoint. Since this fixpoint T is a subset of the finite set of the initial set of types and there are only finitely many possible mappings π to consider, deciding the criterion in the if-statement (based on T) takes as well only finitely many steps. Therefore, the algorithm terminates with one of the return-statements that give as a result either true *or* false.

The following lemma is a key element of the soundness and completeness proof and shows that by successively removing bad types we can indeed ensure that types encode possibility degree assignments to concepts that coincide with the canonical interpretation, and that any such canonical interpretation is a model of the $\mathcal{T}$.

Let T be the set of types that is computed as the fixpoint in the algorithm $\mathbf{FixIt}(\mathbb{ALC})$, i.e. $badtypes(T) = \emptyset$ and let $\mathcal{I}_T$= $(T, \cdot^{\mathcal{I}_T})$ be a canonical interpretation for T as defined above.

Lemma 1. *For each $\mathcal{K}$-type τ, concept $C \in \mathsf{cl}(\mathcal{K})$ and $d \in \mathsf{PossDeg}(\mathcal{K})$ it holds that $C^{\mathcal{I}_T}(\tau) = d$ iff. $\langle C, d\rangle \in \tau$. Further, $\mathcal{I}_T \models \mathcal{T}$.*

Proof. For the first part of the lemma, let τ be any $\mathcal{K}$-type and $d \in \mathsf{PossDeg}(\mathcal{K})$ be any relevant possibility degree.

We show by induction over the structure of concepts $C \in \mathsf{cl}(\mathcal{K})$ that $\langle C, d\rangle \in \tau$ iff. $C^{\mathcal{I}_T}(\tau) = d$: the base case (i.e. $C \in \mathsf{cl}(\mathcal{K})$ is an atomic concept name $C \in \mathbf{C}$ or $C = \top$) is trivially satisfied by our definition of $\mathcal{I}_T$. For the induction step, we consider the different cases of compound concept expressions $C \in \mathsf{cl}(\mathcal{K})$ one-by-one:

1. for $C = C_1 \sqcap C_2 \in \mathsf{cl}(\mathcal{K})$, we know that $C_1, C_2 \in \mathsf{cl}(\mathcal{K})$. By clause (3) in our definition of types, we know that $\langle C, d\rangle = \langle C_1 \sqcap C_2, d\rangle \in \tau$ iff. $\langle C_1, d_1\rangle \in \tau$ and $\langle C_2, d_2\rangle \in \tau$ and $d = min(d_1, d_2)$. Applying the induction hypothesis to C_1 and C_2, we know that this is the case iff. ${C_1}^{\mathcal{I}_T}(\tau) = d_1$ and ${C_2}^{\mathcal{I}_T}(\tau) = d_2$ and $d = min(d_1, d_2)$ iff. $d = min({C_1}^{\mathcal{I}_T}(\tau), {C_2}^{\mathcal{I}_T}(\tau))$ iff. $d = (C_1 \sqcap C_2)^{\mathcal{I}_T} = C^{\mathcal{I}_T}$ by the by the semantics of $\sqcap$ in $\mathbb{ALC}$.

2. for $C = C_1 \sqcup C_2 \in \mathsf{cl}(\mathcal{K})$ the proof is analogous.

3. for $C = \neg D \in \mathsf{cl}(\mathcal{K})$, we know that $D \in \mathsf{cl}(\mathcal{K})$ by the definition of $\mathsf{cl}(\mathcal{K})$. Because of clause (2) and the maximality requirement in our definition of $\mathcal{K}$-types, we know that $\langle C, d\rangle = \langle \neg D, d\rangle \in \tau$ iff. $\langle D, 1 - d\rangle \in \tau$. Applying the induction hypothesis for D, we know that this holds iff. $D^{\mathcal{I}_T}(\tau) = 1 - d$ iff. $(\neg D)^{\mathcal{I}_T}(\tau) = C^{\mathcal{I}_T}(\tau) = d$ by the semantics of concept negation in $\mathbb{ALC}$.

4. for $C = \forall R.D \in \mathsf{cl}(\mathcal{K})$, $D \in sub(\mathcal{K})$ holds and hence $D \in \mathsf{cl}(\mathcal{K})$ by the definition of $\mathsf{cl}(\mathcal{K})$.

First, we show one direction, i.e. that $C^{\mathcal{I}_T}(\tau) = d$ if $\langle C, d\rangle \in \tau$: Assume that $\langle C, d\rangle = \langle \forall R.D, d\rangle \in \tau$. According to the semantics of the universal role restriction in $\mathbb{ALC}$ and our definition of $\mathcal{I}_T$, we have $C^{\mathcal{I}_T}(\tau) = (\forall R.D)^{\mathcal{I}_T}(\tau) = inf_{\tau' \in T}\{max(1 - R^{\mathcal{I}_T}(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\} = inf_{\tau' \in T}\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\}$. We show that d is a lower bound for $\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau')) | \tau' \in T\}$: Assume there exists a $\tau' \in T$ s.t. $d > max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))$. Let $D^{\mathcal{I}_T}(\tau') = d'$. Applying the induction hypothesis to $D \in \mathsf{cl}(\mathcal{K})$, we know $\langle D, d'\rangle \in \tau'$. Hence, both $d > 1 - \Delta_R(\tau, \tau')$ and $d > d'$ must hold. Hence $\Delta_R(\tau, \tau') > 1 - d$. But, since $\langle \forall R.D, d\rangle \in \tau$ this is not possible by our by our definition of Δ_R, because $\Delta_R(\tau, \tau') \leq 1 - d$. From the contradiction we can conclude that d is in fact a lower bound for the considered set. Therefore, $d \leq inf_{\tau' \in T}\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\}$

Next, we show that $inf_{\tau' \in T}\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\} \leq d$ too, by proving that there exists a $\tau' \in T$ s.t. $max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau')) \leq d$. Assume, the contrary, i.e. for all $\tau' \in T$: $max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau')) > d$ (‡). By applying our induction hypothesis to D and τ', we know that this is the case iff. for all $\tau' \in T$: if $\langle D, d'\rangle \in \tau'$ then $max(1 - \Delta_R(\tau, \tau'), d') > d$. But then, τ would be a bad type which contradicts the fact that T the computed fixpoint which can not contain any bad types (i.e. $\tau \in badtypes(T) = \emptyset$). Hence our assumption (‡) must be wrong, and we can conclude that $inf_{\tau' \in T}\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\} \leq d$. Therefore, $d = inf_{\tau' \in T}\{max(1 - \Delta_R(\tau, \tau'), D^{\mathcal{I}_T}(\tau'))\}$, and hence $d = C^{\mathcal{I}_T}(\tau)$.

The other direction of the induction hypothesis (i.e. that $\langle \forall R.D, d\rangle \in \tau$ if $(\forall R.D)^{\mathcal{I}_T}(\tau) = d$) can now be proven as follows: Assume that $(\forall R.D)^{\mathcal{I}_T}(\tau) = d$ (†) but $\langle \forall R.D, d\rangle \notin \tau$. By the maximality requirement in our definition of $\mathcal{K}$-types there must hence exist a $d' \in \mathsf{PossDeg}(\mathcal{K})$ s.t. $\langle \forall R.D, d'\rangle \in \tau$ and $d' \neq d$. Using the same argument as for the if-direction in this case, we can therefore conclude that $(\forall R.D)^{\mathcal{I}_T}(\tau) = d' \neq d$ which contradicts (†). Hence, our assumption must be wrong and $\langle \forall R.D, d\rangle \in \tau$ must hold whenever $(\forall R.D)^{\mathcal{I}_T}(\tau) = d$.

For the second part of the lemma, to show that $\mathcal{I}_T \models \mathcal{T}$, assume that for some $\alpha = C \sqsubseteq C' \in \mathcal{T}$ and some $\tau \in T$ it holds that $C^{\mathcal{I}_T}(\tau) > C'^{\mathcal{I}_T}(\tau)$, in other words, if $C^{\mathcal{I}_T}(\tau) = d$ and $C'^{\mathcal{I}_T}(\tau) = d'$ then $d > d'$. Thus, we can deduce (from the first part of this lemma) that, if $\langle C, d\rangle \in \tau$ and $\langle C', d'\rangle \in \tau$ then $d > d'$. However, by our definition of $\mathcal{K}$-type (i.e. clause (5)), we also know that in this case $d \leq d'$ must hold, which is contradictive. Hence, our assumption must be wrong and $\mathcal{I}_T \models \alpha$ for each $\alpha \in \mathcal{T}$ which means that $\mathcal{I}_T \models \mathcal{T}$.

Theorem 2 (Soundness). *If* $\mathbf{FixIt}(\mathbb{ALC})$ *returns true for a* $\mathbb{ALC}$ *knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$*, then* $\mathcal{K}$ *is satisfiable.*

Proof. We show that a canonical interpretation $\mathcal{I}_T$ for the computed fixpoint T can be extended to a model of $\mathcal{K}$. By Lemma 1, we already know that $\mathcal{I}_T \models \mathcal{T}$. We now show, that $\mathcal{I}_T$ can be extended such that $\mathcal{I}_T \models \mathcal{A}$ too, which completes the proof: Since the algorithm returns true, there exist a total function $\pi : Ind_{\mathcal{A}} \rightarrow T$ s.t. $\langle C, d'\rangle \in \pi(o)$ and $d \leq d'$ for each $\langle o : C \geq d\rangle \in \mathcal{A}$ (⋆), and $\Delta_R(\pi(o), \pi(o')) \geq d$ for each $\langle R(o, o') \geq d\rangle \in \mathcal{A}$ (†). We extend the definition of $\mathcal{I}_T$ to the ABox $\mathcal{A}$ as follows: for all Abox individual names $o \in Ind_{\mathcal{A}}$, we set $o^{\mathcal{I}_T} := \pi(o) \in T$. First, consider an ABox axiom of the form $\alpha = \langle o : C \geq d\rangle \in \mathcal{A}$. Then, $\mathcal{I}_T \models \alpha$ iff. $C^{\mathcal{I}_T}(o^{\mathcal{I}_T}) \geq d$,

iff. $C^{\mathcal{I}_T}(\pi(o)) \geq d$ *iff. there exists a* $d' \geq d$ *s.t.* $C^{\mathcal{I}_T}(\pi(o)) = d'$*. By Lemma 1 this is the case iff. there exists a* $d' \geq d$ *s.t.* $\langle C, d' \rangle \in \pi(o)$ *which is satisfied since* $(\star)$ *holds. Second, consider an ABox axiom of the form* $\alpha = \langle R(o, o') \geq d \rangle \in \mathcal{A}$*. Then,* $\mathcal{I}_T \models \alpha$ *iff.* $R^{\mathcal{I}_T}(o^{\mathcal{I}_T}, o'^{\mathcal{I}_T}) \geq d$ *iff.* $\Delta_R(\pi(o), \pi(o')) \geq d$ *(by Def. of the extended* $\mathcal{I}_T$*). The latter is satisfied because of* $(\dagger)$*.*

An second key element for the completeness proof is the following lemma that shows that our canonical way of interconnecting types (in the fixpoint set) is maximal or the strongest possible one in the following sense: the interconnection R of individuals o, o' defined by any model $\mathcal{I}$ of $\mathcal{K}$ is covered by the canonical interconnection Δ_R of the respective types $\tau(o), \tau(o')$ representing o, o' in $\mathcal{I}$.

Lemma 2. *Let* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *be any model of* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$*. For each individual* $o \in \Delta^{\mathcal{I}}$ *we define its corresponding type* $\tau(o) := \{\langle C, d \rangle \in \mathsf{cl}(\mathcal{K}) \times \mathsf{PossDeg}(\mathcal{K}) | C^{\mathcal{I}}(o) = d\}$*. Then,* $\Delta_R(\tau(o), \tau(o')) \geq R^{\mathcal{I}}(o, o')$ *for all* $o, o' \in \Delta^{\mathcal{I}}$*.*

Proof. Assume that there exist $o, o' \in \Delta^{\mathcal{I}}$ *s.t.* $\Delta_R(\tau(o), \tau(o')) < R^{\mathcal{I}}(o, o')$*. By our definition of* Δ_R*, we then know that* $\delta(d, d') < R^{\mathcal{I}}(o, o')$ $(*)$ *for some* $\langle \forall R.C, d \rangle \in \tau(o)$ *and* $\langle C, d' \rangle \in \tau(o')$*. From the definition of* $\tau(\cdot)$*, we know that* $\delta(d, d') < R^{\mathcal{I}}(o, o')$ *for some* $o, o' \in \Delta^{\mathcal{I}}$ *s.t.* $(\forall R.C)^{\mathcal{I}}(o) = d$ *and* $C^{\mathcal{I}}(o') = d'$*. From the semantics of* $\forall R.C$ *in* $\mathbb{ALC}$*, we derive* $d = inf_{o'' \in \Delta^{\mathcal{I}}}\{max(1 - R^{\mathcal{I}}(o, o''), C^{\mathcal{I}_T}(o''))\}$*. Hence, in particular* $d \leq max(1 - R^{\mathcal{I}}(o, o'), C^{\mathcal{I}_T}(o')) = max(1 - R^{\mathcal{I}}(o, o'), d')$ $(\dagger)$*. We consider two cases: first,* $d' < d$*, then in order to satisfy* $(\dagger)$*,* $max(1 - R^{\mathcal{I}}(o, o'), d') = 1 - R^{\mathcal{I}}(o, o')$ *must hold and* $(\dagger)$ *simplifies to* $d \leq 1 - R^{\mathcal{I}}(o, o')$ *iff.* $R^{\mathcal{I}}(o, o') \leq 1 - d$*. Since* $d' < d$*,* $(*)$ *simplifies to* $1 - d < R^{\mathcal{I}}(o, o')$*, hence,* $1 - d < R^{\mathcal{I}}(o, o') \leq 1 - d$ *which is contradictory. In the second case, we assume that* $d' \geq d$*. Hence,* $\delta(d, d') = 1$*. Then,* $(\dagger)$ *simplifies to* $1 < R^{\mathcal{I}}(o, o')$*, which is contradictory, since* $R^{\mathcal{I}}(o, o') \in \mathsf{PossDeg}(\mathcal{K})$ *and* 1 *is the maximum possibility degree in* $\mathsf{PossDeg}(\mathcal{K})$*. Therefore, in both cases we reach a contradiction and can conclude that our assumption must be wrong. This concludes the proof.*

Theorem 3 (Completeness). *If an* $\mathbb{ALC}$ *knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ *is satisfiable, then* $\mathbf{FixIt}(\mathbb{ALC})$ *returns true for* $\mathcal{K}$

Proof. Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *be any model of* $\mathcal{K} = (\mathcal{T}, \mathcal{A})$*, i.e.* $\mathcal{I} \models \mathcal{T}$ *and* $\mathcal{I} \models \mathcal{A}$*. In [13,14] it is shown that a KB in* $\mathbb{ALC}$ *is consistent iff. there is a model* $\mathcal{I}$ *of the KB which only assigns possibility degrees that occur in the ABox* $\mathcal{A}$ *for interpreting atomic concept or role names. Hence, without loss of generality, we can assume in the following that* $\mathcal{I}$ *assigns possibility degrees in* $\mathsf{PossDeg}(\mathcal{K})$ *only.*

For each individual $o \in \Delta^{\mathcal{I}}$ *we define its corresponding type* $\tau(o) := \{\langle C, d \rangle \in \mathsf{cl}(\mathcal{K}) \times \mathsf{PossDeg}(\mathcal{K}) | C^{\mathcal{I}}(o) = d\}$ *and define* $T_{\mathcal{I}} := \{\tau(o) | o \in \Delta^{\mathcal{I}}\}$*. It is easy to see that* $T_{\mathcal{I}}$ *is a set of* $\mathcal{K}$*-types. Further,* $T_{\mathcal{I}} \neq \emptyset$ *since* $\Delta^{\mathcal{I}} \neq \emptyset$*.*

Let $T^{(i)}$ *denote the set of types that is computed after* i *iterations of the repeat-loop in our algorithm. We first show that* $T_{\mathcal{I}} \subseteq T^{(i)}$ *for all* $i \geq 0$ *by induction over the number of iterations* i*:*

In the base case $i = 0$*, our initialization step sets* $T^{(0)}$ *to contain all* $\mathcal{K}$*-types. Since* $T_{\mathcal{I}}$ *consists of* $\mathcal{K}$*-types only,* $T_{\mathcal{I}} \subseteq T^{(0)}$ *must hold. To proof the induction step, we*

assume that $T_{\mathcal{I}} \subseteq T^{(i)}$ but $T_{\mathcal{I}} \not\subseteq T^{(i+1)}$. Therefore, there must be a $\tau(o) \in T_{\mathcal{I}}$ s.t. $\tau(o) \in T^{(i)}$ but not $\tau(o) \in T^{(i+1)}$. From the repeat-loop in the algorithm, we know that $T^{(i+1)} = T^{(i)}\ badtypes(T^{(i)})$. Consequently, $\tau(o)$ must be a bad type $\tau(o) \in badtypes(T^{(i)})$ and we can not have reached a fix-point yet.

From our definition of bad-types we can derive that there must exist a $\langle \forall R.C, d\rangle \in \tau(o)$ and for all $\tau' \in T^{(i)}$: if $\langle C, d'\rangle \in \tau'$ then $max(1 - \Delta_R(\tau, \tau'), d') > d\}$ (‡). Since $T_{\mathcal{I}} \subseteq T^{(i)}$ (‡) must hold in particular for all $\tau(o') \in T_{\mathcal{I}}$. Using our definition of $\tau(\cdot)$ we can rephrase (‡) as follows: there must exist a $\forall R.C \in \mathsf{cl}(\mathcal{K})$ s.t. for all $o' \in \Delta^{\mathcal{I}}$: $max(1 - \Delta_R(\tau(o), \tau(o')), C^{\mathcal{I}}(o')) > (\forall R.C)^{\mathcal{I}}(o)$ (⋆).

By Lemma 2, we know that $\Delta_R(\tau(o), \tau(o')) \geq R^{\mathcal{I}}(o, o')$ for all $o, o' \in \Delta^{\mathcal{I}}$. Hence, $1 - \Delta_R(\tau(o), \tau(o')) \leq 1 - R^{\mathcal{I}}(o, o')$ (∗) for all $o, o' \in \Delta^{\mathcal{I}}$. Since $max(a, b) \leq max(a', b)$ for any a, a', b s.t. $a \leq a'$, we can reformulate (⋆) using (∗) as follows: there must exist a $\forall R.C \in \mathsf{cl}(\mathcal{K})$ s.t. for all $o' \in \Delta^{\mathcal{I}}$: $max(1 - R^{\mathcal{I}}(o, o'), C^{\mathcal{I}}(o')) > (\forall R.C)^{\mathcal{I}}(o)\}$ (♮), which contradicts the fact that $(\forall R.C)^{\mathcal{I}}(o) = inf_{o' \in \Delta^{\mathcal{I}}}\{max(1 - R^{\mathcal{I}}(o, o'), C^{\mathcal{I}}(o'))\}$: Since $R^{\mathcal{I}}(o, o') \in \mathsf{PossDeg}(\mathcal{K})$ and $C^{\mathcal{I}}(o') \in \mathsf{PossDeg}(\mathcal{K})$, we know that $max(1 - R^{\mathcal{I}}(o, o'), C^{\mathcal{I}}(o')) \in \mathsf{PossDeg}(\mathcal{K})$ for all $o' \in \Delta^{\mathcal{I}}$ by our definition of $\mathsf{PossDeg}(\mathcal{K})$. Because $\Delta^{\mathcal{I}} \neq \emptyset$ and $\mathsf{PossDeg}(\mathcal{K})$ is a finite set, there must exist an $o^* \in \Delta^{\mathcal{I}}$ for which $max(1 - R^{\mathcal{I}}(o, o^*), C^{\mathcal{I}}(o^*))$ is minimal, i.e. $max(1 - R^{\mathcal{I}}(o, o^*), C^{\mathcal{I}}(o^*)) \leq max(1 - R^{\mathcal{I}}(o, o'), C^{\mathcal{I}}(o'))$ for all $o' \in \Delta^{\mathcal{I}}$. Hence, $d^* := max(1 - R^{\mathcal{I}}(o, o^*), C^{\mathcal{I}}(o^*)) \in \mathsf{PossDeg}(\mathcal{K})$ is a lower bound for the set $\{max(1 - R^{\mathcal{I}}(o, o'), C^{\mathcal{I}}(o') | o' \in \Delta^{\mathcal{I}}\}$. However, from (♮) we know that $d < d^*$, hence d can not be the greatest lower bound (i.e. the infimum) of the set $\{max(1 - R^{\mathcal{I}}(o, o'), C^{\mathcal{I}}(o') | o' \in \Delta^{\mathcal{I}}\}$, hence $\forall R.C)^{\mathcal{I}}(o) = d^* > d$ which is contradictive.

Therefore, our assumption that $\tau(o)$ is a bad type must be wrong and we have completed the proof of the induction step as well as the induction argument.

We continue the proof of the lemma as follows: since $T = T^{(i)}$ is the fixpoint that is computed in the loop in our algorithm in i steps for some $i \geq 0$, we know that $T_{\mathcal{I}} \subseteq T^{(i)} = T$. Consider the mapping $\pi_{\mathcal{I}} : Ind_{\mathcal{A}} \to T$ defined by $\pi_{\mathcal{I}}(o) := \tau(o^{\mathcal{I}})$ for all $o \in Ind_{\mathcal{A}}$. Then, $\pi_{\mathcal{I}}$ is a well-defined, total function from $Ind_{\mathcal{A}}$ to T. We now show that this specific mapping $\pi_{\mathcal{I}}$ satisfies the condition that is checked in the if-statement of the algorithm:

In the first case, we consider any Abox axiom $\alpha \in \mathcal{A}$ of the form $\alpha = \langle o : C \geq d\rangle$. Since $\mathcal{I} \models \mathcal{A}$, $\mathcal{I} \models \alpha$ must hold. $\mathcal{I} \models \alpha$ iff. $C^{\mathcal{I}}(o^{\mathcal{I}}) \geq d$ iff. $C^{\mathcal{I}}(o^{\mathcal{I}}) = d'$ for some $d' \in \mathsf{PossDeg}(\mathcal{K})$ with $d' \geq d$ iff. $\langle C, d\rangle \in \tau(o^{\mathcal{I}})$ for some $d' \in \mathsf{PossDeg}(\mathcal{K})$ with $d' \geq d$ (by Lemma 1) iff. $\langle C, d\rangle \in \pi_{\mathcal{I}}(o)$ for some $d' \in \mathsf{PossDeg}(\mathcal{K})$ with $d' \geq d$ (by our definition of $\pi_{\mathcal{I}}$). Hence, the respective part of the if-condition for α holds for $\pi_{\mathcal{I}}$. In the second case, we consider any Abox axiom $\alpha \in \mathcal{A}$ of the form $\alpha = \langle R(o, o') \geq d\rangle$. Since, $\mathcal{I} \models \mathcal{A}$, $\mathcal{I} \models \alpha$ must hold. $\mathcal{I} \models \alpha$ holds iff. $R^{\mathcal{I}}(o^{\mathcal{I}}, o'^{\mathcal{I}}) \geq d$. Since $\Delta_R(\tau(o), \tau(o')) \geq R^{\mathcal{I}}(o^{\mathcal{I}}, o'^{\mathcal{I}})$ by Lemma 2, we know that $\Delta_R(\tau(o^{\mathcal{I}}), \tau(o'^{\mathcal{I}})) \geq d$ and therefore $\Delta_R(\pi_{\mathcal{I}}(o), \pi_{\mathcal{I}}(o)) \geq d$ by our definition of $\pi_{\mathcal{I}}$. Hence, the respective part of the if-condition for α is as well satisfied by $\pi_{\mathcal{I}}$. Consequently, the tested if-condition is satisfied by $\pi_{\mathcal{I}}$ and the algorithm returns true.

This leads to the main result, which is an immediate consequence of Theorems 2, 3, and 1:

Corollary 1. *The algorithm* $\mathbf{FixIt}(\mathbb{ALC})$ *is a sound and complete decision procedure for knowledge base satisfiability in* $\mathbb{ALC}$.

4 Related Work

Our method $\mathbf{FixIt}(\mathbb{ALC})$ generalizes the principle (i.e. a type elimination process) underlying the top-down variant of the $\mathcal{KBDD}$ procedure proposed in [6] for the modal logic $\mathbf{K}$ to the (more expressive) FDL $\mathbb{ALC}$. Further, our method integrates (fuzzy) ABoxes and TBoxes in the inference process both of which are *not* dealt with in $\mathcal{KBDD}$.

So far, reasoning in Fuzzy DLs has been mostly based on tableau-methods (e.g., [13,12,4,11]). Most of these methods do not support reasoning with general terminologies as it is possible with $\mathbf{FixIt}(\mathbb{ALC})$. The first method ever to integrate GCIs into FDL reasoning is [12]. A very similar approach is presented in [4] for the fuzzy variant of a more expressive DL, namely $\mathcal{SHI}$. Very recently, [16] proposed a novel and elegant method for reasoning with GCIs (under a more general semantics than here) which is inspired by earlier works on tableau-based reasoning in multi-valued logics. To the best of our knowledge there is no other approach to deal with GCIs in FDLs available at present. $\mathbf{FixIt}(\mathbb{ALC})$ therefore represents an interesting enrichment of inference calculi toolbox for FDLs, since no non-determinism is introduced by considering GCIs. A similar effect is achieved in [16] by the substantial modification of a standard tableau-based method and an extension with an MILP oracle. A very similar approach to [16] that is not fixed to a specific semantics is presented in [3].

Further, [14] demonstrates how to use inference procedures for *classical* DLs to perform reasoning in (some) FDLs. This allows to use algorithms that have been developed for classical DLs in FDL reasoning (for some FDLs) in an indirect way. Please note that the $\mathcal{KBDD}$ procedure can not be used in such an indirect way to perform $\mathbb{ALC}$ reasoning, since both TBoxes and ABoxes are not supported.

5 Conclusions and Future Work

We presented a novel procedure $\mathbf{FixIt}(\mathbb{ALC})$ for deciding knowledge base (KB) satisfiability in the FDL $\mathbb{ALC}$, introducing a *new class* of inference procedures into FDL reasoning. Besides the tableau-based methods [12,4,16,3], it is the only (and the first non-tableau-based) approach to integrate general terminologies in FDL reasoning that we are aware of.

The main research questions that we want to address next are as follows: we will study means of implicit representation of sets of fuzzy types known from Symbolic Model Checking, in particular their implementation by means of Ordered Binary Decision Diagrams (OBDDs) similar to [6], therefore addressing the main obstacle to apply the procedure in practice. A major question concerning optimization is clearly how to implement the final test of the algorithm efficiently, e.g. by heuristic search using the information in the ABox effectively to find the required mapping. The integration of optimizations such as full vs. lean representations or particle vs. types as discussed in [6] should be straightforward. We want to evaluate the effectiveness of the method

by an implementation and comparison to tableau-based systems for FDLs. Moreover, we believe that it is interesting to study a bottom-up variant of $\mathcal{KBDD}$ in the context of FDLs too, and to check if the integration of ABoxes can be done more efficiently in such a variant. Finally, we would like to see to what extend the method can cover other semantics for FDLs (e.g. other t-norms) and extended constructs, such as fuzzy modifiers and concrete domains.

References

1. F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.
2. R. E. Bryant. Symbolic Boolean manipulation with ordered binary-decision diagrams. *ACM Comput. Surv.*, 24(3):293–318, 1992.
3. V. Haarslev, H. Pai, and N. Shiri. Uncertainty Reasoning for Ontologies with General TBoxes in Description Logic. In P. C. G. Costa, C. D'Amato, N. Fanizzi, K. B. Laskey, K. Laskey, T. Lukasiewicz, M. Nickles, and M. Pool, editors, *Uncertainty Reasoning for the Semantic Web I*, LNAI. Springer, 2008.
4. Y. Li, B. Xu, J. Lu, and D. Kang. Discrete Tableau Algorithms for $\mathcal{FSHI}$. In *Proceedings of the International Workshop on Description Logics (DL)*, 2006.
5. K. L. McMillan. *Symbolic Model Checking*. Kluwer Academic Publishers, Norwell, MA, USA, 1993.
6. G. Pan, U. Sattler, and M. Y. Vardi. BDD-based decision procedures for the modal logic K. *Journal of Applied Non-Classical Logics*, 16(1-2):169–208, 2006.
7. P. F. Patel-Schneider, P. Hayes, and I. Horrocks. OWL Web Ontology Language Semantics and Abstract Syntax. Candidate Recommendation 18 August 2003, W3C, 2003.
8. V. R. Pratt. A Near-Optimal Method for Reasoning about Action. *J. Comput. Syst. Sci.*, 20(2):231–254, 1980.
9. K. Schild. A correspondence theory for terminological logics: Preliminary report. In *In Proceedings of the International Joint Conference of Artificial Intelligence (IJCAI 1991)*, pages 466–471, 1991.
10. M. Schmidt-Schauß and G. Smolka. Attributive Concept Descriptions with Complements. *Artif. Intell.*, 48(1):1–26, 1991.
11. G. Stoilos, G. B. Stamou, J. Z. Pan, V. Tzouvaras, and I. Horrocks. Reasoning with very expressive fuzzy description logics. *J. Artif. Intell. Res. (JAIR)*, 30:273–320, 2007.
12. G. Stoilos, U. Straccia, G. Stamou, and J. Pan. General Concept Inclusions in Fuzzy Description Logics. In *Proceedings of the 17th Eureopean Conference on Artificial Intelligence (ECAI-06)*, pages 457–461. IOS Press, 2006.
13. U. Straccia. Reasoning within Fuzzy Description Logics. *Journal of Artificial Intelligence Research*, 14:137–166, 2001.
14. U. Straccia. Transforming Fuzzy Description Logics into Classical Description Logics. In *Proceedings of the 9th European Conference on Logics in Artificial Intelligence (JELIA-04)*, number 3229 in Lecture Notes in Computer Science, pages 385–399, Lisbon, Portugal, 2004. Springer Verlag.
15. U. Straccia. A Fuzzy Description Logic for the Semantic Web. In E. Sanchez, editor, *Fuzzy Logic and the Semantic Web*, Capturing Intelligence, chapter 4, pages 73–90. Elsevier, 2006.
16. U. Straccia and F. Bobillo. Mixed integer programming, general concept inclusions and fuzzy description logics. *Mathware & Soft Computing*, 14(3):247–259, 2007.

DeLorean: A Reasoner for Fuzzy OWL 1.1

Fernando Bobillo, Miguel Delgado, and Juan Gómez-Romero

Department of Computer Science and Artificial Intelligence, University of Granada
C. Periodista Daniel Saucedo Aranda, 18071 Granada, Spain
Phone: +34 958243194; Fax: +34 958243317
Email: fbobillo@decsai.ugr.es, mdelgado@ugr.es, jgomez@decsai.ugr.es

Abstract. Classical ontologies are not suitable to represent imprecise or vague pieces of information, which has led to fuzzy extensions of Description Logics. In order to support an early acceptance of the OWL 1.1 ontology language, we present DeLorean, the first reasoner that supports a fuzzy extension of the Description Logic $\mathcal{SROIQ}$, closely equivalent to it. It implements some interesting optimization techniques, whose usefulness is shown in a preliminary empirical evaluation.

1 Introduction

Ontologies are a core element in the layered architecture of the Semantic Web. The current standard language for ontology representation is the Web Ontology Language (OWL). However, since its first development, several limitations on the expressiveness of OWL have been identified, and consequently several extensions to the language have been proposed. Among them, the most significant is OWL 1.1 [1] which is its most likely immediate successor. Description Logics (DLs for short) [2] are a family of logics for representing structured knowledge. They have proved to be very useful as ontology languages, and the DL $\mathcal{SROIQ}(\mathbf{D})$ is actually closely equivalent to OWL 1.1.

It has been widely pointed out that classical ontologies are not appropriate to deal with imprecise and vague knowledge, which is inherent to several real-world domains. Since fuzzy logic is a suitable formalism to handle these types of knowledge, several fuzzy extensions of DLs have been proposed [3].

The broad acceptance of the forthcoming OWL 1.1 ontology language will largely depend on the availability of editors, reasoners, and other numerous tools that support the use of OWL 1.1 from a high-level/application perspective [4]. With this idea in mind, this work reports the implementation of DeLorean, the first reasoner that supports the fuzzy DL $\mathcal{SROIQ}$. We also present a new optimization (handling superfluous elements before applying crisp reasoning) and a preliminary evaluation of the optimizations in the reduction.

This paper is organized as follows. Section 2 describes the fuzzy DL $\mathcal{SROIQ}$, which is equivalent to the fuzzy language supported by DeLorean. Then, Section 3 describes the reasoning algorithm, based on a reduction into crisp $\mathcal{SROIQ}$. Section 4 presents our implementation, some of the implemented optimizations (including handling of superfluous elements), and a preliminary evaluation. Finally, Section 6 sets out some conclusions and ideas for future work.

2 A Quick View to Fuzzy $\mathcal{SROIQ}$

In this section we recall the definition of fuzzy $\mathcal{SROIQ}$ [6], which extends $\mathcal{SROIQ}$ to the fuzzy case by letting concepts denote fuzzy sets of individuals and roles denote fuzzy binary relations. Axioms are also extended to the fuzzy case and some of them hold to a degree. We will assume a set of degrees of truth which are rational numbers of the form $\alpha \in (0,1]$, $\beta \in [0,1)$ and $\gamma \in [0,1]$. Moreover, we will assume a set of inequalities $\bowtie \in \{\geq, <, \leq, >\}$, $\rhd \in \{\geq, <\}$, $\lhd \in \{\leq, >\}$. For every operator $\bowtie$, we define: *(i)* its symmetric operator $\bowtie^-$, defined as $\geq^-=\leq, >^-=<, \leq^-=\geq, <^-=>$; *(ii)* its negation operator $\neg \bowtie$, defined as $\neg \geq=<, \neg >=\leq, \neg \leq=>, \neg <=\geq$.

Syntax. In fuzzy $\mathcal{SROIQ}$ we have three alphabets of symbols, for concepts ($\mathbf{C}$), roles ($\mathbf{R}$), and individuals ($\mathbf{I}$). The set of roles is defined by $R_A \cup U \cup \{R^- | R \in R_A\}$, where $R_A \in \mathbf{R}$, U is the universal role and R^- is the inverse of R. Assume $A \in \mathbf{C}$, $R, S \in \mathbf{R}$ (where S is a simple role [7]), $o_i \in \mathbf{I}$ for $1 \leq i \leq m$, $m \geq 1$, $n \geq 0$. Fuzzy concepts are defined inductively as follows: $C, D \rightarrow \top \mid \bot \mid A \mid C \sqcap D \mid C \sqcup D \mid \neg C \mid \forall R.C \mid \exists R.C \mid \{\alpha_1/o_1, \ldots, \alpha_m/o_m\} \mid (\geq m\ S.C) \mid (\leq n\ S.C) \mid \exists S.Self$. Let $a, b \in \mathbf{I}$. The axioms in a fuzzy Knowledge Base $\mathcal{K}$ are grouped in a fuzzy ABox $\mathcal{A}$, a fuzzy TBox $\mathcal{T}$, and a fuzzy RBox $\mathcal{R}$[1] as follows:

ABox	
Concept assertion	$\langle a{:}C \geq \alpha\rangle, \langle a{:}C > \beta\rangle, \langle a{:}C \leq \beta\rangle, \langle a{:}C < \alpha\rangle$
Role assertion	$\langle (a,b){:}R \geq \alpha\rangle$, $\langle (a,b){:}R > \beta\rangle$, $\langle (a,b){:}R \leq \beta\rangle, \langle (a,b){:}R < \alpha\rangle$
Inequality assertion	$\langle a \neq b\rangle$
Equality assertion	$\langle a = b\rangle$
TBox	
Fuzzy GCI	$\langle C \sqsubseteq D \geq \alpha\rangle, \langle C \sqsubseteq D > \beta\rangle$
Concept equivalence	$C \equiv D$, equivalent to $\{\langle C \sqsubseteq D \geq 1\rangle, \langle D \sqsubseteq C \geq 1\rangle\}$
RBox	
Fuzzy RIA	$\langle R_1 R_2 \ldots R_n \sqsubseteq R \geq \alpha\rangle, \langle R_1 R_2 \ldots R_n \sqsubseteq R > \beta\rangle$
Transitive role axiom	$trans(R)$
Disjoint role axiom	$dis(S_1, S_2)$
Reflexive role axiom	$ref(R)$
Irreflexive role axiom	$irr(S)$
Symmetric role axiom	$sym(R)$
Asymmetric role axiom	$asy(S)$

Semantics. A fuzzy interpretation $\mathcal{I}$ is a pair $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non empty set (the interpretation domain) and $\cdot^{\mathcal{I}}$ a fuzzy interpretation function mapping: *(i)* every individual a onto an element $a^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$; *(ii)* every concept C onto a function $C^{\mathcal{I}} : \Delta^{\mathcal{I}} \rightarrow [0,1]$; *(iii)* every role R onto a function $R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0,1]$. $C^{\mathcal{I}}$ (resp. $R^{\mathcal{I}}$) denotes the membership function of the fuzzy concept C

[1] The syntax of role axioms is restricted to guarantee the decidability of the logic [6].

(resp. fuzzy role R) w.r.t. $\mathcal{I}$. $C^{\mathcal{I}}(x)$ (resp. $R^{\mathcal{I}}(x,y)$) gives us the degree of being the individual x an element of the fuzzy concept C (resp. the degree of being (x,y) an element of the fuzzy role R) under the fuzzy interpretation $\mathcal{I}$. Given a t-norm $\otimes$, a t-conorm $\oplus$, a negation function $\ominus$ and an implication function $\Rightarrow$ [8], the interpretation is extended to complex concepts and roles as:

$$\begin{aligned}
\top^{\mathcal{I}}(x) &= 1\\
\bot^{\mathcal{I}}(x) &= 0\\
(C \sqcap D)^{\mathcal{I}}(x) &= C^{\mathcal{I}}(x) \otimes D^{\mathcal{I}}(x)\\
(C \sqcup D)^{\mathcal{I}}(x) &= C^{\mathcal{I}}(x) \oplus D^{\mathcal{I}}(x)\\
(\neg C)^{\mathcal{I}}(x) &= \ominus C^{\mathcal{I}}(x)\\
(\forall R.C)^{\mathcal{I}}(x) &= \inf_{y \in \Delta^{\mathcal{I}}} \{R^{\mathcal{I}}(x,y) \Rightarrow C^{\mathcal{I}}(y)\}\\
(\exists R.C)^{\mathcal{I}}(x) &= \sup_{y \in \Delta^{\mathcal{I}}} \{R^{\mathcal{I}}(x,y) \otimes C^{\mathcal{I}}(y)\}\\
\{\alpha_1/o_1, \ldots, \alpha_m/o_m\}^{\mathcal{I}}(x) &= \sup_{i \mid x = o_i^{\mathcal{I}}} \alpha_i\\
(\geq m\ S.C)^{\mathcal{I}}(x) &= \sup_{y_1,\ldots,y_m \in \Delta^{\mathcal{I}}} [(\otimes_{i=1}^{n} \{S^{\mathcal{I}}(x,y_i) \otimes C^{\mathcal{I}}(y_i)\}) \bigotimes (\otimes_{j<k} \{y_j \neq y_k\})]\\
(\leq n\ S.C)^{\mathcal{I}}(x) &= \inf_{y_1,\ldots,y_{n+1} \in \Delta^{\mathcal{I}}} [(\otimes_{i=1}^{n+1} \{S^{\mathcal{I}}(x,y_i) \otimes C^{\mathcal{I}}(y_i)\}) \Rightarrow (\oplus_{j<k} \{y_j = y_k\})]\\
(\exists S.Self)^{\mathcal{I}}(x) &= S^{\mathcal{I}}(x,x)\\
(R^-)^{\mathcal{I}}(x,y) &= R^{\mathcal{I}}(y,x)\\
U^{\mathcal{I}}(x,y) &= 1
\end{aligned}$$

For example, $Z\ \mathcal{SROIQ}$ uses Zadeh logic: minimum t-norm ($\alpha \otimes \beta = \min\{\alpha,\beta\}$), maximum t-conorm ($\alpha \oplus \beta = \max\{\alpha,\beta\}$), Łukasiewicz negation ($\ominus\alpha = 1-\alpha$), and Kleene-Dienes implication ($\alpha \Rightarrow \beta = \max\{1-\alpha,\beta\}$). $G\ \mathcal{SROIQ}$ uses Gödel logic: minimum t-norm ($\alpha \otimes \beta = \min\{\alpha,\beta\}$), maximum t-conorm ($\alpha \oplus \beta = \max\{\alpha,\beta\}$), Gödel negation ($\ominus\alpha = 1$ if $\alpha = 0$, or 0 otherwise) and Gödel implication ($\alpha \Rightarrow \beta = 1$ if $\alpha \leq \beta$, or β otherwise) [8].

A fuzzy interpretation $\mathcal{I}$ satisfies (is a model of):

- $\langle a\!:\!C \bowtie \gamma\rangle$ iff $C^{\mathcal{I}}(a^{\mathcal{I}}) \bowtie \gamma$,
- $\langle (a,b)\!:\!R \bowtie \gamma\rangle$ iff $R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}) \bowtie \gamma$,
- $\langle a \neq b\rangle$ iff $a^{\mathcal{I}} \neq b^{\mathcal{I}}$,
- $\langle a = b\rangle$ iff $a^{\mathcal{I}} = b^{\mathcal{I}}$,
- $\langle C \sqsubseteq D \rhd \gamma\rangle$ iff $\inf_{x \in \Delta^{\mathcal{I}}} \{C^{\mathcal{I}}(x) \Rightarrow D^{\mathcal{I}}(x)\} \rhd \gamma$,
- $\langle R_1 \ldots R_n \sqsubseteq R \rhd \gamma\rangle$ iff $\sup_{x_1 \ldots x_{n+1} \in \Delta^{\mathcal{I}}} \bigotimes [R_1^{\mathcal{I}}(x_1,x_2), \ldots, R_n^{\mathcal{I}}(x_n,x_{n+1})] \Rightarrow R^{\mathcal{I}}(x_1,x_{n+1}) \rhd \gamma$,
- $trans(R)$ iff $\forall x,y \in \Delta^{\mathcal{I}}, R^{\mathcal{I}}(x,y) \geq \sup_{z \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x,z) \otimes R^{\mathcal{I}}(z,y)$,
- $dis(S_1,S_2)$ iff $\forall x,y \in \Delta^{\mathcal{I}}, S_1^{\mathcal{I}}(x,y) = 0$ or $S_2^{\mathcal{I}}(x,y) = 0$,
- $ref(R)$ iff $\forall x \in \Delta^{\mathcal{I}}, R^{\mathcal{I}}(x,x) = 1$,
- $irr(S)$ iff $\forall x \in \Delta^{\mathcal{I}}, S^{\mathcal{I}}(x,x) = 0$,
- $sym(R)$ iff $\forall x,y \in \Delta^{\mathcal{I}}, R^{\mathcal{I}}(x,y) = R^{\mathcal{I}}(y,x)$,
- $asy(S)$ iff $\forall x,y \in \Delta^{\mathcal{I}}$, if $S^{\mathcal{I}}(x,y) > 0$ then $S^{\mathcal{I}}(y,x) = 0$,
- a fuzzy KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R}\rangle$ iff it satisfies each element in $\mathcal{A}$, $\mathcal{T}$ and $\mathcal{R}$.

Irreflexive, transitive and symmetric role axioms are syntactic sugar for every R-implication (and consequently it can be assumed that they do not appear in fuzzy KBs) due to the following equivalences: $irr(S) \equiv \langle \top \sqsubseteq \neg\exists S.Self \geq 1\rangle$, $trans(R) \equiv \langle RR \sqsubseteq R \geq 1\rangle$ and $sym(R) \equiv \langle R \sqsubseteq R^- \geq 1\rangle$.

In the rest of the paper we will only consider fuzzy KB satisfiability, since (as in the crisp case) most inference problems can be reduced to it [9].

3 A Crisp Representation for Fuzzy $\mathcal{SROIQ}$

In this section we show how to reduce a fuzzy Z $\mathcal{SROIQ}$ fuzzy KB into a crisp KB (see [5, 6] for details). The procedure preserves reasoning, in such a way that existing $\mathcal{SROIQ}$ reasoners could be applied to the resulting KB. The basic idea is to create some new crisp concepts and roles, representing the α-cuts of the fuzzy concepts and roles, and to rely on them. Next, some new axioms are added to preserve their semantics, and finally every axiom in the ABox, the TBox and the RBox is represented using these new crisp elements.

Adding new elements. Let $\mathcal{A}^{\mathcal{K}}$ and $\mathcal{R}^{\mathcal{K}}$ be the set of atomic concepts and roles occurring in a fuzzy KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$. The set of the degrees which must be considered for any reasoning task is defined as $N^{\mathcal{K}} = \gamma, 1-\gamma \mid \langle \tau \bowtie \gamma \rangle \in \mathcal{K}$.

Now, for each $\alpha, \beta \in N^{\mathcal{K}}$ with $\alpha \in (0, 1]$ and $\beta \in [0, 1)$, for each $A \in \mathcal{A}^{\mathcal{K}}$ and for each $R_A \in \mathcal{R}^{\mathcal{K}}$, two new atomic concepts $A_{\geq\alpha}, A_{>\beta}$ and two new atomic roles $R_{\geq\alpha}, R_{>\beta}$ are introduced. $A_{\geq\alpha}$ represents the crisp set of individuals which are instance of A with degree higher or equal than α i.e the α-cut of A.

The semantics of these newly introduced atomic concepts and roles is preserved by some terminological and role axioms. For each $1 \leq i \leq |N^{\mathcal{K}}| - 1, 2 \leq j \leq |N^{\mathcal{K}}| - 1$ and for each $A \in \mathcal{A}^{\mathcal{K}}$, $T(N^{\mathcal{K}})$ is the smallest terminology containing these two axioms: $A_{\geq\gamma_{i+1}} \sqsubseteq A_{>\gamma_i}, A_{>\gamma_j} \sqsubseteq A_{\geq\gamma_j}$. Similarly, for each $R_A \in \mathcal{R}^{\mathcal{K}}$, $R(N^{\mathcal{K}})$ contains these axioms: $R_{\geq\gamma_{i+1}} \sqsubseteq R_{>\gamma_i}, R_{>\gamma_i} \sqsubseteq R_{\geq\gamma_i}$.

Example 1. Consider the fuzzy KB $\mathcal{K} = \{\tau\}$, where $\tau = \langle \mathsf{StGenevieveTexasWhite} : \mathsf{WhiteWine} \geq 0.75 \rangle\}$. We have that $N^{\mathcal{K}} = \{0, 0.25, 0.5, 0.75, 1\}$ and $T(N^{\mathcal{K}}) = \{\mathsf{WhiteWine}_{\geq 0.25} \sqsubseteq \mathsf{WhiteWine}_{>0}, \mathsf{WhiteWine}_{>0.25} \sqsubseteq \mathsf{WhiteWine}_{\geq 0.25}, \mathsf{WhiteWine}_{\geq 0.5} \sqsubseteq \mathsf{WhiteWine}_{>0.25}, \mathsf{WhiteWine}_{>0.5} \sqsubseteq \mathsf{WhiteWine}_{\geq 0.5}, \mathsf{WhiteWine}_{\geq 0.5} \sqsubseteq \mathsf{WhiteWine}_{>0.25}, \mathsf{WhiteWine}_{>0.5} \sqsubseteq \mathsf{WhiteWine}_{\geq 0.5}, \mathsf{WhiteWine}_{\geq 0.75} \sqsubseteq \mathsf{WhiteWine}_{>0.5}, \mathsf{WhiteWine}_{>0.75} \sqsubseteq \mathsf{WhiteWine}_{\geq 0.75}, \mathsf{WhiteWine}_{\geq 1} \sqsubseteq \mathsf{WhiteWine}_{>0.75}\}$. □

Mapping fuzzy concepts, roles and axioms. Concept and role expressions are reduced using mapping ρ, as shown in the first part of Table 1. Given a fuzzy concept C, $\rho(C, \geq \alpha)$ is a crisp set containing all the elements which belong to C with a degree greater or equal than α (the other cases are similar). For instance, the 1-cut of the fuzzy concept $\forall \mathsf{madeFromFruit}.(\mathsf{NonSweetFruit} \sqcup \mathsf{SweetFruit})$ is $\rho(\forall \mathsf{madeFromFruit}.(\mathsf{NonSweetFruit} \sqcup \mathsf{SweetFruit}), \geq 1) = \forall \mathsf{madeFromFruit}_{>0}.\mathsf{NonSweetFruit}_{\geq 1} \sqcup \mathsf{SweetFruit}_{\geq 1}$.

