I N F S Y S

R E S E A R C H

R E P O R T

INSTITUT FÜR INFORMATIONSSYSTEME

ARBEITSBEREICH WISSENSBASIERTE SYSTEME

AN OVERVIEW OF UNCERTAINTY AND VAGUENESS IN DESCRIPTION LOGICS FOR THE SEMANTIC WEB

THOMAS LUKASIEWICZ and UMBERTO STRACCIA

INFSYS RESEARCH REPORT 1843-06-07

OCTOBER 2006

Institut für Informationssysteme
AB Wissensbasierte Systeme
Technische Universität Wien
Favoritenstraße 9-11
A-1040 Wien, Austria
Tel: +43-1-58801-18405
Fax: +43-1-58801-18493
sek@kr.tuwien.ac.at
www.kr.tuwien.ac.at

TU VIENNA

INFSYS RESEARCH REPORT
INFSYS RESEARCH REPORT 1843-06-07, OCTOBER 2006

AN OVERVIEW OF UNCERTAINTY AND VAGUENESS IN DESCRIPTION LOGICS FOR THE SEMANTIC WEB

OCTOBER 25, 2006

Thomas Lukasiewicz [1] Umberto Straccia [2]

Abstract. Ontologies play a crucial role in the development of the Semantic Web as a means for defining shared terms in Web resources. They are formulated in web ontology languages, which are based on expressive description logics. Significant research efforts are recently directed towards representing and reasoning with uncertainty and vagueness in ontologies for the Semantic Web. In this paper, we give an overview of probabilistic uncertainty, possibilistic uncertainty, and vagueness in expressive description logics for the Semantic Web.

[1]Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", Via Salaria 113, I-00198 Rome, Italy; e-mail: lukasiewicz@dis.uniroma1.it. Institut für Informationssysteme, Technische Universität Wien, Favoritenstraße 9-11, A-1040 Vienna, Austria; e-mail: lukasiewicz@kr.tuwien.ac.at.
[2]ISTI-CNR, Via G. Moruzzi 1, I-56124 Pisa, Italy; e-mail: straccia@isti.cnr.it.

Acknowledgements: This work has been partially supported by a Heisenberg Professorship of the German Research Foundation (DFG).

Contents

1 Introduction

The *Semantic Web* [3, 4, 25, 49] has recently attracted much attention, both from academia and industry, and is widely regarded as the next step in the evolution of the World Wide Web. It aims at an extension of the current Web by standards and technologies that help machines to understand the information on the Web so that they can support richer discovery, data integration, navigation, and automation of tasks. The main ideas behind it are to add a machine-readable meaning to web pages, to use ontologies for a precise definition of shared terms in web resources, to use KR technology for automated reasoning from web resources, and to apply cooperative agent technology for processing the information of the Web.

The development of the Semantic Web proceeds in several hierarchical layers, where the *Ontology layer*, in form of the *OWL Web Ontology Language* [49, 120] (recommended by the W3C), is currently the highest layer of sufficient maturity. OWL consists of three increasingly expressive sublanguages, namely *OWL Lite*, *OWL DL*, and *OWL Full*. Hence, *ontologies* [29] play a key role in the Semantic Web, and a major effort has been put by the Semantic Web community into this issue. Informally, an ontology consists of a hierarchical description of important and *precisely* defined concepts in a particular domain, along with the description of the properties (of the instances) of each concept. Web content is then annotated by relying on the concepts defined in a specific domain ontology.

OWL Lite and OWL DL are essentially very expressive description logics with an RDF syntax [49]. As shown in [48], ontology entailment in OWL Lite and OWL DL reduces to knowledge base (un)satisfiability in the expressive description logics $\mathcal{SHIF}(\mathbf{D})$ and $\mathcal{SHOIN}(\mathbf{D})$, respectively. Hence, these expressive description logics play an important role in the Semantic Web, since they are essentially the theoretical counterparts of OWL Lite and OWL DL, respectively. More generally, description logics are a logical reconstruction of frame-based knowledge representation languages, with the aim of providing a decidable first-order formalism with a simple well-established declarative semantics to capture the meaning of the most popular features of structured representation of knowledge.