Finally, we map the axioms in the ABox, TBox and RBox. Axioms are reduced as shown in the second part of Table 1, where σ maps fuzzy axioms into crisp assertions, and κ maps fuzzy TBox (resp. RBox) axioms into crisp TBox (resp. RBox) axioms. Recall that we are assuming that irreflexive, transitive and symmetric role axioms do not appear in the RBox. For example, assuming $N^{\mathcal{K}} = \{0, 0.25, 0.5, 0.75, 1\}$, the reduction of the fuzzy GCI $\langle \mathsf{Port} \sqsubseteq \mathsf{RedWine} \geq 1 \rangle$ is $\kappa(\langle \mathsf{Port} \sqsubseteq \mathsf{RedWine} \geq 1 \rangle) = \{\mathsf{Port}_{>0} \sqsubseteq \mathsf{RedWine}_{>0}, \mathsf{Port}_{\geq 0.25} \sqsubseteq \mathsf{RedWine}_{\geq 0.25}, \mathsf{Port}_{>0.25} \sqsubseteq \mathsf{RedWine}_{>0.25}, \mathsf{Port}_{\geq 0.5} \sqsubseteq \mathsf{RedWine}_{\geq 0.5}, \mathsf{Port}_{>0.5} \sqsubseteq \mathsf{RedWine}_{>0.5}, \mathsf{Port}_{\geq 0.75} \sqsubseteq \mathsf{RedWine}_{\geq 0.75}, \mathsf{Port}_{>0.75} \sqsubseteq \mathsf{RedWine}_{>0.75}, \mathsf{Port}_{\geq 1} \sqsubseteq \mathsf{RedWine}_{\geq 1}\}$.

Table 1. Mapping of concept and role expressions, and reduction of the axioms. The semantics of $\sqsubseteq_G$ uses Gödel implication, that of $\sqsubseteq_{KD}$ uses Kleene-Dienes implication.

Fuzzy concepts	
$\rho(\top, \rhd\gamma)$	$\top$
$\rho(\top, \lhd\gamma)$	$\bot$
$\rho(\bot, \rhd\gamma)$	$\bot$
$\rho(\bot, \lhd\gamma)$	$\top$
$\rho(A, \rhd\gamma)$	$A_{\rhd\gamma}$
$\rho(A, \lhd\gamma)$	$\neg A_{\neg\lhd\gamma}$
$\rho(\neg C, \bowtie\gamma)$	$\rho(C, \bowtie^- 1-\gamma)$
$\rho(C \sqcap D, \rhd\gamma)$	$\rho(C, \rhd\gamma) \sqcap \rho(D, \rhd\gamma)$
$\rho(C \sqcap D, \lhd\gamma)$	$\rho(C, \lhd\gamma) \sqcup \rho(D, \lhd\gamma)$
$\rho(C \sqcup D, \rhd\gamma)$	$\rho(C, \rhd\gamma) \sqcup \rho(D, \rhd\gamma)$
$\rho(C \sqcup D, \lhd\gamma)$	$\rho(C, \lhd\gamma) \sqcap \rho(D, \lhd\gamma)$
$\rho(\exists R.C, \rhd\gamma)$	$\exists\rho(R, \rhd\gamma).\rho(C, \rhd\gamma)$
$\rho(\exists R.C, \lhd\gamma)$	$\forall\rho(R, \neg\lhd\gamma).\rho(C, \lhd\gamma)$
$\rho(\forall R.C, \{\geq, >\}\gamma)$	$\forall\rho(R, \{>, \geq\}1-\gamma).\rho(C, \{\geq, >\}\gamma)$
$\rho(\forall R.C, \lhd\gamma)$	$\exists\rho(R, \lhd^-1-\gamma).\rho(C, \lhd\gamma)$
$\rho(\{\alpha_1/o_1, \ldots, \alpha_m/o_m\}, \bowtie\gamma)$	$\{o_i \mid \alpha_i \bowtie \gamma, 1 \leq i \leq m\}$
$\rho(\geq m\ S.C, \rhd\gamma)$	$\geq m\ \rho(S, \rhd\gamma).\rho(C, \rhd\gamma)$
$\rho(\geq m\ S.C, \lhd\gamma)$	$\leq m-1\ \rho(S, \neg\lhd\ \gamma).\rho(C, \neg\lhd\ \gamma)$
$\rho(\leq n\ S.C, \{\geq, >\}\ \gamma)$	$\leq n\ \rho(S, \{>, \geq\}\ 1-\gamma).\rho(C, \{>, \geq\}\ 1-\gamma)$
$\rho(\leq n\ S.C, \lhd\gamma)$	$\geq n+1\ \rho(S, \lhd^-\ 1-\gamma).\rho(C, \lhd^-\ 1-\gamma)$
$\rho(\exists S.Self, \rhd\gamma)$	$\exists\rho(S, \rhd\gamma).Self$
$\rho(\exists S.Self, \lhd\gamma)$	$\neg\exists\rho(S, \neg\lhd\gamma).Self$
Fuzzy roles	
$\rho(R_A, \rhd\gamma)$	$R_{A\rhd\gamma}$
$\rho(R_A, \lhd\gamma)$	$\neg R_{A\neg\lhd\gamma}$
$\rho(R^-, \bowtie\gamma)$	$\rho(R, \bowtie\gamma)^-$
$\rho(U, \rhd\gamma)$	U
$\rho(U, \lhd\gamma)$	$\neg U$
Axioms	
$\sigma(\langle a\!:\!C \bowtie\gamma\rangle)$	$\{a\!:\!\rho(C, \bowtie\gamma)\}$
$\sigma(\langle (a,b)\!:\!R \bowtie\gamma\rangle)$	$\{(a,b)\!:\!\rho(R, \bowtie\gamma)\}$
$\sigma(\langle a \neq b\rangle)$	$\{a \neq b\}$
$\sigma(\langle a = b\rangle)$	$\{a = b\}$
$\kappa(C \sqsubseteq_G D \geq \alpha)$	$\bigcup_{\gamma\in N^{fK}\setminus\{0\}\ \mid\ \gamma\leq\alpha}\{\rho(C, \geq\gamma) \sqsubseteq \rho(D, \geq\gamma)\}$ $\bigcup_{\gamma\in N^{fK}\ \mid\ \gamma<\alpha}\{\rho(C, >\gamma) \sqsubseteq \rho(D, >\gamma)\}$
$\kappa(C \sqsubseteq_G D > \beta)$	$\kappa(C \sqsubseteq D \geq \beta) \cup \{\rho(C, >\beta) \sqsubseteq \rho(D, >\beta)\}$
$\kappa(C \sqsubseteq_{KD} D \geq \alpha)$	$\{\rho(C, >1-\alpha) \sqsubseteq \rho(D, \geq\alpha)\ \}$
$\kappa(C \sqsubseteq_{KD} D > \beta)$	$\{\rho(C, \geq 1-\beta) \sqsubseteq \rho(D, >\beta)\ \}$
$\kappa(\langle R_1 \ldots R_n \sqsubseteq_G R \geq \alpha\rangle)$	$\bigcup_{\gamma\in N^{fK}\setminus\{0\}\ \mid\ \gamma\leq\alpha}\{\rho(R_1, \geq\gamma) \ldots \rho(R_n, \geq\gamma) \sqsubseteq \rho(R, \geq\gamma)\}$ $\bigcup_{\gamma\in N^{fK}\ \mid\ \gamma<\alpha}\{\rho(R_1, >\gamma) \ldots \rho(R_n, >\gamma) \sqsubseteq \rho(R, >\gamma)\}$
$\kappa(\langle R_1 \ldots R_n \sqsubseteq_G R > \beta\rangle)$	$\kappa(\langle R_1 \ldots R_n \sqsubseteq R \geq \beta\rangle) \cup$ $\{\rho(R_1, >\beta) \ldots \rho(R_n, >\beta) \sqsubseteq \rho(R, >\beta)\}$
$\kappa(\langle R_1 \ldots R_n \sqsubseteq_{KD} R \geq \alpha\rangle)$	$\{\rho(R_1, >1-\alpha) \ldots \rho(R_n, >1-\alpha) \sqsubseteq \rho(R, \geq\alpha)\}$
$\kappa(\langle R_1 \ldots R_n \sqsubseteq_{KD} R > \beta\rangle)$	$\{\rho(R_1, \geq 1-\beta) \ldots \rho(R_n, \geq 1-\beta) \sqsubseteq \rho(R, >\beta)\}$
$\kappa(dis(S_1, S_2))$	$\{dis(\rho(S_1, >0), \rho(S_2, >0))\}$
$\kappa(ref(R))$	$\{ref(\rho(R, \geq 1))\}$
$\kappa(asy(S))$	$\{asy(\rho(S, >0)\}$

Properties. Summing up, a fuzzy KB $\mathcal{K} = \langle \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$ is reduced into a KB $\texttt{crisp}(\mathcal{K}) = \langle \sigma(\mathcal{A}), T(N^{\mathcal{K}}) \cup \kappa(\mathcal{K}, \mathcal{T}), R(N^{\mathcal{K}}) \cup \kappa(\mathcal{K}, \mathcal{R}) \rangle$. The following theorem shows that the reduction preserves reasoning:

Theorem 1. *A Z $\mathcal{SROIQ}$ fuzzy KB $\mathcal{K}$ is satisfiable iff* $\texttt{crisp}(\mathcal{K})$ *is [6].*

The resulting KB is quadratic because it depends on the number of relevant degrees $|N^{\mathcal{K}}|$, or linear if we assume a fixed set. An interesting property is that the reduction of an ontology can be reused when adding a new axiom. If the new axioms does not introduce new atomic concepts, atomic roles nor a new degree of truth, we just need to add the reduction of the axiom.

4 DeLorean Reasoner

This section describes the prototype implementation of our reasoner, which is called DeLorean (DEscription LOgic REasoner with vAgueNess).

Initially, we developed a first version based on Jena API[2] [6]. This version was developed in Java, using the parser generator JavaCC[3], and DIG 1.1 interface [10] to communicate with crisp DL reasoners. An interesting property is the possibility of using any crisp reasoner thanks to the DIG interface. However, DIG interface does not yet support full $\mathcal{SROIQ}$, so the logic supported by DeLorean was restricted to Z $\mathcal{SHOIN}$ (OWL DL). From a historical point of view, this version was the first reasoner that supported a fuzzy extension of the OWL DL language. It implemented the reduction described in [11], and applied the optimization in the number of new elements and axioms described below.

With the aim of augmenting the expressivity of the logic, in the current version we have changed the subjacent API to OWL API for OWL 2[4] [4]. Now, DeLorean supports both Z $\mathcal{SROIQ}(\mathbf{D})$ and G $\mathcal{SROIQ}(\mathbf{D})$, which correspond to fuzzy versions of OWL 1.1 under Zadeh and Gödel semantics, respectively.

Since DIG interface does not currently allow the full expressivity of OWL 1.1, our solution was to integrate directly DeLorean with a concrete crisp ontology reasoner: Pellet [12], which can be directly used from the current version of the OWL API. This way, the user is free to choose to use either a generic crisp reasoner (restricting the expressivity to $\mathcal{SHOIQ}$) or Pellet with no expressivity limitations. DeLorean is the first reasoner that supports a fuzzy extension of OWL 1.1.

Figure 1 illustrates the architecture of the system:

- The *Parser* reads an input file with a fuzzy ontology and translates it into an internal representation. The point here is that we can use any language to encode the fuzzy ontology, as long as the *Parser* can understand the representation and the reduction is properly implemented. Consequently we will not get into details of our particular choice.

[2] `http://jena.sourceforge.net/`
[3] `https://javacc.dev.java.net`
[4] `http://owlapi.sourceforge.net`

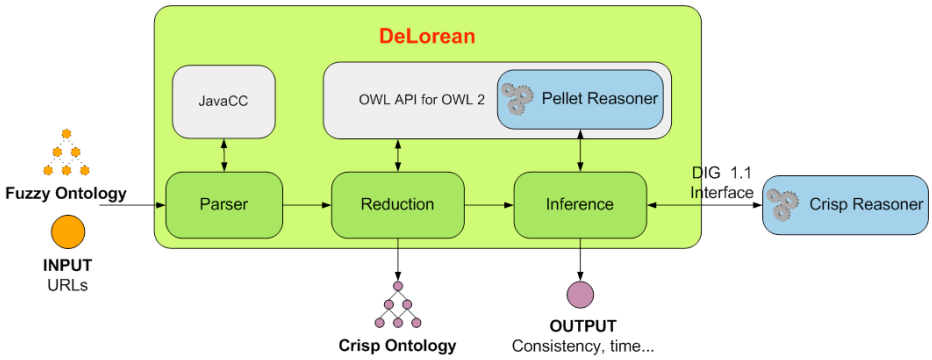


Fig. 1. Architecture of DeLorean reasoner.

In order to make the representation of fuzzy KBs easier, DeLorean also allows the possibility of importing OWL 1.1 ontologies. These (crisp) ontologies are saved into a text file which the user can edit and extend, for example adding membership degrees to the fuzzy axioms or specifying a particular fuzzy operator (Zadeh or Gödel family) for some complex concept.

- The *Reduction* module implements the reduction procedures described in the previous section, building an OWL API model with an equivalent crisp ontology which can be exported to an OWL file. The implementation also takes into account all the optimizations already discussed along this document.
- The *Inference* module tests this ontology for consistency, using either Pellet or any crisp reasoner through the DIG interface. Crisp reasoning does not take into account superfluous elements as we explain below.
- Inputs (the path of the fuzzy ontology) and outputs (the result of the reasoning and the elapsed time) are managed by an *User interface*.

The reasoner implements the following optimizations:

Optimizing the number of new elements and axioms. Previous works use two more atomic concepts $A_{\leq\beta}, A_{<\alpha}$ and some additional axioms $A_{<\gamma_k} \sqsubseteq A_{\leq\gamma_k}, A_{\leq\gamma_i} \sqsubseteq A_{<\gamma_{i+1}}, A_{\geq\gamma_k} \sqcap A_{<\gamma_k} \sqsubseteq \bot, A_{>\gamma_i} \sqcap A_{\leq\gamma_i} \sqsubseteq \bot, \top \sqsubseteq A_{\geq\gamma_k} \sqcup A_{<\gamma_k}, \top \sqsubseteq A_{>\gamma_i} \sqcup A_{\leq\gamma_i}$, $2 \leq k \leq |N^{\mathcal{K}}|$. In [6] it is shown that they are unnecessary.

Optimizing GCI reductions. In some particular cases, the reduction of fuzzy GCIs can be optimized [6]. For example, in *range* role axioms of the form $\langle \top \sqsubseteq \forall R.C \geq 1 \rangle$, *domain* role axioms of the form $\langle \top \sqsubseteq \forall R^{-}.C \geq 1 \rangle$ and *functional* role axioms of the form $\langle \top \sqsubseteq \leq 1\ R.\top \geq 1 \rangle$ we can use that $\kappa(\langle \top \sqsubseteq D \bowtie \gamma \rangle) = \top \sqsubseteq \rho(D, \bowtie \gamma)$. Also, in disjoint concept axioms of the form $\langle C \sqcap D \sqsubseteq \bot \geq 1 \rangle$, we can use that $\kappa(C \sqsubseteq \bot \bowtie \gamma) = \rho(C, > 0) \sqsubseteq \bot$. Furthermore, if the resulting TBox contains $A \sqsubseteq B$, $A \sqsubseteq C$ and $B \sqsubseteq C$, then $A \sqsubseteq C$ is unnecessary since it can be deduced from the other two axioms.

Allowing crisp concepts and roles. Suppose that A is a fuzzy concept. Then, we need $N^{\mathcal{K}} - 1$ concepts of the form $A_{\geq\alpha}$ and another $N^{\mathcal{K}} - 1$ concepts of the form $A_{>\beta}$ to represent it, as well as $2 \cdot (|N^{\mathcal{K}}| - 1) - 1$ axioms to preserve their semantics. Fortunately, in real applications not all concepts and roles will be fuzzy. If A is declared to be crisp, we just need one concept to represent it and no new axioms. The case for fuzzy roles is exactly the same. Of course, this optimization requires some manual intervention.

Reasoning ignoring superfluous elements. Our reduction is designed to promote reusing. For instance, consider the fuzzy KB $\mathcal{K}$ in Example 1. The reduction of $\mathcal{K}$ contains $\sigma(\tau) = \mathsf{StGenevieveTexasWhite} : \mathsf{WhiteWine}_{\geq 0.75}$, but also the axioms in $T(N^{\mathcal{K}})$. It can be seen that the concepts $\mathsf{WhiteWine}_{>0}, \mathsf{WhiteWine}_{\geq 0.25}$, $\mathsf{WhiteWine}_{>0.25}, \mathsf{WhiteWine}_{\geq 0.5}, \mathsf{WhiteWine}_{>0.5}, \mathsf{WhiteWine}_{>0.75}, \mathsf{WhiteWine}_{\geq 1}$ are *superfluous* in the sense that cannot cause a contradiction. Hence, for a satisfiability test of $\mathtt{crisp}(\mathcal{K})$, we can avoid the axioms in $T(N^{\mathcal{K}})$ where they appear.

But please note that if additional axioms are added to $\mathcal{K}$, $\mathtt{crisp}(\mathcal{K})$ will be different and previous superfluous concept and roles may not be superfluous any more. For example, if we want to check if $\mathcal{K} \cup \langle \mathsf{StGenevieveTexasWhite} : \mathsf{WhiteWine} \geq 0.5 \rangle$ is satisfiable, then the concept $\mathsf{WhiteWine}_{\geq 0.5}$ is no longer superfluous. In this case, it is enough to consider $T'(N^{\mathcal{K}}) = \{\mathsf{WhiteWine}_{\geq 0.75} \sqsubseteq \mathsf{WhiteWine}_{\geq 0.5}\}$. The case of atomic roles is similar to that of atomic concepts.

5 Use Case: A Fuzzy Wine Ontology

This section considers a concrete use case, a fuzzy extension of the well-known Wine ontology[5], a highly expressive ontology (in $\mathcal{SHOIN}(\mathbf{D})$). Some metrics of the ontology are shown in the first column of Table 2. In an empirical evaluation of the reductions of fuzzy DLs to crisp DLs, P. Cimiano et al. wrote that "the Wine ontology showed to be completely intractable both with the optimized and unoptimized reduction even using only 3 degrees" [13]. They only considered there what we have called here "optimization of the number of new elements and axioms". We will show that the rest of the optimizations, specially the (natural) assumption that there are some crisp elements, reduce significantly the number of axioms, even if tractability of the reasoning is to be verified.

A fuzzy extension of the ontology. We have defined a fuzzy version of the Wine ontology by adding a degree to the axioms. Given a variable set of degrees $N^{\mathcal{K}}$, the degrees of the truth for fuzzy assertions is randomly chosen in $N^{\mathcal{K}}$. In the case of fuzzy GCIs and RIAs, the degree is always 1 in special GCIs (namely concept equivalences and disjointness, domain, range and functional role axioms) or if there is a crisp element in the left side; otherwise, the degree is 0.5.

In most of the times fuzzy assertions are of the form $\langle \tau \rhd \beta \rangle$ with $\beta \neq 1$. Clearly, this favors the use of elements of the forms $C_{\rhd\beta}$ and $R_{\rhd\beta}$, reducing the number of superfluous concepts. Once again, we are in the worst case from the

[5] `http://www.w3.org/TR/2003/CR-owl-guide-20030818/wine.rdf`

point of view of the size of the resulting crisp ontology. Nonetheless, in practice we will be often able to say that an individual fully belongs to a fuzzy concept, or that two individuals are fully related by means of a fuzzy role.

Crisp concepts and roles. A careful analysis of the fuzzy KB brings about that most of the concepts and the roles should indeed be interpreted as crisp. For example, most of the subclasses of the class Wine refer to a well-defined geographical origin of the wines. For instance, Alsatian wine is a wine which has been produced in the French region of Alsace: AlsatianWine $\equiv$ Wine $\sqcap$ $\exists$locatedAt.{alsaceRegion}. In other applications there could exist examples of fuzzy regions, but this is not our case. Another important number of subclasses of Wine refer to the type of grape used, which is also a crisp concept. For instance, Riesling is a wine which has been produced from Riesling grapes: Riesling $\equiv$ Wine $\sqcap$ $\exists$madeFromGrape.{RieslingGrape}$\sqcap \geq 1$ madeFromGrape.$\top$.

Clearly, there are other concepts with no sharp boundaries (for instance, those derived from the vague terms "dry", "sweet", "white" or"heavy"). The result of our study has identified 50 fuzzy concepts in the Wine ontology, namely: WineColor, RedWine, RoseWine, WhiteWine, RedBordeaux, RedBurgundy, RedTableWine, WhiteBordeaux, WhiteBurgundy, WhiteLoire, WhiteTableWine, WineSugar, SweetWine, SweetRiesling, WhiteNonSweetWine, DryWine, DryRedWine, DryRiesling, DryWhiteWine, WineBody, FullBodiedWine, WineFlavor, WineTaste, LateHarvest, EarlyHarvest, NonSpicyRedMeat, NonSpicyRedMeatCourse, SpicyRedMeat, PastaWithSpicyRedSauce, PastaWithSpicyRedSauceCourse, PastaWithNonSpicyRedSauce, PastaWithNonSpicyRedSauceCourse, SpicyRedMeatCourse, SweetFruit, SweetFruitCourse, SweetDessert, SweetDessertCourse, NonSweetFruit, NonSweetFruitCourse, RedMeat, NonRedMeat, RedMeatCourse, NonRedMeatCourse, PastaWithHeavyCreamSauce, PastaWithLightCreamSauce, Dessert, CheeseNutsDessert, DessertCourse, CheeseNutsDessertCourse, DessertWine.

Furthermore, we identified 5 fuzzy roles: hasColor, hasSugar, hasBody, hasFlavor, and hasWineDescriptor (which is a super-role of the other four).

Measuring the importance of the optimizations. The reduction under Gödel semantics is still to be published [14], so we focus our experimentation in $Z\,\mathcal{SROIQ}$ (omitting the concrete role yearValue), but allowing the use of both Kleene-Dienes and Gödel implications in the semantics of fuzzy GCIs and RIAs.

Table 2 shows some metrics of the crisp ontologies obtained in the reduction of the fuzzy ontology after applying different optimizations.

1. Column "Original" shows some metrics of the original ontology.
2. "None" considers the reduction obtained after applying no optimizations.
3. "(NEW)" considers the reduction obtained after optimizing the number of new elements and axioms.
4. "(GCI)" considers the reduction obtained after optimizing GCI reductions.
5. "(C/S)" considers the reduction obtained after allowing crisp concepts and roles and ignoring superfluous elements.
6. Finally, "All" applies all the previous optimizations.

Table 2. Metrics of the Wine ontology and its fuzzy versions using 5 degrees

	Original	None	(NEW)	(GCI)	(C/S)	All
Individuals	206	206	206	206	206	206
Named concepts	136	2176	486	2176	800	191
Abstract roles	16	128	128	128	51	20
Concept assertions	194	194	194	194	194	194
Role assertions	246	246	246	246	246	246
Inequality assertions	3	3	3	3	3	3
Equality assertions	0	0	0	0	0	0
New GCIs	0	4352	952	4352	1686	324
Subclass axioms	275	1288	1288	931	390	390
Concept equivalences	87	696	696	696	318	318
Disjoint concepts	19	152	152	19	152	19
Domain role axioms	13	104	104	97	104	97
Range role axioms	10	80	80	10	80	10
Functional role axioms	6	48	48	6	48	6
New RIAs	0	136	119	136	34	34
Sub-role axioms	5	40	40	40	33	33
Role equivalences	0	0	0	0	0	0
Inverse role axioms	2	16	16	16	2	2
Transitive role axioms	1	8	8	8	1	1

We have put together the optimizations of crisp and superfluous elements because in this ontology handling superfluous concepts is not always useful, due to the existence of a lot of concept definitions, as we will see in the next example.

Example 2. Consider the fuzzy concept $\mathsf{NonRedMeat}$. Firstly, this concept appears as part of a fuzzy assertion stating that pork is a non read meat: $\sigma(\langle \mathsf{Pork} : \mathsf{NonRedMeat} \rhd \alpha_1 \rangle) = \mathsf{Pork} : \mathsf{NonRedMeat}_{\rhd\alpha_1}$. Secondly, non read meat is declared to be disjoint from read meat: $\kappa(\langle \mathsf{RedMeat} \sqcap \mathsf{NonRedMeat} \sqsubseteq \bot \geq 1 \rangle) = \mathsf{RedMeat}_{>0} \sqcap \mathsf{NonRedMeat}_{>0} \sqsubseteq \bot$. Thirdly, non read meat is a kind of meat: $\kappa(\langle \mathsf{NonRedMeat} \sqsubseteq \mathsf{Meat} \geq \alpha_2 \rangle) = \mathsf{NonRedMeat}_{>0} \sqsubseteq \mathsf{Meat}$. If these were the only occurrences of $\mathsf{NonRedMeat}$, then the reduction would create only two non-superfluous crisp concepts, namely $\mathsf{NonRedMeat}_{>0}$ and $\mathsf{NonRedMeat}_{\rhd\alpha_1}$, and in order to preserve the semantics of them we would need to add just one axiom during the reduction: $\mathsf{NonRedMeat}_{\rhd\alpha_1} \sqsubseteq \mathsf{NonRedMeat}_{>0}$.

However, this is not true because $\mathsf{NonRedMeat}$ appears in the definition of the fuzzy concept $\mathsf{NonRedMeatCourse}$. In fact, $\kappa(\mathsf{NonRedMeatCourse} \equiv \mathsf{MealCourse} \sqcap \forall \mathsf{hasFood}.\mathsf{NonRedMeat})$ introduces non-superfluous crisp concepts for the rest of the degrees in $N^{\mathcal{K}}$. Consequently, for each $1 \leq i \leq |N^{\mathcal{K}}| - 1, 2 \leq j \leq |N^{\mathcal{K}}| - 1$, the reduction adds to $T(N^{\mathcal{K}})$ the following axioms: $\mathsf{NonRedMeat}_{\geq\gamma_{i+1}} \sqsubseteq \mathsf{NonRedMeat}_{>\gamma_i}$; $\mathsf{NonRedMeat}_{>\gamma_j} \sqsubseteq \mathsf{NonRedMeat}_{\geq\gamma_j}$. □

Note that the size of the ABox is always the same, because every axiom in the fuzzy ABox generates exactly one axiom in the reduced ontology.

The number of new GCIs and RIAs added to preserve the semantics of the new elements is much smaller in the optimized versions. In particular, we reduce from 4352 to 324 GCIs (7.44%) and from 136 to 34 RIAs (25%). This shows the importance of reducing the number of new crisp elements and their corresponding axioms, as well as of defining crisp concepts and roles and (to a lesser extent) handling superfluous concepts.

Optimizing GCI reductions turns out to be very useful in reducing the number of disjoint concepts, domain, range and functional role axioms: 152 to 19 (12.5 %), 104 to 97 (93.27 %), 80 to 10 (12.5 %), and 48 to 6 (12.5 %), respectively. In the case of domain role axioms the reduction is not very high because we need an inverse role to be defined in order to apply the reduction, and this happens only in one of the axioms.

Every fuzzy GCI or RIA generates several axioms in the reduced ontology. Combining the optimization of GCI reductions with the definition of crisp concepts and roles reduces the number of new axioms, from 1288 to 390 subclass axioms (30.28 %), from 696 to 318 concept equivalences (45.69 %) and from 40 to 33 sub-role axioms (82.5 %).

Finally, the number of inverse and transitive role axioms is reduced in the optimized version because fuzzy roles interpreted as crisp introduce one inverse or transitive axiom instead of several ones. This allows a reduction from 16 to 2 axioms, and from 8 to 1, respectively, which corresponds to the 12.5 %.

Table 3 shows the influence of the number of degrees on the size of the resulting crisp ontology, as well as on the reduction time (which is shown in seconds), when all the described optimizations are used. The reduction time is small enough to allow to recompute the reduction of an ontology when necessary, thus allowing superfluous concepts and roles in the reduction to be avoided.

Table 3. Influence of the number of degrees in the reduction.

	Crisp	3	5	7	9	11	21
Number of axioms	811	1166	1674	2182	2690	3198	5738
Reduction time	-	0.343	0.453	0.64	0.782	0.859	1.75

6 Conclusions and Future Work

This paper has presented DeLorean, the more expressive fuzzy DL reasoner that we are aware of (it supports fuzzy OWL 1.1), and the optimizations that it implements. Among them, the current version enables the definition of crisp concepts and roles, as well as handling superfluous concepts and roles before applying crisp reasoning. A preliminary evaluation shows that these optimizations help to reduce significantly the size of the resulting ontology. In future work we plan to develop a more detailed benchmark by relying on the hypertableau reasoner HermiT, which seems to outperform other DL reasoners [15], and, eventually, to compare it against other fuzzy DL reasoners.

Acknowledgements

This research has been partially supported by the project TIN2006-15041-C04-01 (Ministerio de Educación y Ciencia). Fernando Bobillo holds a FPU scholarship from Ministerio de Educación y Ciencia. Juan Gómez-Romero holds a scholarship from Consejería de Innovación, Ciencia y Empresa (Junta de Andalucía).

References

1. Patel-Schneider, P.F., Horrocks, I.: OWL 1.1 Web Ontology Language overview (2006) [Online] Available: `http://www.w3.org/Submission/owl11-overview/`.
2. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.F.: The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press (2003)
3. Lukasiewicz, T., Straccia, U.: Managing uncertainty and vagueness in Description Logics for the Semantic Web. Journal of Web Semantics (To appear)
4. Horridge, M., Bechhofer, S., Noppens, O.: Igniting the OWL 1.1 touch paper: The OWL API. In: Proc. of the 3rd International Workshop on OWL: Experiences and Directions (OWLED 2007). Volume 258., CEUR Workshop Proceedings (2007)
5. Straccia, U.: Transforming fuzzy Description Logics into classical Description Logics. In: Proceedings of the 9th European Conference on Logics in Artificial Intelligence (JELIA-04). Volume 3229 of Lecture Notes in Computer Science., Springer-Verlag (2004) 385–399
6. Bobillo, F., Delgado, M., Gómez-Romero, J.: Optimizing the crisp representation of the fuzzy Description Logic $\mathcal{SROIQ}$. In: Proceedings of the 3rd ISWC Workshop on Uncertainty Reasoning for the Semantic Web (URSW 2007). Volume 327., CEUR Workshop Proceedings (2007)
7. Horrocks, I., Kutz, O., Sattler, U.: The even more irresistible $\mathcal{SROIQ}$. In: Proceedings of the 10th International Conference of Knowledge Representation and Reasoning (KR 2006). (2006) 452–457
8. Hájek, P.: Metamathematics of Fuzzy Logic. Kluwer (1998)
9. Straccia, U.: Reasoning within fuzzy Description Logics. Journal of Artificial Intelligence Research **14** (2001) 137–166
10. Bechhofer, S., Möller, R., Crowther, P.: The DIG Description Logic interface: DIG / 1.1. In: Proc. of the 16th Int. Workshop on Description Logics (DL 2003). (2003)
11. Bobillo, F., Delgado, M., Gómez-Romero, J.: A crisp representation for fuzzy $\mathcal{SHOIN}$ with fuzzy nominals and General Concept Inclusions. In: Proceedings of the 2nd ISWC Workshop on Uncertainty Reasoning for the Semantic Web (URSW 2006). Volume 218., CEUR Workshop Proceedings (2006)
12. Sirin, E., Parsia, B., Cuenca-Grau, B., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. Journal of Web Semantics **5**(2) (2007) 51–53
13. Cimiano, P., Haase, P., Ji, Q., Mailis, T., Stamou, G.B., Stoilos, G., Tran, T., Tzouvaras, V.: Reasoning with large A-Boxes in fuzzy Description Logics using DL reasoners: An experimental valuation. In: Proceedings of ARea2008. Volume 350., CEUR Workshop Proceedings (2008)
14. Bobillo, F., Delgado, M., Gómez-Romero, J., Straccia, U.: Fuzzy Description Logics under Gödel semantics. (Submitted)
15. Motik, B., Shearer, R., Horrocks, I.: Optimized reasoning in Description Logics using hypertableaux. In: Proc. of the 21st International Conference on Automated Deduction (CADE-21). Lecture Notes in Artificial Intelligence 4603 (2007) 67–83

Describing and Communicating Uncertainty within the Semantic Web

Matthew Williams[1] (williamw@aston.ac.uk), Lucy Bastin[1], Dan Cornford[1], and Ben Ingram[1]

Knowledge Engineering Group, Aston University, Birmingham, United Kingdom

Abstract. The Semantic Web relies on carefully structured, well defined data to allow machines to communicate and understand one another. In many domains (e.g. geospatial) the data being described contains some uncertainty, often due to bias, observation error or incomplete knowledge. Meaningful processing of this data requires these uncertainties to be carefully analysed and integrated into the process chain. Currently, within the Semantic Web there is no standard mechanism for interoperable description and exchange of uncertain information, which renders the automated processing of such information implausible, particularly where error must be considered and captured as it propagates through a processing sequence. In particular we adopt a Bayesian perspective and focus on the case where the inputs / outputs are naturally treated as random variables.

This paper discusses a solution to the problem in the form of the Uncertainty Markup Language (UncertML). UncertML is a conceptual model, realised as an XML schema, that allows uncertainty to be quantified in a variety of ways: i.e. realisations, statistics and probability distributions. The INTAMAP (INTeroperability and Automated MAPping) project provides a use case for UncertML. This paper demonstrates how observation errors can be quantified using UncertML and wrapped within an Observations & Measurements (O&M) Observation. An interpolation Web Processing Service (WPS) uses the uncertainty information within these observations to influence and improve its prediction outcome. The output uncertainties from this WPS may also be encoded in a variety of UncertML types, e.g. a series of marginal Gaussian distributions, a set of statistics, such as the first three marginal moments, or a set of realisations from a Monte Carlo treatment. Quantifying and propagating uncertainty in this way allows such interpolation results to be consumed by other services. This could form part of a risk management chain or a decision support system, and ultimately paves the way for complex data processing chains in the Semantic Web.

1 Introduction

As the Semantic Web evolves, increasing quantities of data are being formatted to allow distribution, discovery and consumption by machines operating over networks. This approach requires clear conceptualisation of real-world objects

and phenomena, their attributes and relationships, to allow rich datasets to be fully exploited by automated parsers and processes. Uncertainty in measurement (for example of objects' bounds and parameters) is sometimes considered as a part of the metadata taxonomy, but rarely in any significant detail, and currently no uniform standard exists for capturing and communicating the errors and uncertainties which are inherent in almost all real-world datasets. We would argue that, in many cases, data without quantified uncertainty has severely reduced value for analysis and decision making.

Currently, there is a trend in software engineering to move away from tightly coupled legacy systems and towards loosely coupled, interoperable, services [1] based on XML. The Web Services approach, whereby functionality is exposed and consumed over networks, is a particular context where standardised descriptions of capabilities and outputs ensures interoperability, and allows data to be passed sequentially through Services in processing chains. As Semantic Web Services evolve, these descriptions will become richer, but even now there is a need for uncertainty information to be passed between and 'understood' by automated processes. This is especially important where error propagates through a processing sequence — for example, in the case of automatically monitored and interpolated temperature data, where sensor error and random noise in the original measurements can combine with artefacts from the techniques used to characterise and interpolate the data, to produce significant levels of posterior uncertainty. This uncertainty should ideally be explicitly estimated and quantified, either as simple means and variances or as fully-characterised probability distributions, over all inputs, parameters and the final outputs. It is also critical to communicate data uncertainty where the outputs are to be used for decision-making — for example, where national radiation data is used to plan for evacuations after a critical incident. In this case, the uncertainty in predicted radiation at any location might be represented as exceedance probabilities, showing the probability that a critical threshold is exceeded at any location, or as sets of realised samples from the predicted distribution.

The above two examples have a spatial component, and utilise Web Service standards specifically designed to handle geospatial data (for example, the OGC Web Coverage Service, Web Feature Service and Web Processing Service standards). Error propagation in geospatial data and geostatistics has been well documented, particularly in natural resources and decision making contexts [2–4]. However, there is a pressing need, in the context of the Semantic Web, to represent uncertainty far more generically, using a clear and flexible standard which can be incorporated into a variety of existing ontologies and schemata. Our proposal is UncertML, an XML schema designed for communicating uncertainty in an interoperable way, based on a conceptual model which allows data uncertainty to be flexibly represented in combination with any other structured data model, including commonly-used XML schemata such as O&M (Observations and Measurements) and GML (Geography Markup Language). These uncertainty representations currently include sets of summary statistics, marginal or joint distributions, and sets of realisations generated by sampling, and it is

anticipated that they will be extended to other representations such as fuzzy sets.

In order to maintain flexibility and extensibility within UncertML, we have made considerable use of Uniform Resource Identifiers (URIs) in combination with a weak-typed design pattern to allow elements such as statistical distributions and algorithmic sampling techniques to be fully described in dictionaries, rather than encoded as concrete types. These dictionaries could be written in GML (the current option within UncertML), Resource Definition Framework (RDF) or Web Ontology Language (OWL). This paper describes the conceptual model for UncertML, with examples of how one might encode uncertainty in XML, illustrated by examples arising within the INTAMAP project.

2 UncertML Conceptual Model

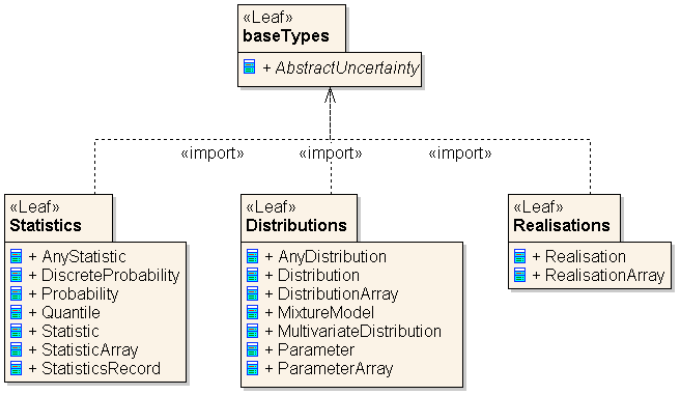


Fig. 1. Package overview of UncertML. Each package contains a set of elements for describing uncertainty.

UncertML is divided into three distinct packages. Each package is tailored toward describing uncertainty using a specific mechanism; either through realisations, statistics or probability distributions. Sections 2.1– 2.3 introduce the conceptual outline for each package and discuss the component types.

2.1 Realisations

In some situations the user may not be able to parametrically describe uncertainties in their data. Typically, in such a situation they may provide a sample, often using Markov Chain Monte Carlo methods, from the probability distribution of the data which allows the uncertainties to be described implicitly.

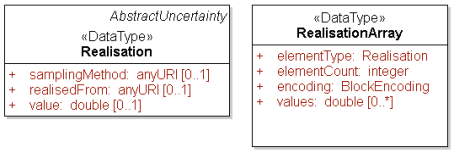


Fig. 2. Realisations can either be encoded singly using the `Realisation` type or aggregated in the `RealisationArray`.

However, a sufficiently large sample of data is required to properly assess uncertainty, therefore efficient encapsulation of large data volumes is an important issue for UncertML.

Realisation. As with all uncertainty types, a `Realisation` inherits the `definition` property from the `AbstractUncertainty` type. In this instance the URI should resolve to a definition of the concept of a realisation. Greater information about any particular realisation may be included with the `realisedFrom` and `samplingMethod` properties. Both these properties are URIs that link to dictionaries, providing information about the distribution the sample was realised from and the method by which the data was sampled, respectively. The final property of a `Realisation` is the `value`. This property contains the actual value of the realisation; i.e. the number generated by the sampling mechanism.

RealisationArray. Working with large arrays of realisations is more common practice, since we are often dealing with joint distributions. UncertML provides the `RealisationArray` type for such purposes. As with all other array types in UncertML, the `RealisationArray` is based around the SWE Common `DataArray` type [5]. The `elementType` property describes the element that is contained within the array, in this instance it is a `Realisation`. The `elementCount` property is an integer value that defines the number of elements, or realisations, within the array. The SWE Common encoding schema [5] provides an efficient and flexible solution to encoding these data arrays. Loosely speaking, the format of the data (binary, ASCII, XML etc) is described in an `encoding` property, while a `values` property contains the data which relates to the `elementType`, or realisations.

2.2 Statistics

There is an extensive range of options available in UncertML for describing 'summary statistics'. Such statistics are used to provide a summary of a random variable, ranging from measures of location (mean, mode, median etc) to measures of dispersion (range, standard deviation, variance etc). While certain statistics (e.g. mean, mode) do not provide any information about uncertainty in isolation, they are often used in conjunction with other statistics (e.g. variance, standard deviation) to provide a concise summary. It should be noted that

providing a location value which is explicitly defined as the mean conveys significantly more information than simply providing a value, since the value might represent many things including the mean, mode, median or even a realisation.

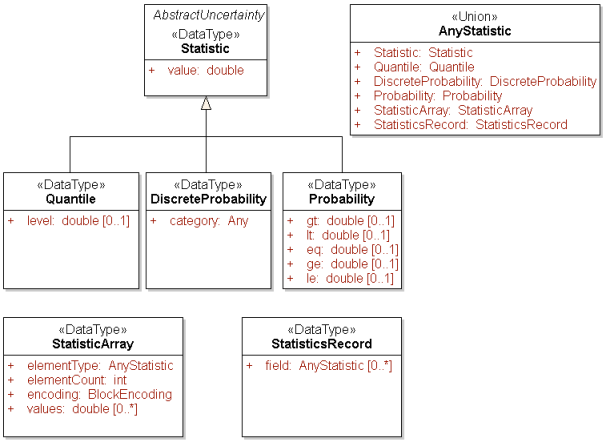


Fig. 3. UncertML model for summary statistics.

The `Statistic` type extends the `AbstractUncertainty` type, inheriting a `definition` property, which in this instance should resolve to a definition of the particular statistic, e.g. mean, variance or mode etc. The other property of a `Statistic` is the `value` which contains the actual value of the statistic, encoded as a double. This generic and concise concept of a statistic allows *most* statistics to be encoded, but for certain statistics more information is required.

One such example is a quantile; here the user needs to know which quantile is being referred to. UncertML provides a specific `Quantile` type which extends the `Statistic` type and provides an additional property, `level`. Continuous and discrete probabilities follow a similar pattern; extending the `Statistic` type with additional properties, and allowing encoding of histograms, exceedance probabilities and discrete random variables.

Due to the soft-typed approach of UncertML all simple statistics will look identical. What separates a 'mean' from a 'median' is the URI (and definition upon resolving) of the `definition` property. Assuming the existence of a dictionary containing definitions of the most common statistics, only the URI is needed in order for an application to 'understand' how to process the data.

StatisticsRecord. A grouped set of summary statistics provides a mechanism for summarising a particular variable's uncertainty. UncertML provides the `StatisticsRecord` type for such use cases. As with all 'record' types within UncertML, the `StatisticsRecord` is closely modelled on the SWE Common `DataRecord` type [5].

A `StatisticsRecord` consists of a number of `field` properties. Each `field` of a `StatisticsRecord` may be a `Statistic`, `Quantile`, `DiscreteProbability`, `Probability`, `StatisticsArray` or `StatisticsRecord`. Grouping statistics into a single structure can be an efficient mechanism for describing the uncertainty surrounding a particular variable. For example, a user might wish to convey the mean value of a variable, and the probability that it exceeds a certain threshold.

StatisticsArray. Arrays of statistics are useful when describing a variable at several locations, or several variables at a given location. The `StatisticsArray` type in UncertML, closely modelled on the `DataArray` of SWE Common, provides such a mechanism. Unlike the `RealisationArray` type, the `elementType` property of a `StatisticsArray` may be any type from within the `AnyStatistic` union. This flexibility allows arrays of single statistics, or an array of `StatisticsRecords` to provide multiple summaries. More complex structures such as two dimensional arrays are also possible.

2.3 Distributions

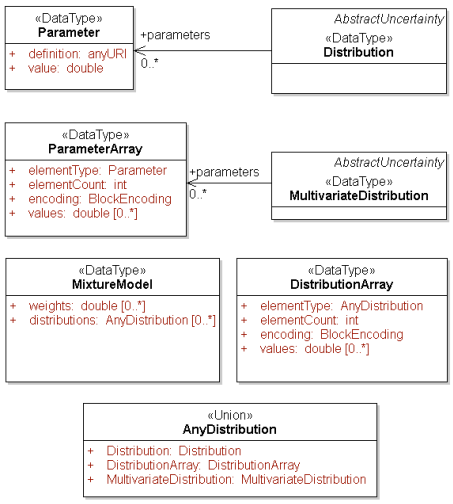


Fig. 4. Distributions in UncertML are encoded using one of the types above.

When the uncertainties of a dataset are more clearly understood, it may be desirable to describe them through the use of probability distributions. The types contained within this section of UncertML are specifically designed to allow a concise encapsulation of *all* probability distributions without sacrificing the simplicity of UncertML.

Distribution. In the simplest case, where a user wishes to describe the probability distribution of a single variable, UncertML provides the `Distribution` type. In the case of distributions the definition may contain both a textual description, and a complex mathematical description of the distribution's functions (for example cumulative distribution function and probability density function).

Complementing the `definition` property is a `parameters` property that contains a number of `Parameter` types. Each `Parameter` of a distribution is not considered to be an uncertainty type, however, it contains a `definition` property which can be used to specify this particular parameter. Each `Parameter` also has a `value` property holding the actual value of that parameter.

It is important to note that the `Distribution` type is not a mechanism for completely describing a probability distribution in terms of its functions, parameters and how they relate to each other; it should be thought of as a mechanism for describing an *instance* of a distribution — which is defined elsewhere. Generating a weak-typed framework such as this allows *any* distribution to be encoded in one generic 'distribution' type. Providing the processing applications understand which distribution is being described (by resolving the URIs) then there exists no need to include any functions. The decision to extract all mathematical functions from the encoding of a distribution has enabled a complex notion such as a Gaussian distribution to be encoded in a simple framework.

DistributionArray. The `DistributionArray` type is similar to both the `StatisticsArray` and `RealisationArray`. However, in this instance the `elementType` property is realised as a type from the `AnyDistribution` union. The rest of the properties remain the same as in the `StatisticsArray` & `RealisationArray`, but one subtle difference exists. Distributions often have numerous parameters that help describe them (e.g. a Gaussian distribution has both a mean and a variance parameter). In this instance the `Distribution` contained within the `elementType` property acts as a form of 'record'. Therefore, when encoding the distributions within the `values` property, care should be taken in interpretation to clearly understand which values refer to which parameter.

MixtureModel. A `MixtureModel` is a specialised form of record. When describing a variable using a mixture of distributions, a specific weight is assigned to each distribution specifying the relative importance of that distribution. This constraint meant that a simple 'DistributionRecord' type would not have been sufficient, so a dedicated `MixtureModel` was designed.

The `distributions` property is equivalent to the `fields` property of a standard record type which may contain a type from the `AnyDistribution` union. The addition of a `weights` property allows a weight (double) to be assigned to each distribution within the `distributions` property.

MultivariateDistribution. The final type provided by UncertML is the `MultivariateDistribution` type. A typical use case for a multivariate (or joint)

distribution is when two variables are correlated. As this scenario (usually) requires the inclusion of a covariance matrix the `DistributionArray` is not sufficient to describe the variable.

A `MultivariateDistribution` is similar to the `Distribution` type, containing both a `definition` and `parameters` property. However, a significant difference is that the `parameters` property of a `MultivariateDistribution` now contains a number of `ParameterArray`s rather than `Parameter` types, due to the fact that multivariate distributions, by definition, always deal with arrays of parameters.

The `ParameterArray` type is similar to all other array types within UncertML, consisting of an `elementType`, `elementCount`, `encoding` and `values` properties. The `elementType` property contains a `Parameter` type which provides a `definition` property. The `values` property then contains all values for that given parameter. A collection of such arrays allows the description of complex joint distributions in an efficient manner.

3 Integrating UncertML into Existing Taxonomies - the INTAMAP Example

The INTAMAP project aims to provide sophisticated functionality across the Web, exposing data cleaning, outlier detection and geostatistical interpolation functions via a Web Processing Service (WPS). The approach prioritises interoperability, with a particular focus on the future consumption of data from automatic monitoring networks via Sensor Observation Services. Our specific case study involves the processing of radiation data from stations across Europe (the European Radiological Data Exchange Platform (EURDEP)) whose spacing, sensitivity and error characteristics are patchy and heterogeneous, and real-time prediction of radiation values to unknown locations between the sampling locations by specialised methods such as Projected Process Kriging [6]. In this context, it is vital that the uncertainty in the monitoring data and the predicted outputs is clearly and fully characterised and communicated.

Several XML schemata exist which are of value in representing this data as it is collected: The Observations & Measurements schema [7] allows results recorded from a sensing instrument to be encoded along with information on the observation time, the specific phenomenon being observed and the spatial extent of the feature of interest. Two important pieces of XML can be used as property values to enrich the information encoded in an Observation. Firstly, an UncertML type, rather than a simple value, can be given as the 'result' property of the Observation, to describe the uncertainty inherent in observed values. This allows a wide range of uncertainty information to be supplied, from a simple marginal mean and variance to a joint distribution with full covariance information. Secondly, the 'procedure' property will typically contain a sensor model encoded in SensorML [5] allowing users a fuller understanding of the physical methods by which the observation was collected.

The INTAMAP WPS interface accepts requests for interpolation, each of which includes a collection of observations, encoded in the O&M schema. The availability of both the error characteristics of a sensor and the observation uncertainty in a machine-parsable form allows us to employ flexible, powerful techniques that take into account the different characteristics and uncertainties, based on a Bayesian framework, to perform the interpolation request. Depending on user preferences made in the request, the result of an interpolation can take several forms. The bulk of the data will be encoded in any one of the uncertainty types within UncertML and additional information may be added by separate schemata. A typical result may consist of a regular grid, possibly defined in GML [8], of some variable defined by a series of Gaussian distributions encoded in UncertML. Figure 5 shows an example of the WPS workflow.

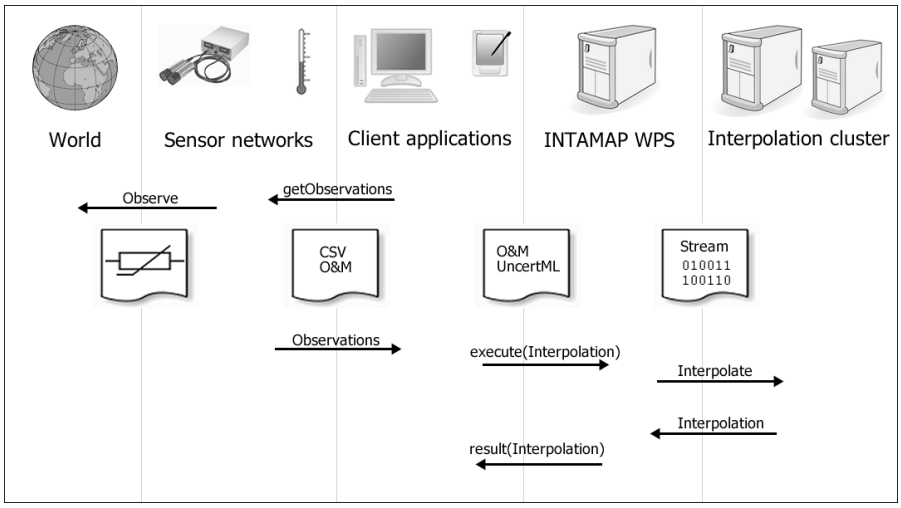


Fig. 5. Example workflow for an interpolation request in the INTAMAP project. A client may obtain observations from multiple sensor systems before submitting them to INTAMAP for processing. Within this Service Oriented Architecture, clients may also be services, forming 'process chains'.

In the INTAMAP example, geographic information (usually in the form of GML) is added to the Observation as a separate layer (see Figure 6). Uncertainty in the spatial location of an Observation could, in theory, be added by nesting UncertML records within an adapted form of GML. Our intention is to make UncertML generic and usable within a large variety of applications, which can replace existing value types with UncertML types.

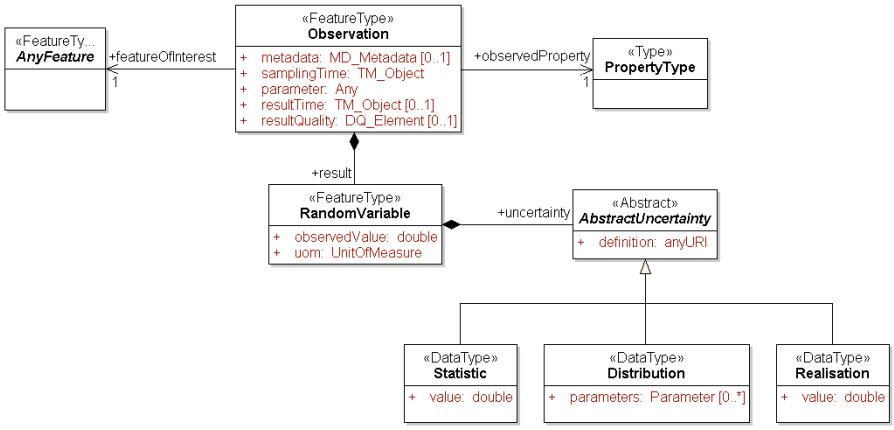


Fig. 6. UncertML can be used in combination with other schemata for specific contexts - here, an environmental measurement with a location is encoded.

4 UncertML in use - adding value to automated environmental measurements

Here we present a geospatial use case from the INTAMAP project which illustrates how, in this context, uncertainty information can be interoperably exchanged and used to improve the outputs of automatic interpolation. A set of radiation measurements for a given area are collected from two overlapping sensor networks with very different error characteristics (see Figure 7): In network A, the error can be characterised as additive positive exponential noise, while measurements from network B tend to vary around the true value according to a Gaussian distribution with known parameters. In practice, each measurement (encoded as an O&M observation) has a location in 2- or 3-D space, but for clarity, only points along a 1-D transect are considered for this illustration. Figure 7a shows the case where an automatic interpolation algorithm has attempted to allow for uncertainty in the measurements, but, in the absence of specific uncertainty information for each observation, has been forced to assume that all measurements have Gaussian noise. Sections of the transect where measurements come from network A are very badly predicted. Using UncertML, a representation of the distinct error for each measurement can be encoded and communicated to the INTAMAP WPS, which can utilise the known error distribution for each specific observation to produce a far more accurate prediction, as shown in Figure 7b. The communication of uncertainty in this example is two-way: for every prediction location, the uncertainty of prediction is returned as an UncertML type (this uncertainty is summarised in the figures as a light-grey confidence envelope). In this case, the uncertainty returned is a Gaussian variance, but a wide variety of metrics and measures can be requested by the user according to the decisions they must make. For example, exceedance probabilities are of value for evacuation planning in environmental emergencies, while

sets of Monte Carlo realisations might be requested as an input to a sensitivity analysis, (which could in turn feasibly be carried out by a separate chained Web Service).

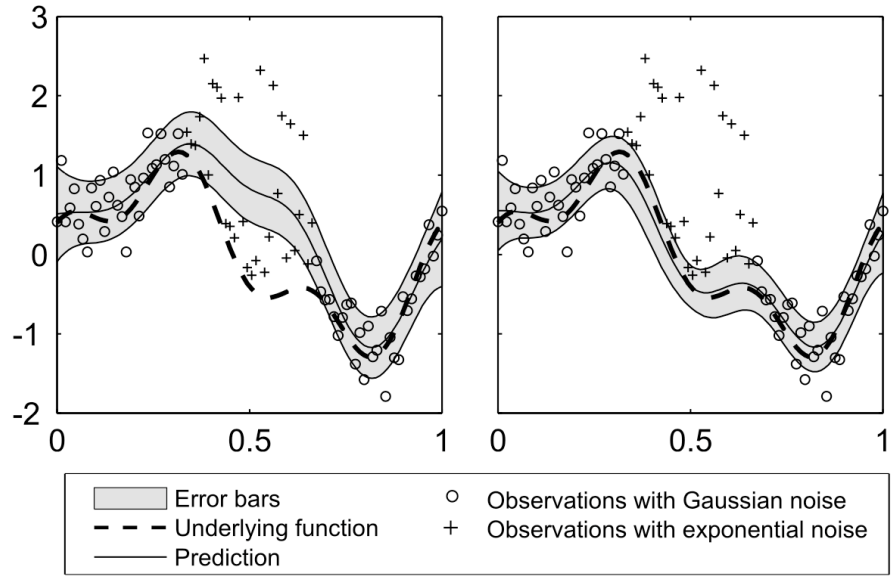


Fig. 7. a) Without specific error information on individual measurements, an automated Bayesian interpolation algorithm is forced to assume Gaussian noise on all measurements, and thus achieves a bad estimate of the true environmental state as shown in the left-hand figure. (b) When observation-specific error characteristics are supplied via UncertML, the performance of the automated interpolator is much improved, as shown on the right.

5 Conclusion

As the Semantic Web and Web Services evolve into a loosely coupled, interoperable framework, sophisticated processing functions such as the geo-processing example described here will become more widely available, along with detailed and rich datasets for analysis. Machine-readable summaries of data quality will become increasingly important, both as a metric on which 'discovered' datasets can be judged for their suitability, and as statistical inputs into analyses where the risks of being wrong need to be quantified. Already, sophisticated users recognise that a single summary of error or precision across an entire dataset (for example, the Root Mean Square registration error commonly supplied as the 'accuracy' metadata on a registered aerial photograph) is rarely representative,

and that excellent use may be made of more detailed error estimates, stratified by time, space, measurement instrument or even by individual measurements. In order to filter and judge the wealth of data which will become available via the Web in coming years, clear and standardised semantic descriptions of data uncertainty are vital, and we believe that UncertML can fulfil this need.

However, for true interoperability, several areas require greater attention. A conceptual model for extending the use of UncertML to random functions is under way, and further work on conditional distributions (or graphical models / belief networks) is envisaged. Other extensions to the UncertML model will include the addition of fuzzy memberships.

Currently, we are undergoing discussions with the Open Geospatial Consortium with the view of making the UncertML specification an official, governed, standard. A working interpolation service using UncertML will be available for testing online shortly. More information and latest developments can be found at the INTAMAP website (http://www.intamap.org).

Acknowledgements

This work is funded by the European Commission, under the Sixth Framework Programme, by Contract 033811 with DG INFSO, action Line IST-2005-2.5.12 ICT for Environmental Risk Management.

References

1. Erl, T.: Service-Oriented Architecture : Concepts, Technology, and Design. Prentice Hall PTR (August 2005)
2. Atkinson, P.M.: Geographical information science: geostatistics and uncertainty. Progress in Physical Geography **23** (1999) 134–142
3. Couclelis, H.: The Certainty of Uncertainty: GIS and the Limits of Geographic Knowledge. Transactions in GIS **7**(2) (2003) 165–175
4. Heuvelink, G.B.M., Goodchild, M.F.: Error Propagation in Environmental Modelling with GIS. CRC Press (1998)
5. Botts, M., Robin, A.: OpenGIS Sensor Model Language (SensorML) Implementation Specification. OpenGIS standard 07-000, Open Geospatial Consortium Inc (July 2007) http://www.opengeospatial.org/standards/sensorml.
6. Ingram, B., Cornford, D., Evans, D.: Fast algorithms for automatic mapping with space–limited covariance functions. Stochastic Environmental Research and Risk Assessment **22**(5) (2008) 661–670
7. Cox, S.: Observations and Measurements – Part 1 - Observation schema. OpenGIS standard 07-022r1, Open Geospatial Consortium Inc (December 2007) http://www.opengeospatial.org/standards/om.
8. Portele, C.: OpenGIS Geography Markup Language (GML) Encoding Standard. OpenGIS standard 07-036, Open Geospatial Consortium Inc (August 2007) http://www.opengeospatial.org/standards/gml.

DL-Media: an Ontology Mediated Multimedia Information Retrieval System

Umberto Straccia and Giulio Visco

ISTI-CNR
Pisa, ITALY,
straccia@isti.cnr.it

Abstract. We outline DL-Media, an ontology mediated multimedia information retrieval system, which combines logic-based retrieval with multimedia feature-based similarity retrieval. An ontology layer is used to define (in terms of a fuzzy DLR-Lite like description logic) the relevant abstract concepts and relations of the application domain, while a content-based multimedia retrieval system is used for feature-based retrieval. We will illustrate its logical model, its architecture, its representation and query language and the preliminary experiments we conducted.

1 Introduction

Multimedia Information Retrieval (MIR) concerns the retrieval of those multimedia objects of a collection that are relevant to a user information need.

In this paper we outline DL-MEDIA [7], an ontology mediated MIR system, which combines logic-based retrieval with multimedia feature-based similarity retrieval. An ontology layer is used to define (in terms of a DLR-Lite like description logic) the relevant abstract concepts and relations of the application domain, while a content-based multimedia retrieval system is used for feature-based retrieval. We will illustrate its logical model, its architecture, its representation and query language and the preliminary experiments we conducted.

Overall, DL-MEDIA lies in the context of *Logic-based Multimedia Information Retrieval* (LMIR) (see [11] for an extensive overview on LMIR literature. A recent work is also *e.g.* [9], see also [10] and [4] for a more complex multimedia ontology model).

2 The DL-MEDIA architecture

In DL-MEDIA, from each multimedia object $o \in \mathbb{O}$ (such as pieces of text, images regions, etc.) we automatically extract low-level features such as text index term weights (object of type text), colour distribution, shape, texture, spatial relationships (object of type image), mosaiced video-frame sequences and time relationships (object of type video). The data are stored in MPEG-7 format [12]. All this pieces of data belong to the *multimedia data layer*. On top of it we have the so-called *ontology layer* in which we define the relevant concepts of our application domain through which we may retrieve the multimedia objects $o \in \mathbb{O}$. In DL-MEDIA this layer consists of an ontology of concepts defined in a fuzzy variant of DLR-Lite like description logic with concrete domains (see Section 3 for details).

The DL-MEDIA architecture has two basic components: the DL-based ontology component and the (feature-based) multimedia retrieval component (see Figure 1).

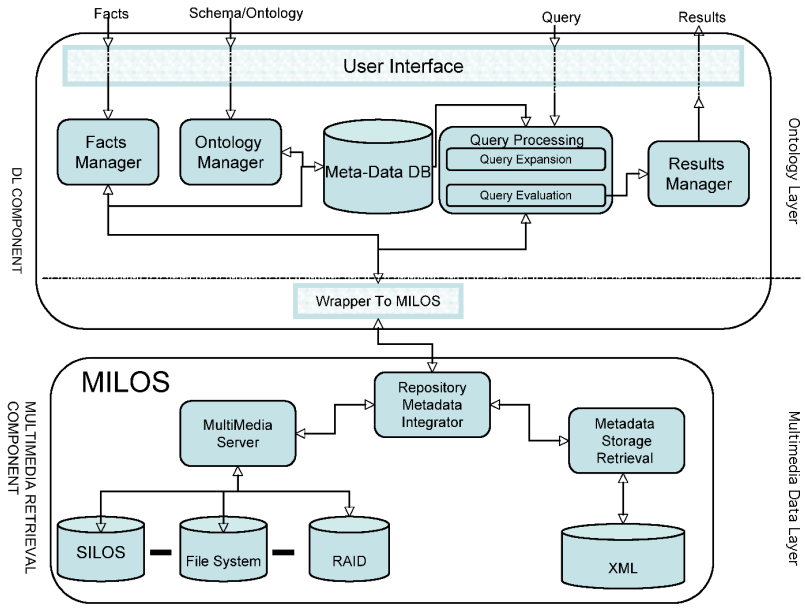


Fig. 1. DL-MEDIA architecture.

The DL-component supports both the definition of the ontology and query answering. In particular, it provides a logical query and representation language, which is an extension of the DL language DLR-Lite [6, 15, 14, 16] without negation (see Section 3 for details).

The (feature-based) multimedia retrieval component, supports the retrieval of text and images based on low-level feature indexing. Specifically, we rely on our MIR system MILOS [1]. MILOS (Multimedia Content Management System) is a general purpose software component that supports the storage and content-based retrieval of any multimedia documents whose descriptions are provided by using arbitrary metadata models represented in XML. MILOS is flexible in the management of documents containing different types of data and content descriptions; it is efficient and scalable in the storage and content-based retrieval of these documents [1–3]. In addition to support XML query language standards such as XPath and XQuery, MILOS offers advanced multimedia search and indexing functionality with new operators that deal with approximate match and ranking of XML and multimedia data (see the MILOS web page for more about it). Approximate match of multimedia data is based on metric spaces theory [17].

The query answering procedure is as follows: a user submits a conceptual query (a conjunctive query) to the the DL-component. The DL-component will then use the

[1] `http://milos.isti.cnr.it/`

ontology to *reformulate* the initial query into one or several queries to be submitted to MILOS (that acts as a Web Service), which then provides back the top-k answers for each of the issued queries. The ranked lists will then be merged into one final top-k result list and displayed to the user.

3 The DL-MEDIA query and representation language

For computational reasons the particular logic DL-MEDIA adopts is based on an extension of the DLR-Lite [6] Description Logic (DL) [5] without negation. The DL will be used in order to define the relevant abstract concepts and relations of the application domain. On the other hand, conjunctive queries will be used to describe the information needs of a user. The DL-MEDIA logic extends DLR-Lite by enriching it with build-in predicates allowing to address three categories of retrieval: feature-based, semantic-based and their combination.

DL-MEDIA **syntax**. DL-MEDIA supports concrete domains with specific predicates on it. The *concrete predicates* that DL-MEDIA allows are not only relational predicates such as $([i] \leq 1500)$ (*e.g.* the value of the i-th column is less or equal than 1500), but also similarity predicates such as $([i]\ simTxt\ 'logic, image, retrieval')$, which given a piece of text x appearing in the i-th column of a tuple returns the system's degree (in $[0, 1]$) of being x about the keywords 'logic, image, retrieval' (keyword-based search).

Formally, a *concrete domain* in DL-MEDIA is a pair $\langle \Delta_{\mathtt{D}}, \Phi_{\mathtt{D}} \rangle$, where $\Delta_{\mathtt{D}}$ is an interpretation domain and $\Phi_{\mathtt{D}}$ is the set of *domain predicates* d with a predefined arity n and an interpretation $d^{\mathtt{D}}: \Delta_{\mathtt{D}}^n \rightarrow [0, 1]$ (see also [13]). The list of the specific domain predicates is presented below.

DL-MEDIA allows to specify the ontology by relying on axioms. Consider an alphabet of n-ary relation symbols (denoted R) and an alphabet of unary relations, called *atomic concepts* (and denoted A). A DL-MEDIA *ontology* $\mathcal{O}$ consists of a set of axioms. An *axiom* is of the form

$$Rl_1 \sqcap \ldots \sqcap Rl_m \sqsubseteq Rr\ ,$$

where $m \geq 1$, all Rl_i and Rr have the same arity and where each Rl_i is a so-called *left-hand relation* and Rr is a *right-hand relation*. They have the following syntax ($h \geq 1$):

$$\begin{array}{lll} Rr & \longrightarrow & A \mid \exists[i_1, \ldots, i_k]R \\ Rl & \longrightarrow & A \mid \exists[i_1, \ldots, i_k]R \mid \exists[i_1, \ldots, i_k]R.(Cond_1 \sqcap \ldots \sqcap Cond_h) \\ Cond & \longrightarrow & ([i] \leq v) \mid ([i] < v) \mid ([i] \geq v) \mid ([i] > v) \mid ([i] = v) \mid ([i] \neq v) \mid \\ & & ([i]\ simTxt'k_1, \ldots, k'_n) \mid ([i]\ simImgURN) \end{array}$$

where A is an atomic concept, R is an n-ary relation with $1 \leq i_1, i_2, \ldots, i_k \leq n$, $1 \leq i \leq n$ and v is a value of the concrete interpretation domain of the appropriate type.

Informally, $\exists[i_1, \ldots, i_k]R$ is the projection of the relation R on the columns $i_1, \ldots, i_k$ (the order of the indexes matters). Hence, $\exists[i_1, \ldots, i_k]R$ has arity k.

On the other hand, $\exists[i_1, \ldots, i_k]R.(Cond_1 \sqcap \ldots \sqcap Cond_l)$ further restricts the projection $\exists[i_1, \ldots, i_k]R$ according to the conditions specified in $Cond_i$. For instance, $([i] \leq v)$ specifies that the values of the i-th column have to be less or equal than the value v. So, *e.g.* suppose we have a relation $\mathsf{Person}(\mathsf{firstname}, \mathsf{lastname}, \mathsf{age}, \mathsf{email}, \mathsf{sex})$ then

$$\exists[2, 4]\mathsf{Person}.(([3] \geq 25))$$

corresponds to the set of tuples $\langle \mathsf{lastname}, \mathsf{email} \rangle$ such that the person's age is equal or greater than 25. Instead, $([i]\, simTxt'k_1 \ldots k'_n)$ evaluates the degree of being the text of the i-th column similar to the list of keywords $k_1 \ldots k_n$, while $([i]\, simImg\, URN)$ returns the system's degree of being the image identified by the i-th column similar to the object o identified by the URN (*Uniform Resource Name* [2]). For instance, the following are axioms:

$$\exists[2,3]\mathsf{Person} \sqsubseteq \exists[1,2]\mathsf{hasAge}$$
$$\exists[2,4]\mathsf{Person} \sqsubseteq \exists[1,2]\mathsf{hasEmail}$$
$$\exists[2,1,4]\mathsf{Person}.(([3] \geq 18) \sqcap ([5] =' \mathsf{male}')) \sqsubseteq \exists[1,2,3]\mathsf{AdultMalePerson}$$

Note that in the last axiom, we require that the age is greater or equal than 18 and the gender is female. This axiom defines the relation $\mathsf{AdultMalePerson}(\mathsf{lastname}, \mathsf{firstname}, \mathsf{email})$. Examples axioms involving similarity predicates are,

$$(\exists[1]\mathsf{ImageDescr}.(([2]\, \mathsf{simImg}\, \mathsf{urn1}))) \sqcap (\exists[1]\mathsf{Tag}.(([2] = \mathsf{sunrise}))) \sqsubseteq \mathsf{Sunrise\_On\_Sea} \quad (1)$$
$$\exists[1]\mathsf{Title}.([2]\, \mathsf{simTxt}\, '\mathsf{lion}') \sqsubseteq \mathsf{Lion} \quad (2)$$

where $urn1$ identifies the image in Fig. 2. The former axiom (axiom 1) assumes that we have an ImageDescr relation, whose first column is the application specific image identifier and the second column contains the image URN. We use also a binary relation Tag. Then, this axiom (informally) states that an image similar to the image depicted in Fig. 2 with a tag labelled 'sunrise' is about a Sunrise_On_Sea (to a system computed degree in $[0, 1]$). Similarly, in axiom (2) we assume that an image is annotated with a

Fig. 2. Sun rise

metadata format, *e.g.* MPEG-7, the attribute Title is seen as a binary relation, whose first column is the identifier of the metadata record, and the second column contains the title (piece of text) of the annotated image. Then, this axiom (informally) states that an image whose metadata record contains an attribute Title which is about 'lion' is about a Lion.

Concerning queries, a DL-MEDIA *query* consists of a conjunctive query of the form

$$q(\mathbf{x}) \leftarrow R_1(\mathbf{z}_1) \wedge \ldots \wedge R_l(\mathbf{z}_l)\ ,$$

where q is an n-ary predicate, every R_i is an n_i-ary predicate, $\mathbf{x}$ is a vector of variables, and every $\mathbf{z_i}$ is a vector of constants, or variables. We call $q(\mathbf{x})$ its *head* and $R_1(\mathbf{z}_1) \wedge \ldots, \wedge R_l(\mathbf{z}_l)$ its *body*. $R_i(\mathbf{z_i})$ may also be a concrete unary predicate of the form $(z \leq v), (z < v), (z \geq v), (z > v), (z = v), (z \neq v), (z\, simTxt'k_1, \ldots, k'_n), (z\, simImg\, URN)$,

[2] http://en.wikipedia.org/wiki/Uniform_Resource_Name

where z is a variable, v is a value of the appropriate concrete domain, k_i is a keyword and URN is an URN. Example queries are:

$\mathsf{q}(\mathsf{x}) \leftarrow \mathsf{Sunrise\_On\_Sea}(\mathsf{x})$
// find objects about a sunrise on the sea

$\mathsf{q}(\mathsf{x}) \leftarrow \mathsf{CreatorName}(\mathsf{x}, \mathsf{y}) \wedge (\mathsf{y} =' \mathsf{paolo}') \wedge \mathsf{Title}(\mathsf{x}, \mathsf{z}), (\mathsf{z}\, \mathsf{simTxt}\, '\mathsf{tour}')$
// find images made by Paolo whose title is about 'tour'

$\mathsf{q}(\mathsf{x}) \leftarrow \mathsf{ImageDescr}(\mathsf{x}, \mathsf{y}) \wedge (\mathsf{y}\, \mathsf{simImg}\, \mathsf{urn2})$
// find images similar to a given image identified by $urn2$

$\mathsf{q}(\mathsf{x}) \leftarrow \mathsf{ImageObject}(\mathsf{x}) \wedge \mathsf{isAbout}(\mathsf{x}, \mathsf{y}_1) \wedge \mathsf{Car}(\mathsf{y}_1) \wedge \mathsf{isAbout}(\mathsf{x}, \mathsf{y}_2) \wedge \mathsf{Racing}(\mathsf{y}_2)$
// find image objects about cars racing

We note that a query may also be written as

$$q(\mathbf{x}) \leftarrow \exists \mathbf{y} \phi(\mathbf{x}, \mathbf{y}) \ ,$$

where $\phi(\mathbf{x}, \mathbf{y})$ is $R_1(\mathbf{z}_1) \wedge \ldots \wedge R_l(\mathbf{z}_l)$ and no variable in $\mathbf{y}$ occurs in $\mathbf{x}$ and vice-versa. Here, $\mathbf{x}$ are the so-called *distinguished variables*, while $\mathbf{y}$ are the so-called *non distinguished variables*, which are existentially quantified.

For a query atom q, we will write $\langle q(\mathbf{c}), s \rangle$ to denote that the tuple $\mathbf{c}$ is instance of the query atom q to degree at least s.

DL-MEDIA **semantics**. From a semantics point of view, DL-MEDIA is based on mathematical fuzzy logic [8] as the underlying MIR system MILOS is based on fuzzy aggregation operators to combine the similarity degrees among low-level image and textual features. Additionally, the DL-component allows for low data-complexity reasoning (LogSpace).

Given a concrete domain $\langle \Delta_{\mathtt{D}}, \Phi_{\mathtt{D}} \rangle$, an *interpretation* $\mathcal{I} = \langle \Delta, \cdot^{\mathcal{I}} \rangle$ consists of a *fixed infinite domain* Δ, containing $\Delta_{\mathtt{D}}$, and an *interpretation function* $\cdot^{\mathcal{I}}$ that maps

- every atom A to a function $A^{\mathcal{I}}: \Delta \rightarrow [0, 1]$
- maps an n-ary predicate R to a function $R^{\mathcal{I}}: \Delta^n \rightarrow [0, 1]$
- constants to elements of Δ such that $a^{\mathcal{I}} \neq b^{\mathcal{I}}$ if $a \neq b$ (unique name assumption).

Intuitively, rather than being an expression (*e.g.* $R(\mathbf{c})$) either true or false in an interpretation, it has a degree of truth in $[0, 1]$. So, given a constant c, $A^{\mathcal{I}}(c)$ determines to which degree the individual c is an instance of atom A. Similarly, given an n-tuple of constants $\mathbf{c}$, $R^{\mathcal{I}}(\mathbf{c})$ determines to which degree the tuple $\mathbf{c}$ is an instance of the relation R.

We also assume to have one object for each constant, denoting exactly that object. In other words, we have standard names, and we do not distinguish between the alphabet of constants and the objects in Δ. Furthermore, we assume that the relations have a typed signature and the interpretations have to agree on the relation's type. For instance, the second argument of the Title relation (see axiom 2) is of type String and any interpretation function requires that the second argument of $\mathsf{Title}^{\mathcal{I}}$ is of type String. To the easy of presentation, we omit the formalization of this aspect and leave it at the intuitive level.

In the following, we use $\mathbf{c}$ to denote an n-tuple of constants, and $\mathbf{c}[i_1, ..., i_k]$ to denote the $i_1, \ldots, i_k$-th components of $\mathbf{c}$. For instance, $(a, b, c, d)[3, 1, 4]$ is (c, a, d).

Concerning concrete comparison predicates, the interpretation function $\cdot^{\mathcal{I}}$ has to satisfy

$$([i] \leq v)^{\mathcal{I}}(\mathbf{c}') = \begin{cases} 1 & \text{if} \quad \mathbf{c}'[i] \leq v \\ 0 & \text{otherwise} \end{cases}$$

and similarly for the other comparison constructs, $([i] < v), ([i] \geq v), ([i] > v)$ and $([i] = v) \mid ([i] \neq v)$.

Concerning the concrete similarity predicates, the interpretation function $\cdot^{\mathcal{I}}$ has to satisfy

$$([i]\, simTxt\,' k_1, \ldots, k_n')^{\mathcal{I}}(\mathbf{c}') = simTxt^{\mathtt{D}}(\mathbf{c}'[i],' k_1, \ldots, k_n') \in [0,1]$$
$$([i]\, simImg\, URN)^{\mathcal{I}}(\mathbf{c}') = simImg^{\mathtt{D}}(\mathbf{c}'[i], URN) \in [0,1]\ .$$

where $simTxt^{\mathtt{D}}$ and $simImg^{\mathtt{D}}$ are the textual and image similarity predicates supported by the underlying MIR system MILOS.

Concerning axioms, as in an interpretation each $Rl_i(\mathbf{c})$ has a degree of truth, we have to specify how to combine them to determine the degree of truth of the conjunction $Rl_1 \sqcap \ldots \sqcap Rl_m$. Usually, in fuzzy logic one uses a so-called T-norm $\otimes$ to combine the truth of "conjunctive" expressions [3] (see [8]). Some typical T-norms are

$x \otimes y = \min(x, y)$	Gödel conjunction
$x \otimes y = \max(x + y - 1, 0)$	Łukasiewicz conjunction
$x \otimes y = x \cdot y$	Product conjunction .

In DL-MEDIA, to be compliant with the underlying MILOS system, the T-norm is fixed to be Gödel conjunction.

The interpretation function $\cdot^{\mathcal{I}}$ has to satisfy: for all $\mathbf{c} \in \Delta^k$ and n-ary relation R:

$$(\exists[i_1, \ldots, i_k]R)^{\mathcal{I}}(\mathbf{c}) = \sup_{\mathbf{c}' \in \Delta^n,\, \mathbf{c}'[i_1,\ldots,i_k]=\mathbf{c}} R^{\mathcal{I}}(\mathbf{c}')$$

$$(\exists[i_1, \ldots, i_k]R.(Cond_1 \sqcap \ldots \sqcap Cond_l))^{\mathcal{I}}(\mathbf{c}) = \sup_{\mathbf{c}' \in \Delta^n,\, \mathbf{c}'[i_1,\ldots,i_k]=\mathbf{c}} \min(R^{\mathcal{I}}(\mathbf{c}'), Cond_1{}^{\mathcal{I}}(\mathbf{c}'), \ldots, Cond_l{}^{\mathcal{I}}(\mathbf{c}'))$$

Some explanation is in place. Consider $(\exists[i_1, \ldots, i_k]R)$. Informally, from a classical semantics point of view, $(\exists[i_1, \ldots, i_k]R)$ is the projection of the relation R over the columns $i_1, \ldots, i_k$ and, thus, corresponds to the set of tuples

$$\{\mathbf{c} \mid \exists \mathbf{c}' \in R \text{ s.t. } \mathbf{c}'[i_1, \ldots, i_k] = \mathbf{c}\}\ .$$

Note that for a fixed tuple $\mathbf{c}$ there may be several tuples $\mathbf{c}' \in R$ such that $\mathbf{c}'[i_1, \ldots, i_k] = \mathbf{c}$. Now, if we switch to fuzzy logic, for a fixed tuple $\mathbf{c}$ and interpretation $\mathcal{I}$, each of the previous mentioned $\mathbf{c}'$ is instance of R to a degree $R^{\mathcal{I}}(\mathbf{c}')$. It is usual practice in mathematical fuzzy logic to consider the supremum among these degrees (the existential is interpreted as supremum), which motivates the expression $\sup_{\mathbf{c}' \in \Delta^n,\, \mathbf{c}'[i_1,\ldots,i_k]=\mathbf{c}} R^{\mathcal{I}}(\mathbf{c}')$. The argument is similar for the $\exists[i_1, \ldots, i_k]R.(Cond_1 \sqcap \ldots \sqcap Cond_l)$ construct except that we consider also the additional conditions as conjuncts.

Now given an interpretation $\mathcal{I}$, the notion of $\mathcal{I}$ *is a model of* (*satisfies*) an axiom τ, denoted $\mathcal{I} \models \tau$, is defined as follows:

$$\mathcal{I} \models Rl_1 \sqcap \ldots \sqcap Rl_m \sqsubseteq Rr \text{ iff for all } \mathbf{c} \in \Delta^n, \min(Rl_1{}^{\mathcal{I}}(\mathbf{c}), \ldots, Rl_l{}^{\mathcal{I}}(\mathbf{c})) \leq Rr^{\mathcal{I}}(\mathbf{c})\ ,$$

[3] Given truth degrees x and y, the conjunction of x and y is $x \otimes y$. $\otimes$ has to be symmetric, associative, monotone in its arguments and such that $x \otimes 1 = x$.

where we assume that the arity of Rr and all Rl_i is n. An interpretation $\mathcal{I}$ *is a model of* (*satisfies*) an ontology $\mathcal{O}$ iff it satisfies each element in it.

Concerning queries, an interpretation $\mathcal{I}$ *is a model of* (*satisfies*) a query q the form $q(\mathbf{x}) \leftarrow \exists \mathbf{y} \phi(\mathbf{x}, \mathbf{y})$, denoted $\mathcal{I} \models q$, iff for all $\mathbf{c} \in \Delta^n$:

$$q^{\mathcal{I}}(\mathbf{c}) \geq \sup_{\mathbf{c}' \in \Delta \times \cdots \times \Delta} \phi^{\mathcal{I}}(\mathbf{c}, \mathbf{c}') \, ,$$

where $\phi^{\mathcal{I}}(\mathbf{c}, \mathbf{c}')$ is obtained from $\phi(\mathbf{c}, \mathbf{c}')$ by replacing every R_i by $R_i^{\mathcal{I}}$, and Gödel conjunction is used to combine all the truth degrees $R_i^{\mathcal{I}}(\mathbf{c}'')$ in $\phi^{\mathcal{I}}(\mathbf{c}, \mathbf{c}')$. Furthermore, we say that an interpretation $\mathcal{I}$ *is a model of* (*satisfies*) $\langle q(\mathbf{c}), s \rangle$, denoted $\mathcal{I} \models \langle q(\mathbf{c}), s \rangle$, iff $q^{\mathcal{I}}(\mathbf{c}) \geq s$.

We say $\mathcal{O}$ *entails* $q(\mathbf{c})$ to degree s, denoted $\mathcal{O} \models \langle q(\mathbf{c}), s \rangle$, iff each model $\mathcal{I}$ of $\mathcal{O}$ is a model of $\langle q(\mathbf{c}), s \rangle$. The *greatest lower bound* of $q(\mathbf{c})$ relative to $\mathcal{O}$ is

$$glb(\mathcal{O}, q(\mathbf{c})) = \sup\{s \mid \mathcal{O} \models \langle q(\mathbf{c}), s \rangle\} \, .$$

As now each answer to a query has a degree of truth, the basic inference problem that is of interest in DL-MEDIA is the top-k retrieval problem, formulated as follows. Given $\mathcal{O}$ and a query with head $q(\mathbf{x})$, retrieve k tuples $\langle \mathbf{c}, s \rangle$ that instantiate the query predicate q with maximal degree, and rank them in decreasing order relative to the degree s, denoted

$$ans_k(\mathcal{O}, q) = \mathtt{Top}_k\{\langle \mathbf{c}, s \rangle \mid s = glb(\mathcal{O}, q(\mathbf{c}))\} \, .$$

From a query answering point of view, the DL-MEDIA system extends the DL-Lite/DLR-Lite reasoning method [6] to the fuzzy case. The algorithm is an extension of the one described in [6, 15, 14]). Roughly, given a query $q(\mathbf{x}) \leftarrow R_1(\mathbf{z}_1) \wedge \ldots \wedge R_l(\mathbf{z}_l)$,

1. by considering $\mathcal{O}$, the user query q is *reformulated* into a set of conjunctive queries $r(q, \mathcal{O})$. Informally, the basic idea is that the reformulation procedure closely resembles a top-down resolution procedure for logic programming, where each axiom is seen as a logic programming rule. For instance, given the query $q(x) \leftarrow A(x)$ and suppose that $\mathcal{O}$ contains the axioms $B_1 \sqsubseteq A$ and $B_2 \sqsubseteq A$, then we can reformulate the query into two queries $q(x) \leftarrow B_1(x)$ and $q(x) \leftarrow B_2(x)$, exactly as it happens for top-down resolution methods in logic programming;
2. from the set of reformulated queries $r(q, \mathcal{O})$ we remove redundant queries;
3. the reformulated queries $q' \in r(q, \mathcal{O})$ are translated to MILOS queries and evaluated. The query evaluation of each MILOS query returns the top-k answer set for that query;
4. all the $n = |r(q, \mathcal{O})|$ top-k answer sets have to be merged into the unique top-k answer set $ans_k(\mathcal{O}, q)$. As $k \cdot n$ may be large, we apply the *Disjunctive Threshold Algorithm* (DTA, see [15] for the details) to merge all the answer sets.

4 DL-MEDIA at work

A prototype of the DL-MEDIA system has been implemented. The main interface is shown in Fig. 3.

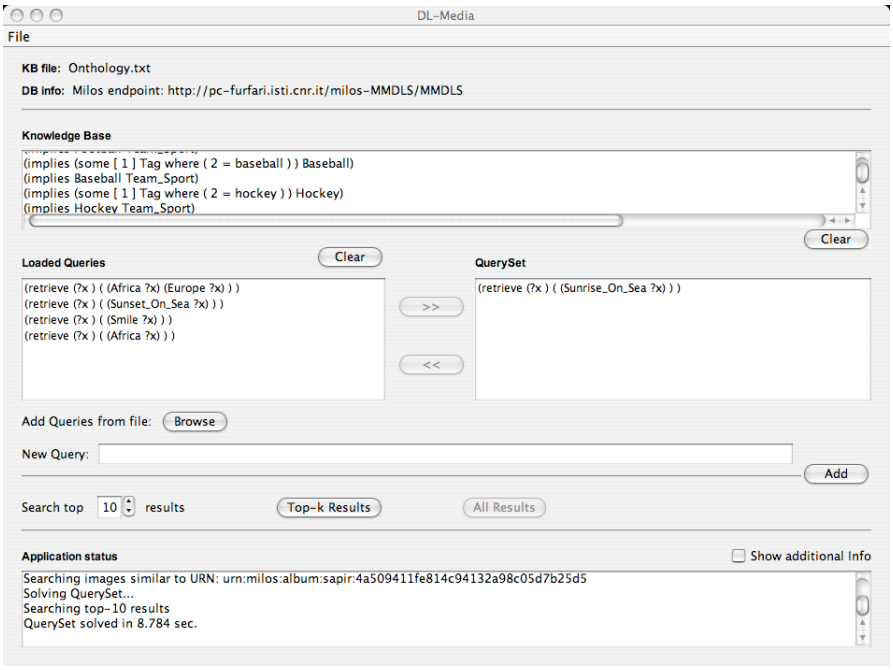


Fig. 3. DL-MEDIA main interface.

In the upper pane, the currently loaded ontology component $\mathcal{O}$ is shown. Below it and to the right, the current query is shown ("find images about sunrises on the sea", we also do not report here the concrete syntax of the DL-MEDIA DL).

So far, in DL-MEDIA, given a query, it will be transformed, using the ontology, into several queries (according to the query reformulation step described above) and then the conjunctive queries are transformed into appropriate queries (this component is called wrapper) in order to be submitted to the underlying database and multimedia engine. To support the query rewriting phase, DL-MEDIA allows also to write *schema mapping* rules, which map *e.g.* a relation name R into the concrete name of a XML tag (see Fig. 4) and excerpt of the metadata format is shown in Fig. 5.

```
(map-role ImageDescr     /Mpeg7/Description/MultimediaContent/Image (docID[string] image[string]))
(map-role MediaUri       /MediaLocator/MediaUri (docID[string] mediauri[string]))
(map-role Title          /photo/title (docID[string] title[string]))
(map-role OwnerUserName  /photo/owner/@username (docID[string] creator-username[string]))
(map-role OwnerRealName  /photo/owner/@realname (docID[string] creator-realname[string]))
(map-role OwnerLocation  /photo/owner/@location (docID[string] creator-location[string]))
(map-role Mpeg7Content   /Mpeg7/Description (docID[string] data[string]))
(map-role Location       /photo/owner/@location (docID[string] location[string]))
(map-role CreationTime   /photo/dates/@taken (docID[string] date[string]))
(map-role Description    /photo/description (docID[string] descr[string]))
(map-role Tag            /photo/tags/tag  (docID[string] tag[string]))
(map-role Comment        /photo/comments/comment  (docID[string] comment[string]))
```

Fig. 4. DL-MEDIA mapping rules.

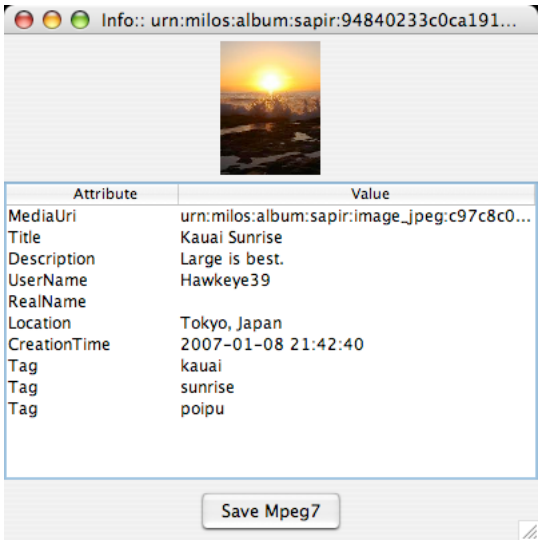


Fig. 5. Image metadata.

For instance, the execution of the query shown in Fig. 3 produces the ranked list of images shown in Fig. 6.

Related to each image, we may also access to its metadata, which is in our case an excerpt of MPEG-7 (the data can be edited by the user as well). We may also select an image of the result pane and further refine the query to retrieve images similar to the selected one.

5 Experiments

We conducted an experiment with the DL-MEDIA system. We considered an image set of around 560.000 images together with their MPEG-7 metadata. The data have been provided by Flickr [4] as a courtesy and for experimental purposes only. In MILOS we have indexed the images' low-level features as well as their associated XML metadata. We built an ontology with 356 concept definitions, 12 relations. Totally, we have 746 DL-MEDIA axioms. We built 10 queries to be submitted to the system and measured for each of them

1. the precision at 10, *i.e.* the percentage of relevant images within the top-10 results.
2. the number of queries generated after the reformulation process (q'_{ref});
3. the number of reformulated queries after redundancy elimination (q_{ref});
4. the time of the reformulation process (t_{ref});
5. the number of queries effectively submitted to MILOS (q_{MILOS});
6. the query answering time of MILOS for each submitted query (t_{MILOS});
7. the time of merging process using the DTA (t_{DTA});

[4] `http://www.flickr.com/`.

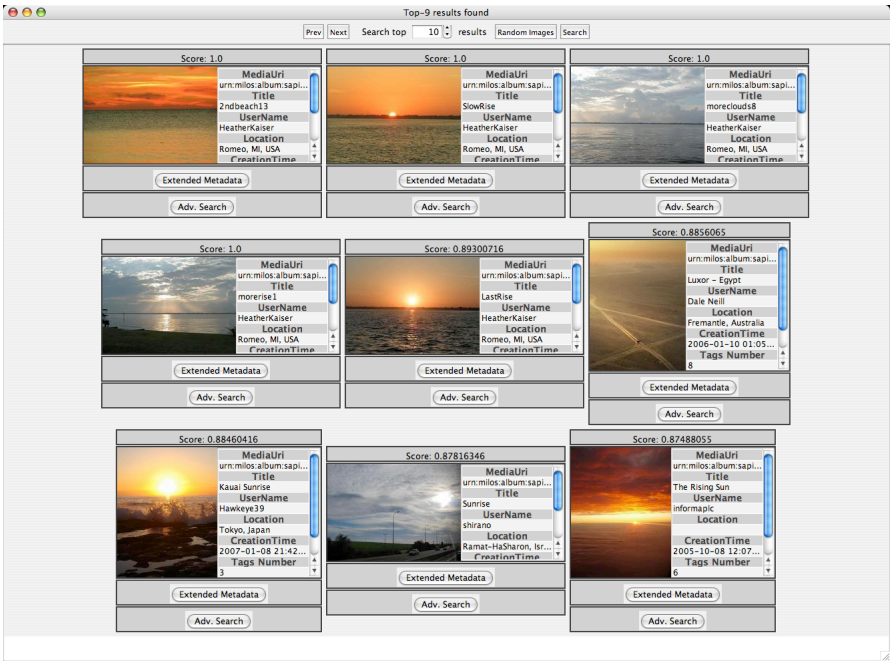


Fig. 6. DL-MEDIA results pane.

8. the time needed to visualize the images in the user interface (t_{Img});
9. the total time from the submission of the initial query to the visualization of the final result (t_{tot}).

The results are shown in Table 1 below (time is measured in seconds). Let's comment some points. The number of queries generated after query reformulation varies significantly and depends both on the structure of the ontology and the concepts involved in the original query. For instance, a query about African animals formulated as

$$q_8(x) \leftarrow Animal(x) \wedge Africa(x)$$

will be reformulated into several queries involving the sub-concepts of both Animal and Africa, which in our case is quite large. Also interesting is that, e.g. for query 8, we may remove more than 100 queries from $r(q_8, \mathcal{O})$ by a simple query subsumption test check. Besides the possibility to have large query reformulation sets, the query reformulation time is quite low (less than 0.5 seconds). Also negligible is the time spent by the DTA merging algorithm. The MILOS response time is quite reasonable once we submit one query only (the answer is provided within some seconds). Clearly, as we submit the queries sequentially to the MILOS system, the total time sums up. Of course, an improvement may be expected once we submit the queries to MILOS in parallel. This part is under development as a joint activity with the MILOS development group.

Also note that the effective number of queries q_{MILOS} may not coincide with $q_{ref} =$, as we do not submit queries to MILOS which involve abstract concepts only, as they do

Query	Precision	q'_{ref}	q_{ref}	t_{ref}	q_{MILOS}	t_{MILOS}	t_{DTA}	t_{Img}	t_{tot}
Q1	1.0	2	2	0.005	1	0.3	0	0.613	1.045
Q2	0.8	48	48	2.125	1	0.327	0	0.619	3.073
Q3	0.9	3	2	0.018	1	2.396	0	0.617	3.036
Q4	0.8	6	6	0.03	1	0.404	0	0.642	1.147
Q5	0.9	10	6	0.113	1	0.537	0	0.614	1.359
Q6	0.8	10	6	0.254	1	1.268	0	0.86	2.387
Q7	1.0	4	4	0.06	3	15.101	0.004	0.635	15.831
Q8	0.9	522	420	0.531	7	13.620	0.009	0.694	14.895
Q9	0.1	360	288	0.318	20	40.507	0.029	0.801	41.631
Q10	0.9	37	36	0.056	20	36.073	0.018	0.184	36.320

Table 1. Experimental evaluation.

not have a translation into a MILOS query (for instance, the query q_8, which despite belonging to the set of reformulated queries $r(q_8, \mathcal{O})$ is not submitted, while the reformulated query $q_{8_1}(x) \leftarrow Tag(x, animal) \wedge Tag(x, africa)$ is). Also, if we have already retrieved 10 images with score 1.0, we stop the MILOS query submission phase.