However, classical ontology languages and description logics are less suitable in all those domains where the information to be represented is *imperfect*, that is, either *uncertain*, or *vague/imprecise*, or both. In particular, web content is very likely to be imperfect, and thus there is a strong need to deal with imperfect knowledge in the Semantic Web. This need to deal with uncertainty and vagueness in ontologies for the Semantic Web has been recognized by a large number of research efforts in this direction. In particular, dealing with uncertainty and vagueness in ontologies has been successfully applied in ontology mapping and information retrieval.

Due to the rising popularity of description logics and their use, the emergence of dealing with uncertain and vague information is increasingly attracting the attention of many researcher and practitioners towards description logics able to cope with this lack of expressive power. The goal of this paper is to provide an overview of the current state of the art about the management of uncertainty and vagueness in description logics for the Semantic Web, which should help the reader to get insights on main features of the formalisms proposed in the literature.

The rest of this paper is organized as follows. In Section 2, we give a brief introduction to uncertainty and vagueness at the propositional level. In Section 3, we describe the classical description logic $\mathcal{SHOIN}(\mathbf{D})$, which is the reference language in this paper. Sections 4 and 5 show how to extend classical description logics by probabilistic and possibilistic uncertainty, respectively, while Section 6 describes how to extend classical description logics for the management of vague/imprecise knowledge. In Section 7, we give a summary and an outlook on open research.

2 Uncertainty and Vagueness

There has been a long-lasting misunderstanding in the literature of artificial intelligence and uncertainty modeling, regarding the role of probability/possibility theory and fuzzy/many-valued theory. A clarifying paper is [19]. We recall here salient notes, which may clarify the role of these theories for the inexpert reader.

A standard example that points out the difference between degrees of uncertainty and degrees of truth is that of a bottle [19]. In terms of binary truth values, a bottle is viewed as full or empty. If one accounts for the quantity of liquid in the bottle, one may say the bottle is "half-full" for instance. Under this way of speaking, "full" becomes a fuzzy predicate [125] and the degree of truth of "the bottle is full" reflects the amount of liquid in the bottle. The situation is quite different when expressing our ignorance about whether the bottle is either full or empty (given that we know only one of the two situations is the true one). To say that the probability that the bottle is full is 0.5 does not mean that the bottle is half full.

We recall that under *uncertainty theory* fall all those approaches in which statements rather than being either true or false, are true or false to some *probability* or *possibility/necessity* (for instance, "it will rain tomorrow"). That is, a statement is true or false in any world, but we are "uncertain" about which world to consider as the right one, and, thus, we speak e.g. about a probability distribution or a possibility distribution over the worlds. For instance, we cannot exactly establish whether it will rain tomorrow or not, due to our *incomplete* knowledge about our world, but we can estimate to which degree this is probable, possible, and necessary.

On the other hand, under *vagueness/imprecision theory* fall all those approaches in which statements (for instance, "the tomato is ripe") are true to some degree, which is taken from a truth space. That is, an interpretation maps a statement to a truth degree, as we are unable to establish whether a statement is completely true or false due to the involvement of vague concepts, such as "ripe", which do not have a *precise* definition. For instance, we cannot exactly say whether a tomato is ripe or not, but rather just can say that the tomato is ripe to some degree. Usually, such statements involve so-called *vague/fuzzy predicates* [125].

Note that vague statements are truth-functional, i.e., the degree of truth of a statement can be calculated from the degrees of truth of its constituents, while uncertain statements cannot be a function of the uncertainties of its constituents [18].

In the following, we illustrate a typical formalization of uncertain statements and vague statements. In the former case, we consider a basic probabilistic/possibilistic logic, while in the latter, we consider a basic many-valued logic.

2.1 Probabilistic Logic

Probabilistic logic has its origin in philosophy and logic. Its roots can be traced back to already Boole in 1854 [6]. There is a wide spectrum of formal languages that have been explored in probabilistic logic, ranging from constraints for unconditional and conditional events to rich languages that specify linear inequalities over events (see especially the work by Nilsson [85], Fagin et al. [24], Dubois and Prade et al. [17, 21, 1, 20], Frisch and Haddawy [26], and the first author [65, 66, 68]; see also the survey on sentential probability logic by Hailperin [33]). Recently, nonmonotonic generalizations of probabilistic logic have been developed and explored; see especially [70] for an overview. In this section, for illustrative purposes, we recall only the simple probabilistic logic described in [85].