From a qualitative point of view of the retrieved images, the precision is satisfactory, though more extensive experiments are needed to assess the effectiveness of the DL-MEDIA system. Worth noting is query 9

$$q_9(x) \leftarrow Europe(x) \wedge Africa(x)$$

in which we considered as relevant one image only, which dealt with a postcard sent from Johannesburg (South Africa) to Norwich (UK).

6 Conclusions

In this work, we have outlined the DL-MEDIA system, *i.e.* an ontology mediated multimedia retrieval system. Main features (so far) of DL-MEDIA are that: (i) it uses an extension of DLR-Lite like language as query and ontology representation language; (ii) it supports feature-based queries, semantic-based queries and their combination; and (iii) is promisingly scalable.

There are several points, which we are further investigating:

- so far, we consider all reformulated queries as equally relevant in response to information need. However, it seems reasonable to assume that the more specific the reformulated query becomes the less relevant its answers may be;
- multithreading of reformulated queries;
- from a language point of view, we would like to extend it by using rules on top of axioms and adding more concrete predicates.

Currently we are investigating how to scale both to a DL-component with 10^3 concepts and to a MIR component indexing 10^6 images.

References

1. Giuseppe Amato, Paolo Bolettieri, Franca Debole, Fabrizio Falchi, Fausto Rabitti, and Pasquale Savino. Using MILOS to build a multimedia digital library application: The PhotoBook experience. In *10th European Conference on Research and Advanced Technology for Digital Libraries*, LNCS 4172, pages 379–390. Springer Verlag, 2006.
2. Giuseppe Amato and Franca Debole. A native XML database supporting approximate match search. In *ECDL*, pages 69–80, 2005.
3. Giuseppe Amato, Claudio Gennaro, Fausto Rabitti, and Pasquale Savino. MILOS: A multimedia content management system for digital library applications. In *Proceedings of the 8th European Conference Research and Advanced Technology for Digital Libraries (ECDL-04)*, pages 14–25, 2004.
4. Richard Arndt, Raphaël Troncy, Steffen Staab, Lynda Hardman, and Miroslav Vacura. COMM: Designing a well-founded multimedia ontology for the web. In *6th International Semantic Web Conference, 2nd Asian Semantic Web Conference (ISWC-07, ASWC-07)*, LNCS 4825, pages 30–43. Springer Verlag, 2007.
5. Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.
6. Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Data complexity of query answering in description logics. In *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning (KR-06)*, pages 260–270, 2006.
7. DL-Media. `http://gaia.isti.cnr.it/~straccia/software/DL-Media/DL-Media.html`.
8. Petr Hájek. *Metamathematics of Fuzzy Logic*. Kluwer, 1998.
9. Samira Hammiche, Salima Benbernou, and Athena Vakali. A logic based approach for the multimedia data representation and retrieval. In *Seventh IEEE International Symposium on Multimedia (ISM-05)*, pages 241–248. IEEE Computer Society, 2005.
10. J. S. Hare, P. A. S. Sinclair, P. H. Lewis, K. Martinez, P. G. B. Enser, and C. J. Sandom. Bridging the semantic gap in multimedia information retrieval: Top-down and bottom-up approaches. In *3rd European Semantic Web Conference (ESWC-06)*, LNCS 4011. Springer Verlag, 2006.
11. Carlo Meghini, Fabrizio Sebastiani, and Umberto Straccia. A model of multimedia information retrieval. *Journal of the ACM*, 48(5):909–970, 2001.
12. IEEE MultiMedia. MPEG-7: The generic multimedia content description standard, part 1. *IEEE MultiMedia*, 9(2):78–87, 2002.
13. Umberto Straccia. Description logics with fuzzy concrete domains. In Fahiem Bachus and Tommi Jaakkola, editors, *21st Conference on Uncertainty in Artificial Intelligence (UAI-05)*, pages 559–567, Edinburgh, Scotland, 2005. AUAI Press.
14. Umberto Straccia. Answering vague queries in fuzzy DL-Lite. In *Proceedings of the 11th International Conference on Information Processing and Managment of Uncertainty in Knowledge-Based Systems, (IPMU-06)*, pages 2238–2245. E.D.K., Paris, 2006.
15. Umberto Straccia. Towards top-k query answering in description logics: the case of DL-Lite. In *Proceedings of the 10th European Conference on Logics in Artificial Intelligence (JELIA-06)*, LNCS 4160, pages 439–451, Liverpool, UK, 2006. Springer Verlag.
16. Umberto Straccia and Giulio Visco. DL-Media: an ontology mediated multimedia information retrieval system. In *Proceeedings of the International Workshop on Description Logics (DL-07)*, volume 250, Insbruck, Austria, 2007. CEUR.
17. Pavel Zezula, Giuseppe Amato, Vlastislav Dohnal, and Michal Batko. *Similarity Search: The Metric Space Approach (Advances in Database Systems)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

Inference in Probabilistic Ontologies with Attributive Concept Descriptions and Nominals

Rodrigo Bellizia Polastro and Fabio Gagliardi Cozman

Escola Politécnica, Universidade de São Paulo - São Paulo, SP - Brazil
rodrigopolastro@usp.br, fgcozman@usp.br

Abstract. This paper proposes a probabilistic description logic that combines (i) constructs of the well-known $\mathcal{ALC}$ logic, (ii) probabilistic assertions, and (iii) limited use of nominals. We start with our recently proposed logic CR$\mathcal{ALC}$, where any ontology can be translated into a relational Bayesian network with partially specified probabilities. We then add nominals to restrictions, while keeping CR$\mathcal{ALC}$'s interpretation-based semantics. We discuss the clash between a domain-based semantics for nominals and an interpretation-based semantics for queries, keeping the latter semantics throughout. We show how inference can be conducted in CR$\mathcal{ALC}$ and present examples with real ontologies that display the level of scalability of our proposals.

Key words: $\mathcal{ALC}$ logic, nominals, Bayesian/credal networks.

1 Introduction

Semantic web technologies typically rely on the theory of description logics, as these logics offer reasonable flexibility and decidability at an computational cost that seems to be acceptable [1, 2]. Recent literature has examined ways to enlarge description logics with uncertainty representation and management. In this paper we focus on two challenges in uncertainty representation: we seek to define coherent semantics for a probabilistic description logic and to derive algorithms for inference in this logic. More precisely, we wish to attach sensible semantics to sentences such as

$$P(\mathsf{Merlot}(\mathsf{a})|\exists\mathsf{Color}.\{\mathsf{red}\}) = \alpha, \tag{1}$$

that should refer to the probability that a particular wine is Merlot, given that its color is red. Note the presence of the nominal red in this expression, a feature that complicates matters considerably. Also, we wish to *compute* the smallest α that makes Expression (1) true with respect to a given ontology. This latter calculation is an "inference"; that is, an inference is the calculation of a tight bound on the probability of some assertion.

We have recently proposed a probabilistic description language, referred to as "credal $\mathcal{ALC}$" or simply CR$\mathcal{ALC}$ [3], that combines the well-known Attributive Concept Description ($\mathcal{ALC}$) logic and probabilistic inclusions such as

$P(\mathsf{hasWineSugarDry}|\mathsf{WineSugar}) = \beta$. One of the main features of CR$\mathcal{ALC}$ is that it adopts an interpretation-based semantics that allows it to handle probabilistic queries involving Aboxes i.e., sets of assertions. In CR$\mathcal{ALC}$ there is a one-to-one correspondence between a consistent set of sentences and a relational *credal* network; that is, a relational Bayesian network with partially specified probabilities. An inference in CR$\mathcal{ALC}$ is equivalent to an inference in such a network.

In this paper we wish to extend our previous effort [3] by handling realistic examples, such as the Wine and the Kangaroo ontologies, and by adding to CR$\mathcal{ALC}$ a limited version of nominals (that is, reference to individuals in concept descriptions). Nominals often appear in ontologies; besides, the study of nominals touches on central issues in probabilistic logic, as discussed later.

Section 2 summarizes the main features of CR$\mathcal{ALC}$ and reviews existing probabilistic description logics, emphasizing the differences between them and CR$\mathcal{ALC}$. Section 3 presents the challenges created by nominals, and introduces our proposal for dealing with (some of) them. Section 4 described our experiments with real ontologies in the literature. While our previous effort [3] was mainly directed at theoretical analysis of CR$\mathcal{ALC}$, in this paper we move to evaluation of inference methods in real ontologies. Finally, Section 5 evaluates the results and draws some thoughts on the next steps in the creation of a complete probabilistic semantic web.

2 Probabilistic Description Logics and CR$\mathcal{ALC}$

In this section we review a few important concepts, the literature on probabilistic description logics, and the logic CR$\mathcal{ALC}$. This section is based on our previous work [3].

2.1 A few definitions

Assume a vocabulary containing *individuals*, *concepts*, and *roles* [1]. Concepts and roles are combined to form new concepts using a set of *constructors*. In $\mathcal{ALC}$ [4], constructors are *conjunction* ($C \sqcap D$), *disjunction* ($C \sqcup D$), *negation* ($\neg C$), *existential* restriction ($\exists r.C$) and *value* restriction ($\forall r.C$). A *concept inclusion* is denoted by $C \sqsubseteq D$ and *concept definition* is denoted by $C \equiv D$, where C and D are concepts. Usually one is interested in *concept subsumption*: whether $C \sqsubseteq D$ for concepts C and D. A set of concept inclusions and definitions is called a *terminology*. If an inclusion/definition contains a concept C in its left hand side and a concept D in its right hand side, the concept C *directly uses* D. The transitive closure of "*directly uses*" is indicated by "*uses*". A terminology is *acyclic* if it is a set of concept inclusions and definitions such that no concept in the terminology uses itself [1]. Typically terminologies only allow the left hand side of a concept inclusion/definition to contain a concept name (and no constructors). Concept $C' \sqcup \neg C'$ is denoted by $\top$ and concept $C' \sqcap \neg C'$ is denoted by $\bot$, where C' is a dummy concept that does not appear anywhere else; also, $r.\top$ is abbreviated by r (for instance, $\exists r$).

A set of *assertions* about individuals may be associated to a terminology. An assertion $C(a)$ directly uses assertions of concepts (resp. roles) directly used by C instantiated by a (resp. by (a, b) for $b \in \mathcal{D}$), and likewise for the "uses" relation. As an example, we may have the assertions such as $\mathsf{Fruit}(\mathsf{appleFromJohn})$ and $\mathsf{buyFrom}(\mathsf{houseBob}, \mathsf{John})$.

The semantics of a description logic is almost always given by a *domain* $\mathcal{D}$ and an *interpretation* $\mathcal{I}$. The domain $\mathcal{D}$ is a nonempty set; we often assume its cardinality to be given as input. Note that in description logics the cardinality of the domain is usually left unspecified, while in probabilistic description logics this cardinality is usually specified (Section 2.2). The interpretation function $\mathcal{I}$ maps each individual to an element of the domain, each concept name to a subset of the domain, each role name to a binary relation on $\mathcal{D} \times \mathcal{D}$. The interpretation function is extended to other concepts as follows: $\mathcal{I}(C \sqcap D) = \mathcal{I}(C) \cap \mathcal{I}(D)$, $\mathcal{I}(C \sqcup D) = \mathcal{I}(C) \cup \mathcal{I}(D)$, $\mathcal{I}(\neg C) = \mathcal{D} \backslash \mathcal{I}(C)$, $\mathcal{I}(\exists r.C) = \{x \in \mathcal{D} | \exists y : (x, y) \in \mathcal{I}(r) \wedge y \in \mathcal{I}(C)\}$, $\mathcal{I}(\forall r.C) = \{x \in \mathcal{D} | \forall y : (x, y) \in \mathcal{I}(r) \rightarrow y \in \mathcal{I}(C)\}$. An inclusion $C \sqsubseteq D$ is entailed iff $\mathcal{I}(C) \subseteq \mathcal{I}(D)$, and $C \equiv D$ iff $\mathcal{I}(D) = \mathcal{I}(D)$.

Some logics in the literature offer significantly larger sets of features, such as numerical restrictions, role hierarchies, inverse and transitive roles (the OWL language contains several such features [2]). And most description logics have direct translations into multi-modal logics [5] or fragments of first-order logic [6]. The translation of $\mathcal{ALC}$ to first-order logic is: each concept C is interpreted as a unary predicate $C(x)$; each role r is interpreted as a binary predicate $r(x, y)$; the other constructs have direct translations into first-order logic, (e.g. $\exists r.C$ is translated to $\exists y : \ r(x, y) \wedge C(y)$ and $\forall r.C$ to $\forall y : \ r(x, y) \rightarrow C(y)$).

2.2 Probabilistic description logics

There are several probabilistic description logics in the literature. Heinsohn [7], Jaeger [8] and Sebastiani [9] consider probabilistic inclusion axioms such as $P_{\mathcal{D}}(\mathsf{Plant}) = \alpha$, meaning that a randomly selected individual is a Plant with probability α. This interpretation characterizes a *domain-based* semantics. Sebastiani also allows assessments as $P(\mathsf{Plant}(\mathsf{Tweety})) = \alpha$, specifying probabilities over the interpretations themselves, characterizing an *interpretation-based* semantics. Most proposals for probabilistic description logics adopt a domain-based semantics [7–14, 16], while relatively few adopt an interpretation-based semantics [9, 17].

Direct inference refers to the transfer of statistical information about domains to specific individuals [18, 19]. Direct inference is a problem for domain-based semantics; for instance, from $P(\mathsf{FlyingBird}) = 0.3$ there is nothing to be concluded over $P(\mathsf{FlyingBird}(\mathsf{Tweety}))$. We discuss direct inference further in Section 3. Due to the difficulties in solving direct inference, most proposals for probabilistic description logics with a domain-based semantics simply do not handle assertions. Dürig and Studer avoid direct inference by only allowing probabilities over assertions [11]. Also note that Lukasiewicz has proposed another strategy, where expressive logics are combined with probabilities through an entailment relation with non-monotonic properties, *lexicographic entailment* [12, 14, 15].

The probabilistic description logics mentioned so far do not encode independence relations, neither syntactically nor semantically. A considerable number of proposals for probabilistic description logics that represent independence through graphs has appeared in the last decade or so, in parallel with work on statistical relational models [20, 21]. Logics such as P-CLASSIC [13], Yelland's Tiny Description Logic [16], Ding and Peng's BayesOWL language [10], and Staker's logic [22] all employ Bayesian networks and various constructs of description logics to define probabilities over domains — that is, they have domain-based semantics. Costa and Laskey's PR-OWL language [17] uses an interpretation-based semantics inherited from Multi-entity Bayesian networks (MEBNs) [23]. Related and notable efforts by Nottelmann and Fuhr [24] and Hung et al [25] should be mentioned (note also the existence of several non-probabilistic variants of description logics [26]).

The logic CR$\mathcal{ALC}$, proposed previously by the authors [3], adopts an interpretation-based semantics, so as to avoid direct inference and to handle individuals smoothly (this is discussed in more detail later). The closest existing proposal is Costa and Laskey's PR-OWL; indeed one can understand CR$\mathcal{ALC}$ as a trimmed down version of PR-OWL where the focus is on the development of scalable inference methods. The next section summarizes the main features of CR$\mathcal{ALC}$.

2.3 CR$\mathcal{ALC}$

The logic CR$\mathcal{ALC}$ starts with all constructs of $\mathcal{ALC}$: concepts and roles combined through *conjunction* $C \sqcap D$, *disjunction* $C \sqcup D$, *negation* $\neg C$, *existential* restriction $\exists r.C$, and *value* restriction $\forall r.C$; concept *inclusions* $C \sqsubseteq D$ and concept *definitions* $C \equiv D$; individuals and assertions. An inclusion/definition can only have a concept name in its left hand side; also, restrictions $\exists r.C$ and $\forall r.C$ can only use a concept name C (an auxiliary definition may specify a concept C of arbitrary complexity). A set of assertions is called an *Abox*. The semantics is given by a domain $\mathcal{D}$ and an interpretation $\mathcal{I}$, just as in $\mathcal{ALC}$.

Probabilistic inclusions are then added to the language. A probability inclusion reads $P(C|D) = \alpha$, where D is a concept and C is a concept name. If D is $\top$, then we simply write $P(C) = \alpha$. Probabilistic inclusions are required to only have concept names in their conditioned concept (that is, an inclusions such as $P(\forall r.C|D)$ is not allowed). Given a probabilistic inclusion $P(C|D) = \alpha$, say that C "directly uses" B if B appears in the expression of D; again, "uses" is the transitive closure of "directly uses", and a terminology is acyclic if no concept uses itself. The semantics of a probabilistic inclusion is:

$$\forall x : P(C(x)|D(x)) = \alpha, \tag{2}$$

where it is understood that probabilities are over the set of all interpretation mappings $\mathcal{I}$ for a domain $\mathcal{D}$. We also allow assessments such as $P(r) = \beta$ to be made for roles, with semantics

$$\forall x, y : P(r(x, y)) = \beta, \tag{3}$$

where again the probabilities are over the set of all interpretation mappings.

These probabilistic assessments and their semantics allow us to smoothly interpret a query $P(A(a)|B(b))$ for concepts A and B and individuals a and b. Note that asserted facts must be conditioned upon; there is no contradiction between $\forall x : P(C(x)) = \alpha$ and observation $C(a)$ holds, as we can have $P(C(a)|C(a)) = 1$ while $P(C(a)) = \alpha$. As argued by Bacchus [18], for such a semantics to be useful, an assumption of rigidity for individuals must be made (that is, an element of the domain is associated with the same individual in all interpretations).

An *inference* is the calculation of a *query* $P(A(a)|\mathcal{A})$, where A is a concept, a is an individual, and $\mathcal{A}$ is an Abox.

Concept inclusions (including probabilistic ones) and definitions are assumed acyclic: a concept never uses itself. The acyclicity assumption allows one to draw any terminology $\mathcal{T}$ as a directed acyclic graph $\mathcal{G}(\mathcal{T})$ defined as follows. Each concept (even a restriction) is a node, and if a concept C directly uses concept D, then D is a *parent* of C in $\mathcal{G}(\mathcal{T})$. Also, each restriction $\exists r.C$ or $\forall r.C$ also appears as a node in the graph $\mathcal{G}(\mathcal{T})$, and the graph must contain a node for each role r, and an edge from r to each restriction directly using it.

The next step in the definition of CR$\mathcal{ALC}$ is a *Markov condition*. This Markov condition indicates which independence relations should be read off of a set of sentences. The Markov condition is similar to Markov conditions adopted in probabilistic description logics such as P-CLASSIC, BayesOWL and PR-OWL, but in those logics, a set of sentences is specified with the help of a directed acyclic graph, while in CR$\mathcal{ALC}$ a set of sentences $\mathcal{T}$ specifies a directed acyclic graph $\mathcal{G}(\mathcal{T})$. The Markov condition for CR$\mathcal{ALC}$ refers to this directed acyclic graph $\mathcal{G}(\mathcal{T})$. More details on the various possible Markov conditions can be found elsewhere [3].

The idea in CR$\mathcal{ALC}$ is that the structure of the "directly uses" relation encodes stochastic independence through a Markov condition: (i) for every concept $C \in \mathcal{T}$ and for every $x \in \mathcal{D}$, $C(x)$ is independent of every assertion that does not use $C(x)$, given assertions that directly use C; (ii) for every $(x, y) \in \mathcal{D} \times \mathcal{D}$, $r(x, y)$ is independent of all other assertions, except ones that use $r(x, y)$.

A terminology in CR$\mathcal{ALC}$ does not necessarily specify a single probability measure over interpretations. The following *homogeneity condition* is assumed. Consider a concept C with parents $D_1, \ldots, D_m$. For any conjunction of the m concepts $\pm D_i$, where $\pm$ indicates that D_i may be negated or not, we have that $P(C| \pm D_1 \sqcap \pm D_2 \sqcap \cdots \sqcap \pm D_m)$ is a constant. Consequently, any terminology can be translated into a non-recursive relational Bayesian network [28] where some probabilities are not fully specified. Indeed, for a fixed finite domain $\mathcal{D}$, the propositionalization of a terminology $\mathcal{T}$ produces a *credal network* [29].

In this paper we also adopt the *unique names assumption* (distinct elements of the domain refer to distinct individuals), and the assumption that the cardinality of the domain is fixed and known (*domain closure*). While the rigidity, acyclicity and Markov conditions are essential to the meaning of CR$\mathcal{ALC}$, the homogeneity, unique names, and domain closure assumptions seem less motivated, but are necessary for computational reasons at this point.

3 CR$\mathcal{ALC}$ and nominals

The logic $\mathcal{ALC}$ does not allow *nominals*; that is, it does not allow individuals to appear in concept definitions. Nominals are difficult to handle even in standard description logics. Several optimization techniques employed in description logics fail with nominals, and indeed few algorithms and packages do support nominals correctly at this point. For one thing, nominals introduce connections between a terminology and an Abox, thus complicating inferences. To some extent, nominals cause reasoning to require at least partial grounding of a terminology, a process that may incur significant cost. Still, nominals appear in many real ontologies; an important example is the Wine Ontology that has been alluded to in the Introduction [30].

In the context of uncertainty handling, nominals are particularly interesting as they highlight differences between domain-based and interpretation-based semantics. Consider for instance a domain-based semantics, and suppose that a nominal Tweety is used to define a class $\{\mathsf{Tweety}\}$ such that $P(\{\mathsf{Tweety}\}) = 0.3$. Presumably the assessment indicates that Tweety is "selected" with probability 0.3; this is a natural way to interpret nominals. However, now we face the challenge of *direct inference*; for instance, what is $P(\mathsf{Fly}(\mathsf{Tweety}))$? The difficulty is that for every interpretation mapping $\mathcal{I}$, $\mathsf{Fly}(\mathsf{Tweety})$ either holds or not; that is, Tweety either flies or not. Once we fix an interpretation mapping, as required by a domain-based semantics, the probability $P(\mathsf{Fly}(\mathsf{Tweety}))$ gets fixed at 0 or 1. We might then try to consider the set of all interpretation mappings; this takes us back to an interpretation-based semantics. Worse, with the set of interpretations mappings we have mappings fixing the behavior of Tweety either way (flying or otherwise). Thus we cannot conclude anything about the probability that Tweety flies, unless we make additional assumptions about the connection between domains and interpretations. Several proposals exist for connecting domains and interpretations, but the matter is still quite controversial at this point [19].

Our approach is to stay within the interpretation-based semantics of CR$\mathcal{ALC}$, allowing some situations to have nominals and interpreting those situations through an interpretation-based semantics as well. We do not allow general constructs such as

$$\mathsf{WineFlavor} \equiv \{\mathsf{delicate}, \mathsf{moderate}, \mathsf{strong}\}.$$

Rather, we allow nominals only as domains of roles in restrictions. That is, the semantics for $r.\{a\}$ is not based on quantification over the domain, as the semantics given by Expression (2). Instead, we wish to interpret this construct directly either as (in existential restrictions):

$$\exists x : r\,(x, y) \wedge (y = a), \tag{4}$$

or as (in universal restrictions):

$$\forall x : r\,(x, y) \rightarrow (y = a). \tag{5}$$

In restrictions containing more than one nominal as in $r.\{a, b, c\}$, the resulting restriction considers the disjunction of the various assignments to a, b, c and so on.

Inference in CR$\mathcal{ALC}$, as presented previously [3], grounds a terminology into a credal network. The various conditions previously adopted (acyclicity, domain closure, homogeneity) guarantee that this is always possible. Inference is then the calculation of tight lower and upper bounds on some probability $P(A(a)|\mathcal{A})$ of interest, where A is a concept, a is an individual, and $\mathcal{A}$ is an Abox. Inference can be conducted in the grounded credal network using either exact [31–33] or approximate [34] algorithms.

In the presence of nominals, this grounding of a terminology in CR$\mathcal{ALC}$ may generate huge networks. To avoid this problem, the grounded network must be instantiated only at its relevant nominals; that is, the nominals present in the roles must have specific domains. So, if the role $\mathsf{hasProperty}(x, y)$ indicates that the element x has one specific property with value y, then x must be one object being described and y must be a nominal that describes the property indicated by the role. For instance, $\exists\mathsf{hasColor}.\{\mathsf{red}\}$ is interpreted as:

$$\exists x \in D : \mathsf{hasColor}\,(x, y) \wedge (y = \mathsf{red}), \tag{6}$$

where $\mathcal{D}$ is the domain with the elements being described and y ranges over all the nominals that "are" colors. This approach is very close to *Datatypes*, but its most significant characteristic is the definition of the semantic given by Expressions 4 and 5.

Nominals are often used to define mutually exclusive individuals. Although CR$\mathcal{ALC}$ does not have any construct to express this situation, it can be easily done through the inclusion of a probabilistic node that has the mutually exclusive nodes as its parents and a conditional probability table that mimics the behavior of a XOR logic gate. This node must be set as an observed node with value `true` so that all of its parents become inter-dependent.

4 Experiments

We now report on two experiments with well-known networks. The first one was done with the large Wine Ontology, and the second one was done with the not so famous Kangaroo ontology.

The Wine Ontology was extracted from a OWL file available at the ontology repository of the Temporal Knowledge Base Group from Universitat Jaume I (at http://krono.act.uji.es/Links/ontologies/wine.owl/view). It is a ontology that relies extensively in nominals for describing the different kind of wines and their properties. These nominals were represented as indicated in Section 3. Probability inclusions were added to the terminology; assertions were made on properties of an unspecified wine and the wine type was then inferred. Figure 1 shows the network generated for a domain of size 1. We have:

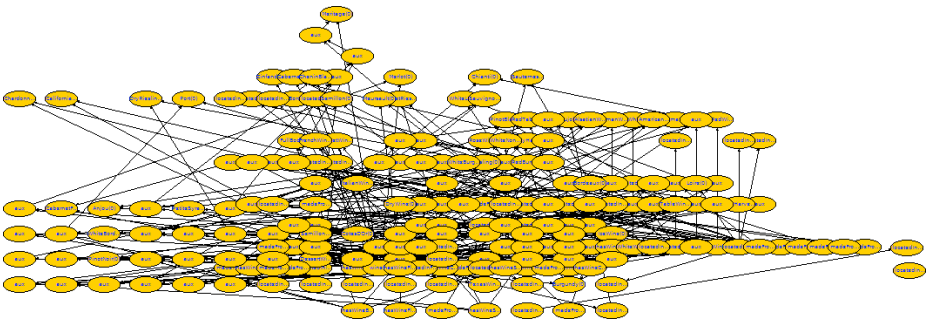


Fig. 1. Network generated from the Wine ontology with domain size 1.

Example 1. The probability of a wine to be Merlot given its body is medium, its color is red, its flavor is moderate, its sugar is dry and it is made from merlot grape:

$$P(\mathsf{Merlot}(a) \mid \mathit{medium}(a), \mathit{red}(a), \mathit{moderate}(a), \mathit{dry}(a), \mathit{merlotGrape}(a)) = 1.0.$$

Example 2. The probability of a wine to be Merlot given its body is medium, its color is red, its flavor is moderate and its sugar is dry:

$$P(\mathsf{Merlot}(b) \mid \mathit{medium}(b), \mathit{red}(b), \mathit{moderate}(b), \mathit{dry}(b)) = 0.5.$$

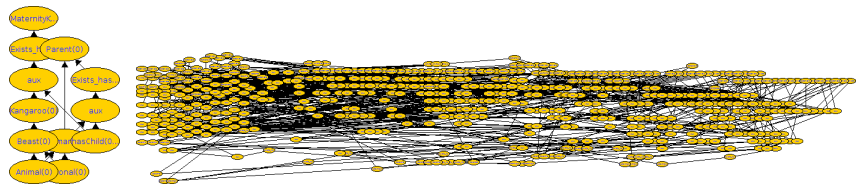
Example 3. The probability of a wine to be Merlot given it is made from merlot grape and its sugar is sweet:

$$P(\mathsf{Merlot}(c) \mid \mathit{merlotGrape}(c), \mathit{sweet}(c)) = 0.0.$$

The Wine ontology only presents restrictions over roles and properties, not having any restriction over individuals. That is, there is no connection between individuals other than the constraints imposed on restrictions by nominals. Consequently, the whole ontology can be translated into a single credal network of fixed size regardless of the actual size of the domain, as far as inference is concerned. Hence there are no qualms about scalability and computational cost when the domain grows. In fact, it was possible to run exact inference in this experiment, even with big domains, since we can separate only the necessary nodes using the Markov condition (we have run exact inferences using the SamIam package, available at http://reasoning.cs.ucla.edu/samiam/).

The second experiment was done with the Kangaroo ontology, adapted from a KRSS file available among the toy ontologies for the CEL System[1] at http://lat.inf.tu-dresden.de/meng/ontologies/kangaroo.cl. Although this ontology does not contain nominals, it uses restrictions amongst individuals in the

[1] A polynomial-time Classifier for the description logic $\mathcal{EL}+$, http://lat.inf.tu-dresden.de/systems/cel/.



(a) $|\mathcal{D}| = 1$ (b) $|\mathcal{D}| = 10$

Fig. 2. Network generated from kangaroo ontology for various domain sizes.

domain, leading to possible concerns on scalability issues as the domain grows. For instance, consider some of the definitions in this ontology:

$$\mathsf{Parent} \equiv \mathsf{Human} \sqcap \forall \mathsf{hasChild}.\mathsf{Human}.$$

$$\mathsf{MaternityKangaroo} \equiv \mathsf{Kangaroo} \sqcap \forall \mathsf{hasChild}.\mathsf{Kangaroo}.$$

In this case, the size of the grounded credal network is proportional to $|\mathcal{D}|^2$; that is, it is quadratic on domain size.

It was not possible to run exact inference in this ontology with big domains, but the *L2U* algorithm [34] produced approximate inferences with reasonable computational cost. Table 1 shows some results for a growing domain. In Figure 2 we can see the size of the network generated for different domain sizes: in Fig.2(a) the domain size is 1 while in Fig.2(b) the domain size is 10.

Table 1. Results from the L2U algorithm for the inference $P\left(Parent\left(0\right) \mid Human\left(1\right)\right)$ for various domain sizes

N	2	5	10	20	30	40	50
L2U	0.2232	0.3536	0.4630	0.5268	0.5377	0.5396	0.5399

5 Conclusion

In this paper we have continued our efforts to develop a probabilistic description logic that can handle both probabilistic inclusions and queries containing Aboxes. This may seem a modest goal, but it touches on the central question concerning semantics in probabilistic logics; that is, whether the semantics is a domain-based or interpretation-based one. We have kept our preference for an interpretation-based semantics in this paper, as it seems to be the only way to avoid the challenges of direct inference. Without an interpretation-based semantics, it is hard to imagine how an inference involving Aboxes could be defined.

Most existing probabilistic description logics do adopt domain-based semantics, but it seems that the cost in avoiding inferences with Aboxes is high.

In this paper we have shown that the algorithms outlined in a previous publication [3] do scale up to realistic ontologies in the literature. Obviously, there is a trade-off between expressivity and complexity in any description logic, and it is difficult to know which features can be added to a description logic before making it intractable in practice. In this paper we have examined the challenges in adding nominals to the CR$\mathcal{ALC}$ logic. Nominals are both useful in practice, and interesting on theoretical grounds. The discussion of nominals can shed light on issues of semantics and direct inference, and one of the goals of this paper was to start a debate in this direction. We have presented relatively simple techniques that handle nominals in a limited setting; that is, as domains of restrictions. Much more work must be done before the behavior of nominals in probabilistic description logics becomes well understood. The inclusion of nominals intro CRALC, however limited, moves us towards the SHOIN logic, and therefore closer to OWL, the recommended standard for the Semantic Web. We hope to gradually close the remaining gap and a complete probabilistic version of OWL in future work

Acknowledgements

This work was partially funded by FAPESP (04/09568-0); the first author is supported by HP Brazil R&D.; the second author is partially supported by CNPq. We thank all these organizations.

References

1. F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, and P.F. Patel-Schneider. *Description Logic Handbook*. Cambridge University Press, 2002.
2. I. Horrocks, P. F. Patel-Schneider, and F. van Harmelen. From SHIQ and RDF to OWL: The making of a web ontology language. *Journal of Web Semantics*, 1(1):7–26, 2003.
3. F. G. Cozman, and R. B. Polastro. Loopy Propagation in a Probabilistic Description Logic. In *Int. Conf. on Scalable Uncertainty Management*, to appear, 2008.
4. M. Schmidt-Schauss and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48:1–26, 1991.
5. K. Schild. A correspondence theory for terminological logics: Preliminary report. In *Int. Joint Conf. on Artificial Intelligence*, pages 466–471, 1991.
6. A. Borgida. On the relative expressiveness of description logics and predicate logics. *Artificial Intelligence*, 82(1-2):353–367, 1996.
7. J. Heinsohn. Probabilistic description logics. In *Conf. on Uncertainty in Artificial Intelligence*, page 311318, 1994.
8. M. Jaeger. Probabilistic reasoning in terminological logics. In *Principles of Knowledge Representation (KR)*, pages 461–472, 1994.

9. F. Sebastiani. A probabilistic terminological logic for modelling information retrieval. In W.B. Croft and C.J. van Rijsbergen, editors, *17th Annual Int. ACM Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 122–130, Dublin, Ireland, 1994. Springer-Verlag.
10. Z. Ding, Y Peng, and R. Pan. BayesOWL: Uncertainty modeling in semantic web ontologies. In *Soft Computing in Ontologies and Semantic Web*, volume 204 of *Studies in Fuzziness and Soft Computing*, pages 3–29. Springer, Berlin/Heidelberg, 2006.
11. M. Dürig and T. Studer. Probabilistic ABox reasoning: preliminary results. In *Description Logics*, pages 104–111, 2005.
12. R. Giugno and T. Lukasiewicz. P-SHOQ(D): A probabilistic extension of SHOQ(D) for probabilistic ontologies in the semantic web. In *Proceedings of the 8th European Conf. on Logics in Artificial Intelligence (JELIA)*, volume 2424, pages 86–97, Cosenza, Italy, September 2002. Lecture Notes in Artificial Intelligence, Springer.
13. D. Koller and A. Pfeffer. Object-oriented Bayesian networks. In *Conf. on Uncertainty in Artificial Intelligence*, pages 302–313, 1997.
14. T. Lukasiewicz. Expressive probabilistic description logics. *Artificial Intelligence*, to appear, 2008.
15. C. d'Amato, N. Fanizzi, T. Lukasiewicz Tractable Reasoning with Bayesian Description Logics. In Proceedings SUM-2008.
16. P. M. Yelland. Market analysis using a combination of Bayesian networks and description logics. Technical Report SMLI TR-99-78, Sun Microsystems Laboratories, 1999.
17. P. C. G. Costa and K. B. Laskey. PR-OWL: A framework for probabilistic ontologies. In *Conf. on Formal Ontology in Information Systems*, 2006.
18. F. Bacchus. *Representing and Reasoning with Probabilistic Knowledge: A Logical Approach*. MIT Press, Cambridge, 1990.
19. H. E. Kyburg Jr. and C. M. Teng. *Uncertain Inference*. Cambridge University Press, 2001.
20. L. Getoor and B. Taskar. *Introduction to Statistical Relational Learning*. MIT Press, 2007.
21. B. Milch and S. Russell. First-order probabilistic languages: into the unknown. In *Int. Conf. on Inductive Logic Programming*, 2007.
22. R. Staker. Reasoning in expressive description logics using belief networks. In *Int. Conf. on Information and Knowledge Engineering*, pages 489–495, Las Vegas, USA, 2002.
23. P. C. G. da Costa and K. B. Laskey. Of Klingons and starships: Bayesian logic for the 23rd century. In *Conf. on Uncertainty in Artificial Intelligence*, 2005.
24. H. Nottelmann and N. Fuhr. Adding probabilities and rules to OWL lite subsets based on probabilistic datalog. *Int. Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, 14(1):17–42, 2006.
25. E. Hung, L. Getoor, and V. S. Subrahmanian. Probabilistic interval XML. *ACM Transactions on Computational Logic*, 8(4), 2007.
26. T. Lukasiewicz and U. Straccia. Managing uncertainty and vagueness in description logics for the semantic web. *submitted*, 2008.
27. T. Lukasiewicz. Probabilistic logic programming. In *European Conf. on Artificial Intelligence*, pages 388–392, 1998.
28. M. Jaeger. Relational Bayesian networks. In Dan Geiger and Prakash Pundalik Shenoy, editors, *Conf. on Uncertainty in Artificial Intelligence*, pp. 266–273, San Francisco, California, 1997. Morgan Kaufmann.

29. F. G. Cozman. Credal networks. *Artificial Intelligence*, 120:199–233, 2000.
30. M. Smith, C. Welty, and D. McGuiness. OWL Web Ontology Language Guide (W3C Recommendation), 2004.
31. K. A. Andersen and J. N. Hooker. Bayesian logic. *Decision Support Systems*, 11:191–210, 1994.
32. F. G. Cozman, C. P. de Campos, and J. C. Ferreira da Rocha. Probabilistic logic with independence. *Int. Journal of Approximate Reasoning*, in press, (available online 7 September 2007, doi: 10.1016/j.ijar.2007.08.002).
33. C. P. de Campos, F. G. Cozman, and J. E. Ochoa Luna. Assessing a consistent set of sentences in relational probabilistic logic with stochastic independence. *Journal of Applied Logic*, to appear, 2008.
34. J. S. Ide and F. G. Cozman Approximate algorithms for credal networks with binary variables. Int. Journal of Approximate Reasoning, 48(1):275-296, 2008.

Introducing Fuzzy Trust for Managing Belief Conflict over Semantic Web Data

Miklos Nagy[1], Maria Vargas-Vera[2], and Enrico Motta[1]

[1] Knowledge Media Institute (KMi)
The Open University
Walton Hall, Milton Keynes
MK7 6AA, United Kingdom
mn2336@student.open.ac.uk,e.motta@open.ac.uk
[2] Computing Department
The Open University
Walton Hall, Milton Keynes
MK7 6AA, United Kingdom
m.vargas-vera@open.ac.uk

Abstract. Interpreting Semantic Web Data by different human experts can end up in scenarios, where each expert comes up with different and conflicting ideas what a concept can mean and how they relate to other concepts. Software agents that operate on the Semantic Web have to deal with similar scenarios where the interpretation of Semantic Web data that describes the heterogeneous sources becomes contradicting. One such application area of the Semantic Web is ontology mapping where different similarities have to be combined into a more reliable and coherent view, which might easily become unreliable if the conflicting beliefs in similarities are not managed effectively between the different agents. In this paper we propose a solution for managing this conflict by introducing trust between the mapping agents based on the fuzzy voting model.

1 Introduction

Assessing the performance and quality of different ontology mapping algorithms, which operate in the Semantic Web environment has gradually been evolved during the recent years. One remarkable effort is the Ontology Alignment Evaluation Initiative [3] , which provides a possibility to evaluate and compare the mapping quality of different systems. However it also points out the difficulty of evaluating ontologies with large number of concepts i.e. the library track where due to the size of the vocabulary only a sample evaluation is carried out by a number of domain experts. Once each expert has assessed the correctness of the sampled mappings their assessment is discussed and they produce a final assessment, which reflects their collective judgment. Our ontology mapping algorithm DSSim [1] tries to mimic the aforementioned process, using different software

[3] http://oaei.ontologymatching.org/

agents as experts to evaluate and use beliefs over similarities of different concepts in the source ontologies. Our mapping agents use WordNet as background knowledge to create a conceptual context for the words that are extracted from the ontologies and employ different syntactic and semantic similarities to create their subjective beliefs over the correctness of the mapping. DSSim addresses the uncertain nature of the ontology mapping by considering different similarity measures as subjective probability for the correctness of the mapping. It employs the Dempster-Shafer theory of evidence in order to create and combine beliefs that has been produced by the different similarity algorithms. For the detailed description of the DSSim algorithm one can refer to [2]. Using belief combination has their advantages compared to other combination methods . However the belief combination has received a verifiable criticism from the research community. There is a problem with the belief combination if agents have conflicting beliefs over the solution. The main contribution of this paper is a novel trust management approach for resolving conflict between beliefs in similarities, which is the core component of the DSSim ontology mapping system.

The paper is organized as follows. Section 2 provides the description of the problem and its context. Section 3 describes the voting model and how it is applied for determining trust during the ontology mapping. In section 4 we present our experiments that have been carried out with the benchmarks of the Ontology Alignment Initiative. Section 5 gives and overview of the related work. Finally, section 6 describes our future work.

2 Problem description

In the context of the Semantic Web trust can have different meaning therefore before we describe the problem let us define the basic notions of our argument.

Definition 1 *Trust: One mapping agent's measurable belief in the competence of the other agents' belief over the established similarities.*

Definition 2 *Content related trust: Dynamic trust measure that is dependent on the actual vocabulary of the mappings, which has been extracted from the ontologies and can change from mapping to mapping.*

Definition 3 *Belief: The state in which a software agent holds a proposition or premise over a possible mapping of selected concept pair combination to be true. Numerical representation of belief can be assigned to a value between [0..1].*

If we assume that in the Semantic Web environment it is not possible to deduct an absolute truth from the available sources then we need to evaluate content dependent trust levels by each application that processes the information on the Semantic Web e.g. how a particular information coming from one source compares the same or similar information that is coming from other sources.

Dominantly the existing approaches that address the problem of the trustworthiness of the available data on the Semantic Web are reputation based e.g.

using digital signatures that would state who the publisher of the ontology is. However another and probably most challenging aspect of trust appears when we process the available information on the Semantic Web and we discover contradictory information from the evidences. Consider an example from ontology mapping. When we assess similarity between two terms, ontology mapping can use different linguistic and semantic[3] information in order to determine the similarity level e.g. background knowledge or concept hierarchy. In practice any similarity algorithm will produce good and bad mappings for the same domain depending of the actual interpretation of the terms in the ontologies e.g. using different background knowledge descriptions or class hierarchy. In order to overcome this shortcoming the combination of different similarity measures are required. During the recent years a number of methods and strategies have been proposed[3] to combine these similarities. In practice considering the overall results these combination methods will perform well under different circumstances except when contradictory evidence occurs during the combination process.

In our ontology mapping framework different agents assess similarities and their beliefs on the similarities need to be combined into a more coherent result. However these individual beliefs in practice are often conflicting. A conflict between two beliefs in Dempster-Shafer theory can be interpreted qualitatively as one source strongly supports one hypothesis and the other strongly supports another hypothesis, where the two hypotheses are not compatible. In this scenario applying Dempster's combination rule to conflicting beliefs can lead to an almost impossible choice, because the combination rule strongly emphasizes the agreement between multiple sources and ignores all the conflicting evidences.

We argue that the problem of contradictions can only be handled from case to case by introducing trust for the similarity measures, which is applied only for the selected mapping and can change from mapping to mapping during the process depending on the available evidences. We propose evaluating trust in the different beliefs that does not depend on the credentials of the ontology owner but it purely represents the trust in a proposed subjective belief that has been established by using different similarity algorithms.

3 Fuzzy trust management for conflicting belief combination

In ontology mapping the conflicting results of the different beliefs in similarity can be resolved if the mapping algorithm can produce an agreed solution, even though the individual opinions about the available alternatives may vary. We propose a solution for reaching this agreement by evaluating fuzzy trust between established beliefs through voting, which is a general method of reconciling differences. Voting is a mechanism where the opinions from a set of votes are evaluated in order to select the alternatives that best represent the collective preferences. Unfortunately deriving binary trust like trustful or not trustful from the difference of belief functions is not so straightforward since the different voters express their opinion as subjective probability over the similarities. For a

particular mapping this always involves a certain degree of vagueness hence the threshold between the trust and distrust cannot be set definitely for all cases that can occur during the process. Additionally there is no clear transition between characterising a particular belief highly or less trustful.

Fuzzy model is based on the concept of linguistic or "fuzzy" variables. These variables correspond to linguistic objects or words, rather than numbers e.g. trust or belief conflict . The fuzzy variables themselves are adjectives that modify the variable (e.g. "high" trust, "small" trust). The membership function is a graphical representation of the magnitude of participation of each input. It associates a weighting with each of the inputs that are processed, define functional overlap between inputs, and ultimately determines an output response. The membership function can be defined differently and can take different shapes depending on the problem it has to represent. Typical membership functions are trapezoidal, triangle or exponential. The selection of our membership function is not arbitrary but can be derived directly from fact that our input the belief difference has to produce the trust level as an output. Each input has to produce output, which requires a trapezoidal and overlapping membership function. Therefore our argument is that the trust membership value, which is expressed by different voters, can be modelled properly by using fuzzy representation as depicted on Fig. 1.

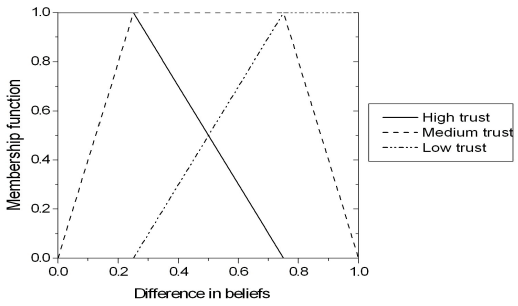


Fig. 1. Trust representation

Imagine the scenario where before each agent evaluates the trust in other agent's belief over the correctness of the mapping it calculates the difference between its own and the other agent's belief. The belief functions for each agent are derived from different similarity measures therefore the actual value might differ from agent to agent. Depending on the difference it can choose the available trust levels e.g. one agent's measurable belief over the similarity is 0.85 and an another agent's belief is 0.65 then the difference in beliefs is 0.2 which can lead to high and medium trust levels. We model these trust levels as fuzzy membership functions.

In fuzzy logic the membership function $\mu(x)$ is defined on the universe of discourse U and represents a particular input value as a member of the fuzzy set i.e. $\mu(x)$ is a curve that defines how each point in the U is mapped to a membership value (or degree of membership) between 0 and 1.

For representing trust in beliefs over similarities we have defined three overlapping trapezoidal membership functions, which represents high, medium and low trust in the beliefs over concept and property similarities in our ontology mapping system.

3.1 Fuzzy voting model

The fuzzy voting model was developed by Baldwin [4] and has been used in Fuzzy logic applications. However, to our knowledge it has not been introduced in the context of trust management on the Semantic Web. In this section, we will briefly introduce the fuzzy voting model theory using a simple example of 10 voters voting against or in favour of the trustfulness of an another agent's belief over the correctness of mapping. In our ontology mapping framework each mapping agent can request a number of voting agents to help assessing how trustful the other mapping agent's belief is.

According to Baldwin [4] a linguistic variable is a quintuple $(L, T(L), U, G, \mu)$ in which L is the name of the variable, $T(L)$ is the term set of labels or words (i.e. the linguistic values), U is a universe of discourse, G is a syntactic rule and μ is a semantic rule or membership function. We also assume for this work that G corresponds to a null syntactic rule so that $T(L)$ consists of a finite set of words. A formalization of the fuzzy voting model can be found in [5].
Consider the set of words { Low_trust (L_t), Medium_trust (M_t) and High_trust (H_t) } as labels of a linguistic variable trust with values in $U = [0, 1]$. Given a set "m" of voters where each voter is asked to provide the subset of words from the finite set $T(L)$, which are appropriate as labels for the value u. The membership value $\chi_{\mu_(w)(u)}$ is taking the proportion of voters who include u in their set of labels which is represented by w.
We need to introduce more opinions to the system i.e. we need to add the opinion of the other agents in order to vote for the best possible outcome. Therefore we assume for the purpose of our example that we have 10 voters (agents). Formally, let us define

$$\begin{aligned} V &= A1, A2, A3, A4, A5, A6, A7, A8, A9, A10 \\ \Theta &= L_t, M_t, H_t \end{aligned} \tag{1}$$

The number of voters can differ however assuming 10 voters can ensure that

1. The overlap between the membership functions can proportionally be distributed on the possible scale of the belief difference [0..1]
2. The work load of the voters does not slow the mapping process down

Let us start illustrating the previous ideas with a small example - By definition consider our linguistic variable L as TRUST and T(L) the set of linguistic values as $T(L) = (Low\_trust, Medium\_trust, High\_trust)$. The universe of discourse is U, which is defined as $U = [0, 1]$. Then, we define the fuzzy sets $\mu(Low\_trust), \mu(Medium\_trust)$ and $\mu(High\_trust)$ for the voters where each voter has different overlapping trapezoidal membership functions as described on Table 1.