We first define probabilistic formulas and probabilistic knowledge bases. We assume a set of *basic events* $\Phi = \{p_1, \ldots, p_n\}$ with $n \geqslant 1$. We use $\bot$ and $\top$ to denote *false* and *true*, respectively. We define *events* by induction as follows. Every element of $\Phi \cup \{\bot, \top\}$ is an event. If ϕ and ψ are events, then also $\neg\phi$, $(\phi \wedge \psi)$, $(\phi \vee \psi)$, and $(\phi \rightarrow \psi)$ are events. We adopt the usual conventions to eliminate parentheses. A *probabilistic formula* is an expression of the form $(\phi \geqslant l)$, where ϕ is an event, and l is a real number from the unit interval $[0, 1]$. Informally, $(\phi \geqslant l)$ says that ϕ is true with a probability of at least l. For example, $(rain\_tomorrow \geqslant 0.7)$ may express that it will rain tomorrow with a probability of at least 0.7. Notice also that $(\neg\phi \geqslant 1 - u)$ encodes that ϕ is true with a probability of at most u. A *probabilistic knowledge base* $\mathcal{K}$ is a finite set of probabilistic formulas.

We next define worlds and probabilistic interpretations. A *world* I associates with every basic event in Φ a binary truth value. We extend I by induction to all events as usual. We denote by $\mathcal{I}_\Phi$ the (finite) set of all worlds for Φ. A world I *satisfies* an event ϕ, or I is a *model* of ϕ, denoted $I \models \phi$, iff $I(\phi) = \mathbf{true}$. A *probabilistic interpretation* Pr is a probability function on $\mathcal{I}_\Phi$ (that is, a mapping $Pr\colon \mathcal{I}_\Phi \rightarrow [0, 1]$ such that all $Pr(I)$ with $I \in \mathcal{I}_\Phi$ sum up to 1). Intuitively, $Pr(I)$ is the degree to which the world $I \in \mathcal{I}_\Phi$ is probable, i.e., the probability function Pr encodes our "uncertainty" about which world is the right one. The *probability* of an event ϕ in Pr, denoted $Pr(\phi)$, is the sum of all $Pr(I)$ such that $I \in \mathcal{I}_\Phi$ and $I \models \phi$. The following theorem is an immediate consequence of the above definitions.

Theorem 2.1 *For all probabilistic interpretations Pr and events ϕ and ψ:*

$$\begin{array}{lcl} Pr(\phi \wedge \psi) & = & Pr(\phi) + Pr(\psi) - Pr(\phi \vee \psi) \\ Pr(\phi \wedge \psi) & \leqslant & \min(Pr(\phi), Pr(\psi)) \\ Pr(\phi \wedge \psi) & \geqslant & \max(0, Pr(\phi) + Pr(\psi) - 1) \\ Pr(\phi \vee \psi) & = & Pr(\phi) + Pr(\psi) - Pr(\phi \wedge \psi) \\ Pr(\phi \vee \psi) & \leqslant & \min(1, Pr(\phi) + Pr(\psi)) \\ Pr(\phi \vee \psi) & \geqslant & \max(Pr(\phi), Pr(\psi)) \\ Pr(\neg\phi) & = & 1 - Pr(\phi) \\ Pr(\bot) & = & 0 \\ Pr(\top) & = & 1 \end{array} \qquad (1)$$

A probabilistic interpretation Pr *satisfies* a probabilistic formula $(\phi \geqslant l)$, or Pr is a *model* of $(\phi \geqslant l)$, denoted $Pr \models (\phi \geqslant l)$, iff $Pr(\phi) \geqslant l$. We say Pr *satisfies* a probabilistic knowledge base $\mathcal{K}$, or Pr is a model of $\mathcal{K}$, iff Pr satisfies all $F \in \mathcal{K}$. We say $\mathcal{K}$ is *satisfiable* iff a model of $\mathcal{K}$ exists. A probabilistic formula F is a *logical consequence* of $\mathcal{K}$, denoted $\mathcal{K} \models F$, iff every model of $\mathcal{K}$ satisfies F. We say $(\phi \geqslant l)$ is a *tight logical consequence* of $\mathcal{K}$ iff l is the infimum of $Pr(\phi)$ subject to all models Pr of $\mathcal{K}$. Notice that the latter is equivalent to $l = \sup\{r \mid \mathcal{K} \models (\phi \geqslant r)\}$.