Table 1. Fuzzy set definitions

Voters	$\mu(Low\_trust)$	$\mu(Medium\_trust)$	$\mu(High\_trust)$
A1	[0.25:0,0.75:1,1:1]	[0:0,0.25:1,0.75:1,1:0]	[0:1,0.25:1,0.75:0]
A2	[0.25:0,0.70:1,1:1]	[0:0,0.30:1,0.70:1,1:0]	[0:1,0.30:1,0.75:0]
A3	[0.25:0,0.65:1,1:1]	[0:0,0.35:1,0.65:1,1:0]	[0:1,0.35:1,0.75:0]
A4	[0.25:0,0.60:1,1:1]	[0:0,0.40:1,0.60:1,1:0]	[0:1,0.40:1,0.75:0]
A5	[0.25:0,0.55:1,1:1]	[0:0,0.45:1,0.55:1,1:0]	[0:1,0.45:1,0.75:0]
A6	[0.25:0,0.50:1,1:1]	[0:0,0.50:1,0.50:1,1:0]	[0:1,0.50:1,0.75:0]
A7	[0.30:0,0.75:1,1:1]	[0.5:0,0.50:1,0.50:1,0.95:0]	[0:1,0.25:1,0.70:0]
A8	[0.35:0,0.75:1,1:1]	[0.10:0,0.50:1,0.50:1,0.90:0]	[0:1,0.25:1,0.65:0]
A9	[0.40:0,0.75:1,1:1]	[0.15:0,0.50:1,0.50:1,0.85:0]	[0:1,0.25:1,0.60:0]
A10	[0.45:0,0.75:1,1:1]	[0.20:0,0.50:1,0.50:1,0.80:0]	[0:1,0.25:1,0.55:0]

The data in Table 1 are demonstrative only for the purpose of an example, which is presented in this paper. The difference in the membership functions represented by the different vertices of the trapezoid in Table 1 ensures that voters can introduce different opinions as they pick the possible trust levels for the same difference in belief.
The possible set of trust levels $L = TRUST$ is defined by the Table 2. Note that in the table we use a short notation L_t means Low_trust, M_t means Medium_trust and H_t means High_trust. Once the fuzzy sets (membership functions) have been defined the system is ready to assess the trust memberships for the input values. Based on the difference of beliefs in similarities the different voters will select the words they view as appropriate for the difference of belief. Assuming that the difference in beliefs(x) is 0.67(one agent's belief over similarities is 0.85 and an another agent's belief is 0.18) the voters will select the labels representing the trust level as described in Table 2. Note that each voter has its own membership

Table 2. Possible values for the voting

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
L_t	L_t	L_t	L_t	L_t	L_t	L_t	L_t	L_t	L_t
M_t	M_t	M_t	M_t	M_t	M_t				
H_t	H_t	H_t							

function where the level of overlap is different for each voter. As an example the belief difference 0.67 can represent high, medium and low trust level for the first voter(A1) and it can only represent low trust for the last voter(A10).
Then we compute the membership value for each of the elements on set $T(L)$.

$$\chi_{\mu(Low\_trust)(u)} = 1 \tag{2}$$

$$\chi_{\mu(Medium\_trust)(u)} = 0.6 \tag{3}$$

$$\chi_{\mu(High\_trust)(u)} = 0.3 \tag{4}$$

and

$$L = \frac{Low\_trust}{1} + \frac{Medium\_trust}{0.6} + \frac{High\_trust}{0.3} \tag{5}$$

A value x(actual belief difference between two agents) is presented and voters randomly pick exactly one word from a finite set to label x as depicted in Table 3. The number of voters will ensure that a realistic overall response will prevail during the process.

Table 3. Voting

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
H_t	M_t	L_t	L_t	M_t	M_t	L_t	L_t	L_t	L_t

Taken as a function of x these probabilities form probability functions. They should therefore satisfy:

$$\sum_{w \in T(L)} Pr(L = w|x) = 1 \tag{6}$$

which gives a probability distribution on words:

$$\sum Pr(L = Low\_trust|x) = 0.6 \tag{7}$$

$$\sum Pr(L = Medium\_trust|x) = 0.3 \tag{8}$$

$$\sum Pr(L = High\_trust|x) = 0.1 \tag{9}$$

As a result of voting we can conclude that given the difference in belief $x = 0.67$ the combination should not consider this belief in the similarity function since based on its difference compared to another beliefs it turns out to be a distrustful assessment. The before mentioned process is then repeated as many times as many different beliefs we have for the similarity i.e. as many as different similarity measures exist in the ontology mapping system.

3.2 Introducing trust into ontology mapping

The problem of trustworthiness in the context of ontology mapping can be represented in different ways. In general, trust issues on the Semantic Web are associated with the source of the information i.e. who said what and when and what credentials they had to say it. From this point of view the publisher of the ontology could greatly influence the outcome of the trust evaluation and the mapping process can prefer mappings that came from a more "trustful" source.

However we believe that in order to evaluate trust it is better to look into our processes that map these ontologies, because from the similarity point of view it is more important to see how the information in the ontologies are "conceived" by our algorithms than who have created them e.g. do our algorithms exploit all the available information in the ontologies or just part of it. The reason why we propose such trust evaluation is because ontologies of the Semantic Web usually represent a particular domain and support a specific need. Therefore even if two ontologies describe the same concepts and properties their relation to each other can differ depending on the conceptualisation of their creators, which is independent from the organisation where they belong. In our ontology mapping method we propose that the trust in the provided similarity measures, which is assessed between the ontology entities are associated to the actual understanding of the mapping entities, which differs from case to case e.g. a similarity measure can be trusted in one case but not trustful in an another case during the same process. Our mapping algorithm that incorporates trust management into the process is described by Algorithm 1.

```
   Input: Similarity belief matrixes S_{n×m} = {S_1, .., S_k}
   Output: Mapping candidates
 1 for i=1 to n do
 2 |   BeliefVectors BeliefVectors ← GetBeliefVectors(S[i, 1 − m]) ;
 3 |   Concepts ← GetBestBeliefs(BeliefVectors BeliefVectors) ;
 4 |   Scenario ← CreateScenario(Concepts) ;
 5 |   for j=1 to size(Concepts) do
 6 |   |   Scenario ← AddEvidences (Concepts) ;
 7 |   end
 8 |   if Evidences are contradictory then
 9 |   |   for count=1 to numberOf(Experts) do
10 |   |   |   Voters ← CreateVoters(10) ;
11 |   |   |   TrustValues ← VoteTrustMembership(Evidences) ;
12 |   |   |   ProbabilityDistribution ← CalculateTrustProbability(TrustValues) ;
13 |   |   |   Evidences ← SelectTrustedEvidences(ProbabilityDistribution) ;
14 |   |   end
15 |   end
16 |   Scenario ← CombineBeliefs(Evidences) ;
17 |   MappingList ← GetMappings(Scenario) ;
18 end
```

Algorithm 1: Belief combination with trust

Our mapping algorithm receives the similarity matrixes(both syntactic and semantic) as an input and produces the possible mappings as an output. The similarity matrixes represent the assigned similarities between all concepts in ontology 1 and 2. Our mapping algorithm iterates through all concepts in ontology 1 and selects the best possible candidate terms from ontology 2 which is represented as a vector of best beliefs(step 2). Once we have selected the best

beliefs we get the terms that corresponds to these beliefs and create a mapping scenario. This scenario contains all possible mapping pairs between the selected term in ontology 1 and the possible terms from ontology 2(step 3 and 4). Once we have build our mapping scenario we start adding evidences from the similarity matrixes(step 6). These evidences might contradict because different similarity algorithms can assign different similarity measure for the same mapping candidates. In these evidences are contradictory we need to evaluate which measure i.e. mapping agent's belief we trust in this particular scenario(step 8-15). The trust evaluation(see details in section 3.1) is invoked which invalidates the evidences(agent beliefs) which cannot be trusted in this scenario. Once the conflict resolution routine is finished, the valid beliefs can be combined and the possible mapping candidates can be selected from the scenario.

The advantage of our proposed solution is that the evaluated trust is independent from the source ontologies themselves and can change depending on the available information in the context.

4 Empirical evaluation

The evaluation was measured with recall and precision, which are useful measures that have a fixed range and meaningful from the mapping point of view. Before we present our evaluation let us discuss what improvements one can expect considering the mapping precision or recall. Most people would expect that if the results can be doubled i.e. increased by 100% then this is a remarkable achievement. This might be the case for anything but ontology mapping. In reality researchers are trying to push the limits of the existing matching algorithms and anything between 10% and 30% is considered a good improvement. The objective is always to make improvement in preferably both in precision and recall

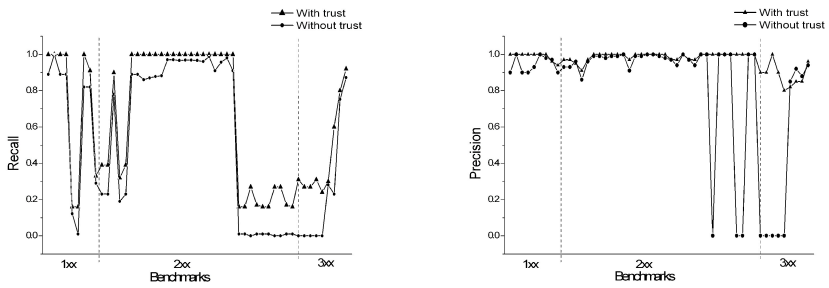
We have carried out experiments with the benchmark ontologies of the Ontology Alignment Evaluation Initiative(OAEI), [4] which is an international initiative that has been set up for evaluating ontology matching algorithms. The experiments were carried out to assess how trust management influences results of our mapping algorithm. Our main objective was to evaluate the impact of establishing trust before combining beliefs in similarities between concepts and properties in the ontology. The OAEI benchmark contains tests, which were systematically generated starting from some reference ontology and discarding a number of information in order to evaluate how the algorithm behave when this information is lacking. The bibliographic reference ontology (different classifications of publications) contained 33 named classes, 24 object properties, 40 data properties. Further each generated ontology was aligned with the reference ontology. The benchmark tests were created and grouped by the following criteria:

- Group 1xx: simple tests such as comparing the reference ontology with itself, with another irrelevant ontology or the same ontology in its restriction to OWL-Lite

[4] http://oaei.ontologymatching.org/

– Group 2xx: systematic tests that were obtained by discarding some features from some reference ontology e.g. name of entities replaced by random strings or synonyms
– Group 3xx: four real-life ontologies of bibliographic references that were found on the web e.g. BibTeX/MIT, BibTeX/UMBC

As a basic comparison we have modified our algorithm (without trust), which does not evaluate trust before conflicting belief combination just combine them using Dempster's combination rule. The recall and precision graphs for the algorithm with trust and without trust over the whole benchmarks are depicted on Fig. 2. Experiments have proved that with establishing trust one can reach higher average precision and recall rate.



(a) Recall (b) Precision

Fig. 2. Recall and Precision graphs

Figure 2 shows the improvement in recall and precision that we have achieved by applying our trust model for combining contradictory evidences. From the precision point of view the increased recall values have not impacted the results significantly, which is good because the objective is always the improvement of both recall and precision together. We have measured the average improvement for the whole benchmark test set that contains 51 ontologies. Based on the experiments the average recall has increased by 12% and the precision is by 16%. The relative high increase in precision compared to recall is attributed to the fact that in some cases the precision has been increased by 100% as a consequence of a small recall increase of 1%. This is perfectly normal because if the recall increases from 0 to 1% and the returned mappings are all correct (which is possible since the number of mappings are small) then the precision is increases from 0 to 100%. Further the increase in recall and precision greatly varies from test to test. Surprisingly the precision have decreased in some cases(5 out of 51). The maximum decrease in precision was 7% and maximum increase was 100%. The recalls have never decreased in any of the tests and the minimum increase was 0.02% whereas the maximum increase was 37%.

As mentioned in our scenario in our ontology mapping algorithm there are number of mapping agents that carry out similarity assessments hence create belief mass assignments for the evidence. Before the belief mass function is combined each mapping agent need to calculate dynamically a trust value, which describes how confident the particular mapping agent is about the other mapping agent's assessment. This dynamic trust assessment is based on the fuzzy voting model and depending on its own and other agents' belief mass function. In our ontology mapping framework we assess trust between the mapping agents' beliefs and determine which agent's belief cannot be trusted, rejecting the one, which is as the result of trust assessment become distrustful.

5 Related work

To date trust has not been investigated in the context of ontology mapping. Ongoing research has mainly been focusing on how trust can be modelled in the Semantic Web context [6] where the trust of user's belief in statements supplied by any other user can be represented and combined. Existing approaches for resolving belief conflict are based on either negotiation or the definition of different combination rules that consider the possibility of belief conflict. Negotiation based techniques are mainly proposed in the context of agent communication. For conflicting ontology alignment an argumentation based framework has been proposed [7] , which can be applied for agent communication and web services where the agents are committed to a ontology and they try to negotiate with other agent over the meaning of their concepts. Considering multi-agent systems on the Web existing trust management approaches have successfully used fuzzy logic to represent trust between the agents from both individual[8] and community[9] perspective. However the main objective of these solutions is to create a reputation of an agent, which can be considered in future interactions. Considering the different variants [10] [11] of combination rules that considers conflicting belief a number of alternatives have been proposed. These methods are based on well founded theoretical base but they all modify the combination rule itself and such these solutions do not consider the process in which these combinations take place. We believe that the conflict needs to be treated before the combination occurs. Further our approach does not assume that any agent is committed to a particular ontology but our agents are considered as "experts" in assessing similarities of terms in different ontologies and they need to reach conclusion over conflicting beliefs in similarities.

6 Conclusion

In this paper we have shown how the fuzzy voting model can be used to evaluate trust, and determine which belief is contradictory with other beliefs before combining them into a more coherent state. We have proposed new levels of trust in the context of ontology mapping, which is a prerequisite for any systems that makes use of information available on the Semantic Web. Our system is

flexible because the membership functions for the voters can be changed dynamically in order to influence the outputs according to the different similarity measures that can be used in the mapping system. We have described initial experimental results with the benchmarks of the Ontology Alignment Initiative, which demonstrates the effectiveness of our approach through the improved recall and precision rates. There are many areas of ongoing work, with our primary focus being additional experimentation to investigate different kind of membership functions for the different voters and to consider the effect of the changing number of voters and the impact on precision and recall.

References

1. Nagy, M., Vargas-Vera, M., Motta, E.: Dssim - managing uncertainty on the semantic web. In: Proceedings of the 2nd International Workshop on Ontology Matching. (2007)
2. Nagy, M., Vargas-Vera, M., Motta, E.: Multi-agent ontology mapping with uncertainty on the semantic web. In: Proceedings of the 3rd IEEE International Conference on Intelligent Computer Communication and Processing. (2007)
3. Euzenat, J., Shvaiko, P.: Ontology matching. Springer-Verlag, Heidelberg (DE) (2007)
4. Baldwin, J.F. In: Mass assignment Fundamentals for computing with words. Volume 1566 of Selected and Invited Papers from the Workshop on Fuzzy Logic in Artificial Intelligence ,Lecture Notes In Computer Science. Springer-Verlag (1999) 22–44
5. Lawry, J.: A voting mechanism for fuzzy logic. International Journal of Approximate Reasoning **19** (1998) 315–333
6. Richardson, M., Agrawal, R., Domingos, P.: Trust management for the semantic web. In: Proceedings of the 2nd International Semantic Web Conference. (2003) 351–368
7. Laera, L., Blacoe, I., Tamma, V., Payne, T., Euzenat, J., Bench-Capon, T.: Argumentation over ontology correspondences in mas. In: AAMAS '07: Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems, New York, NY, USA, ACM (2007) 1–8
8. Griffiths, N.: A fuzzy approach to reasoning with trust, distrust and insufficient trust. In: Proceedings of the 10th International Workshop on Cooperative Information Agents. (2006) 360–374
9. Rehak, M., Pechoucek, M., Benda, P., Foltyn, L.: Trust in coalition environment: Fuzzy number approach. In: Proceedings of The 4th International Joint Conference on Autonomous Agents and Multi Agent Systems - Workshop Trust in Agent Societies. (2005) 119–131
10. Yamada, K.: A new combination of evidence based on compromise. Fuzzy Sets Syst. **159**(13) (2008) 1689–1708
11. Josang, A.: The consensus operator for combining beliefs. Artificial Intelligence **141**(1) (2002) 157–170

Representing Uncertain Concepts in Rough Description Logics via Contextual Indiscernibility Relations

Nicola Fanizzi[1], Claudia d'Amato[1],
Floriana Esposito[1], and Thomas Lukasiewicz[2,3]

[1] LACAM – Dipartimento di Informatica, Università degli Studi di Bari
Via Orabona 4, 70125 Bari, Italy
{fanizzi|claudia.damato|esposito}@di.uniba.it
[2] Computing Laboratory, University of Oxford
Wolfson Building, Parks Road, Oxford OX1 3QD, UK
thomas.lukasiewciz@comlab.ox.ac.uk

Abstract. We investigate on modeling uncertain concepts via *rough description logics*, which are an extension of traditional description logics by a simple mechanism to handle approximate concept definitions through lower and upper approximations of concepts based on a rough-set semantics. This allows to apply rough description logics for modeling uncertain knowledge. Since these approximations are ultimately grounded on an indiscernibility relationship, the paper explores possible logical and numerical ways for defining such relationships based on the considered knowledge. In particular, the notion of context is introduced, allowing for the definition of specific equivalence relationships, to be used for approximations as well as for determining similarity measures, which may be exploited for introducing a notion of tolerance in the indiscernibility.

1 Introduction

Modeling uncertain concepts in description logics (DLs) [1] is generally done via numerical approaches, such as probabilistic or possibilistic ones [2]. A drawback of these approaches is that uncertainty is introduced in the model, which often has the consequence that the approach becomes conceptually and/or computationally more complex. An alternative (simpler) approach is based on the theory of *rough sets* [3], which gave rise to new representations and *ad hoc* reasoning procedures [4]. These languages are based on the idea of *indiscernibility*.

Among these recent developments, *rough description logics* (RDLs) [5] have introduced a complementary mechanism that allows for modeling uncertain knowledge by means of crisp approximations of concepts. RDLs extend classical DLs with two modal-like operators, the lower and the upper approximation. In the spirit of rough set theory, two concepts approximate an underspecified (uncertain) concept as particular

[3] Alternative address: Institut für Informationssysteme, Technische Universität Wien, Favoritenstr. 9-11, 1040 Wien, Austria; email: lukasiewicz@kr.tuwien.ac.at.

sub- and super-concepts, describing which elements are definitely and possibly, respectively, elements of the concept.

The approximations are based on capturing uncertainty as an indiscernibility relation R among individuals, and then formally defining the upper approximation of a concept as the set of individuals that are indiscernible from at least one that is known to belong to the concept:

$$\overline{C} := \{a \mid \exists b : R(a,b) \wedge b \in C\}\,.$$

Similarly, one can define the lower approximation as

$$\underline{C} := \{a \mid \forall b : R(a,b) \rightarrow b \in C\}\,.$$

Intuitively, the upper approximation of a concept C covers the elements of a domain with the typical properties of C, whereas the lower approximation contains the prototypical elements of C.

This may be described in terms of necessity and possibility. These approximations are to be defined in a crisp way. RDLs can be simulated with standard DLs without added expressiveness. This means that reasoning can be performed by translation to standard DLs using a standard DL reasoner.

The pressing issue of efficiency of the reasoning has to be solved. So far, reasoners are not optimized for reasoning with equivalence classes, which makes reasoning sometimes inefficient. To integrate equivalence relations into RDL ABoxes, other ways may be investigated. Inspired by recent works on semantic metrics [6] and kernels, we propose to exploit semantic similarity measures, which can be optimized in order to maximize their capacity of discerning really different individuals. This naturally induces ways for defining an equivalence relation based on indiscernibility criteria.

The rest of this paper is organized as follows. The basics of RDLs are presented in the next section. Then, in Section 3, contextual indiscernibility relations are introduced. In Section 4, a family of similarity measures based on such contexts is proposed along with a suggestion on their optimization. This also allows for the definition of tolerance degrees of indiscernibility. Conclusions and further applications of ontology mining methods are finally outlined in Section 5.

2 Rough Description Logics

In the following, we assume some familiarity with the basics of standard DL languages and their inference services (see [1] for further details).

As mentioned above, the basic idea behind RDLs is rather straightforward: one can approximate an uncertain concept C by giving upper and lower bounds. The upper approximation of C, denoted $\overline{C}$, is the set of all individuals that possibly belong to C. Orthogonally, the lower approximation of C, denoted $\underline{C}$, is the set of all individuals that definitely belong to C. Traditionally, this is modeled using primitive definitions, i.e., subsumption axioms. In pure DL modeling, the relation between C and its approximations $\underline{C}$ and $\overline{C}$ is $\underline{C} \sqsubseteq C \sqsubseteq \overline{C}$.

RDLs are not restricted to particular DLs, and can be defined for an arbitrary DL language $\mathcal{DL}$. Its RDL language $\mathcal{RDL}$ has the lower and upper approximation as additional unary concept constructors, that is, if C is a concept in $\mathcal{RDL}$, then also $\overline{C}$ and $\underline{C}$ are concepts in $\mathcal{RDL}$. The notions of *rough TBox* and *ABox*, as well as *rough knowledge base* canonically extend the usual notions.

Example 2.1 (Advertising Campaign). Suppose that we want to use some pieces of data collected from the Web to find a group of people to serve as addressees for the advertising campaign of a new product. Clearly, the collected pieces of data are in general highly incomplete and uncertain. The DL concept *Addressee* may now be approximated from below by all the definite addressees and from above by all the potential addressees. So, we can use a DL language to specify the TBox knowledge about the concept *Addressee*, and in the same time specify the TBox and ABox knowledge about which people are definite and potential addressees, i.e., belong to the two concepts $\underline{Addressee}$ and $\overline{Addressee}$, respectively.

A *rough interpretation* is a triple $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}, R^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a domain of objects, $\cdot^{\mathcal{I}}$ is an interpretation function, and $R^{\mathcal{I}}$ is an equivalence relation over $\Delta^{\mathcal{I}}$. The function $\cdot^{\mathcal{I}}$ maps RDL concepts to subsets and role names to relations over the domain $\Delta^{\mathcal{I}}$. Formally, $\mathcal{I}$ extends to the new constructs as follows:

- $\overline{C}^{\mathcal{I}} = \{a^{\mathcal{I}} \in \Delta^{\mathcal{I}} \mid \exists b^{\mathcal{I}} \in \Delta^{\mathcal{I}} : R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}) \wedge b^{\mathcal{I}} \in C^{\mathcal{I}}\}$,
- $\underline{C}^{\mathcal{I}} = \{a^{\mathcal{I}} \in \Delta^{\mathcal{I}} \mid \forall b^{\mathcal{I}} \in \Delta^{\mathcal{I}} : R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}) \rightarrow b^{\mathcal{I}} \in C^{\mathcal{I}}\}$.

Example 2.2 (Advertising Campaign cont'd). In order to define the definite and potential addressees for the advertising campaign of a new product, we may exploit a classification of the people into equivalence classes. For example, people with an income above 1 million dollars may be definite addressees for the advertising campaign of a new Porsche, while people with an income above 100 000 dollars may be potential addressees, and people with an income below 10 000 dollars may not be addressees of such an advertising campaign.

One of the advantages of this way of modeling uncertain concepts is that reasoning comes for free. Indeed, reasoning with approximations can be reduced to standard DL reasoning, by translating rough concepts into pure DL concepts with a special reflexive, transitive, and symmetric role.

A translation function for concepts $\cdot^t : \mathcal{RDL} \mapsto \mathcal{DL}$ is defined as follows (introducing the new role symbol R for the indiscernibility relation):

- $A^t = A$, for all atomic concepts A in $\mathcal{RDL}$,
- $(\overline{C})^t = \exists R.C$, and $(\underline{C})^t = \forall R.C$, for all other concepts C in $\mathcal{RDL}$.

The translation function is recursively applied on subconcepts for all other constructs. This definition can be extended to subsumption axioms and TBoxes.

For any DL language $\mathcal{DL}$ with universal and existential quantification, and symmetric, transitive, and reflexive roles, there is no increase in expressiveness, i.e., RDLs can be simulated in (almost) standard DLs: an $\mathcal{RDL}$ concept C is satisfiable in a rough interpretation relative to $\mathcal{T}^t$ iff the $\mathcal{DL}$ concept C^t is satisfiable relative to $\mathcal{T}^t$ [5].

Other reasoning services, such as subsumption, can be reduced to satisfiability (and finally to ABox consistency) in the presence of negation. As the translation is linear, the complexity of reasoning in an RDL is the same as of reasoning in its DL counterpart with quantifiers, symmetry, and transitivity.

Since RDLs do not specify the nature of the indiscernibility relation, except prescribing its encoding as a (special) new equivalent relation, we introduce possible ways for defining it. The first one makes the definition depend on a specific set of concepts determining the indiscernibility of the individuals relative to a specific context described by the concepts in the knowledge base. Then, we also define the relations in terms of a similarity measure (based on a context of features) which allows for relaxing the discernibility using a tolerance threshold.

3 Contextual Indiscernibility Relations

In this section, we first define the notion of a context via a collection of DL concepts. We then introduce indiscernibility relations based on such contexts. We finally define upper and lower approximations of DL concepts using these notions, and we provide some theoretical results about them.

It is well known that classification by analogy cannot be really general-purpose, since the number of features on which the analogy is made may be very large [7]. The key point is that indiscernibility is not absolute but, rather, an induced notion which depends on the specific contexts of interest. Instead of modeling indiscernibility through a single relation in the interpretation, one may consider diverse contexts each giving rise to a different equivalence relation which determines also different ways of approximating uncertain concepts.

We first recall the notion of projection function [8]:

Definition 3.1 (projection). Let $\mathcal{I}$ be a DL interpretation, and let F be a DL concept. The projection function $\pi_F^{\mathcal{I}} : \Delta^{\mathcal{I}} \mapsto \{0, \frac{1}{2}, 1\}$ is defined as follows:

$$\forall a \in \Delta^{\mathcal{I}} : \quad \pi_F^{\mathcal{I}}(a) = \begin{cases} 1 & \mathcal{I} \models F(a); \\ 0 & \mathcal{I} \models \neg F(a); \\ \frac{1}{2} & \text{otherwise.} \end{cases}$$

We define a *context* as a finite set of relevant features in the form of DL concepts, which may encode a kind of context information for the similarity to be measured [9].

Definition 3.2 (context). A *context* is a set of DL concepts $\mathsf{C} = \{F_1, \ldots, F_m\}$.

Example 3.1 (Advertising Campaign cont'd). One possible context C for the advertising campaign of a new product is given as follows:

$$\mathsf{C} = \{\mathit{SalaryAboveMillion}, \mathit{HouseOwner}, \mathit{Manager}\},$$

where *SalaryAboveMillion*, *HouseOwner*, and *Manager* are DL concepts.

Two individuals, say a and b, are indiscernible relative to the context C iff $\forall i \in \{1, \ldots, m\} : \pi_i(a) = \pi_i(b)$. This easily induces an equivalence relation:

Definition 3.3 (indiscernibility relation). Let $\mathsf{C} = \{F_1, \ldots, F_m\}$ be a context. The indiscernibility relation R_{C} induced by C is defined as follows:

$$R_{\mathsf{C}} = \left\{(a, b) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}) \mid \forall i \in \{1, \ldots, m\} : \pi_i^{\mathcal{I}}(a) = \pi_i^{\mathcal{I}}(b)\right\}$$

Hence, one may define multiple such relations by considering different contexts.

Any indiscernibility relation splits $\Delta^{\mathcal{I}}$ in a partition of equivalence classes (also known as *elementary sets*) denoted $[a]_{\mathsf{C}}$, for a generic individual a. Each class naturally induces a concept, denoted C_a.

Example 3.2 (Advertising Campaign cont'd). Consider again the context C of Example 3.1. Observe that C defines an indiscernibility relation on the set of all people, which is given by the extensions of all atomic concepts constructed from C as its equivalence classes. For example, one such atomic concept is the conjunction of *SalaryAboveMillion*, *HouseOwner*, and *Manager*; another one is the conjunction of *SalaryAboveMillion*, *HouseOwner*, and $\neg$ *Manager*.

Thus, a C-*definable* concept has an extension that corresponds to the union of elementary sets. The other concepts may be approximated as usual (we give a slightly different definition of the approximations relative to those in Section 2).

Definition 3.4 (contextual approximations). Let $\mathsf{C} = \{F_1, \ldots, F_m\}$ be a context, let D be a generic DL concept, and let $\mathcal{I}$ be an interpretation. Then, the *contextual upper and lower approximations* of D relative to C, denoted $\overline{D}^{\mathsf{C}}$ and $\underline{D}_{\mathsf{C}}$, respectively, are defined as follows:

- $(\overline{D}^{\mathsf{C}})^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \mathsf{C}_a \sqcap D \not\models \bot\}$,
- $(\underline{D}_{\mathsf{C}})^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \mathsf{C}_a \sqsubseteq D\}$.

Fig. 1 depicts these approximations. The partition is determined by the feature concepts included in the context, each block standing for one of the C-definable concepts. The block inscribed in the concept polygon represent its lower approximation, while the red-hatched ones stand for its upper approximation.

These approximations can be encoded in a DL knowledge base through special indiscernibility relationships, as in [5], so to exploit standard reasoners for implementing inference services (with crisp answers). Alternatively new constructors for contextual rough approximation may be defined to be added to the standard ones in the specific DL language.

It is easy to see that a series properties hold for these operators:

Proposition 3.1 (properties). *Given a context $\mathsf{C} = \{F_1, \ldots, F_m\}$ and two concepts D and E, it holds that:*

1. $\underline{\bot}_{\mathsf{C}} = \overline{\bot}^{\mathsf{C}} = \bot$,
2. $\underline{\top}_{\mathsf{C}} = \overline{\top}^{\mathsf{C}} = \top$,
3. $\underline{D \sqcup E}_{\mathsf{C}} \sqsupseteq \underline{D}_{\mathsf{C}} \sqcup \underline{E}_{\mathsf{C}}$,

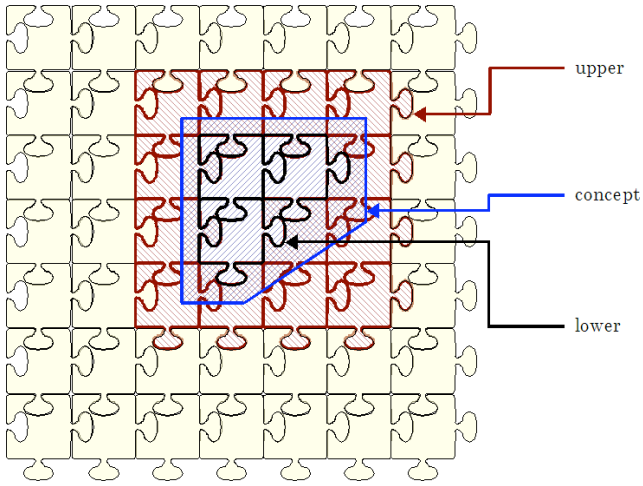


Fig. 1. Lower and upper approximations of rough concepts.

4. $\overline{D \sqcup E}^{\mathsf{C}} = \overline{D}^{\mathsf{C}} \sqcup \overline{E}^{\mathsf{C}}$,
5. $\underline{D \sqcap E}_{\mathsf{C}} = \underline{D}_{\mathsf{C}} \sqcap \underline{E}_{\mathsf{C}}$,
6. $\overline{D \sqcap E}^{\mathsf{C}} \sqsubseteq \overline{D}^{\mathsf{C}} \sqcap \overline{E}^{\mathsf{C}}$,
7. $\underline{\neg D}_{\mathsf{C}} = \neg \overline{D}^{\mathsf{C}}$,
8. $\overline{\neg D}^{\mathsf{C}} = \neg \underline{D}_{\mathsf{C}}$,
9. $\underline{\underline{D}_{\mathsf{C}}}_{\mathsf{C}} = \underline{D}_{\mathsf{C}}$,
10. $\overline{\overline{D}^{\mathsf{C}}}^{\mathsf{C}} = \overline{D}^{\mathsf{C}}$.

4 Numerical Extensions

We now first define rough membership functions. We then introduce contextual similarity measures, and we discuss the aspect of finding optimal contexts. We finally describe how indiscernibility relations can be defined on top of tolerance functions.

4.1 Rough Membership Functions

A rough concept description may include boundary individuals which cannot be ascribed to a concept with absolute certainty. As uncertainty is related to the membership to a set, one can define (rough) membership functions. This can be considered a numerical measure of the uncertainty:

Definition 4.1 (rough membership function). Let $\mathsf{C} = \{F_1, \ldots, F_m\}$ be a context. The C-rough membership function of an individual a to a concept D is defined by:

$$\mu_{\mathsf{C}}(a, D) = \frac{|(\mathsf{C}_a \sqcap D)^{\mathcal{I}}|}{|(\mathsf{C}_a)^{\mathcal{I}}|},$$

where $\mathcal{I}$ is the canonical interpretation [1].

Of course, this measure suffers from being related to the known individuals which conflicts with the open-world semantics of DL languages (unless an epistemic operator is adopted [10] or domain closure is assumed).

4.2 Contextual Similarity Measures

Since indiscernibility can be graded in terms of the similarity between individuals, we propose a new set of similarity functions, based on ideas that inspired a family of inductive distance measures [8, 6]:

Definition 4.2 (family of similarity functions). Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a knowledge base. Given a context $\mathsf{C} = \{F_1, F_2, \ldots, F_m\}$, a family of similarity functions

$$s_p^{\mathsf{C}} : \mathsf{Ind}(\mathcal{A}) \times \mathsf{Ind}(\mathcal{A}) \mapsto [0, 1]$$

is defined as follows ($\forall a, b \in \mathsf{Ind}(\mathcal{A})$):

$$s_p^{\mathsf{C}}(a, b) := \sqrt[p]{\sum_{i=1}^{m} \left| \frac{\sigma_i(a, b)}{m} \right|^p},$$

where $p > 0$ and the *basic similarity function* σ_i ($\forall i \in \{1, \ldots, m\}$) is defined by:

$$\forall a, b \in \mathsf{Ind}(\mathcal{A}) : \qquad \sigma_i(a, b) = 1 - |\pi_i(a) - \pi_i(b)|.$$

This corresponds to defining these functions model-theoretically as follows:

$$\sigma_i(a, b) = \begin{cases} 1 & (\mathcal{K} \models F_i(a) \wedge \mathcal{K} \models F_i(b)) \vee (\mathcal{K} \models \neg F_i(a) \wedge \mathcal{K} \models \neg F_i(b)); \\ 0 & (\mathcal{K} \models \neg F_i(a) \wedge \mathcal{K} \models F_i(b)) \vee (\mathcal{K} \models F_i(a) \wedge \mathcal{K} \models \neg F_i(b)); \\ \frac{1}{2} & \text{otherwise.} \end{cases}$$

Alternatively, in case of densely populated knowledge bases, this can be efficiently approximated, defining the functions as follows ($\forall a, b \in \mathsf{Ind}(\mathcal{A})$):

$$\sigma_i(a, b) = \begin{cases} 1 & (F_i(a) \in \mathcal{A} \wedge F_i(b) \in \mathcal{A}) \vee (\neg F_i(a) \in \mathcal{A} \wedge \neg F_i(b) \in \mathcal{A}); \\ 0 & (F_i(a) \in \mathcal{A} \wedge \neg F_i(b) \in \mathcal{A}) \vee (\neg F_i(a) \in \mathcal{A} \wedge F_i(b) \in \mathcal{A}); \\ \frac{1}{2} & \text{otherwise.} \end{cases}$$

The rationale for these functions is that similarity between individuals is determined relative to a given context [9]. Two individuals are maximally similar relative to a given concept F_i if they exhibit the same behavior, i.e., both are instances of the concept or

of its negation. Conversely, the minimal similarity holds when they belong to opposite concepts. By the open-world semantics, sometimes a reasoner cannot assess the concept-membership, hence, since both possibilities are open, an intermediate value is assigned to reflect such uncertainty.

As mentioned, instance-checking is to be employed for assessing the value of the simple similarity functions. As this is known to be computationally expensive (also depending on the specific DL language), alternatively a simple look-up may be sufficient, as suggested by the first definition of the σ_i functions, especially for ontologies that are rich of explicit class-membership information (assertions).

The parameter p was borrowed from the form of the Minkowski's measures [11]. Once the context is fixed, the possible values for the similarity function are determined, hence p has an impact on the granularity of the measure.

Furthermore, the uniform choice of the weights assigned to the similarity related to the various features in the sum ($1/m^p$) may be replaced by assigning different weights reflecting the importance of a certain feature in discerning the various instances. A good choice may be based on the amount of *entropy* related to each feature concept (then the weight vector has only to be normalized) [6].

4.3 Optimization of the Contexts

It is worthwhile to note that this is indeed a family of functions parameterized on the choice of features. Preliminary experiments regarding instance-based classification, demonstrated the effectiveness of the similarity function using the very set of both primitive and defined concepts found in the knowledge bases. But the choice of the concepts to be included in the context C is crucial and may be the object of a preliminary learning problem to be solved (*feature selection*).

As performed for inducing the pseudo-metric that inspired the definition of the similarity function [8], a preliminary phase may concern finding optimal contexts. This may be carried out by means of randomized optimization procedures.

Since the underlying idea in the definition of the functions is that similar individuals should exhibit the same behavior relative to the concepts in C, one may assume that the context C represents a sufficient number of (possibly redundant) features that are able to discriminate different individuals (in terms of a discernibility measure).

Namely, since the function is strictly dependent on the context C, two immediate heuristics arise:

- the *number* of concepts of the context,
- their discriminating power in terms of a *discernibility factor*, i.e., a measure of the amount of difference between individuals.

Finding optimal sets of discriminating features, should also profit by their composition, employing the specific constructors made available by the DL representation language of choice.

These objectives can be accomplished by means of randomized optimization techniques, especially when knowledge bases with large sets of individuals are available [8]. Namely, part of the entire data can be drawn in order to learn optimal feature sets, in advance with respect to the successive usage for all other purposes.

4.4 Approximation by Tolerance

In [4], a less strict type of approximation is introduced, based on the notion of *tolerance*. Exploiting the similarity functions that have been defined, it is easy to extend this kind of (contextual) approximation to the case of RDLs.

Let a *tolerance function* on a set U be any function $\tau : U \times U \mapsto [0, 1]$ such that for all $a, b \in U, \tau(a, b) = 1$ and $\tau(a, b) = \tau(b, a)$. Considering a tolerance function τ on U and a *tolerance threshold* $\theta \in [0, 1]$, a *neighborhood function* $\nu : U \mapsto 2^U$ is defined as follows:

$$\nu_\theta(a) = \{b \in U \mid \tau(a, b) \geq \theta\}.$$

For each element $a \in U$, the set $\nu_\theta(a)$ is also called the neighborhood of a.

Now, let us consider the domain $\Delta^\mathcal{I}$ of an interpretation $\mathcal{I}$ as a universal set, a similarity function s_p^C on $\Delta^\mathcal{I}$ (for some context C) as a tolerance function, and a threshold $\theta \in [0, 1]$. It is easy to derive an equivalence relationship on $\Delta^\mathcal{I}$, where the classes consist of individuals within a certain degree of similarity, indicated by the threshold: $[a]_\mathsf{C} = \nu_\theta(a)$. The notions of upper and lower approximation relative to the induced equivalence relationship descend straightforwardly.

Not that these approximations depend on the threshold. Thus, we have a numerical way to control the degree of indiscernibility that is needed to model uncertain concepts. This applies both in the standard RDL setting and in the new contextual one presented in the previous section.

5 Summary and Outlook

Inspired by previous works on dissimilarity measures in DLs, we have defined a notion of context, which allows to extend the indiscernibility relationship adopted by rough DLs, thus allowing for various kinds of approximations of uncertain concepts within the same knowledge base. It also saves the advantage of encoding the relation in the same DL language thus allowing for reasoning with uncertain concepts through standard tools obtaining crisp answers to queries.

Alternatively, these approximations can be implemented as new modal-like language operators. Some properties of the approximations deriving from rough sets theory have also been investigated.

A novel family of semantic similarity functions for individuals has also been defined based on their behavior relative to a number of features (concepts). The functions are language-independent being based on instance-checking (or ABox look-up). This allows for defining further kinds of graded approximations based on the notion of tolerance relative to a certain threshold.

Since data can be classified into indiscernible clusters, unsupervised learning methods for grouping individuals on the grounds of their similarity may be used for the definition of the equivalence relation [12, 8, 13]. Besides, it may also possible to learn rough DL concepts from the explicit definitions of the instances of particular concepts [14, 15, 16].

Acknowledgments. This work has been partially supported by the German Research Foundation (DFG) under the Heisenberg Programme.

References

[1] Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P., eds.: The Description Logic Handbook. Cambridge University Press (2003)
[2] Lukasiewicz, T., Straccia, U.: Managing uncertainty and vagueness in description logics for the Semantic Web. Journal of Web Semantics (2008) in press.
[3] Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning About Data. Kluwer Academic Publishers (1991)
[4] Doherty, P., Grabowski, M., Lukaszewicz, W., Szalas, A.: Towards a framework for approximate ontologies. Fundamenta Informaticae **57** (2003) 147–165
[5] Schlobach, S., Klein, M.C.A., Peelen, L.: Description logics with approximate definitions - precise modeling of vague concepts. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007). (2007) 557–562
[6] d'Amato, C., Fanizzi, N., Esposito, F.: Query answering and ontology population: An inductive approach. In: Proceedings of the 5th European Semantic Web Conference (ESWC 2008). Volume 5021 of LNCS, Springer (2008) 288–302
[7] Mitchell, T.: Machine Learning. McGraw-Hill (1997)
[8] Fanizzi, N., d'Amato, C., Esposito, F.: Randomized metric induction and evolutionary conceptual clustering for semantic knowledge bases. In: Proceedings of the 16th International Conference on Knowledge Management (CIKM 2007), ACM Press (2007) 51–60
[9] Goldstone, R., Medin, D., Halberstadt, J.: Similarity in context. Memory and Cognition **25** (1997) 237–255
[10] Donini, F., Lenzerini, M., Nardi, D., Nutt, W.: An epistemic operator for description logics. Artificial Intelligence **100** (1998) 225–274
[11] Zezula, P., Amato, G., Dohnal, V., Batko, M.: Similarity Search – The Metric Space Approach. Advances in Database Systems. Springer (2007)
[12] Hirano, S., Tsumoto, S.: An indiscernibility-based clustering method. In: 2005 IEEE International Conference on Granular Computing, IEEE Computer Society (2005) 468–473
[13] Fanizzi, N., d'Amato, C., Esposito, F.: Conceptual clustering for concept drift and novelty detection. In: Proceedings of the 5th European Semantic Web Conference (ESWC 2008). Volume 5021 of LNCS, Springer (2008) 318–332
[14] Iannone, L., Palmisano, I., Fanizzi, N.: An algorithm based on counterfactuals for concept learning in the Semantic Web. Applied Intelligence **26** (2007) 139–159
[15] Lehmann, J., Hitzler, P.: A refinement operator based learning algorithm for the $\mathcal{ALC}$ description logic. In: Proceedings of the 17th International Conference on Inductive Logic Programming (ILP 2007). Volume 4894 of LNCS, Springer (2008) 147–160
[16] Fanizzi, N., d'Amato, C., Esposito, F.: DL-FOIL: Concept learning in description logics. In: Proceedings of the 18th International Conference on Inductive Logic Programming (ILP 2008). Volume 5194 of LNCS, Springer (2008) 107–121

Storing and Querying Fuzzy Knowledge in the Semantic Web

Nick Simou, Giorgos Stoilos, Vassilis Tzouvaras,
Giorgos Stamou, and Stefanos Kollias

Department of Electrical and Computer Engineering,
National Technical University of Athens,
Zographou 15780, Greece
{nsimou,gstoil,tzouvaras,gstam}@image.ntua.gr

Abstract. The great evolution of ontologies during the last decade, bred the need for storage and querying for the Semantic Web. For that purpose, many RDF tools capable of storing a knowledge base, and also performing queries on it, were constructed. Recently, fuzzy extensions to description logics have gained considerable attention especially for the purposes of handling vague information in many applications. In this paper we investigate on the issue of using classical RDF storing systems in order to provide persistent storing and querying over large-scale fuzzy information. To accomplish this we first propose a novel way for serializing fuzzy information into RDF triples thus classical storing systems can be used without any extensions. Additionally, we extend the existing query languages of RDF stores in order to support expressive fuzzy queries proposed in the literature. These extensions are implemented through the FiRE fuzzy reasoning engine, which is a fuzzy DL reasoner for fuzzy-$\mathcal{SHIN}$. Finally, the proposed architecture is evaluated using an industrial application scenario about casting for TV commercials and spots.

1 Introduction

Ontologies, through the OWL language [11], are expected to play a significant role in the Semantic Web. OWL is mainly based on Description Logics (DLs) [2], a popular family of knowledge representation languages. However, despite their rich expressiveness, they are insufficient to deal with vague and uncertain information which is commonly found in many real-world applications such as multimedia content, medical informatics etc. For that purpose a variety of DLs capable of handling imprecise information in many flavors, like probabilistic [13] and fuzzy [15, 14] have been proposed.

Fuzzy ontologies are envisioned to be very useful in the Semantic Web. Similar to crisp ontologies, they can serve as basic semantic infrastructure, providing shared understanding of certain domains across different applications. Furthermore, the need for handling fuzzy and uncertain information is crucial to the Web. This is because information and data along the Web may often be uncertain or imperfect.

Therefore sophisticated uncertainty representation and reasoning are necessary for the alignment and integration of Web data from different sources. This requirement for uncertainty representation has led W3C to set up the Uncertainty Reasoning for the World Wide Web XG[1]. Recently, fuzzy DL reasoners such as fuzzyDL[2] and FiRE[3] that can handle imprecise information have been implemented. Despite these implementations of expressive fuzzy DLs there is still no other work on persistent storage and querying, besides the work of Straccia [16] and Pan [10], which are based on fuzzy DL-lite and can be considered as closely related to databases, but on the other hand they don't use RDF triple store technologies.

The main contributions of this paper are the following:

1. It presents a novel framework for persistent storage and querying of expressive fuzzy knowledge bases,
2. It presents the first ever integration of fuzzy DL reasoners with RDF triple stores, and
3. It provides experimental evaluation of the proposed architecture using a real-world industrial strength use-case scenario.

The rest of the paper is organized as follows. Firstly, in section 2 a short theoretical description of the fuzzy DL f-$\mathcal{SHIN}$ [6] is made. In section 3 the proposed triples syntax accommodating the fuzzy element used for storing a fuzzy knowledge base in RDF-Stores, is presented. Additionally, the syntax and the semantics of expressive queries that have been proposed in the literature [10] to exploit fuzziness are briefly presented. In the following section (4) the fuzzy reasoning engine FiRE which is based on the fuzzy DL f-$\mathcal{SHIN}$ and the way in which it was integrated with the RDF-Store Sesame are presented. In the last section (5) the applicability of the proposed architecture is demonstrated, presenting a use case based on a database of human models. This database was used by a production company for the purposes of casting for TV commercials and spots. Some entries of the database were first fuzzified and then using an expressive knowledge base, abundant implicit knowledge was extracted. The extracted knowledge was stored to a Sesame repository, and various expressive queries were performed in order to benchmark the proposed architecture.

2 Preliminaries

2.1 The Fuzzy DL $\mathbf{f}_{KD}$-$\mathcal{SHIN}$

In this section we briefly present the notation of DL f-$\mathcal{SHIN}$ which is a fuzzy extension of DL $\mathcal{SHIN}$ [7]. Similar to crisp description logic languages, a fuzzy description logic language consist of an alphabet of distinct concepts names ($\mathbf{C}$), role names ($\mathbf{R}$) and individual names ($\mathbf{I}$), together with a set of constructors to

[1] `http://www.w3.org/2005/Incubator/urw3/`
[2] `http://gaia.isti.cnr.it/~straccia/software/fuzzyDL/fuzzyDL.html`
[3] `http://www.image.ece.ntua.gr/~nsimou/FiRE/`

construct concept and role descriptions. If R is a role then R^- is also a role, namely the inverse of R. f-$\mathcal{SHIN}$-concepts are inductively defined as follows,

1. If $C \in \mathbf{C}$, then C is a f-$\mathcal{SHIN}$-concept,
2. If C and D are concepts, R is a role, S is a simple role and $n \in \mathbb{N}$, then $(\neg C)$, $(C \sqcup D)$, $(C \sqcap D)$, $(\forall R.C)$, $(\exists R.C)$, $(\geq nS)$ and $(\leq nS)$ are also f-$\mathcal{SHIN}$-concepts.

In contrast to crisp DLs, the semantics of fuzzy DLs are provided by a *fuzzy interpretation* [15]. A fuzzy interpretation is a pair $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ where $\Delta^{\mathcal{I}}$ is a non-empty set of objects and $\cdot^{\mathcal{I}}$ is a fuzzy interpretation function, which maps an individual name $\mathtt{a}$ to elements of $\mathtt{a}^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ and a concept name A (role name R) to a membership function $\mathsf{A}^{\mathcal{I}} : \Delta^{\mathcal{I}} \rightarrow [0,1]$ ($R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0,1]$).

By using fuzzy set theoretic operations the fuzzy interpretation function can be extended to give semantics to complex concepts, roles and axioms [8]. FiRE uses the standard fuzzy operators of $1 - x$ for fuzzy negation and max, min for fuzzy union and intersection, respectively.

A f-$\mathcal{SHIN}$ knowledge base Σ is a triple $\langle \mathcal{T}, \mathcal{R}, \mathcal{A} \rangle$, where $\mathcal{T}$ is a fuzzy $TBox$, $\mathcal{R}$ is a fuzzy $RBox$ and $\mathcal{A}$ is a fuzzy $ABox$. $TBox$ is a finite set of fuzzy concept axioms which are of the form $C \sqsubseteq D$ called fuzzy concept inclusion axioms and $C \equiv D$ called fuzzy concept equivalence axioms, where C, D are concepts, saying that C is a sub-concept or C is equivalent of D, respectively. Similarly, $RBox$ is a finite set of fuzzy role axioms of the form $\mathsf{Trans}(R)$ called fuzzy transitive role axioms and $R \sqsubseteq S$ called fuzzy role inclusion axioms saying that R is transitive and R is a sub-role of S respectively. Finally, $ABox$ is a finite set of fuzzy assertions of the form $\langle a : C \bowtie n \rangle$, $\langle (a,b) : R \bowtie n \rangle$, where $\bowtie$ stands for $\geq, >, \leq$ or $<$, or $a \neq b$, for $a, b \in \mathbf{I}$. Intuitively, a fuzzy assertion of the form $\langle a : C \geq n \rangle$ means that the membership degree of a to the concept C is at least equal to n.

3 Storing and Querying a Fuzzy Knowledge Base

3.1 Fuzzy OWL Syntax in triples

In order to use the existing RDF storing systems to store fuzzy knowledge without enforcing any extensions we have to provide a way of serializing fuzzy knowledge into RDF triples. Some work has already been done in this issue. In [9] the authors use RDF *reification*, in order to store membership degrees, while the authors in [17] use datatypes. Our goal is to neither use reification nor datatypes. On the one hand, it is well-known that reification has weak, ill-defined model theoretic semantics and its support by RDF tools is doubtful while on the other hand, we do not want to use a concrete feature like datatypes to represent abstract information such as fuzzy assertions. For those reasons we propose a more clarified way based on the use of blank nodes. First, we define three new entities, namely `frdf:membership`, `frdf:degree` and `frdf:ineqType` as types (i.e. `rdf:type`) of `rdf:Property`.

Our syntax becomes obvious in the following example. Suppose that we want to represent the assertion $\langle(paul : Tall) \geq n\rangle$. The RDF triples representing this information are the following:

```
paul              frdf:membership    _:paulmembTall .
_:paulmembTall    rdf:type           Tall .
_:paulmembTall    frdf:degree        "n^^xsd:float" .
_:paulmembTall    frdf:ineqType      "=" .
```

where `_:paulmembPaul` is a blank node used to represent the fuzzy assertion of `paul` with the concept Tall.

On the other hand, mapping fuzzy role assertions is more tricky since RDF does not allow the use of blank nodes in the predicate position. Thus, we have to use new properties for each assertion. Thus, the assertion $\langle(paul, frank) : FriendOf \geq n\rangle$ is mapped to

```
paul                        frdf:paulFriendOffrank    frank .
frdf:paulFriendOffrank      rdf:type                  FriendOf .
frdf:paulFriendOffrank      frdf:degree               "n^^xsd:float" .
frdf:paulFriendOffrank      frdf:ineqType             "=" .
```

3.2 Extensions to Query Languages

One of the main advantages of persistent storage systems, like relational databases and RDF storing systems, is their ability to support *conjunctive queries*. Conjunctive queries generalize the classical inference problem of *realization* of Description Logics [2], i.e. get me all individuals of a given concept C, by allowing for the combination (conjunction) of concepts and roles. Formally, a conjunctive query is of the following form:

$$q(X) \leftarrow \exists Y.conj(X, Y) \tag{1}$$

or simply $q(X) \leftarrow conj(X, Y)$, where $q(X)$ is called the head, $conj(X, Y)$ is called the body, X are called the *distinguished variables*, Y are existentially quantified variables called the *non-distinguished variables*, and $conj(X, Y)$ is a conjunction of atoms of the form $\mathsf{A}(v)$, $R(v_1, v_2)$, where A, R are respectively concept and role names, v, v_1 and v_2 are *individual* variables in X and Y or individuals from the ontology.

Since in our case we extend classical assertions to fuzzy assertions, new methods of querying such fuzzy information are possible. More precisely, in [10] the authors extend ordinary conjunctive queries to a family of significantly more expressive query languages, which are borrowed from the fields of fuzzy information retrieval [5]. These languages exploit the membership degrees of fuzzy assertions by introducing weights or thresholds in query atoms. In particular, the authors first define *conjunctive threshold queries*(CTQs) as:

$$q(X) \leftarrow \exists Y. \bigwedge_{i=1}^{n} (atom_i(X, Y) \geq k_i), \tag{2}$$

where $k_i \in [0,1]$, $1 \leq i \leq n$, $atom_i(X,Y)$ represents either a fuzzy-DL concept or role and all $k_i \in (0,1]$ are thresholds. Intuitively, an evaluation $[X \mapsto S]$ (where S is a set of individuals) is a solution if $atom_i^{\mathcal{I}}(X,Y)_{[X \mapsto S, Y \mapsto S']} \geq k_i$ for some S and for $1 \leq i \leq n$. As it is obvious answers of CTQs is a matter of true or false, in other words an evaluation either is or is not a solution to a query. The authors also propose *General Fuzzy Conjunctive Queries* (GFCQs) that further exploit fuzziness and support degrees in query results. The syntax of a GFCQ is the following:

$$q(X) \leftarrow \exists Y. \bigwedge_{i=1}^{n} (atom_i(X,Y) : k_i), \tag{3}$$

where $atom_i(X,Y)$ and k_i are as above. As it is shown in [10], this syntax is general enough to allow various choices of semantics, which emerge by interpreting differently the association of the degree of each fuzzy-DL atom ($atom_i(X,Y)$) with the degree associated weight (k_i). For example if this association is interpreted by a fuzzy implication ($\mathcal{J}$) [8] then we obtain fuzzy threshold queries:

$$d = \sup_{S' \in \Delta^{\mathcal{I}} \times \ldots \times \Delta^{\mathcal{I}}} \{ t_{i=1}^{n} \; \mathcal{J}(k_i, atom_i^{\mathcal{I}}(\bar{v})_{[X \mapsto S, Y \mapsto S']}) \}.$$

Similarly we can use fuzzy aggregation functions [8] or fuzzy weighted t-norms [4]. Variations of semantics of GFCQs can be effectively used to model importance of query atoms, preferences, and many more. The interested reader is referred to [10] for more details on the semantics of GFCQs.

4 Implementation with FiRE and Sesame

FiRE is a JAVA implementation of a fuzzy reasoning engine for imperfect knowledge currently supporting f-$\mathcal{SHIN}$. It can be found at `http://www.image.ece.ntua.gr/~nsimou/FiRE/` together with installation instructions and examples. Its syntax is based the Knowledge Representation System Specification [1] proposal which has been extended to fit uncertain knowledge. In this section the inference services of FiRE are presented and the way in which it was integrated with RDF Store Sesame [4] in order to support CTQs and GFCQs is demonstrated.

4.1 Inference services

Crisp DL reasoners offer reasoning services such as deciding satisfiability, subsumption and entailment of concepts and axioms w.r.t. an ontology. In other words, these tools are capable of answering queries like "Can the concept C have any instances in models of the ontology T?" (satisfiability of C), "Is the concept D more general than the concept C in models of the ontology T ?" (subsumption $C \sqsubseteq D$) or does axiom Ψ logically follows from the ontology (entailment of Ψ).

[4] `http://www.openrdf.org/`

These reasoning services are also available by f-$\mathcal{SHIN}$ together with *greatest lower bound queries* which are specific to fuzzy assertions. FiRE uses the tableau algorithm of f-$\mathcal{SHIN}$, presented by Stoilos et al [6], in order to decide the key inference problems of a fuzzy ontology. In the case of fuzzy DL, satisfiability queries are of the form "Can the concept C have any instances with degree of participation $\bowtie n$ in models of the ontology T ?". Furthermore, it is in our interest to compute the best lower and upper truth-value bounds of a fuzzy assertion. The term *greatest lower bound* of a fuzzy assertion w.r.t. Σ was defined in [15]. Roughly speaking, greatest lower bound are queries like "What is the greatest degree n that our ontology entails an individual a to participate in a concept C?". Entailment queries ask whether our knowledge base logically entails the membership of an individual to a specific concept to a certain degree.

Finally, FiRE allows the user to make greatest lower bound queries (GLB). GLB queries are evaluated by FiRE performing entailment queries of the individual participating in concept of interest for all the degrees contained in the ABox, using the binary search algorithm in order to reduce the degrees search space [15]. Furthermore a user can perform global GLB for a fuzzy knowledge base. Global GLB service of FiRE, creates a file containing the greatest lower bound degree of all the concepts of Σ participating in all the individuals of Σ.

4.2 Sesame Integration with FiRE

FiRE was enhanced by the functionalities of the RDF-Store Sesame (Sesame 2 beta 6). The RDF Store is used as a back-end for storing and querying RDF triples in a sufficient and convenient way. In this architecture the reasoner is the front-end which the user can use in order to store and query a fuzzy knowledge base. Additionally, a user is able to access data from a repository, apply any of the available reasoning services on this data and then store the implicit knowledge extracted from them back in the repository.

Another important benefit from this integration is the use of the query language SPARQL [12] in the implementation of the fuzzy queries of section 3. These queries are performed using the *Queries* inference tab of FiRE, and in the case of generalized fuzzy conjunctive queries, users can choose among semantics, such as fuzzy threshold queries, fuzzy aggregation and fuzzy weighted queries.

Example 1. A threshold query that reveals their syntax in FiRE follows:

```
x,y <- Tall(x) >= 0.8 ^ has-friend(x,y) >= 0.4 ^ Short(y) >= 0.7
```

Queries consist of two parts: the first one specifies the individuals that will be evaluated while the second one states the condition that has to be fulfilled for the individuals. This query asks for individuals x and y, x has to participate in concept Tall to at least the given degree, it also has to be the subject of a has-friend assertion with participation greater than 0.4, having as a role-filler individual y which has to participate in concept Short to at least the given degrees.

Example 2. We can issue a GFCQ by using the symbol ":" followed by the importance of participation for each condition statement instead of inequality types. Hence we can get all female models and rank those who have long hair higher than those who are good-looking:

```
x <- Female(x) : 1 ^ Goodlooking(x) : 0.6
       ^ has-hairLength(x,y) : 1 ^ Long(y) : 0.8
```

In case of CTQs, a query is firstly converted from the FiRE conjunctive query syntax to SPARQL query language. Based on the fuzzy OWL syntax in triples that we have defined in section 3.1 the query of **Example** 1 is as follows in SPARQL. The query results are evaluated by Sesame engine and visualized by FiRE.

```
SELECT ?x WHERE {
    ?x ns5:membership ?Node1 .
    ?Node1 rdf:type ?Concept1 .
    ?Node1 ns5:ineqType ?IneqType1 .
    ?Node1 ns5:degree ?Degree1 .
    FILTER regex (?Concept1 , "CONCEPTS#Tall")
    FILTER regex (?IneqType1 ,">")
    FILTER (?Degree1 >= "0.8^^xsd:float")

    ?BlankRole2 ns5:ineqType ?IneqType2 .
    ?BlankRole2 ns5:degree ?Degree2 .
    ?BlankRole2 rdf:type ?Role2 .
    ?x BlankRole2 ?y .
    FILTER regex (?Role2 , "ROLES#has-friend")
    FILTER regex (?IneqType1 ,">")
    FILTER (?Degree2 >= "1.0^^xsd:float")
    ...
}
```

In case of general fuzzy conjuctive queries the operation is different. The SPARQL query is constructed in a way that retrieves the participation degrees of every Role or Concept used in the atoms criterions, for the results that satisfy all of the atoms. The participation degrees retrieved for each query atom are then used together with the degree associated weight by FiRE for the ranking procedure of the results according to the selected semantics.

It is worth mentioning that the proposed architecture obviously does not provide a complete query answering system for f-$\mathcal{SHIN}$ since queries are issued against the stored assertions of the RDF repository. Hence queries that include, for example, transitive or inverse roles are not correctly evaluated. On the one hand, query answering for fuzzy-DLs is still an open problem even for inexpressive fuzzy-DLs while on the other hand, even for classical DLs it is known that the algorithms are highly complex [3] and no practically scalable system is known. However, in order to limit the effects of incompleteness, the f-$\mathcal{SHIN}$

expressibility is used by FiRE for the extraction of implicit knowledge that is stored in the repository performing GLB tests.

5 Evaluation

5.1 Models use case

In this section we present the use case of human models utilized for the evaluation of our proposal. The data were taken from a production company database containing 2140 human models. The database contained information on each model regarding their height, age, body type, fitness type, tooth condition, eye-condition and color, hair quality, style, color and length, ending with the hands' condition. Apart from the above, there were some additional, special-appearance characteristics for certain models such as good-looking, sexy, smile, sporty, etc. introduced by the casting producer. Finally for a minority of models, a casting-video was stored in the database. The main objective of the production company was to pick a model, based on the above features, who would be suitable for a certain commercial spot. Furthermore, depending on the spot type, inquiries about models with some profession-like characteristics (like teacher, chef, mafia etc.) were also of interest.

Despite the fact that the database information on each model was rich enough, there was great difficulty in querying models of appropriate characteristics. The main reason for that was that the database information was not semantically organized. The various tables of a database made the searching for combined characteristics antiliturgical. Additionally, retrieval of models based on threshold criteria for their age or height was most of the times inaccurate since this kind of information is clearly fuzzy. Hence, selection was restricted among models that had videotaped castings or those that had worked with the producers in previous spots, thus not taking advantage of their database information.

In order to eliminate these limitations we have implemented a fuzzy knowledge base using f-$\mathcal{SHIN}$. For the generation of the fuzzy $ABox$ the characteristics given by numerical values being the height and age, were fuzzified defining new concepts, while the remaining characteristics were used as crisp assertions. Therefore, the fuzzification process of age was made by setting fuzzy partitions depending on age by defining the concepts Kid, Teen, Baby, 20s, 30s, 40s, 50s, 60s and Old. Hence, as can be observed from the age fuzzification graph, a model who is 29 years old participates in both concepts 20s and 30s with degrees 0.35 and 0.65 respectively. Similarly for the fuzzification process of height, the concepts Very_Short, Short, Normal_Height, Tall and Very_Tall were defined. In the case of the height characteristic, the fuzzy partition used for female models was different from the one used for males, since the average height of females is lower than that of males. The fuzzification graphs of age and men's height are shown in **Figure** 1. An example of the produced assertions is shown in *Example* 3.

Example 3. An excerpt of ABox for the model michalis1539 is

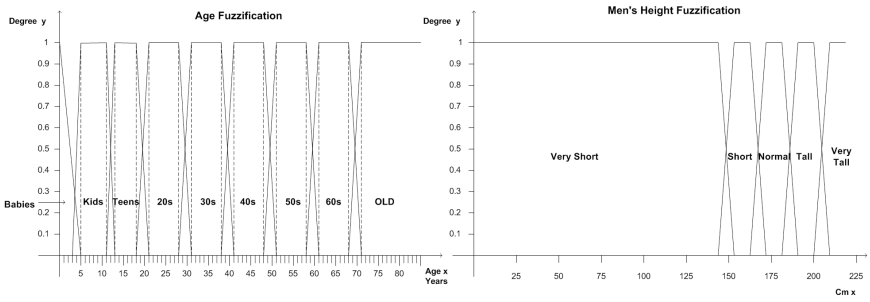


Fig. 1. Fuzzification graphs

$\langle michalis1539 : 20s \geq 0.66\rangle$, $\langle michalis1539 : 30s \geq 0.33\rangle$
$\langle michalis1539 : Normal\_Height \geq 0.5\rangle$
$\langle michalis1539 : Tall \geq 0.5\rangle$, $\langle michalis1539 : GoodLooking \geq 1\rangle$
$\langle (michalis1539, good) : has - toothCondition \geq 1\rangle$, $\langle good : Good \geq 1\rangle$

5.2 The fuzzy knowledge base

In order to permit knowledge-based retrieval of human models we have implemented an expressive terminology for a fuzzy knowledge base. The alphabet of concepts used for the fuzzy knowledge base consists of the features described above while some characteristics like hair length, hair condition etc. were represented by the use of roles.

The effective extraction of implicit knowledge from the explicit one requires an expressive terminology capable of defining higher concepts. In our case the higher domain concepts defined for human models lie into five categories: age, height, family, some special categories and the professions. Hence, the profession Scientist has been defined as male, between their 50s or 60s, with classic appearance who also wears glasses. In a similar way we have defined 33 domain concepts; an excerpt of the $TBox$ can be found in Table 1.

At this point we must mention the fact that the proposed fuzzy knowledge base does not fully utilize the expressivity of f-$\mathcal{SHIN}$. This restriction is due to the application domain (i.e transitive and inverse roles or number restrictions are not applicable in this domain), but nevertheless it is more expressive that an fuzzy DL-Lite ontology.

5.3 Results

All the experiments were conducted under Windows XP on a Pentium 2.40 GHz computer with 2. GB of RAM.

The described fuzzy knowledge base was used in the evaluation of our approach. Implicit knowledge was extracted using the greatest lower bound service of FiRE, asking for the degree of participation of all individuals, in all the defined

domain concepts. The average number of assertions per individual was 13 while the defined concepts were 33, that together with the 2140 individuals (i.e entries of the database) resulted to 29460 explicit assertions and the extraction of 2430 implicit. These results, together with concept and role axioms, were stored to a Sesame repository using the proposed fuzzy OWL triples syntax to form a repository of 529.926 triples.

The average time for the GLB reasoning process and the conversion of explicit and implicit knowledge to fuzzy OWL syntax in triples was 1112 milliseconds. The time required for uploading the knowledge to a Sesame repository depends on the type of repository (Memory or Native) and also on repository's size. Based on our experiments, we have observed that the upload time is polynomial to the size of the repository but without significant differences. Therefore, the average minimum upload time to an almost empty repository (0-10.000 triples) is 213 milliseconds while the average maximum upload time to a full repository (over 500.000 triples) is 700 milliseconds.

FiRE and Sesame were also examined in the use of expressive fuzzy queries. The performance in this case mainly depended on the complexity of the query but also on the type and size of the repository. Queries using role names in combination with large repositories can dramatically slow down the response. Table 2 illustrates the response times in milliseconds using both types of repositories and different repository sizes. Repository sizes was set by adjusting the number of assertions. As it can be observed, very expressive queries seeking for young female models with beautiful legs and eyes as well as long hair, a popular demand in commercial spots, can be easily performed. It is worth mentioning that these queries consist of higher domain concepts defined in our fuzzy knowledge base.

Since our system is not a sound and complete query answering system for f-$\mathcal{SHIN}$, the GLB service performed before uploading the triples is employed in order to use as much of the expressivity of the language as possible producing new implied assertions.

Furthermore, the results regarding query answering time are also very encouraging, at least for the specific application. Although, compared to crisp querying, over crisp knowledge bases, our method might require several more seconds to be answered (mainly due to post processing steps for GFCQs or due to very lengthy SPARQL queries for CTQs) this time is significantly less, compared to

$$
\begin{aligned}
\mathcal{T} = \{&\mathsf{MiddleAged} \equiv \mathsf{40s} \sqcup \mathsf{50s},\\
&\mathsf{TallChild} \equiv \mathsf{Child} \sqcap (\mathsf{Short} \sqcup \mathsf{Normal\_Height}),\\
&\mathsf{Father} \equiv \mathsf{Male} \sqcap (\mathsf{30s} \sqcup \mathsf{MiddleAged}),\\
&\mathsf{Legs} \equiv \mathsf{Female} \sqcap (\mathsf{Normal\_Height} \sqcup \mathsf{Tall})\\
&\qquad \sqcap (\mathsf{Normal} \sqcup \mathsf{Perfect}) \sqcap (\mathsf{Fit} \sqcup \mathsf{PerfectFitness}),\\
&\mathsf{Teacher} \equiv (\mathsf{30s} \sqcup \mathsf{MiddleAged}) \sqcap \mathsf{Elegant} \sqcap \mathsf{Classic},\\
&\mathsf{Scientist} \equiv \mathsf{Male} \sqcap \mathsf{Classic} \sqcap (\mathsf{50s} \sqcup \mathsf{60s})\\
&\qquad \sqcap \mathsf{Serious} \sqcap \exists \mathsf{has} - \mathsf{eyeCondition.Glasses}\ \}
\end{aligned}
$$

Table 1. An excerpt of the Knowledge Base ($TBox$).

the time spent by producers on casting (usually counted in days), since they usually have to browse through a very large number of videos and images before they decide.

6 Conclusions

Due to the fact that imperfect information is spread along the web, the effective management of imperfect knowledge is very important for the substantial evolution of the Semantic Web. In this paper, we have proposed an architecture that can be used for storing and querying fuzzy knowledge bases for the semantic web. Our proposal which is based on DL f-$\mathcal{SHIN}$, consists of the RDF triples syntax accommodating the fuzzy element, the fuzzy reasoning engine FiRE and its integration with RDF Store Sesame which permits very expressive fuzzy queries.

The proposed architecture was evaluated using an industrial application scenario about casting for TV commercials and spots. The obtained results are very promising from the querying perspective. From the initial 29460 explicit assertions made by database instances for models, 2430 new implicit assertions where extracted and both uploaded in the Sesame repository. In this way expressive semantic queries like "Find me young female models with beautiful legs and eyes as well as long hair", that might have proved very difficult or even impossible using the producing company's database, are applicable through FiRE. This reveals both the strength of knowledge-based applications, and technologies for managing fuzzy knowledge, since a wealth of the information of the databases, like age, height, as well as many high level concepts of the specific application, like "beautiful eyes", "perfect fitness" and "scientist look" are inherently fuzzy.

As far as future directions are concerned, we intend to further investigate on different ways of performing queries using expressive fuzzy description logics. Finally, it would be of great interest to attempt a comparison between the proposed architecture and approaches using fuzzy DL-lite ontologies and approximation.

Query	Native			Memory		
	100.000	250.000	500.000	100.000	250.000	500.000
$x \leftarrow \mathsf{Scientist}(x)$	1042	2461	3335	894	2364	3332
$x \leftarrow \mathsf{Father}(x) \geq 1 \wedge \mathsf{Teacher}(x) \geq 0.8$ $\wedge \mathsf{Normal\_Height}(x) \geq 0.5.$	1068	2694	3935	994	2524	3732
$x \leftarrow \mathsf{Scientist}(x) : 0.8$	2562	4173	5235	3042	4543	6027
$x \leftarrow \mathsf{Father}(x) : 0.6 \wedge \mathsf{Teacher}(x) : 0.7$ $\wedge \mathsf{Normal\_Height}(x) : 0.8.$	4318	6694	8935	4341	7896	9306

Table 2. Fuzzy queries evaluation. Queries performed on repositories of size 100.000 250.000 and 500.000. The response time is in milliseconds

Acknowledgements.

This work is supported by the FP6 Network of Excellence EU project X-Media (FP6-026978) and K-space (IST-2005-027026).

References

1. Description-logic knowledge representation system specification from the KRSS group of the ARPA knowledge sharing effort. `http://dl.kr.org/krss-spec.ps`.
2. F. Baader, D. McGuinness, D. Nardi, and P.F. Patel-Schneider. *The Description Logic Handbook: Theory, implementation and applications*. Cambridge University Press, 2002.
3. B.Glimm, I.Horrocks, C.Lutz, and U.Sattler. Conjunctive query answering for $\mathcal{SHIQ}$. Technical report, University of Manchester, 2006.
4. A. Chortaras, Giorgos Stamou, and Andreas Stafylopatis. Adaptation of weighted fuzzy programs. In *Proc. of the International Conference on Artificial Neural Networks (ICANN 2006)*, pages 45–54. Springer, 2006.
5. V. Cross. Fuzzy information retrieval. *Journal of Intelligent Information Systems*, 3:29–56, 1994.
6. G.Stoilos, G.Stamou, V.Tzouvaras, J.Z.Pan, and I.Horrocks. Reasoning with very expressive fuzzy description logics. *Journal of Artificial Intelligence Research*, 30(5):273–320, 2007.
7. I. Horrocks, U. Sattler, and S. Tobies. Reasoning with Individuals for the Description Logic $\mathcal{SHIQ}$. In David MacAllester, editor, *CADE-2000*, number 1831 in LNAI, pages 482–496. Springer-Verlag, 2000.
8. G. J. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice-Hall, 1995.
9. M.Mazzieri and A.F.Dragoni. A fuzzy semantics for semantic web languages. In *ISWC-URSW*, pages 12–22, 2005.
10. J.Z. Pan, G. Stamou, G. Stoilos, and E. Thomas. Expressive querying over fuzzy DL-Lite ontologies. In *Proceedings of the International Workshop on Description Logics (DL 2007)*, 2007.
11. P.F.Patel-Schneider, P.Hayes, and I.Horrocks. Owl web ontology language semantics and abstract syntax. Technical report, World Wide Web Consortium, 2004.
12. E. Prud'hommeaux and A. Seaborne. SPARQL query language for RDF, 2006. W3C Working Draft, http://www.w3.org/TR/rdf-sparql-query/.
13. R.Giugno and T.Lukasiewicz. P-SHOQ(D): A probabilistic extension of SHOQ(D) for probabilistic ontologies in the semantic web. In *JELIA '02: Proceedings of the European Conference on Logics in Artificial Intelligence*, pages 86–97, London, UK, 2002. Springer-Verlag.
14. G. Stoilos, G. Stamou, V. Tzouvaras, J. Z. Pan, and I. Horrocks. Fuzzy OWL: Uncertainty and the Semantic Web. In *Proc. of the OWL-ED 2005*.
15. U. Straccia. Reasoning within fuzzy description logics. *Journal of Artificial Intelligence Research*, 14:137–166, 2001.
16. U.Straccia and G.Visco. DLMedia: an ontology mediated multimedia information retrieval system. In *Proceeedings of the International Workshop on Description Logics (DL 07)*, volume 250, Insbruck, Austria, 2007. CEUR.
17. V. Vaneková, J. Bella, P. Gurský, and T. Horváth. Fuzzy RDF in the semantic web: Deduction and induction. In *Proceedings of Workshop on Data Analysis (WDA 2005)*, pages 16–29, 2005.

Uncertainty Treatment in the Rule Interchange Format: From Encoding to Extension

Jidi Zhao[1], Harold Boley[2]

[1] Faculty of Computer Science, University of New Brunswick, Fredericton, NB, E3B 5AC, Canada
Judy.Zhao AT unb.ca

[2] Institute for Information Technology, National Research Council of Canada, Fredericton, NB, E3B 9W4 Canada
Harold.Boley AT nrc.gc.ca

Abstract. The Rule Interchange Format (RIF) is an emerging W3C format that allows rules to be exchanged between rule systems. Uncertainty is an intrinsic feature of real world knowledge, hence it is important to take it into account when building logic rule formalisms. However, the set of truth values in the Basic Logic Dialect (RIF-BLD) currently consists of only two values (t and f). In this paper, we first present two techniques of encoding uncertain knowledge and its fuzzy semantics in RIF-BLD presentation syntax. We then propose an extension leading to an Uncertainty Rule Dialect (RIF-URD) to support a direct representation of uncertain knowledge. In addition, rules in Logic Programs (LP) are often used in combination with the other widely-used knowledge representation formalism of the Semantic Web, namely Description Logics (DL), in order to provide greater expressive power. To prepare DL as well as LP extensions, we present a fuzzy extension to Description Logic Programs (DLP), called Fuzzy DLP, and discuss its mapping to RIF. Such a formalism not only combines DL with LP, as in DLP, but also supports uncertain knowledge representation.

1. Introduction

Description Logics (DL) and Logic Programs (LP) are the two main categories of knowledge representation formalisms for the Semantic Web, both of which are based on subsets of first-order logic [1]. DL and LP cover different but overlapping areas of knowledge representation. They are complementary to some degree; for example, DL cannot express LP's n-ary function applications (complex terms) while LP cannot express DL's disjunctions (in the head). Combining DL with LP in order to "build rules on top of ontologies" or, "build ontologies on top of rules" has become an emerging topic for various applications of the Semantic Web. It is therefore important to research the combination of DL and LP with different strategies. There have been various achievements in this area, including several proposed combination frameworks [2-6]. As a minimal approach in this area, the Description Logic Program (DLP) 'intersection' of DL and LP has been studied, along with mappings from DL to LP [2]. Both [3] and [5] studied the combination of standard Datalog inference procedures with intermediate ALC DL satisfiability checking.

On the other hand, as evidenced by Fuzzy RuleML [7] and W3C's Uncertainty Reasoning for the World Wide Web (URW3) Incubator Group [8], handling uncertain knowledge is becoming a critical research direction for the (Semantic) Web. For example, many concepts needed in business ontology

modeling lack well-defined boundaries or, precisely defined criteria of relationships with other concepts. To take care of these knowledge representation needs, different approaches for integrating uncertain knowledge into traditional rule languages and DL languages have been studied [1, 9-17].

The Rule Interchange Format (RIF) is being developed by W3C's Rule Interchange Format (RIF) Working Group to support the exchange of rules between rule systems [18]. In particular, the Basic Logic Dialect (RIF-BLD) [19] corresponds to the language of definite Horn rules with equality and a standard first-order semantics. While RIF's Framework for Logic-based Dialects (RIF-FLD) [20] permits multi-valued logics, the current version of RIF-BLD instantiates RIF-FLD with the set of truth values consisting of only two values, t and f, hence is not designed for expressing uncertain knowledge.

According to the final report from the URW3 Incubator group, uncertainty is a term intended to include different types of uncertain knowledge, including incompleteness, vagueness, ambiguity, randomness, and inconsistency [8]. Mathematical theories for representing uncertain knowledge include, but are not limited to, Probability, Fuzzy Sets, Belief Functions, Random Sets, Rough Sets, and combinations of several models (Hybrid). The uncertain knowledge representations and interpretations discussed in this paper are limited to Fuzzy set theory and Fuzzy Logic (a multi-valued logic based on Fuzzy set theory); other types of uncertainty will be studied in future work.

The main contributions of this paper are: (1) two techniques of encoding uncertain information in RIF as well as an uncertainty extension to RIF; (2) an extension of DLP to Fuzzy DLP and the mapping of Fuzzy DLP to RIF. Two earlier uncertainty extensions to the combination of DL and LP that we can expand on are [21] and [22]. While our approach allows DL atoms in the head of hybrid rules and DL subsumption axioms in hybrid rules, the approach of [21] excludes them. Our approach deals with fuzzy subsumption of fuzzy concepts of the form $C \sqsubseteq D \ = c$ whereas [22] deals with crisp subsumption of fuzzy concepts of the form $C \sqsubseteq D$. Also, we do not limit hybrid knowledge bases to the intersection of (fuzzy) DL and (fuzzy) LP. We extend [22] and study the decidable union of DL and LP. In this paper, we only consider the Horn logic subset of LP.

The rest of this paper is organized as follows. Section 2 reviews earlier work on the interoperation between DL and LP in the intersection of these two formalisms (known as DLP) and represents the DL-LP mappings in RIF. Section 3 addresses the syntax and semantics of fuzzy Logic Programs, and then presents two techniques of bringing uncertainty into the current version of RIF presentation syntax (hence its semantics and XML syntax), using encodings as RIF functions and RIF predicates. Section 4 adapts the definition of the set of truth values in RIF-FLD for the purpose of representing uncertain knowledge directly, and proposes the new Uncertainty Rule Dialect (RIF-URD), extending RIF-BLD. Section 5 extends DLP to Fuzzy DLP, supporting mappings between fuzzy DL and fuzzy LP, and gives representations of Fuzzy DLP in RIF and RIF-URD. Finally, Section 6 summarizes our main results and gives an outlook on future research.

2. Description Logic Programs and Their Representation in RIF

In this section, we summarize the work on Description Logic Programs (DLP) [2] and then show how to represent the mappings between DL and LP in RIF presentation syntax.

The paper [2] studied the intersection between the leading Semantic Web approaches to rules in LP and ontologies in DL, and showed how to interoperate between DL and LP in the intersection known as DLP. A DLP knowledge base consists of axioms of the following kinds: $\underline{C} \sqsubseteq \underline{D}$, $\underline{C} \equiv \underline{D}$, $\mathrm{T} \sqsubseteq \forall \underline{R}.\underline{C}$,

$\mathrm{T} \sqsubseteq \forall \underline{R}^{-}.\underline{C}$, $\underline{R} \sqsubseteq \underline{P}$, $\underline{P} \equiv \underline{R}$, $\underline{P} \equiv \underline{R}^{-}$, $\underline{R}^{+} \sqsubseteq \underline{R}$, $\underline{C}(a)$ and $\underline{R}(a,b)$, where $\underline{C}, \underline{D}$ are concepts, T is the universal concept, $\underline{P}, \underline{R}$ are roles, $\underline{R}^{-}$ and $\underline{R}^{+}$ are the inverse role and the transitive role of $\underline{R}$, respectively, and a, b are individuals.

In RIF presentation syntax, the quantifiers Exists and Forall are made explicit, rules are written with a ":-" infix, variables start with a "?" prefix, and whitespace is used as a separator.

Table 1 summarizes the mappings in [2] between DL and LP in the DLP intersection, and shows its representation in RIF. In Table 1, C, D, C_1, C_2 are atomic concepts, P, R, R_1, R_2 are atomic roles, R^{-} and R^{+} are the inverse role and the transitive role of R, respectively, and T, a, b are defined as above. Note that in DLP, a complex concept expression which is a disjunction (e.g. $C_1 \sqcup C_2$) or an existential (e.g. $\exists R.C$) is not allowed in the right side of a concept subsumption axiom.

Table 1. Mapping between LP and DL

LP syntax	DL syntax	RIF
$D(x) \leftarrow C(x)$	$C \sqsubseteq D$	Forall ?x (D(?x) :- C(?x))
$D(x) \leftarrow C(x)$, $C(x) \leftarrow D(x)$	$C \equiv D$	Forall ?x (D(?x) :- C(?x)) Forall ?x (C(?x) :- D(?x))
$R(x,y) \wedge C(y)$	$\exists R.C$	Forall ?x (Exists ?y (And(R(?x ?y) C(?y))))
$C(y) \leftarrow R(x,y)$	$\mathrm{T} \sqsubseteq \forall R.C$	Forall ?x ?y (C(?y) :- R(?x ?y))
$C(x) \leftarrow R(x,y)$	$\mathrm{T} \sqsubseteq \forall R^{-}.C$	Forall ?x ?y (C(?x) :- R(?x ?y))
$C(a)$	$C(a)$	C(a)
$R(a,b)$	$R(a,b)$	R(a b)
$R(x,y) \leftarrow P(x,y)$, $P(x,y) \leftarrow R(x,y)$	$P \equiv R$	Forall ?x ?y (R(?x ?y) :- P(?x ?y)) Forall ?x ?y (P(?x ?y) :- R(?x ?y))
$R(x,y) \leftarrow P(y,x)$, $P(y,x) \leftarrow R(x,y)$	$P \equiv R^{-}$	Forall ?x ?y (R(?x ?y) :- P(?y ?x)) Forall ?x ?y (P(?y ?x) :- R(?x ?y))
$R(x,z) \leftarrow R(x,y), R(y,z)$	$R^{+} \sqsubseteq R$	Forall ?x ?y ?z (R(?x ?z) :- And(R(?x ?y) R(?y ?z)))
$P(x,y) \leftarrow R(x,y)$	$R \sqsubseteq P$	Forall ?x ?y (P(?x ?y) :- R(?x ?y))
$D(x) \leftarrow C_1(x) \wedge C_2(x)$	$C_1 \sqcap C_2 \sqsubseteq D$	Forall ?x (D(?x) :- And(C_1(?x) C_2(?x))
$P(x,y) \leftarrow R_1(x,y) \wedge R_2(x,y)$	$R_1 \sqcap R_2 \sqsubseteq P$	Forall ?x ?y (P(?x ?y) :- And(R_1(?x ?y) R_2(?x ?y))

3. Encoding Uncertainty in RIF

Fuzzy set theory was introduced in [23] as an extension of the classical notion of sets to capture the inherent vagueness (the lack of crisp boundaries) of real-world sets. Formally, a fuzzy set A with respect to a set of elements X (also called a universe) is characterized by a membership function $\mu_A(x)$ which assigns a value in the real unit interval [0,1] to each element $x \in X$. $\mu_A(x)$ gives the degree to which an element x belongs to the set A. Fuzzy logic is a form of multi-valued logic derived from fuzzy set theory to deal with reasoning that is approximate rather than precise. In Fuzzy Logic the degree of truth of a statement can range between 0 and 1 and is not constrained to the two truth values, t and f, as in classic predicate logic [24]. Such degrees can be computed based on various specific membership functions, for example, a trapezoidal function.

In this section, we first present the syntax and semantics for fuzzy Logic Programs based on Fuzzy Sets and Fuzzy Logic [23] and on previous work on fuzzy LP [15, 16, 25], and then propose two

techniques of encoding the semantics of uncertain knowledge based on Fuzzy Logic in the presentation syntax of RIF-BLD using BLD functions and BLD predicates respectively.

3.1. Fuzzy Logic Programs

Rules in van Emden's formalism for fuzzy LP have the syntactic form

$$H \leftarrow_c B_1, \cdots, B_n \qquad (1)$$

where H, B_i are atoms, $n \geq 0$, and the factor c is a real number in the interval (0,1] [15]. For $n = 0$, such fuzzy rules degenerate to fuzzy facts.

The fuzzy LP language proposed by [16, 25] is a generalization of van Emden's work [15]. Rules are constructed from an implication ($\leftarrow$) with a corresponding t-norm adjunction operator (f_1), and another t-norm operator denoted by f_2. A t-norm is a generalization to the many-valued setting of the conjunction connective. In their setting, a rule is of the form $H \leftarrow_{f1} f_2(B_1, \cdots, B_n)\ with-cf\ c$, where the confidence factor c is a real number in the unit interval [0,1] and H, B_i are atoms with truth values in (0, 1]. If we take the operator f_1 as the product following Goguen implication and the operator f_2 as the Gödel t-norm (minimum), this is exactly of the form by van Emden [15].

In the current paper, we follow this work and use the following form to represent a fuzzy rule.

$$H(\vec{x}) \leftarrow B_1(\vec{x}_1), \cdots, B_n(\vec{x}_n) \quad / c \qquad (2)$$

Here $H(\vec{x}), B_i(\vec{x}_i)$ are atoms, $\vec{x}, \vec{x}_i$ are vectors of variables or constants, $n \geq 0$ and the confidence factor c (also called certainty degree) is a real number in the interval (0,1]. For the special case of fuzzy facts this becomes $H \quad / c$. These forms with a "/" symbol have the advantages of avoiding possible confusion with the equality symbol usually used for functions in logics with equality, as well as using a unified and compact format to represent fuzzy rules and fuzzy facts.

The semantics of such fuzzy LP is an extension of classical LP semantics. Let B_R stand for the Herbrand base of a fuzzy knowledge base KB_{LP}. A fuzzy Herbrand interpretation H_I for KB_{LP} is defined as a mapping $B_R \rightarrow [0,1]$. It is a fuzzy subset of B_R under Zadeh's semantics and can be specified by a function val with two arguments: a variable-free atom H (or $B_1, \cdots, B_n$) and a fuzzy Herbrand interpretation H_I. The returned result of the function val is the membership value of H (or $B_1, \cdots, B_n$) under H_I, denoted as $val(H, H_I)$ (or $val(B_i, H_I)$).

Therefore, a fuzzy knowledge base KB_{LP} is true under H_I iff every rule in KB_{LP} is true under H_I. Such a Herbrand interpretation H_I is called a Herbrand model of KB_{LP}. Furthermore, a rule is true under H_I iff each variable-free instance of this rule is true under H_I. A variable-free instance of a rule (**3**) is true under H_I iff $val(H, H_I) \geq c \times \min\{val(B_i, H_I) \mid i \in \{1, \cdots, n\}\}$ ($\min\{\} = 1$ *if* $n = 0$). In other words, such an interpretation can be separated into the following two parts [26-28].

(1) The body of the rule consists of n atoms. Our confidence that all these atoms are true is interpreted under Gödel's semantics for fuzzy logic:
$val((B_1, \cdots, B_n), H_I) = \min\{val(B_i, H_I) \mid i \in \{1, \cdots, n\}\}$

(2) The implication is interpreted as the product:
$val(H, H_I) = c \times val((B_1, \cdots, B_n), H_I)$

For a fuzzy knowledge base KB_{LP}, the reasoning task is a fuzzy entailment problem written as $KB_{LP} \models H \quad / c$ ($H \in B_R, c \in (0,1]$).

Example 3.1. Consider the following fuzzy LP knowledge base:

$cheapFlight(x, y) \leftarrow affordableFlight(x, y) \quad / 0.9$ (1)
$affordableFlight(x, y) \quad / left\_shoulder0k4k1k3k(y)$ (2)

Fig. 1 shows the left__shoulder membership function $left\_shoulder(0, 4000, 1000, 3000)$. We use the name $left\_shoulder0k4k1k3k$ for this parameterization. The function has the mathematical form

$$left\_shoulder0k4k1k3k(y) = \begin{cases} 1 & 0 \le y \le 1000 \\ -0.0005y + 1.5 & 1000 < y \le 3000 \\ 0 & 3000 < y \le 4000 \end{cases}.$$

For example, the certainty degree computed by this function for the fact $affordableFlight(flight0001, 1800)$ is 0.7.

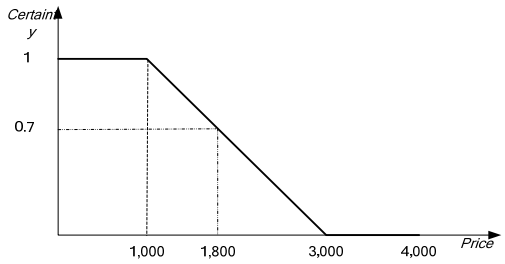


Fig. 1. A Left__shoulder Membership Function

Applying the semantics we discussed, $val(cheapFlight(flight0001, 1800), H_I) = 0.9 * 0.7 = 0.63$, so we have that $KB_{LP} \models cheapFlight(flight0001, 1800) \quad / 0.63$.

Example 3.2. Consider the following fuzzy LP knowledge base:

$A(x) \leftarrow B(x), C(x) \quad / 0.5$ (1)
$C(x) \leftarrow D(x) \quad / 0.5$ (2)
$B(d) \quad / 0.5$ (3)
$D(d) \quad / 0.8$ (4)

We have that $KB_{LP} \models A(d) \quad / 0.2$. The reasoning steps of example 3.2 are described as follows:

$$\begin{aligned} val(A(d), H_I) &= 0.5 \times \min(val(B(d), H_I), val(C(d), H_I)) && **according\ to\ (1) \\ &= 0.5 \times \min(val(B(d), H_I), 0.5 \times val(D(d), H_I)) && **according\ to\ (2) \\ &= 0.5 \times \min(0.5, 0.5 \times val(D(d), H_I)) && **according\ to\ (3) \\ &= 0.5 \times \min(0.5, 0.5 \times 0.8) && **according\ to\ (4) \\ &= 0.5 \times 0.4 \\ &= 0.2 \end{aligned}$$

3.2. Encoding Uncertainty Using RIF Functions

One technique to encode uncertainty in logics with equality such as the current RIF-BLD (where equality in the head is "At Risk") is mapping all predicates to functions and using equality for letting them return uncertainty values [29]. We assume that H, B_i of the fuzzy rule of equation (2) from Section 3.1 contain variables in {?x_1, …, ?x_k} and that the head and body predicates are applied to terms t_1 … t_r and $t_{j,1}$ … $t_{j,sj}$ ($1 \le j \le n$) respectively, which can all be variables, constants or complex terms. A fuzzy rule in the form of equation (2) from Section 3.1 can then be represented in RIF-BLD as (for simplicity, we will omit prefix declarations)

```
Document(
   Group
   (      Forall ?x1 … ?xk (
            h(t1 … tr)=?ch :- And(b1(t1,1 … t1,s1)=?c1  ⋯  bn(tn,1 … tn,sn)=?cn
                                 ?ct =External(numeric-minimum(?c1  ⋯  ?cn))
                                 ?ch=External(numeric-multiply(c ?ct)) )
   )    )
```

Each predicate in the fuzzy rule thus becomes a function with a return value between 0 and 1. The semantics of the fuzzy rules is encoded in RIF-BLD using the built-in functions numeric-multiply from RIF-DTB[30] and an aggregate function numeric-minimum proposed here as an addition to RIF-DTB (this could also be defined using rules).

A fact of the form $H \quad / c$ can be represented in RIF-BLD presentation syntax as

```
h(t1 … tr)=c
```

Example 3.3 We can rewrite example 3.1 using RIF functions as follows:

```
(* <http://example.org/fuzzy/membershipfunction > *)
Document(
   Group
   (      (* "Definition of membership function  left _ shoulder(0,4000,1000,3000) "[] *)
      Forall ?y(
            left__shoulder0k4k1k3k(?y)=1 :- And(External(numeric-less-than-or-equal(0 ?y))
                                               External(numeric-less-than-or-equal(?y 1000))))
      Forall ?y(
      left__shoulder0k4k1k3k(?y)=External(numeric-add(External(numeric-multiply(-0.0005?y)) 1.5))
            :- And(External(numeric-less-than(1000 ?y))
                   External(numeric-less-than-or-equal(?y 3000))))
      Forall ?y(
            left__shoulder0k4k1k3k(?y)=0 :- And(External(numeric-less-than(3000 ?y))
                                               External(numeric-less-than-or-equal(?y 4000))))
         . .
   )    )
```

Note that membership function $left\_shoulder(0,4000,1000,3000)$ is encoded as three rules.

```
Document(
   Import (<http://example.org/fuzzy/membershipfunction >)
   Group
   (      Forall ?x ?y(
            cheapFlight(?x ?y)=?ch :- And(affordableFlight(?x ?y)=?c1
                                          ?ch=External(numeric-multiply(0.4 ?c1))))
         Forall ?x ?y(affordableFlight(?x ?y)=left__shoulder0k4k1k3k(?y))
   )    )
```

The Import statement loads the left__shoulder0k4k1k3k function defined at the given "<…>" IRI.

Example 3.4 We can rewrite example 3.2 in RIF functions as follows:

```
Document(
   Group
   (      Forall ?x(
            A(?x)=?ch :- And(B(?x)=?c1 C(?x)=?c2
                             ?ct =External(numeric-minimum(?c1 ?c2))
                             ?ch=External(numeric-multiply(0.5 ?ct))))
         Forall ?x(
            C(?x)= ?ch :- And(D(?x)=?c1 ?ch=External(numeric-multiply(0.5 ?c1))) )
            B(d)=0.5
            D(d)=0.8
   )    )
```

3.3 Encoding Uncertainty Using RIF Predicates

Another encoding technique is making all n-ary predicates into (1+n)-ary predicates, each being functional in the first argument which captures the certainty factor of predicate applications. A fuzzy rule in the form of equation (2) from Section 3.1 can then be represented in RIF-BLD as

```
Document(
  Group
  (     Forall ?x1 … ?xk (
          h(?ch t1 … tr) :- And(b1(?c1 t1,1 … t1,s1) ⋯ bn(?cn tn,1 … tn,sn)
                               ?ct =External(numeric-minimum(?c1 ⋯ ?cn))
                               ?ch=External(numeric-multiply(c ?ct)) )
  )     )
```

Likewise, a fact of the form $H \quad / c$ can be represented in RIF-BLD as

```
h(c t1 … tr)
```

Example 3.5 We can rewrite example 3.1 in RIF predicates as follows,

```
Document(
  Import (<http://example.org/fuzzy/membershipfunction >)
  Group
  (
    Forall ?x ?y(
      cheapFlight(?ch ?x ?y) :- And(affordableFlight(?c1 ?x ?y)
                                    ?ch=External(numeric-multiply(0.4 ?c1)))
    )
    Forall ?x ?y(affordableFlight(?c1 ?x ?y) :- ?c1 =left__shoulder0k4k1k3k(?y))
  )     )
```

4. Uncertainty Extension of RIF

In this section, we adapt the definition of the set of truth values from RIF-FLD and its semantic structure. We then propose a RIF extension for directly representing uncertain knowledge.

4.1 Definition of Truth Values and Truth Valuation

In previous sections, we showed how to represent the semantics of fuzzy LP with RIF functions and predicates in RIF presentation syntax. We now propose to introduce a new dialect for RIF, RIF Uncertainty Rule Dialect (RIF-URD), so as to directly represent uncertain knowledge and extend the expressive power of RIF.

The set TV of truth values in RIF-BLD consists of just two values, t and f. This set forms a two-element Boolean algebra with $t = 1$ and $f = 0$. However, in order to represent uncertain knowledge, all intermediate truth values must be allowed. Therefore, the set TV of truth values is extended to a set with infinitely many truth values ranging between 0 and 1. Our uncertain knowledge representation is specifically based on Fuzzy Logic, thus a member function maps a variable to a truth value in the 0 to 1 range.

Definition 1. (Set of truth values as a specialization of the set in RIF-FLD). In RIF-FLD, $\leq_t$ denotes the truth order, a binary relation on the set of truth values TV. Instantiating RIF-FLD, which just requires a partial order, the set of truth values in RIF-URD is equipped with a total order over the 0 to 1 range. In RIF-URD, we specialize $\leq_t$ to $\leq$, denoting the numerical truth order. Thus, we observe that the following statements hold for any element e_i, e_j or e_k in the set of truth values TV in the 0 to 1 range, justifying to write it as the interval [0,1].

(1) The set TV is a complete lattice with respect to $\leq$, i.e., the least upper bound (lub) and the greatest lower bound (glb) exist for any subset of $\leq$.

(2) Antisymmetry. If $e_i \leq e_j$ and $e_j \leq e_i$ then $e_i = e_j$.

(3) Transitivity. If $e_i \leq e_j$ and $e_j \leq e_k$ then $e_i \leq e_k$.

(4) Totality. Any two elements should satisfy one of these two relations: $e_i \leq e_j$ or $e_j \leq e_i$.

(5) The set TV has an operator of negation, $\sim: TV \rightarrow TV$, such that

a). $\sim e_i = 1 - e_i$.

b). ~ is self-inverse, i.e., $\sim\sim e_i = e_i$.

Let $TVal(\varphi)$ denote the truth value of a non-document formula, φ, in RIF-BLD. $TVal(\varphi)$ is a mapping from the set of all non-document formulas to TV, I denotes an interpretation, and c is a real number in the interval (0,1].

Definition 2. **(*Truth valuation adapted from RIF-FLD*)**. Truth valuation for well-formed formulas in RIF-URD is determined as in RIF-FLD, adapting the following three cases.

(8) Conjunction (glb_t becomes min): $TVal_I(And(B_1 \cdots B_n)) = \min(TVal(B_1) \cdots TVal(B_n))$.

(9) Disjunction (lub_t becomes max): $TVal_I(Or(B_1 \cdots B_n)) = \max(TVal(B_1) \cdots TVal(B_n))$

(11) Rule implication (t becomes 1, f becomes 0, condition valuation is multiplied with c):

$TVal_I(conclusion :- condition \quad / c) = 1$ if $TVal_I(conclusion) \geq c \times TVal_I(condition)$

$TVal_I(conclusion :- condition \quad / c) = 0$ if $TVal_I(conclusion) < c \times TVal_I(condition)$

4.2 Using RIF-URD to Represent Uncertain Knowledge

A fuzzy rule in the form of equation (2) from Section 3.1 can be directly represented in RIF-URD as

```
Document(
   Group
   (
      Forall ?x1 … ?xk (
            h(t1 … tr) :- And(b1(t1,1 … t1,s1)  ⋯  bn(tn,1 … tn,sn))
      )   / c
   )
```

Likewise, a fact of the form $H \quad / c$ can be represented in RIF-URD as

```
h(t1 … tr)   / c
```

Such a RIF-URD document of course cannot be executed by an ordinary RIF-compliant reasoner. RIF-URD-compliant reasoners will need to be extended to support the above semantics and the reasoning process shown in Section 3.1.

Example 3.6 We can directly represent example 3.1 in RIF-URD as follows:

```
Document(
   Import (<http://example.org/fuzzy/membershipfunction >)
   Group
   (
      Forall ?x ?y(
            cheapFlight(?x ?y) :- affordableFlight(?x ?y)
      )   / 0.4
      Forall ?x ?y(affordableFlight(?x ?y))   / left__shoulder0k4k1k3k(?y)
   )      )
```

5. Fuzzy Description Logic Programs and Their Representation in RIF

In this section, we extend Description Logic Programs (DLP) [2] to support mappings between fuzzy DL and fuzzy LP; we also show how to represent such mappings in RIF-BLD and RIF-URD based on the three uncertainty treatment methods addressed in previous sections.

Based on Fuzzy Sets and Fuzzy Logic [23], the semantics for fuzzy DL [12] and fuzzy LP [15], as well as the previous work cited in Section 1 and 3, we extend the work on Description Logic Programs (DLP) [2] to fuzzy Description Logic Programs (Fuzzy DLP).

Since DL is a subset of FOL, it can also be seen in terms of that subset of FOL, where individuals are equivalent to FOL constants, concepts and concept descriptions are equivalent to FOL formulas with one free variable, and roles and role descriptions are equivalent to FOL formulas with two free variables.

A concept inclusion axiom of the form $C \sqsubseteq D$ is equivalent to an FOL sentence of the form $\forall x.C(x) \rightarrow D(x)$, i.e. an FOL implication. In uncertainty representation and reasoning, it is important to represent and compute the degree of subsumption between two fuzzy concepts, i.e., the degree of overlap, in addition to crisp subsumption. Therefore, we consider fuzzy axioms of the form $C \sqsubseteq D \quad = c$ generalizing the crisp $C \sqsubseteq D$. The above equivalence leads to a straightforward mapping from a fuzzy concept inclusion axiom of the form $C \sqsubseteq D = c \quad (c \in (0,1])$ to an LP rule as follows: $D(x) \leftarrow C(x) \quad / c$.

The intersection of two fuzzy concepts in fuzzy DL is defined as $(C_1 \sqcap C_2)^I(x) = \min(C_1^I(x), C_2^I(x))$; therefore, a fuzzy concept inclusion axiom of the form $C_1 \sqcap C_2 \sqsubseteq D \quad = c$ including the intersection of C_1 and C_2 can be transformed to an LP rule $D(x) \leftarrow C_1(x), C_2(x) \quad / c$. Here the certainty degree of (variable-free) instantiations of the atom $D(x)$ is defined by the valuation $val(D, H_I) = c \times \min\{val(C_i, H_I) \mid i \in \{1,2\}\}$. It is easy to see that such a fuzzy concept inclusion axiom can be extended to include the intersection of n concepts ($n > 2$).

Similarly, a role inclusion axiom of the form $R \sqsubseteq P$ is equivalent to an FOL sentence consisting of an implication between two roles. Thus we map a fuzzy role inclusion axiom of the form $R \sqsubseteq P = c$ ($c \in (0,1]$) to a fuzzy LP rule as $P(x,y) \leftarrow R(x,y) \quad / c$. Moreover, $\bigcap_{i=1}^{n} R_i \sqsubseteq P = c$ can be transformed to $P(x,y) \leftarrow R_1(x,y), \cdots, R_n(x,y) \quad / c$.

A concept equivalence axiom of the form $C \equiv D$ can be represented as a symmetrical pair of FOL implications: $\forall x.C(x) \rightarrow D(x)$ and $\forall x.D(x) \rightarrow C(x)$. Therefore, we map the 'fuzzified' equivalence axiom $C \equiv D \quad = c$ into $C(x) \leftarrow D(x) \quad / c$ and $D(x) \leftarrow C(x) \quad / c \quad (c \in (0,1])$. As later examples show, such mappings in hybrid knowledge bases are directed from rules to DL expressions, hence if we have two rules of the forms $C(x) \leftarrow D(x) \quad / c_1$ and $D(x) \leftarrow C(x) \quad / c_2 \quad (c_1, c_2 \in (0,1])$, they are mapped to a DL expression as $C \equiv D \quad = c$ with $c = \min(c_1, c_2)$. Similarly, we map two rules $R(x,y) \leftarrow P(x,y) \quad / c_1$ and $P(x,y) \leftarrow R(x,y) \quad / c_2$ into a role equivalence axiom of the form $R \equiv P \quad = \min(c_1, c_2)$, as well as two rules $R(x,y) \leftarrow P(y,x) \quad / c_1$ and $P(y,x) \leftarrow R(x,y) \quad / c_2$ into an inverse role equivalence axiom of the form $P \equiv R^- \quad = \min(c_1, c_2)$.

A DL assertion $C(a)$ (respectively, $R(a,b)$) is equivalent to an FOL atom of the form $C(a)$ (respectively, $R(a,b)$), where a and b are individuals. Therefore, a fuzzy DL concept-individual assertion of the form $C(a) = c$ corresponds to a ground fuzzy atom $C(a) \quad / c$ in fuzzy LP, while a fuzzy DL role-individual assertion of the form $R(a,b) = c$ corresponds to a ground fuzzy fact $R(a,b) \quad / c$.

Table 2 summarizes the mappings in Fuzzy DLP. For simplicity, in Fuzzy DLP as defined in this paper we do not use fuzzy forms for all of DLP, excluding value restrictions and transitive role axiom, and assuming $c = 1$ whenever $/ c$ is omitted.

Table 2. Representing Fuzzy DLP in RIF

LP syntax	$D(x) \leftarrow C_1(x), \cdots, C_n(x) \quad / c$
DL syntax	$\bigcap_{i=1}^{n} C_i \sqsubseteq D \quad = c$
RIF function	Forall ?x(D(?x)=?c_h :- And(C_1(?x)=?c_1 ⋯ C_n(?x)=?c_n ?c_t =External(numeric-minimum(?c_1 ⋯ ?c_n)) ?c_h=External(numeric-multiply(c ?c_t)))
RIF predicate	Forall ?x(D(?c_h ?x) :- And(C_1(?c_1 ?x) ⋯ C_n(?c_n ?x) ?c_t =External(numeric-minimum(?c_1 ⋯ ?c_n)) ?c_h=External(numeric-multiply(c ?c_t)))

RIF-URD	Forall ?x(D(?x) :- And(C_1(?x) ⋯ C_n(?x))) / c	
LP syntax	$P(x,y) \leftarrow R_1(x,y),\cdots,R_n(x,y) \quad / c$	
DL syntax	$\bigcap_{i=1}^{n} R_i \sqsubseteq P \quad = c$	
RIF function	Forall ?x ?y(P(?x ?y)=?c_h :- And(R_1(?x ?y)=?c_1 ⋯ R_n(?x ?y)=?c_n ?c_t =External(numeric-minimum(?c_1 ⋯ ?c_n)) ?c_h=External(numeric-multiply(c ?c_t)))	
RIF predicate	Forall ?x ?y(P(?c_h ?x ?y) :- And(R_1(?c_1 ?x ?y) ⋯ R_n(?c_n ?x ?y) ?c_t =External(numeric-minimum(?c_1 ⋯ ?c_n)) ?c_h=External(numeric-multiply(c ?c_t)))	
RIF-URD	Forall ?x ?y(P(?x ?y) :- And(R_1(?x ?y) ⋯ R_n(?x ?y))) / c	
LP syntax	$C(x) \leftarrow D(x) \quad / c, \quad D(x) \leftarrow C(x) \quad / c'$	
DL syntax	$C \equiv D \quad = \min(c,c')$	
RIF function	Forall ?x(C(?x)=?c_h :- And(D(?x)=c_1 ?c_h=External(numeric-multiply(c c_1))) Forall ?x(D(?x)=?c_h :- And(C(?x)=c_1 ?c_h=External(numeric-multiply(c' c_1)))	
RIF predicate	Forall ?x(C(?c_h ?x) :- And(D(?c_1 ?x) ?c_h=External(numeric-multiply(c c_1))) Forall ?x(D(?c_h ?x) :- And(C(?c_1 ?x) ?c_h=External(numeric-multiply(c' c_1)))	
RIF-URD	Forall ?x(C(?x) :- D(?x)) / c, Forall ?x(D(?x) :- C(?x)) / c	
LP syntax	$R(x,y) \leftarrow P(x,y) \quad / c, \quad P(x,y) \leftarrow R(x,y) \quad / c'$	
DL syntax	$R \equiv P \quad = \min(c,c')$	
RIF function	Forall ?x ?y(R(?x ?y)=?c_h :- And(P(?x ?y)=c_1 ?c_h=External(numeric-multiply(c c_1))) Forall ?x ?y(P(?x ?y)=?c_h :- And(R(?x ?y)=c_1 ?c_h=External(numeric-multiply(c' c_1)))	
RIF predicate	Forall ?x ?y(R(?c_h ?x ?y) :- And(P(?c_1 ?x ?y) ?c_h=External(numeric-multiply(c c_1))) Forall ?x ?y(P(?c_h ?x ?y) :- And(R(?c_1 ?x ?y) ?c_h=External(numeric-multiply(c' c_1)))	
RIF-URD	Forall ?x ?y(R(?x ?y) :- P(?x ?y)) / c, Forall ?x ?y(P(?x ?y) :- R(?x ?y)) / c	
LP syntax	$R(x,y) \leftarrow P(y,x) \quad / c, \quad P(y,x) \leftarrow R(x,y) \quad / c'$	
DL syntax	$P \equiv R^{-} \quad = \min(c,c')$	
RIF function	Forall ?x ?y(R(?x ?y)=?c_h :- And(P(?y ?x)=c_1 ?c_h=External(numeric-multiply(c c_1))) Forall ?x ?y(P(?y ?x)=?c_h :- And(R(?x ?y)=c_1 ?c_h=External(numeric-multiply(c' c_1)))	
RIF predicate	Forall ?x ?y(R(?c_h ?x ?y) :- And(P(?c_1 ?y ?x) ?c_h=External(numeric-multiply(c c_1))) Forall ?x ?y(P(?c_h ?y ?x) :- And(R(?c_1 ?x ?y) ?c_h=External(numeric-multiply(c' c_1)))	
RIF-URD	Forall ?x ?y(R(?x ?y) :- P(?y ?x)) / c, Forall ?x ?y(P(?y ?x) :- R(?x ?y)) / c	
LP syntax	$C(a) \quad / c$	$R(a,b) \quad / c$
DL syntax	$C(a) \quad = c$	$R(a,b) \quad = c$
RIF function	C(a)=c	R(a b)=c
RIF predicate	C(c a)	R(c a b)
RIF-URD	C(a) /c	R(a b) /c

In summary, Fuzzy DLP is an extension of Description Logic Programs supporting the following concept and role inclusion axioms, range and domain axioms, concept and role assertion axioms to build a knowledge base: $\bigcap_{i=1}^{n} C_i \sqsubseteq D \quad = c$, $C \equiv D \quad = c$, $\top \sqsubseteq \forall R.C$, $\top \sqsubseteq \forall R^{-}.C$,

$\bigcap_{i=1}^{n} R_i \sqsubseteq P \quad = c$, $P \equiv R \quad = c$, $P \equiv R^- \quad = c$, $R^+ \sqsubseteq R$, $C(a) \quad = c$, and $R(a,b) \quad = c$, where $C, D, C_1, \cdots C_n$ are atomic concepts, P, R are atomic roles, a, b are individuals, $c \in (0,1]$ and $n \geq 1$. Notice that the crisp DLP axioms in DLP are special cases of their counterparts in Fuzzy DLP. For example, $C \sqsubseteq D$ is equal to its fuzzy version $\bigcap_{i=1}^{n} C_i \sqsubseteq D \quad = c$ for $n = 1$ and $c = 1$.

In previous sections, we presented two techniques of encoding uncertainty in RIF and proposed a method based on an extension of RIF for uncertainty representation. Subsequently, we also showed how to represent Fuzzy DLP in RIF-BLD and RIF-URD in Table 2.

Layered on Fuzzy DLP, we can build fuzzy hybrid knowledge bases in order to build fuzzy rules on top of ontologies for the Semantic Web and reason on such KBs.

Definition 3. A fuzzy hybrid knowledge base KB_{hf} is a pair $< K_{DL}, K_{LP} >$, where K_{DL} is the finite set of (fuzzy) concept inclusion axioms, role inclusion axioms, and concept and role assertions of a decidable DL defining an ontology. K_{LP} consists of a finite set of (fuzzy) hybrid rules and (fuzzy) facts.

A hybrid rule r in K_{LP} is of the following generalized form (we use the BNF choice bar, |):

$$\left(H(\vec{y}) \,\middle|\, \& H(\vec{z})\right) \leftarrow B_1(\vec{y}_1), \cdots, B_l(\vec{y}_l), \& Q_1(\vec{z}_1), \cdots, \& Q_n(\vec{z}_n) \quad / c \qquad \textbf{(4)}$$

Here, $H(\vec{y}), H(\vec{z}), B_i(\vec{y}_i), Q_j(\vec{z}_j)$ are atoms, & precedes a DL atom, $\vec{y}, \vec{z}, \vec{y}_i, \vec{z}_j$ are vectors of variables or constants, where $\vec{y}$ and each $\vec{y}_i$ have arbitrary lengths, $\vec{z}$ and each $\vec{z}_j$ have length 1 or 2, and $c \in (0,1]$. Also, & atoms and $/c$ degrees are optional (if all & atoms and $/c$ degrees are missing from a rule, it becomes a classical rule of Horn Logic).

Such a fuzzy hybrid rule must satisfy the following constraints:

(1) H is either a DL predicate or a rule predicate ($H \in \sum_T \bigcup \sum_R$). H is a DL predicate with the form $\& H$, while it is a rule predicate without the & operator.

(2) Each $B_i \quad (1 < i \leq l)$ is a rule predicate ($B_i \in \sum_R$), and $B_i(y_i)$ is an LP atom.

(3) Each $Q_j \quad (1 < j \leq n)$ is a DL predicate ($Q_j \in \sum_T$), and $Q_j(z_j)$ is a DL atom.

(4, pure DL rule) If a hybrid rule has head $\& H$, then each atom in the body must be of the form $\& Q_j \, (1 < j \leq n)$; in other words, there is no $B_i \, (l = 0)$. A head $\& H$ without a body ($l = 0$, $n = 0$) constitutes the special case of a pure DL fact.

Example 5.1. The rule $\& CheapFlight(x, y) \leftarrow AffordableFlight(x, y) \quad / c$ is not a pure DL rule according to (4), hence not allowed in our hybrid knowledge base, while $CheapFlight(x, y) \leftarrow \& AffordableFlight(x, y) \quad / c$ is allowed.

A hybrid rule of the form $\& CheapFlight(x, y) \leftarrow \& AffordableFlight(x, y) \quad / c$ can be mapped to a fuzzy DL role subsumption axiom $AffordableFlight \sqsubseteq CheapFlight \quad = c$.

Our approach thus allows DL atoms in the head of hybrid rules which satisfy the constraint (4, pure DL rule), supporting the mapping of DL subsumption axioms to rules. We also deal with fuzzy subsumption of fuzzy concepts of the form $C \sqsubseteq D \quad = c$ as shown in Example 5.1.

An arbitrary hybrid knowledge base cannot be fully embedded into the knowledge representation formalism of RIF with uncertainty extensions. However, in the proposed Fuzzy DLP subset, DL components (DL axioms in LP syntax) can be mapped to LP rules and facts in RIF. A RIF-compliant reasoning engine can be extended to do reasoning on a hybrid knowledge base on top of Fuzzy DLP by adding a module that first maps atoms in rules to DL atoms, and then derives the reasoning answers with a DL reasoner, e.g. Racer or Pellet, or with a fuzzy DL reasoner, e.g. fuzzyDL [31]. The specification of such a reasoning algorithm for a fuzzy hybrid knowledge base KB_{hf} based on Fuzzy DLP and a query q is treated in a companion paper[32].

6. Conclusion

In this paper, we propose two different principles of representing uncertain knowledge, encodings in RIF-BLD and an extension leading to RIF-URD. We also present a fuzzy extension to Description Logic Programs, namely Fuzzy DLP. We address the mappings between fuzzy DL and fuzzy LP within Fuzzy DLP, and give Fuzzy DLP representations in RIF. Since handling uncertain information, such as with fuzzy logic, was listed as a RIF extension in the RIF Working Group Charter [18] and RIF-URD is a manageable extension to RIF-BLD, we propose here a version of URD as a RIF dialect, realizing a fuzzy rule sublanguage for the RIF standard.

Our fuzzy extension directly relates to Lotfi Zadeh's semantics of fuzzy sets and fuzzy logic. We do not yet cover here other researchers' semantics, for example, Jan Lukasiewicz's. Nor do we cover other uncertainty formalisms, based on probability theory, possibilities, or rough sets. Future work will include generalizing our fuzzy extension of hybrid knowledge bases to some of these different kinds of uncertainty, and parameterizing RIF-URD to support different theories of uncertainty in a unified manner.

Complementing the RIF-URD presentation syntax, XML elements and attributes like <degree>, @mapkind, and @kind, following those of Fuzzy RuleML, can be introduced for the RIF-URD XML syntax. Another direction of future work would be the extension of uncertain knowledge to various combination strategies of DL and LP without being limited to DL queries.

References

1. D. Koller, A. Levy and A. Pfeffer, "P-CLASSIC: A tractable probabilistic description logic," in *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97),* 1997, pp. 390-397.
2. B. N. Grosof, I. Horrocks, R. Volz and S. Decker, "Description logic programs: Combining logic programs with description logic," in *Proceedings of the 12th International Conference on World Wide Web,* 2003, pp. 48-57.
3. F. M. Donini, M. Lenzerini, D. Nardi and A. Schaerf, "AL-log: Integrating Datalog and Description Logics," *Journal of Intelligent Information Systems,* vol. 10, pp. 227-252, 1998.
4. I. Horrocks, P. Patel-schneider, S. Bechhofer and D. Tsarkov, "OWL Rules: A Proposal and Prototype Implementation," *Journal of Web Semantics,* vol. 3, pp. 23-40, 2005.
5. J. Mei, Z. Q. Lin, H. Boley, J. Li and V. C. Bhavsar, "The Datalog^{DL} Combination of Deduction Rules and Descriprition Logics," *Computational Intelligence,* vol. 23, pp. 356-372, 2007.
6. B. Motik, U. Sattler and R. Studer, "Query Answering for OWL-DL with rules," *Web Semantics: Science, Services and Agents on the World Wide Web,* vol. 3, pp. 41-60, 7. 2005.
7. C. V. Damasio, J. Pan, G. Stoilos and U. Straccia, "Representing Uncertainty in RuleML," *Fundamenta Informaticae,* vol. 82, pp. 1-24, 2008.
8. K. Laskey, K. Laskey, P. Costa, M. Kokar, T. Martin and T. Lukasiewicz, "W3C incubator group report," W3C, Tech. Rep. http://www.w3.org/2005/Incubator/urw3/wiki/DraftFinalReport, 05 March, 2008.
9. M. Jaeger, "Probabilistic reasoning in terminological logics," in *Proc. of the 4th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'94),* 1994, pp. 305-316.
10. U. Straccia, "A fuzzy description logic," in *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI'98),* 1998, pp. 594-599.
11. U. Straccia, "Reasoning within Fuzzy Description Logics," *Journal of Artificial Intelligence Research,* vol. 14, pp. 137-166, 2001.
12. U. Straccia, "Towards a fuzzy description logic for the semantic web (preliminary report)," in *2nd European Semantic Web Conference (ESWC-05),* 2005, pp. 167-181.

13. G. Stoilos, G. Stamou, J. Pan, V. Tzouvaras and I. Horrocks, "Reasoning with Very Expressive Fuzzy Description Logics," *Journal of Artificial Intelligence Research,* vol. 30, pp. 273-320, 2007.
14. T. Lukasiewicz, "Expressive probabilistic description logics," *Artificial Intelligence,* vol. 172, pp. 852-883, 2008.
15. van Emden, M. H., "Quantitative Deduction and its Fixpoint Theory," *Journal of Logic Programming,* vol. 30, pp. 37-53, 1986.
16. P. Vojtás and L. Paulík, "Soundness and completeness of non-classical SLD-resolution," in *Extensions of Logic Programming,* 1996, pp. 289-301.
17. C. V. Damasio and L. M. Pereira, "Monotonic and residuated logic programs," in *Proceedings of the 6th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty,* 2001, pp. 748-759.
18. RIF Working Group. 2007, Rule interchange format (RIF).
19. H. Boley and M. Kifer, "RIF Basic Logic Dialect," W3C Working Draft (Last Call), Tech. Rep. http://www.w3.org/TR/rif-bld/, 30 July, 2008.
20. H. Boley and M. Kifer, "RIF Framework for Logic Dialects," W3C Working Draft, Tech. Rep. http://www.w3.org/TR/rif-fld/, 30 July, 2008.
21. U. Straccia, "Fuzzy description logic programs," in *Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, (IPMU-06),* 2006, pp. 1818-1825.
22. T. Venetis, G. Stoilos, G. Stamou and S. Kollias, "f-DLPs: Extending description logic programs with fuzzy sets and fuzzy logic," in *Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007. IEEE International,* 2007, pp. 1-6.
23. L. A. Zadeh, "Fuzzy sets," *Information and Control,* vol. 8, pp. 338-353, 1965.
24. V. Novák, *Mathematical Principles of Fuzzy Logic.* Dodrecht: Kluwer Academic, 1999.
25. P. Vojtás, "Fuzzy Logic Programming," *Fuzzy Sets and Systems,* vol. 124, pp. 361-370, 2004.
26. P. Hájek, "Fuzzy logic from the logical point of view," in *SOFSEM '95: Proceedings of the 22nd Seminar on Current Trends in Theory and Practice of Informatics,* 1995, pp. 31-49.
27. P. Hájek, *Metamathematics of Fuzzy Logic.* Kluwer, 1998.
28. P. Hájek, "Fuzzy Logic and Arithmetical Hierarchy III," *Studia Logica,* vol. 68, pp. 129-142, 2001.
29. V. Hall, "Uncertainty-valued Horn Clauses," Tech. Rep. http://www.dfki.uni-kl.de/~vega/relfun+/fuzzy/fuzzy.ps, 1994.
30. A. Polleres, H. Boley and M. Kifer, "RIF datatypes and built-ins 1.0," W3C Working Draft (Last Call), Tech. Rep. http://www.w3.org/2005/rules/wiki/DTB, 30 July, 2008.
31. F. Bobillo and U. Straccia, "fuzzyDL: An expressive fuzzy description logic reasoner," in *Proceedings of the 2008 International Conference on Fuzzy Systems (FUZZ-08),* 2008.
32. J. Zhao and H. Boley, "Combining Fuzzy Description Logics and Fuzzy Logic Programs," in *Proceedings of IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology (WI-IAT 2008) Workshops, To Appear,* 2008.

Uncertainty Reasoning for the World Wide Web: Report on the URW3-XG Incubator Group

Kenneth J. Laskey[#]
MITRE Corporation, M/S H305
7515 Colshire Drive
McLean, VA 22102-7508 USA
klaskey@mitre.org

Kathryn Blackmond Laskey
Department of Systems Engineering
and Operations Research
George Mason University
4400 University Drive
Fairfax, VA 22030-4444 USA
klaskey@gmu.edu

Abstract. The Uncertainty Reasoning for the World Wide Web Incubator Group (URW3-XG) was chartered as a means to explore and better define the challenges of reasoning with and representing uncertain information in the context of the World Wide Web. The objectives of the URW3-XG were: (1) To identify and describe situations on the scale of the World Wide Web for which uncertainty reasoning would significantly increase the potential for extracting useful information; and (2) To identify methodologies that can be applied to these situations and the fundamentals of a standardized representation that could serve as the basis for information exchange necessary for these methodologies to be effectively used. This paper describes the activities undertaken by the URW3-XG, the recommendations produced by the group, and next steps required to carry forward the work begun by the group.

1 Introduction

The Uncertainty Reasoning for the World Wide Web Incubator Group (URW3-XG) was proposed [1] during the 2006 Uncertainty Reasoning for the Semantic Web workshop [2] as a means to explore and better define the challenges of reasoning with and representing uncertain information in the context of the Semantic Web. In addition, it was intended to identify situations in which the combination of semantics and uncertainty could further the Web Services vision of quickly and efficiently composing services and data resources to address the needs of users in an ever-changing world.

The 2006 workshop included a Use Case Challenge [3] to generate an initial collection of use cases and to gauge the interest of the workshop participants in continu-

[#] The author's affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions or viewpoints expressed by the author.

ing as a W3C XG or through some other collaboration venue. The Use Case Challenge generated a lively interchange of ideas, and the participants overwhelmingly agreed to create the XG to continue the work.

2 W3C Incubator (XG) Process

As noted in [1], the World Wide Web Consortium (W3C) [4] created the Incubator process [5] to provide a formal, yet flexible venue to better understand Web-related challenges and their potential solutions. It encourages a public exploration of issues and potential solutions before the solutions are mature enough for standardization. It also provides a "head start" if the Incubator experimental group, the XG, is able to adequately formulate principles and techniques that gain consensus in the wider community.

The URW3-XG was in operation [6] from 5 March 2007 until its final report [7] was published by the W3C on 22 April 2008. The group included 25 participants from North and South America, Europe, and Australia. Participants came from a range of time zones spanning 18 hours. The group conducted over 20 telecons, with an average duration between 90 and 120 minutes. In addition, face-to-face meetings of subsets of the XG were held at the 5th ISWC (Busan - Korea) and the SUM conference in College Park, Maryland USA. The telecons were supported by the W3C resources (e.g. telecon bridge, IRC, RSSAgent, etc). Meeting results and action items were catalogued in online Minutes [6].

The objectives of the URW3-XG were twofold:

- To identify and describe situations on the scale of the World Wide Web for which uncertainty reasoning would significantly increase the potential for extracting useful information; and,
- To identify methodologies that can be applied to these situations and the fundamentals of a standardized representation that could serve as the basis for information exchange necessary for these methodologies to be effectively used.

3 Results of the URW3-XG Effort

The Final Report [7] was the major deliverable of the URW3-XG. It describes the work done by the XG, identifies elements of uncertainty that need to be represented to support reasoning under uncertainty for the World Wide Web, and provides an overview of the applicability to the World Wide Web of various uncertainty reasoning techniques (in particular, probability theory, fuzzy logic, and belief functions) and the information that needs to be represented for effective uncertainty reasoning to be

possible. The report concludes with a discussion on the benefits of standardization of uncertainty representation to the World Wide Web and the Semantic Web and provides a series of recommendations for continued work. The report also includes a Reference List of work relevant to the challenge of developing standardized representations for uncertainty and exploiting them in Web-based services and applications.

A major part of the work was development of a set of use cases illustrating conditions under which uncertainty reasoning is important. Another major effort was the development of an Uncertainty Ontology that was used to categorize uncertainty found in the use cases. These products are described briefly in the following sections. Section 4 then details the conclusions and recommendations from the report.

3.1 The Uncertainty Ontology

The Uncertainty Ontology is a simple ontology developed to demonstrate some basic functionality of exchanging uncertain information. It was used to classify the use cases developed by the URW3-XG with the intent of obtaining a relatively complete coverage of the functionalities related to uncertainty reasoning about information available on the World Wide Web. The top level of the ontology is shown in Figure 1. According to the ontology, uncertainty is associated with sentences that make assertions about the world, and are asserted by agents (human or computer). The uncertainty derivation may be objective (via a formal, repeatable process) or subjective (judgment or guess). Uncertainty type includes ambiguity, empirical uncertainty, randomness, vagueness, inconsistency and incompleteness. Uncertainty models include probability, fuzzy logic, belief functions, rough sets, and other mathematical models for reasoning under uncertainty. Uncertainty nature includes aleatory (chance; inherent in the phenomenon) or epistemic (belief; due to limited knowledge of the agent).

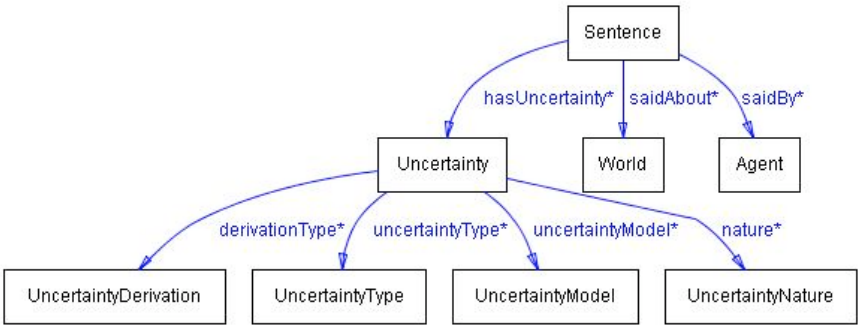


Figure 1 Top level of URW3-XG Uncertainty Ontology

While this ontology served the purpose of focusing discussion of the use cases, allowing use case developers to show examples of annotation of uncertainty, the ontology was only meant to provide a starting point to be refined through an iterative process. Further development of a more complete ontology for annotating uncertainty is one of the XG's recommendations.

3.2 The URW3-XG Uncertainty Use Cases

Building on the work started during the Use Case Challenge, the URW3-XG developed 16 use cases to identify how the representation of uncertainty would help to address issues in Web reasoning that cannot be properly represented with current deterministic approaches. The use cases were developed for the most part using a common template. Occurrences of uncertainty in the use case descriptions were annotated with information from the Uncertainty Ontology. One use case, entitled Buying Speakers, is shown in the Appendix.

The analysis of the use cases indicated that a representation of uncertainty would be required to represent both uncertainty inherent in the data and uncertainty related to the processing of data and the delivery of processing results. This will be discussed further in section 4.

4 Key Conclusions and Recommendations

In automated data processing, we often face situations where Boolean truth-values are unknown, unknowable, or inapplicable. This is true for a wide variety of data and information processing applications, and therefore it should be no surprise that the methodologies considered by the XG are popular in contexts other than the Web. The use cases considered by the XG concerned reasoning challenges specific to the Web, such as discovery of Web Services, order processing via Web Services, and the like. The XG's work confirmed the hypothesis that a unified model for uncertainty annotation of Web resources would provide value for deductive engines, and this could be further facilitated by an ontology characterizing the types and sources of uncertainty.

The work with the Uncertainty Ontology suggested that a finer grained extension may be useful. Such an extension could provide a means to visualize a possible evolution of an upper level Uncertainty Ontology. The conclusions go on to focus especially on finer classification of Machine Agents and uncertainty caused by lack of knowledge of a machine agent.

With respect to the kinds of uncertainty observed in the use cases, it was noted that uncertainty may be an inherent part of the data or may be related to the processing that produces results. In the first case, the standardization should provide a single syntactical system so that people can identify and process this information quickly.

For example, one may want to be able to communicate information that Study X shows that people with property Y have an Z% increased likelihood of disease W. The ability to communicate such information using a common interchange syntax could be extremely useful in a number of web-based applications. Such characterizations of data uncertainty may require something like uncertain extensions to OWL (i.e., probabilistic, fuzzy, belief function, random set, rough set, and hybrid uncertain extensions to OWL).

The second kind of uncertainty involves reasoning on the part of the tools used to access and share web information. For example, if a web service uses uncertainty reasoning to find and rank hotel rooms, the need would be to represent meta-information about the reasoning models and assumptions. This could facilitate the development of trust models, or allow the identification of compatible web services to increase the likelihood that the results are consistent with the user preferences. Here the representation would include determining how to represent the meta-information on processing and deciding how detailed the meta-information would need to be and where it would reside.

The deliberations and conclusions of the URW3-XG led to the following recommendations:

- A principled means for expressing uncertainty will increase the usefulness of Web-based information and a standard way of representing that information should be developed.
- Different use cases appear to lend themselves to different uncertainty formalisms, indicating the standard representation should provide a means to unambiguously identify the formalism providing the context for assigning other uncertainty characteristics and values.
- Different uncertainty formalisms assign values to properties specifically related to the underlying meaning and processing of these values, and the representation should support defining different standard properties for each formalism without requiring changes to the representation itself.
- Sample representations for the most useful formalisms should be developed both as exemplars and for their immediate use, with the ability to expand beyond the initial exemplars as circumstances might indicate to be prudent.
- Given that uncertainty can be present anywhere, the representation should support associating uncertainty with any property or value expressible across the Web.

An open question that remains when considering a standard uncertainty representation is whether existing languages (e.g. OWL, RDFS, RIF) are sufficiently expressive to support the necessary annotations. If so, the development of such annotations might merely require work on a more complete uncertainty ontology and possibly rules; otherwise, the expressiveness of existing languages might need to be extended. As an example of the latter, it might be advisable to develop a probabilistic extension to OWL or a Fuzzy-OWL format or profiles associated with the type of uncertainty to be

represented. Further work is required to investigate the adequacy of the existing languages against the compiled use cases.

The means to associate the uncertainty representation with its subject was also beyond the scope of the URW3-XG. The conclusions noted that a mechanism similar to that specified under Semantic Annotations for WSDL and XML Schema (SAWSDL) [8].

5 Considerations for Next Steps

The work of the URW3-XG provided an important beginning for characterizing the range of uncertainty that affects reasoning on the scale of the World Wide Web, and the issues to be considered in designing a standard representation of that uncertainty. However, the work to date likely falls short of what would be needed to charter an effort to develop that representation. Additional work needed includes the following:

- The conclusions note the value of the Uncertainty Ontology developed thus far, but it also notes the value of further work to extend the ontology;
- A representation is needed for uncertainty models but it was beyond the scope of the current effort to decide whether extensions to existing Semantic Web languages (e.g. OWL, RDFS, RIF) will be sufficient or whether new representation standards will be needed;
- As SAWSDL provides a mechanism to associate semantics with certain Web resources, it might also provide a useful model for associating a standard representation of uncertainty information, but the feasibility of such use has not been adequately considered.

The question to be answered is what future venue should be pursued to tackle these issues and others that may become evident. There are several nonexclusive possibilities, among which are

- Continue with the URSW workshop series, using it as a forum to discuss advances in theory and practice;
- Approach other communities, such as those dealing with health care and life sciences, and form a wider collaboration to both continue the research aspects and to provide concrete problems against which to develop solutions;
- Develop a charter for and establish a new XG to work the items recommended by the URW3-XG;
- Investigate funding opportunities to formalize a dedicated effort to pursue the issues and develop implementable solutions and tools in a reasonable time frame.

This paper provides a summary of work to date. As the discussions of the attendees at the 2nd URSW workshop provided the basis for the URW3-XG work, so the 4th

URSW workshop provides the opportunity to discuss these and possibly other options and assess the consensus of the community for its next steps.

References

[1] Laskey, K. J.; Laskey, K. B.; and Costa, P. C. G. (2006) A Proposal for a W3C XG on Uncertainty Reasoning for the World Wide Web. Proceedings of the second workshop on Uncertainty Reasoning for the Semantic Web (URSW 2006), held at the Fifth International Semantic Web Conference (ISWC 2006), 5-9 November 2006, Athens, Georgia, USA. Available at http://c4i.gmu.edu/ursw/2006/files/papers/URSW06_P5_LaskeyCostaLaskey.pdf.

[2] Second workshop on Uncertainty Reasoning for the Semantic Web (URSW 2006), held at the Fifth International Semantic Web Conference (ISWC 2006), 5 November 2006, Athens, Georgia, USA. http://c4i.gmu.edu/ursw/2006/

[3] Laskey, K. J. (2006) Use Case Challenge. Proceedings of the second workshop on Uncertainty Reasoning for the Semantic Web (URSW 2006), held at the Fifth International Semantic Web Conference (ISWC 2006), 5-9 November 2006, Athens, Georgia, USA. Available at http://c4i.gmu.edu/ursw/2006/files/talks/URSW06_UseCaseChallenge.pdf

[4] World Wide Web Consortium (W3C), http://www.w3.org/

[5] W3C Incubator Activity > About XGs, http://www.w3.org/2005/Incubator/about.html

[6] Uncertainty Reasoning for the World Wide Web Incubator Group (URW3-XG), http://www.w3.org/2005/Incubator/urw3/

[7] URW3-XG Final Report, 31 March 2008. Available at http://www.w3.org/2005/Incubator/urw3/XGR-urw3-20080331/

[8] Semantic Annotations for WSDL and XML Schema. W3C Recommendation, 28 August 2007. Available at http://www.w3.org/2002/ws/sawsdl/spec/.

[9] Agarwal, S.; and Lamparter, S. (2005) sMART - A Semantic Matchmaking Portal for Electronic Markets. Proceedings of the 7th International IEEE Conference on E-Commerce Technology. Munich, Germany, 2005.

Appendix – Buying Speakers Use Case

1 - Purpose/Goals

Customer needs to make a decision on (1) whether to go to a store today or wait until tomorrow to buy speakers, (2) which speakers to buy and (3) at which store. Customer is interested in two speaker features: wattage and price. Customer has a valuation formula that combines the likelihood of availability of speakers on a particular day in a particular store, as well as the two features. The features of wattage and price are fuzzy. Optionally, Customer gets the formulas from CustomerService, a Web based service that collects information about products, stores, statistics, evaluations.

2 - Assumptions/Preconditions

- Customer either relies on the definitions provided by CustomerService or is knowledgeable in both probability and fuzzy sets.
- Stores provide information to CustomerService. CustomerService keeps information on both probabilistic models and fuzzy models.
- Customer has the capability of either obtaining or defining a combination function for combining probabilistic information with fuzzy.

3 - Required Resources

- Data collected by CustomerService on the availability of items, which in turn depends on restocking and rate of selling.
- Ontology of uncertainty that covers both probability and fuzziness.

4 - Successful End

Customer gets necessary information about the availability and types of speakers from stores. This information is sufficient for customer to compute the required metric.

5 - Failed End

Customer does not get necessary information and thus needs to go to multiple stores, wasting in this way a lot of time.

6 - Main Scenario

1. Customer formulates query about availability of speakers in the stores within some radius.
2. Customer sends the query to the CustomerService.
3. CustomerService replies with information about the availability of speakers. CustomerService cannot say for sure whether a given type of speaker will be available in a store tomorrow or not. It all depends on delivery and rate of sell. Thus CustomerService provides the customer only with probabilistic information.
4. Since part of the query involves requests that cannot be answered in crisp terms (vagueness), CustomerService annotates its replies with fuzzy numbers.
5. CustomerService uses the uncertainty annotated information to compute a metric.
6. Customer uses the resulting values of the metric for particular stores and for particular types of speaker to decide whether to buy speakers, what type and which store.

7. Additional background information or references: This use case was inspired by Agarwal and Lamparter [9].
8. General Issues and Relevance to Uncertainty:
    1. There is known probability distribution on the availability of particular speaker type in particular stores on a particular day in the future. Say there are two stores (not too close to each other) and the probability that speakers of type X will be available in stores A and B tomorrow are Pr(X, A)=0.4 and Pr(X, B)=0.6. The probabilities for all types of speakers are represented in the same way.
        - The uncertainty annotation process (UncAnn) was used.
        - The agent issues a query (a sentence): Sentence. It is a complex sentence consisting of three basic sentences. One related to the availability, one to the wattage and one to the price of speakers.
        - Each of these sub-sentences will have uncertainty Uncertainty associated with it.
        - The uncertainty type related to the availability of particular speaker type in the stores is of type UncAnn - UncertaintyType: Empirical.
        - The uncertainty nature is UncAnn - UncertaintyNature: Aleatory.
        - The uncertainty model is UncAnn - UncertaintyModel: Probability.
    2. The customer has (or obtains from CustomerService) definitions of features of wattage and price in terms of fuzzy membership functions. For wattage, Customer has three such functions: weak, medium and strong. These are of "trapezoid shaped" membership functions. Similarly, for price Customer has three such membership functions: cheap, reasonable and expensive.
        - The uncertainty type related to the features of wattage and price is of type UncAnn - UncertaintyType: Vagueness.
        - The uncertainty nature is UncAnn - UncertaintyNature: Epistemic.
        - The uncertainty model is UncAnn - UncertaintyModel: FuzzySets.
    3. The valuation has three possible outcomes, all are expressed as fuzzy membership functions: bad, fair, good and super.
    4. Customer knows the probabilistic information, since the probabilities are provided by CustomerService. CustomerService uses the Uncertainty Ontology for this purpose.
    5. Customer has (or selects) fuzzy definitions of the features of wattage and price. Again, the six membership functions that define these features are annotated with the Uncertainty Ontology.
    6. Customer has (or uses one suggested by CustomerService) a combination function that computes the decision, d, based upon those types of input. This function can be modified by each customer, however the stores need to give input to CustomerService - the probabilities and the (crisp) values of wattage and price for their products. The features are fuzzified by the customer's client software. Customer uses the Uncertainty Ontology to annotate the fuzziness of particular preferences.

Position Papers

A reasoner for generalized Bayesian dl-programs

Livia Predoiu

Institute of Computer Science, University of Mannheim,
A5,6, 68159 Mannheim, Germany
livia@informatik.uni-mannheim.de

Abstract. In this paper, we describe an ongoing reasoner implementation for reasoning with generalized Bayesian dl-programs and thus for dealing with deterministic ontologies and logic programs and probabilistic (mapping) rules in an integrated framework.

1 Introduction

The Semantic Web has been envisioned to enable software tools or web services, respectively, to process information provided on the Web automatically. For this purpose, the information represented in different ontologies needs to be integrated. More specifically, mappings between the ontologies need to be determined. In our framework, mappings are probabilistic rules. A more detailled discussion on the advantages of using rules for mappings and modelling the uncertainty of mappings with bayesian probabilities can be found in [1–3]). We are using generalized Bayesian dl-programs [3] for representing the deterministic ontologies and the uncertain mapping rules in an integrated logical framework.

2 Generalized Bayesian dl-programs

Generalized Bayesian dl-programs are a slightly extended more general and more formal representation of Bayesian Description Logic Programs as published in [1]. A general Bayesian dl-program is a knowledge base $KB = (L, P, \mu, Comb)$ where L is the knowledge base corresponding to the union of the ontologies to be integrated. L is represented in the description logic programming (DLP) fragment [4]. P is a logic program in Datalog without negation, μ associates with each rule r of $ground(P)$[1] and every truth valuation v of the body atoms of r a probability function $\mu(r, v)$ over all truth valuations of the head atom of r. $Comb$ is a *combining rule*, which defines how rules $r \in ground(P)$ with the same head atom can be combined to obtain a single rule. Semantically, a generalized Bayesian dl-program corresponds to a Bayesian Network. Examples and more details can be found in [3].

[1] As usual in the area of logic programming, $ground(P)$ is the set of all ground instances of rules in the logic program P

3 Architecture of the Reasoner

Below, in figure 1, the architecture of our reasoner is depicted. Without loss of generality, two OWL ontologies in the DLP fragment and a user query are the input to our reasoner. We use *dlpConvert* [5] for translating the ontologies into F-Logic for the Ontobroker[2] which is a F-logic programming reasoner. Furthermore, we are using the probabilistic matchers of *oMap* [6] for generating probabilistic level 0 mappings. We translate those mappings and the user query also into F-Logic and feed the translation into a meta reasoner based on the Ontobroker. The user query needs to be translated and fed into the Ontobroker as well before the reasoning process starts. The meta reasoner deduces all atoms needed for the creation of the corresponding Bayesian Network. From the result of the meta reasoner, we can create a Bayesian network which can be dealt with with SamIam[3]. The colored nodes in the architecture below represent knowledge bases or declarative knowledge and the uncolored ones represent tools.

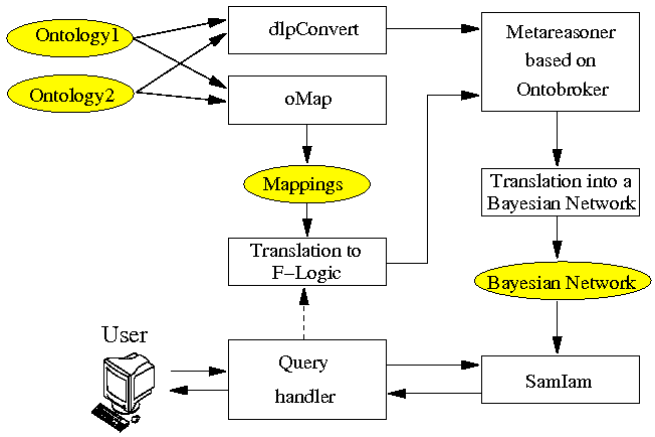


Fig. 1. Architecture of the reasoner

References

1. Predoiu, L., Stuckenschmidt, H.: A probabilistic framework for information integration and retrieval on the semantic web. In: Proc. of the 3rd Workshop on Database Interoperability (InterDB). (2007)
2. Predoiu, L.: Probabilistic information integration and retrieval on the semantic web. In: Proc. of the International Semantic Web Conference (ISWC). (2007)

[2] c.f. http://www.ontoprise.de/
[3] c.f. http://reasoning.cs.ucla.edu/samiam/

3. Cali, A., Lukasiewicz, T., Predoiu, L., Stuckenschmidt, H.: Rule-based Approaches for Representing Probabilistic Ontology Mappings. In: Uncertainty Reasoning for the Semantic Web I, Springer (to appear)
4. Grosof, B.N., Horrocks, I., Volz, R., Decker, S.: Description Logic Programs: combining logic programs with description logic. In: Proc. of the international conference on World Wide Web (WWW). (2003)
5. Motik, B., Vrandecic, D., Hitzler, P., Sure, Y., Studer, R.: dlpconvert. Converting OWL DLP statements to logic programs. In: Proc. of the 3rd European Semantic Web Conference. (2005)
6. Straccia, U., Troncy, R.: Towards Distributed Information Retrieval in the Semantic Web: Query Reformulation Using the oMAP Framework. In: Proc. of the 3rd European Semantic Web Conference. (2006)

Discussion on Uncertainty Ontology for Annotation and Reasoning (a position paper)

J. Dedek, A. Eckhardt, L. Galambos, P. Vojtas

Charles University in Prague, Department of Software Engineering
{jan.dedek,alan.eckhardt,leo.galambos,peter.vojtas}@mff.cuni.cz

In this position paper we discuss the what, who, when, where, why and how of uncertain reasoning based on achievements of URW3XG [2], our experiments and some future plans.

What and *Why* – improving semantic web practice through uncertain reasoning. This vision is described in the URW3XG charter (see [2]), especially the objective is "*to identify and describe situations* […] *for which uncertainty reasoning would significantly increase the potential for extracting useful information; and to identify methodologies that can be applied to these situations and the fundamentals of a standardized representation that could serve as the basis for information exchange necessary for these methodologies to be effectively used.*" A crucial point in this is uncertainty annotation of web (extending W3C standards [3]).

Who and *When* - will create, maintain and use this annotation. Will this annotation be done by a human creator using an annotation supporting tool for web page creation? Or will it be done by a third party annotation? For this, we will discuss a refinement of URW3XG use cases. Possible use of this enriched web will be for humans and services.

Where - will be this annotations stored. Our proposal is based on the web crawler Egothor repository [4] (we have crawled data in size of several TB from .cz domain) and an additional semantic repository build on the top using data pile technology [5].

How – to semantically enrich information and how to measure success and/or progress of such enrichment. This problem consists of two parts, namely, a data mining task and an ontology modeling task. Third party annotation of great size can be done only in an automated way and it should be done according to an ontology.

Our annotation ontology grows out of URW3XG uncertainty ontology and extends some features needed for annotation. Below we

show a part of our annotation ontology in Fig. 1. We start here from an assumption that a part of annotation will be done by a web information extraction and that this is the main source of uncertainty.

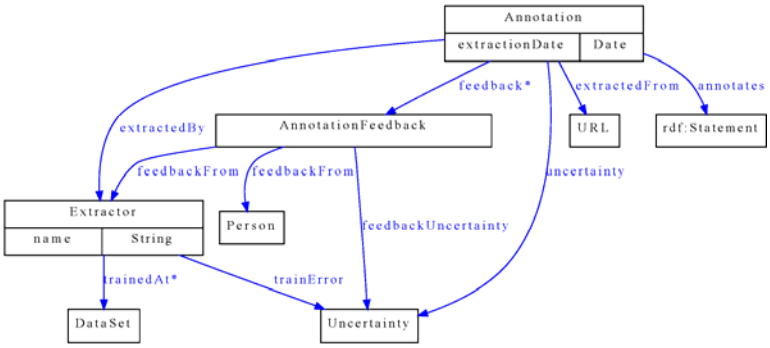


Fig. 1. Part of our uncertainty annotation ontology

Web information extraction splits pages to dominantly tabular and/or textual. Uncertainty issues connected with information extraction (and annotation) from tabular pages were discussed in [1]. Extraction of textual pages will use techniques described in [6]. Both approaches (and any other approach) generate a level of (un)certainty they have about their annotations. Also users, human or agents, can review these uncertainties and provide feedback about them.

Success of this approach can be measured primarily by the advance of semantic web functionalities. This is easier to measure for software agents. More difficult is to design metrics to measure human user satisfaction. All these aspects will be discussed in this presentation.

Acknowledgement. This work was partially supported by Czech projects 1ET100300517, 1ET100300419 and MSM-0021620838.

References.

[1] Eckhardt A., Horváth T., Maruščák D., Novotný R. , Vojtáš P.: Uncertainty Issues in Automating Process Connecting Web and User, in URSW '07 Uncertainty Reasoning for the Semantic Web. CEUR-WS.org/Vol-327/paper9.pdf
[2] Uncertainty Reasoning for the World Wide Web W3C Incubator Group Report 31 March 2008, http://www.w3.org/2005/Incubator/urw3/XGR-urw3/
[3] Search at http://www.w3.org/ for Ruby Annotation, GRDDL, RDFa
[4] Egothor search engine http://www.egothor.org/
[5] Bednárek, D., Obdržálek, D., Yaghob, J., Zavoral, F.: Data Integration Using DataPile Structure, In: Proceedings of the 9th East-European Conference on Advances in Databases and Information Systems, ADBIS 2005, Tallinn, ISBN 9985-59-545-9, 2005, 178-188
[6] J. Dědek, P. Vojtáš. Linguistic extraction for semantic annotation , accepted for IDC 2008, Catania, Sicily, to appear in the Symposium Proceedings, which will be published by Springer as a part of their series Studies in Computational Intelligence

Maximum Entropy in Support of Semantically Annotated Datasets

Paulo Pinheiro da Silva, Vladik Kreinovich, and
Christian Servin

Department of Computer Science
University of Texas, El Paso, TX 79968, USA
paulo@utep.edu, vladik@utep.edu

Abstract. One of the important problems of semantic web is checking whether two datasets describe the same quantity. The existing solution to this problem is to use these datasets' ontologies to deduce that these datasets indeed represent the same quantity. However, even when ontologies seem to confirm the identify of the two corresponding quantities, it is still possible that in reality, we deal with somewhat different quantities. A natural way to check the identity is to compare the numerical values of the measurement results: if they are close (within measurement errors), then most probably we deal with the same quantity, else we most probably deal with different ones. In this paper, we show how to perform this checking.

Key words: semantic web, ontology, uncertainty, probabilistic approach, Maximum Entropy approach

Checking whether two datasets represent the same data: formulation of the problem. In the semantic web, data are often encoded in Resource Description Framework (RDF) [2]. In RDF, every piece of information is represented as a triple consisting of a *subject*, a *predicate*, and an *object*. For example, when we describe the result of measuring the gravitation field, the coordinates at which we perform the measurements for a subject, a predicate is a term indicating that the measured quantity is a gravitational field (e.g., a term *hasGravityReading*), an the actual measurement result is an object.

In general, an RDF-based scientific dataset can be viewed as a (large) graph of RDF triples. One of the hard-to-solve problems is that triples in two different datasets using the same predicate *hasGravityReading* may not mean the same thing just because the predicates have the same name. One way to check this is to use semantics, i.e., to specify the meanings of the terms used in both datasets by an appropriate ontology, and then use reasoning to verify that the meaning of the terms is indeed the same. In the gravity example, we conclude that the predicate *hasGravityReading* has the same meaning in both datasets if in both datasets, this meaning coincides with *sweet:hasGravityReading*, the meaning of this term in one of the the Semantic Web for Earth and Environmental Terminology (SWEET) ontologies [3] that deals with gravity.

Need to take uncertainty into account. Even when ontologies seem to infer that we are dealing with the same concept, there is still a chance that the two datasets talk about slightly different concepts. To clarify the situation, we can use the fact that often, the two datasets contain the values measured at the same (or almost the same) locations. In such cases, to confirm that we are indeed dealing with the same concept, we can compare the corresponding measurement results $x'_1, \ldots, x'_n$ and $x''_1, \ldots, x''_n$. Due to measurement uncertainty, the measured values x'_i and x''_i are, in general, slightly different.

The question is: *Based on the semantically annotated measurement results and the known information about the measurement uncertainty, how can we use the uncertainty information to either reinforce or question whether two datasets namely representing the same data may not be the same data.*

Probabilistic approach to measurement uncertainty. To answer the above question, we must start by analyzing how the measurement uncertainty is represented. In this paper, we consider the traditional probabilistic way of describing measurement uncertainty.

In the engineering and scientific practice, we usually assume that for each measuring instrument, we know the probability distribution of different values of measurement error $\Delta x'_i \stackrel{\text{def}}{=} x'_i - x_i$. This assumption is often reasonable, since we can *calibrate* each measuring instrument by comparing the results of this measuring instrument with the results of a "standard" (much more accurate) one. The differences between the corresponding measurement results form the sample from which we can extract the desired distribution.

Often, after the calibration, it turns out that the tested measuring instrument is somewhat *biased* in the sense that the mean value of the measurement error is different from 0. In such cases, the instrument is usually re-calibrated – by subtracting this bias (mean) from all the measurement results – to make sure that the mean is 0. Thus, without losing generality, we can also assume that the mean value of the measurement error is 0: $E[\Delta x'_i] = 0$.

The degree to which the measured value x'_i differs from the actual value x_i is usually measured by the *standard deviation* $\sigma'_i \stackrel{\text{def}}{=} \sqrt{E[(\Delta x'_i)^2]}$.

Gaussian distribution: justification. The measurement error is usually caused by a large number of different independent factors. It is known that under certain reasonable conditions, the joint effect of a large number of small independent factors has a probability distribution which is close to Gaussian; the corresponding results (*Central Limit Theorems*) are the main reason why Gaussian (normal) distribution is indeed widely spread in practice [4]. So, it is reasonable to assume that the distribution for $\Delta x'_i$ is Gaussian.

Towards a solution. We do not know the actual values x_i, we only know the measurement results x'_i and x''_i from the two datasets. For each i, the difference between these measurement results can be described in terms of the measurement errors: $\Delta x_i \stackrel{\text{def}}{=} x'_i - x''_i = (x'_i - x_i) - (x''_i - x_i) = \Delta x'_i - \Delta x''_i$. It is reasonable to

assume that this difference is also normally distributed. Since the mean values of $\Delta x_i'$ and $\Delta x_i''$ are zeros, the mean value of their difference Δx_i is also 0, so it is sufficient to find the standard deviation $\sigma_i = \sqrt{V_i}$ of Δx_i. In general, for the sum of two Gaussian variables, we have $\sigma_i^2 = (\sigma_i')^2 + (\sigma_i'')^2 + 2r_i \cdot \sigma_i' \cdot \sigma_i''$, where $r_i = \dfrac{E[\Delta x_i' \cdot \Delta x_i'']}{\sigma_i' \cdot \sigma_i''}$ is the correlation between the i-th measurement errors. It is known that the correlation r_i can take all possible values from the interval $[-1, 1]$: the value $r_i = 1$ corresponds to the maximal possible (perfect) positive correlation, when $\Delta x_i'' = a \cdot \Delta x_i' + b$ for some $a > 0$; the value $r_i = 0$ corresponds to the case when measurement errors are independent; the value $r_i = -1$ corresponds to the maximal possible (perfect) negative correlation, when $\Delta x_i'' = a \cdot \Delta x_i' + b$ for some $a < 0$. Other values correspond to imperfect correlation. The problem is that usually, we have no information about the correlation between measurement errors from different datasets.

First idea: assume independence. A usual practical approach to situations in which we have no information about possible correlations is to assume that the measurement errors are independent.

A possible (somewhat informal) justification of this assumption is as follows. Each correlation r_i can take any value from the interval $[-1, 1]$. We would like to choose a single value r_{ij} from this interval.

We have no information why some values are more reasonable than others, whether non-negative correlation is more probable or non-positive correlation is more probable. Thus, our information is invariant with respect to the change $r_i \to -r_i$, and hence, the selected correlation value r_i must be invariant w.r.t. the same transformation. Thus, we must have $r_i = -r_i$, thence $r_i = 0$. A somewhat more formal justification of this selection can be obtained from the Maximum Entropy approach; see, e.g., [1]. Under the independence assumption, we have $(\sigma_i)^2 = (\sigma_i')^2 + (\sigma_i'')^2$.

Once we know the values, we can use the χ^2 criterion (see, e.g., [4]) to check whether with given degree of confidence α, the observed differences are consistent with the assumption that these differences are normally distributed with standard deviations σ_i: $\sum\limits_{i=1}^{n} \dfrac{(\Delta x_i)^2}{(\sigma_i)^2} \le \chi^2_{n,\alpha}$. If this inequality is satisfied, i.e., if $\sum\limits_{i=1}^{n} \dfrac{(\Delta x_i)^2}{(\sigma_i')^2 + (\sigma_i'')^2} \le \chi^2_{n,\alpha}$, then we conclude that the two datasets indeed describe the same quantity. If this inequality is not satisfied, then most probably, the datasets describe somewhat different quantities.

On the other hand, there is another possibility: that the two datasets do describe the same quantity, but the measurement errors are indeed correlated.

An alternative idea: worst-case estimations. If the above inequality holds for some values σ_i, then it holds for larger values σ_i as well. To take into account the possibility of correlations, we should only reject the similarity hypothesis when the above inequality does not hold even for the largest possible values σ_i.

Since $|r_i| \le 1$, we have $(\sigma_i)^2 \le V_i \stackrel{\text{def}}{=} (\sigma_i')^2 + (\sigma_i'')^2 + 2\sigma_i' \cdot \sigma_i''$. The value V_i is attained for $\Delta x_i'' = -\dfrac{\sigma_i''}{\sigma_i'} \cdot \Delta x_i'$. So, the largest possible value of σ_i^2 is equal to V_i. One can easily check that $V_i = (\sigma_i' + \sigma_i'')^2$. Thus, in this case, if $\sum\limits_{i=1}^{n} \left(\dfrac{\Delta x_i}{\sigma_i' + \sigma_i''}\right)^2 \le \chi^2_{n,\alpha}$, then we conclude that the two datasets indeed describe the same quantity. If this inequality is not satisfied, then most probably, the datasets describe somewhat different quantities.

Conclusion. Based on the semantically annotated measurement results and the known information about the measurement uncertainty, how can we use the uncertainty information to either reinforce or question whether two datasets namely representing the same data may not be the same data?

We assume the some values from the two datasets contain the results of measuring the same quantity at the same locations and/or moments of time. Let n denote the total number of such measurements, let $x_1', \ldots, x_n'$ denote the corresponding results from the first dataset, and let $x_1'', \ldots, x_n''$ denote the measurement results from the second dataset. We assume that we know the standard deviations σ_i' and σ_i'' of these measurements, and that we have no information about possible correlation between the corresponding measurement errors. In this case, we apply the Maximum Entropy approach, and conclude that if $\sum\limits_{i=1}^{n} \dfrac{(\Delta x_i)^2}{(\sigma_i')^2 + (\sigma_i'')^2} \le \chi^2_{n,\alpha}$, where $\chi^2_{n,\alpha} \approx n$ is the value of the χ^2-criterion for the desired certainty α, then this reinforces the original conclusion that the two datasets represent the same data. If the above inequality is not satisfied, then we conclude that either the two datasets represent different data (or, alternatively, that the measurement uncertainty values σ_i' and σ_i'' are underestimated).

If we have reasons to suspect that the measurement errors corresponding to two databases may be correlated, then can be more cautious and reinforce the original conclusion even when a weaker inequality is satisfied: $\sum\limits_{i=1}^{n} \left(\dfrac{\Delta x_i}{\sigma_i' + \sigma_i''}\right)^2 \le \chi^2_{n,\alpha}$.

Acknowledgments. This work was partly supported by NSF grant HRD-0734825 and by NIH Grant 1 T36 GM078000-01. The authors are thankful to the anonymous referees for valuable suggestions.

References

1. Jaynes, E. T.: Probability Theory: The Logic of Science, Cambridge University Press (2003)
2. Resource Description Framework (RDF) http://www.w3.org/RDF/
3. Semantic Web for Earth and Environmental Terminology SWEET ontologies http://sweet.jpl.nasa.gov/ontology/
4. Sheskin, D.: Handbook of Parametric and Nonparametric Statistical Procedures, Chapman & Hall/CRC, Boca Raton, Florida (2004)

Position Paper: Relaxing the Basic KR&R Principles to Meet the Emergent Semantic Web*

Vít Nováček

DERI, National University of Ireland, Galway
IDA Business Park, Galway, Ireland
E-mail: vit.novacek@deri.org

Abstract. The paper argues for an alternative, empirical (instead of analytical) approach to a Semantic Web-ready KR&R, motivated by the so far largely untackled need for a feasible emergent content processing.

1 Revisiting the Prevalent KR&R Trends

Since the onset of AI, the knowledge representation and reasoning (KR&R) field has been largely an analytical (in the early Wittgenstein sense) endeavour aimed at producing sound and complete results by algorithmic manipulation of rigorously defined symbol sets (knowledge bases). This works pretty well when the respective domain of interest is closed, deterministic and amenable for complete, indubitable formalisation. Unfortunately, the Semantic Web is not such a neat environment. As has been widely acknowledged in the community, the data one has to manage generally have one or more of the following qualities to them: they are dynamic, noisy, inconsistent, incomplete, intractably abundant, too inexpressive, uncertain and/or context-dependent.

Approaches extending the traditional analytical KR&R accordingly have been investigated recently, however, they seldom take the problem of the actual content acquisition into account as a primary design consideration. To illustrate the issue, we can think of the current RDF/OWL experience – substantially more people generate and use the rather relaxed OWL Full than the rigorous OWL DL flavour. Yet, much larger number of users employ the even simpler RDF(S). It seems to be quite risky to assume that future Semantic Web developers and users will eagerly and happily adopt complex uncertain, paraconsistent or contextualised extensions of the rather OWL-ish (analytical) approach to KR.

Therefore we argue that a truly Semantic Web-ready KR&R should natively tackle noisiness, uncertainty, etc., but also sensibly redefine and/or relax the rigorous assumptions and theoretical groundwork of the analytical approaches in order to follow the WWW success instead of the vapour-ware Xanadu path.

2 Towards the Relaxed, Empirical KR&R

The informatic universe we have to represent within the Semantic Web is very similar (yet simpler) to the perceptual reality of human beings – namely concerning its openness, noisiness and lack of complete, sufficiently formalised data.

* This work has been supported by the EU IST FP6 project 'Nepomuk' (FP6-027705) and by Science Foundation Ireland under Grant No. SFI/02/CE1/I131.

Therefore it can be quite useful to draw inspirations from the features of the human mind. These are, however, in many respects exact opposites of the traditional KR&R basic notions (e.g., entailment or model theory) [1]. Conversely, the high-performance and robust (although quite likely unsound and incomplete) natural reasoning abundantly employs similarity-based incorporation and retrieval of data to and from the memory [2]. The respective reasoning is much rather empirical than analytical then [1].

Expanding on these rough considerations, the proposed alternative KR&R conceptualisation can be described by three general canons: (1) **empirical** nature – everything shall be allowed to a degree once it is supported by an empirical evidence; (2) **relaxed** KR principles – the representation shall be as simple as possible so that even AI-illiterates can safely and efficiently contribute to the empirical knowledge refinement if need be; (3) **similarity-based** reasoning – any inference service shall employ soft analogical concept unification enabling to yield sufficient conclusions even from the relaxed representations. Moreover, we suggest that the particular implementations of these canons should maximally reduce the knowledge acquisition and maintenance burden imposed on the users. An obvious way is to support and reasonably employ automatically extracted knowledge as well as legacy resources, while minimising the necessary amount of modelling to be done by the users themselves.

We have recently started to implement our vision in a respective framework, with which we have already attained promising initial results in integration and "analogical closure" of automatically learned ontologies using a biomedical legacy resource [3]. We address the canon (1) by a mechanism of continuous conceptual change based on ordered weighted operators. The canon (2) is reflected by an intuitive, yet expressive basic knowledge representation (essentially compatible with RDF(S), adding heuristic uncertainty and negation). We support also simple, but already quite powerful user-defined uncertain conjunctive rules and queries. Eventually, the canon (3) is addressed by defining an ordered class of universal metrics on the set of basic KR units, which supports granular analogical concept retrieval and a well-founded soft rule and query evaluation. The implementation of these metrics allows for both closed and open world assumptions (can be chosen according to application needs at will). We are currently developing a packaged Python module comprising the framework (a public release is planned for December, 2008 at latest). Apart of that, we are going to further refine and disseminate the "philosophical" and theoretical principles among the relevant research communities.

References

1. Frith, C.: Making Up the Mind: How the Brain Creates Our Mental World. Blackwell Publishing (2007)
2. Gentner, D., Holyoak, K.J., Kokinov, B.K., eds.: The Analogical Mind: Perspectives from Cognitive Science. MIT Press (2001)
3. Nováček, V.: Empirical KR&R in action: A new framework for the emergent knowledge. Technical Report DERI-TR-2008-04-18, DERI, NUIG (2008) Available at `http://140.203.154.209/~vit/resources/2008/pubs/aerTR0408.pdf`.

Tractable Reasoning Based on the Fuzzy $\mathcal{EL}^{++}$ Algorithm

Theofilos Mailis, Giorgos Stoilos, Nick Simou, and Giorgos Stamou

National Technical University of Athens

Abstract. Fuzzy Description Logics (f-DLs) are extensions of classic DLs that are capable of representing and reasoning about imprecise and vague knowledge. Though reasoning algorithms for very expressive fuzzy DLs have been explored, an open issue in the fuzzy DL community is the study of tractable systems. In this paper we introduce the fuzzy extension of $\mathcal{EL}^{++}$, we provide its syntax and semantics together with a reasoning algorithm for the fuzzy concept subsumption problem, in which other problems related to fuzzy DLs can be reduced.

1 Introduction

Fuzzy Description Logics (f-DLs) [5] are extensions of classic DLs capable of representing and reasoning about imprecise and vague knowledge. Following the progress in the classic DL community, reasoning algorithms for tractable fuzzy DLs have been explored. In [7] Straccia et al. introduced a fuzzy extension of the DL-Lite language while Pan et al. [3] presented the very first efficient and scalable system for f-DL-Lite which is able to answer expressive fuzzy conjunctive queries over millions of data. The current bibliography includes two fuzzy extensions of $\mathcal{EL}$. First Vojtáš presented a fuzzy extension of $\mathcal{EL}$ [8] which differs from most fuzzy DL languages because it interprets conjunction as a fuzzy aggregation rather than fuzzy intersection while in [4] Stoilos et al. examined a fuzzy extension of the tractable algorithm $\mathcal{EL}^{+}$.

In this paper we introduce the fuzzy extension of $\mathcal{EL}^{++}$. Similar to the fuzzy $\mathcal{EL}^{+}$ language, fuzzy $\mathcal{EL}^{++}$ allows for concept axioms with degrees of truth i.e. fuzzy subsumption axioms [6]. Furthermore it allows for nominals and the bottom concept increasing in that way its expressiveness compared to its previous extensions.

2 The Fuzzy $\mathcal{EL}^{++}$ Language

The structural elements of the fuzzy $\mathcal{EL}^{++}$ language are *concept names* N_C, *role names* N_R and *individuals* N_I. As usual individuals represent the objects of our universe, concept names represent fuzzy sets of individuals and role names represent binary fuzzy relationships between individuals. The semantics of fuzzy $\mathcal{EL}^{++}$ are given via a fuzzy interpretation $\mathcal{I} = \left(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}\right)$ that is consisted of a domain $\Delta^{\mathcal{I}}$ which is a non empty set of individuals and a fuzzy interpretation

function $\cdot^{\mathcal{I}}$ which maps each $a \in N_I$ to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, each $A \in N_C$ to a membership function $A^{\mathcal{I}} : \Delta^{\mathcal{I}} \to [0,1]$ and each $r \in N_R$ to a membership function $r^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \to [0,1]$.

Fuzzy $\mathcal{EL}^{++}$ allows us to inductively define complex concept descriptions using the constructors shown in the table below, along with their semantics. Our language, similar to [4], allows for *fuzzy general concept inclusions* (*fuzzy GCIs*, first introduced in [6]) of the form $C \sqsubseteq^d D$ and *role inclusion axioms* (*RIs)* of the form $r_1 \circ \ldots \circ r_k \sqsubseteq s$. The semantics of fuzzy GCIs and RIs are given in the same table where the operator $\circ^t$ corresponds to the sup-t composition described in [4]. The set of fuzzy GCIs and RIs is called a *constraint box (CBox)* $\mathcal{C}$ (similar to [1]). An interpretation $\mathcal{I}$ is a model of a CBox $\mathcal{C}$ iff, for each GCI and RI in $\mathcal{C}$, the conditions described in the middle part of table are satisfied.

The fuzzy $\mathcal{EL}^{++}$ language also allows for an *assertional box (ABox)* $\mathcal{A}$ i.e. a finite set of *concept* and *role assertions* that are used to describe a snapshot of our world. The syntax along with the semantics, of concept and role assertions, is described in the table below. An interpretation $\mathcal{I}$ is a model of an ABox $\mathcal{A}$ iff, each concept and role assertion in $\mathcal{A}$ is satisfied.

Finally an interpretation $\mathcal{I}$ is a model of a fuzzy knowledge base $\mathcal{K} = \{\mathcal{A}, \mathcal{C}\}$ consisting of an ABox $\mathcal{A}$ and a CBox $\mathcal{C}$ iff it is, at the same time, a model of $\mathcal{A}$ and $\mathcal{C}$.

Name	Syntax	Semantics
top	$\top$	$\top^{\mathcal{I}}(x) = 1$
bottom	$\bot$	$\bot^{\mathcal{I}}(x) = 0$
nominal	$\{a\}$	$\{a\}^{\mathcal{I}}(x) = \begin{cases} 1 & \text{when } x = a^{\mathcal{I}} \\ 0 & \text{otherwise} \end{cases}$
conjunction	$C \sqcap D$	$(C \sqcap D)^{\mathcal{I}}(x) = min\left(C^{\mathcal{I}}(x), D^{\mathcal{I}}(x)\right)$
existential restriction	$\exists r.C$	$(\exists r.C)^{\mathcal{I}}(x) = sup_{y \in \Delta^{\mathcal{I}}}\left(min\left(r^{\mathcal{I}}(x,y), C^{\mathcal{I}}(y)\right)\right)$
GCI	$C \sqsubseteq^d D$	$min\left(C^{\mathcal{I}}(x), d\right) \leq D^{\mathcal{I}}(x)$
RI	$r_1 \circ \ldots \circ r_k \sqsubseteq s$	$\left[r_1^{\mathcal{I}} \circ^t \ldots \circ^t r_k^{\mathcal{I}}\right](x,y) \leq s^{\mathcal{I}}(x,y)$
concept assertion	$C(a) \geq d$	$C^{\mathcal{I}}\left(a^{\mathcal{I}}\right) \geq d$
role assertion	$r(a,b) \geq d$	$r^{\mathcal{I}}\left(a^{\mathcal{I}}, b^{\mathcal{I}}\right) \geq d$

3 Deciding Subsumption in fuzzy $\mathcal{EL}^{++}$

The Fuzzy $\mathcal{EL}^{++}$ is an algorithm for deciding fuzzy concept subsumption. Following to [1] other problems can be reduced to the fuzzy concept subsumption problem. The proposed algorithm, similar to that presented in [1], demands for a normalized form of CBoxes. The normalization process operates similarly to that described in [1], having as main difference the use of fuzzy general concept inclusions instead of concept inclusions.

In order to decide for fuzzy subsumption between two concept names C and D w.r.t. a normalized CBox $\mathcal{C}$ i.e. $C \sqsubseteq^d_{\mathcal{C}} D$, it is sufficient to decide for fuzzy subsumption between a nominal $\{o\}$ and a concept B w.r.t. a CBox $\mathcal{C}' =$

$\mathcal{C} \cup \{\{o\} \sqsubseteq C, D \sqsubseteq B\}$, where o is a new individual name and B is a new concept name not appearing in $BC_{\mathcal{C}}$.

Let $R_{\mathcal{C}}$ denote the set of all role names in $\mathcal{C}$, where $\mathcal{C}$ is the normal form of the CBox to be classified. Our algorithm similar to [1,4] is based on two mappings, a mapping S from $BC_{\mathcal{C}} \times BC_{\mathcal{C}}$ to $[0,1]$ and a mapping R from $R_{\mathcal{C}} \times BC_{\mathcal{C}} \times BC_{\mathcal{C}}$ to $[0,1]$. Intuitively each of these two mappings has the purpose of making implicit fuzzy subsumption relationships, explicit as follows: $S(C,D) = d$ implies that $C \sqsubseteq^d_{\mathcal{C}} D$ and $R(r,C,D) = d$ implies that $C \sqsubseteq^d_{\mathcal{C}} \exists r.D$.

In the initialization of S we have that $S(C,D) := 1$ if $D = C$ or $D = \top$, otherwise $S(C,D) = 0$ for each $C, D \in BC_{\mathcal{C}} \cup \{\bot\}$. In the initialization of R we have that $R(r,C,D) = 0$ for each $r \in R_{\mathcal{C}}$, $C, D \in BC_{\mathcal{C}} \cup \{\bot\}$. After the initialization our algorithm proceeds with the application of the following completion rules, until no rule can be applied.

CR1 If $S(C,C') = d_1$, $C' \sqsubseteq^{d_2} D \in \mathcal{C}$ and $S(C,D) < min(d_1,d_2)$
then $S(C,D) = min(d_1,d_2)$

CR2 If $S(C,C_1) = d_1$, $S(C,C_2) = d_2$, $C_1 \sqcap C_2 \sqsubseteq^{d_3} D \in \mathcal{C}$,
and $S(C,D) < min(d_1,d_2,d_3)$
then $S(C) := min(d_1,d_2,d_3)$

CR3 If $S(C,C') = d_1$, $C' \sqsubseteq^{d_2} \exists r.D \in \mathcal{C}$ and $R(r,C,D) < min(d_1,d_2)$
then $R(r,C,D) := min(d_1,d_2)$

CR4 If $R(r,C,D) = d_1$, $S(D,C') = d_2$, $\exists r.C' \sqsubseteq^{d_3} E \in \mathcal{C}$
and $S(C,E) < min(d_1,d_2,d_3)$
then $S(C,E) = min(d_1,d_2,d_3)$

CR5 If $R(r,C,D) > 0$, $S(D,\bot) > 0$ and $S(C,\bot) = 0$,
then $S(C,\bot) = 1$

CR6 If $S(C,\{a\}) = 1$, $S(E,\{a\}) = 1$ and $C \rightsquigarrow_d E$,
then for each $D \in BC_{\mathcal{C}}$, if $S(C,D) < min(d, S(E,D))$
$S(C,D) := min(d, S(E,D))$

CR7 If $R(r,C,D) = d$, $r \sqsubseteq s \in \mathcal{C}$ and $R(s,C,D) < d$
then $R(s,C,D) := d$

CR8 If $R(r_1,C,D) = d_1$, $R(r_2,D,E) = d_2$, $r_1 \circ r_2 \sqsubseteq r_3 \in \mathcal{C}$
and $R(r_3,C,E) < min(d_1,d_2)$
then $\mathrm{R}(r_3,C,D) := min(d_1,d_2)$

CR9 If $S(C,\{a\}) > 0$ for some nominal $\{a\}$ and $S(C,\{a\}) < 1$
then $S(C,\{a\}) := 1$

Definition 1. *The abbreviation $\rightsquigarrow_d$ used in rule CR6 is similar to the abbreviation adopted in [1]. The relation $C \rightsquigarrow_d E$ between two concept names $C, E \in BC_{\mathcal{C}}$ indicates that there exists a set of concept names $C_1, \ldots, C_{k+1} \in BC_{\mathcal{C}}$ and role name $r_1, \ldots, r_k \in R_{\mathcal{C}}$, such that it holds that $min(R(r_1,C_1,C_2), \ldots, R(r_k,C_k,C_{k+1})) = d$ where $C_{k+1} = E$ and either $C_1 = C$ or $C_1 = \{a\}$, where $\{a\}$ is a nominal in $BC_{\mathcal{C}}$.*

Lemma 1. *Let S be the mapping obtained after the exhaustive application of rules for a normalized CBox $\mathcal{C}$ and let $\{o\}$ be a nominal and B be a concept name in $\mathcal{C}$. Then $\{o\} \sqsubseteq^d_{\mathcal{C}} B$ holds iff $S(\{o\}, B) \geq d$ or* there is some nominal $\{a\} \in BC_{\mathcal{C}}$ such that $S(\{a\}, \bot) > 0$.

Theorem 1. *The algorithm we have developed for fuzzy subsumption between a nominal and a concept is sound and complete and operates in polynomial time.*

4 Conclusions and Future Work

In this paper we have presented a fuzzy extension of the tractable DL language $\mathcal{EL}^{++}$, fuzzy-$\mathcal{EL}^{++}$. The main contributions of our algorithm compared to the one of fuzzy-$\mathcal{EL}^{+}$ is that we introduce nominals and the bottom concept. The introduction of nominals allows for reasoning w.r.t. some assertional knowledge in contrast to the fuzzy-$\mathcal{EL}^{+}$ language which only allowed for a CBox. Therefore the instance problem w.r.t. to some ABox and some CBox can be described and solved in fuzzy $\mathcal{EL}^{++}$. Additionally the presence of the bottom concept permits concept satisfiability and ABox consistency reasoning services. Finally the presence of the bottom concept allows to imply disjointness between concepts i.e. $C \sqcap D \sqsubseteq \bot$ and along with the existence of nominals allows to express unique name assumption between two individuals i.e. $\{a\} \sqcap \{b\} \sqsubseteq \bot$.

Further extensions of our language would be lead by the extensions of the corresponding crisp language. We could examine if an extension of our language with concrete domains is possible and the way in which this would affect its complexity. Furthermore we could also examine if our language could be extended, retaining its tractability, with the existence of domain and range properties restrictions similarly to the extension of the crisp algorithm presented in [2].

References

1. Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the el envelope. In *IJCAI*, pages 364–369, 2005.
2. Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the el envelope further. In *In Proceedings of the OWLED 2008 DC Workshop on OWL: Experiences and Directions*, 2008.
3. Jeff Z. Pan, Giorgos B. Stamou, Giorgos Stoilos, and Edward Thomas. Expressive querying over fuzzy dl-lite ontologies. In *Description Logics*, 2007.
4. Giorgos Stoilos, Giorgos B. Stamou, and Jeff Z. Pan. Classifying fuzzy subsumption in fuzzy-el+. In *Description Logics*, 2008.
5. Umberto Straccia. Reasoning within fuzzy description logics. *J. Artif. Intell. Res. (JAIR)*, 14:137–166, 2001.
6. Umberto Straccia. Towards a fuzzy description logic for the semantic web (preliminary report). In *ESWC*, pages 167–181, 2005.
7. Umberto Straccia. Towards top-k query answering in description logics: The case of dl-lite. In *JELIA*, pages 439–451, 2006.
8. Peter Vojtáš. *EL description logic with aggregation of user preference concepts*, volume 154 of *Frontiers in Artificial Intelligence and Applications ISSN 0922-6389*, pages 154–166. IOS Press Amsterdam, 1 edition, 2007.

Which Role for an Ontology of Uncertainty?

Paolo Ceravolo, Ernesto Damiani, Marcello Leida

Dipartimento di Tecnologie dell'Informazione - Università degli studi di Milano
via Bramante, 65 - 26013 Crema (CR), Italy
{ceravolo,damiani,leida}@dti.unimi.it

Abstract. An Ontology of Uncertainty, like the one proposed by the W3C's UR3W-XG incubator group, provides a vocabulary to annotate different sources of information with different types of uncertainty. Here we argue that such annotations should be clearly mapped to corresponding reasoning and representation strategies. This mapping allows the system to analyse the information on the basis of its uncertainty model, running the inference proccess according to the respective uncertainty. As a proof of concepts we present a data integration system implementing a semantics-aware matching strategy based on an ontological representation of the uncertain/inconsistent matching relations generated by the various matching operators. In this scenario the sources of information to be analyzed according to different uncertainty models are independent and no intersection among them is to be managed. This particular case allows a straight-forward use of the Ontology of Uncertainty to drive the reasoning process, although in general the assumption of independence among the source of information is a lucky case. This position paper highlights the need of additional work on the Ontology of Uncertainty in order to support reasoning processes when combinations of uncertainty models are to be applied on a single source of information.

1 Introduction

Information is hardly ever perfect or certain, specially if it belongs to an unsupervised environment. The theories and the models proposed so far for representing and managing information in a effective way can be successfully applied only in small-scale scenarios. Nature and features of information need to be analyzed taking into account several aspects: context, source of information, temporal location, dependencies and so on. Mathematical models for reasoning with uncertain information has been successfully applied in several situations. But a still open issue is to simultaneously consider different models for managing uncertain information and coordinate the different independent reasoning process by an explicit representation of the different uncertainties present in the knowledge base. Here we argue that the scope of an Ontology of Uncertainty should include this coordination. As a proof of concept of this approach we present, in this paper, a data integration system implementing a semantics-aware matching strategy, which is based on an ontological representation of the matching relations generated by the various matching operators where the semantics of each

assertion is also represented explicitly as instances of an ontology. The uncertainty is assigned to each relation using SWRL rules, this allows to divide the knowledge base in sub-parts according with the specific uncertainty. The Ontology of Uncertainty, proposed by W3C's UR3W-XG incubator group, allows an explicit definition of the various types uncertainty. Assigning to each model a reasoner process it then possible to manage different independent sources of information. But the case we present is very particular because the independence of the various source of information. For this reason, our conclusions highlight the need of additional work on the Ontology of Uncertainty in order to support reasoning processes when combinations of uncertainty models are to be applied on a single source of information

2 Uncertain Information Representation and Reasoning

Uncertainty falls at meta-level respect to truth; it arises when the knowledge base does not provide sufficient information to decide if a statement is true or false in the actual situation of the system. Uncertainty can be encoded as the level of certainty of the system about a statement. Nature of uncertainty can be classified as **Epistemic**, if the uncertainty comes from the limited knowledge of the agent that generates the assertion or **Aleatory** if the uncertainty is intrinsic in the observed world. Moreover it is possible to identify two different source of uncertainty: **Objective** if the uncertainty derives from a repeatable observation and **Subjective** if the uncertainty in the information is derived from an informal evaluation. Nature of information can be: **Contingent** if it refers to a particular situation or instant or **Generic** if it refers to situations that summarize trends.Uncertainty moreover can depend on the type of information: **Ambiguous**, **Inconsistent**,**Vague**, **Incomplete** and **Empiric**. Depending on the type of uncertainty to deal with, a certain model is more suitable than another: Fuzzy theories, Probabilistic Theories and Possibility theory.

The need of a unified framework for dealing with gradual truth values and probabilities is arising but, as stated in [5] probability and possibility theories are not fully compositional with respect to all the logical connectives, without a relevant loss of expressiveness. This consideration leads to the consequence that uncertain calculi and degrees of truth are not fully compositional either. Nevertheless some work in this direction has been proposed by imposing restrictions to the expressiveness of the logics. The most relevant studies are: [7,8] where the authors define *probabilistic description logics programs* by combining stratified fuzzy description logics programs with respect to degrees of probabilities in a unified framework. In [4] a definition of possibilistic fuzzy description logics has been proposed by associating weights, representing degrees of uncertainty, to the fuzzy description logic formulas. An extension of the Fuzzy Description Logics in the field of Possibility theory has been presented also in [2]. The models available in literature are then able to deal with uncertainty and the progresses in theories for handling both uncertainty and vague truth values are remarkable. But the uncertainty that these models are able to manage has not been differentiated: as

mentioned in Section 3 uncertainty is generated from different situations and has different semantics. For this reason the URW3-XG[1] proposed an ontology (Ontology of Uncertainty) as a generic meta-model for representing the semantics of the uncertainty in various assertions.This ontology is designed for a flexible environment, where different uncertainties can arise in the same knowledge base, so the selection of the correct model for inference is driven by the information in the ontology. But the URW3-XG incubator group did not specify how to deal with situations where more than one model is involved in the inference process. Hybrid theories are considered in the set of possible models as a separate model and new sub concepts of this category can be easily added when a new model appears. The reasoning becomes complex if the result of a inference in a specific reasoning process is dependent to the result of another reasoning process. In [9] the authors propose a framework for sharing information between three different models of uncertainty, where the fuzzy linguistic truth values are propagated through the three models in a nonmonotonic way, by exploiting the extension principle [12] and aggregation of linguistic values. This approach is promising but it is grounded to fixed fuzzy values (linguistic truth) that are used by all the different models and then aggregated according to nonmonotonic rules. In literature we are not aware of hybrid reasoning processes, which can handle a flexible integration of different models. In [11,9,10,1] the interoperability has been studied and defined on a set of selected inference models. Adding new models to the framework can easily result in a revision of the underlying theory.

3 A Use Case in Data Integration

Let us consider the problem of matching heterogeneous data to externally defined types, such as ontology classes. In the case of our matching strategy, the inference process breaks down in two different steps: first step is to divide the knowledge base in sub sets according to the specific model, and the second step is to aggregate the result of the independent inference processes. In our scenario, the various reasoning processes are independent; this important premise allows us to use the Ontology of Uncertainty to partition the various matching relations according to the model used for the reasoning process. The knowledge base containing the necessary information for our matching strategy, it is composed by a set of statements that are represented by the concept `Sentence` in the Ontology of Uncertainty. To each statement the information about the uncertainty is explicitly defined by the concept `Uncertainty` that defines the correct semantics.

In this example, Ontology of Uncertainty is used basically to drive the reasoning process: each type of uncertainty is processed by its specific reasoner and a subsequent process, based on SWRL rules, integrates the results of the various reasoners. In our system we consider a Probabilistic Description Logic reasoner [6], a Fuzzy Description Logic [3].

[1] `http://www.w3.org/2005/Incubator/urw3/XGR-urw3-20080331/`

The first part of the matching strategy is to assign to the various assertions (`Sentence`), the correct information about its uncertainty semantics. This information is classified according to a set of pre-defined SWRL rules that assigns the correct semantics in relation to the matching operator (`Agent`) that has generated the relation; in relation to the presence of a degree of probability and in relation to the level of inconsistency among matching relations. There is one or more rules for each specific uncertainty type, nature, model and derivation.

According to the information that has been provided to each reasoner, the process has to return back to the matching strategy the set of assertions that they believe to be the most trustable ones.

Once the various reasoning processes come to an end, the results are propagated back to the matching ontology by a Reconciliation process. In the case of our matching strategy we make use of SWRL rules to aggregate the results.

4 Conclusions

We presented the idea of extending the scope of the Ontology of Uncertainty for hybrid reasoning under managing different types of uncertainty and showed a simple application of this idea to Schema Matching. The main constraint for the use of the Ontology of Uncertainty it is to make the reasoning process independent to each other, so that no interdependencies between assertions inferred by the reasoner can happen. This way after the various reasoning processes the Reconciliation Process is reduced to handle possible inconsistencies by SWRL rules. The Ontology of Uncertainty does not specifies for each semantics of uncertainty a particular reasoner to use. This limits the use of the Ontology of Uncertainty in real world situations, which differs from the lucky case of our matching strategy. In our opinion, the Ontology of Uncertainty has to provide further information about how the various reasoning processes exchange dependent information.

References

1. Grigoris Antoniou and Antonis Bikakis. Dr-prolog: A system for defeasible reasoning with rules and ontologies on the semantic web. *IEEE Trans. Knowl. Data Eng.*, 19(2):233–245, 2007.
2. Fernando Bobillo, Miguel Delgado, and Juan Gómez-Romero. Extending fuzzy description logics with a possibilistic layer. In *URSW*, volume 327 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2007.
3. Fernando Bobillo and Umberto Straccia. fuzzydl: An expressive fuzzy description logic reasoner. In *2008 International Conference on Fuzzy Systems (FUZZ-08)*, pages –. IEEE Computer Society, 2008.
4. Didier Dubois, Jérôme Mengin, and Henri Prade. Possibilistic uncertainty and fuzzy features in description logic. A preliminary discussion. In E. Sanchez, editor, *Fuzzy logic and the semantic web*, pages 101–113. Elsevier, http://www.elsevier.com/, 2006. DMenP001.
5. Didier Dubois and Henri Prade. Can we enforce full compositionality in uncertainty calculi. In *In Proc. of the 11th Nat. Conf. on Artificial Intelligence (AAAI-94*, pages 149–154. AAAI Press / MIT Press, 1994.

6. Pavel Klinov. Pronto: Probabilistic dl reasoning with pellet, 2008.
7. Thomas Lukasiewicz and Umberto Straccia. Description logic programs under probabilistic uncertainty and fuzzy vagueness. In *ECSQARU*, pages 187–198, 2007.
8. Thomas Lukasiewicz and Umberto Straccia. Uncertainty and vagueness in description logic programs for the semantic web, 2007.
9. X. Luo, C. Zhang, and N. R. Jennings. A hybrid model for sharing information between fuzzy, uncertain and default reasoning models in multi-agent systems. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(4):401–450, 2002.
10. William James Van Melle. *A domain-independent system that aids in constructing knowledge-based consultation programs*. PhD thesis, Stanford, CA, USA, 1980.
11. E. H. Shortliffe and B. G. Buchanan. A model of inexact reasoning in medicine. pages 259–275, 1990.
12. L.A. Zadeh. Fuzzy sets. *Information Control*, 8:338–353, 1965.

The 7th International Semantic Web Conference

October 26 – 30, 2008

Congress Center, Karlsruhe, Germany