The main decision and optimization problems in probabilistic logic are deciding the satisfiability of probabilistic knowledge bases and logical consequences from probabilistic knowledge bases, as well as computing tight logical consequences from probabilistic knowledge bases, which can be done by deciding the solvability of a system of linear inequalities and by solving a linear optimization problem, respectively. In particular, column generation techniques from operations research have been successfully used to solve large problem instances in probabilistic logic (see especially the work by Jaumard et al. [53] and Hansen et al. [37]).

2.2 Possibilistic Logic

We next recall possibilistic logic; see especially [15]. We first define possibilistic formulas and possibilistic knowledge bases. *Possibilistic formulas* have the form $(\phi, P\, l)$ or $(\phi, N\, l)$, where ϕ is an event, and l is a real number from $[0, 1]$. Informally, such formulas encode to what extent ϕ is *possibly* resp. *necessarily* true. For example, $(rain\_tomorrow, P\,0.7)$ encodes that it will rain tomorrow is possible to degree 0.7, while $(father \rightarrow man, N\,1)$ says that a father is necessarily a man. A *possibilistic knowledge base* $\mathcal{K}$ is a finite set of possibilistic formulas.

A *possibilistic interpretation* is a mapping $\pi\colon \mathcal{I}_\Phi \rightarrow [0, 1]$. Intuitively, $\pi(I)$ is the degree to which the world I is *possible*. In particular, every world I such that $\pi(I) = 0$ is *impossible*, while every world I such that $\pi(I) = 1$ is *totally possible*. We say π is *normalized* iff $\pi(I) = 1$ for some $I \in \mathcal{I}_\Phi$. Intuitively, this guarantees that there exists at least one world, which could be considered as the real one.

The *possibility* of an event ϕ in a possibilistic interpretation π, denoted $Poss(\phi)$, is then defined by $Poss(\phi) = \max\{\pi(I) \mid I \in \mathcal{I}_\Phi,\, I \models \phi\}$ (where $\max \emptyset = 0$). Intuitively, the possibility of ϕ is evaluated in the most possible world where ϕ is true. The dual notion to the possibility of an event ϕ is the *necessity* of ϕ, denoted $Nec(\phi)$, which is defined by $Nec(\phi) = 1 - Poss(\neg\phi)$. It reflects the lack of possibility of $\neg\phi$, i.e., $Nec(\phi)$ evaluates to what extent ϕ is certainly true. The following theorem follows immediately from the above definitions.

Theorem 2.2 *For all possibilistic interpretations π and events ϕ and ψ:*

$$
\begin{array}{lcl}
Poss(\phi \wedge \psi) & \leqslant & \min(Poss(\phi), Poss(\psi)) \\
Poss(\phi \vee \psi) & = & \max(Poss(\phi), Poss(\psi)) \\
Poss(\neg\phi) & = & 1 - Nec(\phi) \\
Poss(\bot) & = & 0 \\
Poss(\top) & = & 1 \quad \textit{(in the normalized case)} \\
Nec(\phi \wedge \psi) & = & \min(Nec(\phi), Nec(\psi)) \\
Nec(\phi \vee \psi) & \geqslant & \max(Nec(\phi), Nec(\psi)) \\
Nec(\neg\phi) & = & 1 - Poss(\phi) \\
Nec(\bot) & = & 0 \quad \textit{(in the normalized case)} \\
Nec(\top) & = & 1
\end{array}
\tag{2}
$$

A possibilistic interpretation π *satisfies* a possibilistic formula $(\phi, P\,l)$ (resp., $(\phi, N\,l)$), or π is a *model* of $(\phi, P\,l)$ (resp., $(\phi, N\,l)$), denoted $\pi \models (\phi, P\,l)$ (resp., $\pi \models (\phi, N\,l)$) iff $Poss(\phi) \geqslant l$ (resp., $Nec(\phi) \geqslant l$). The notions of satisfiability, logical consequence, and tight logical consequence for possibilistic knowledge bases are then defined in the standard way (in the same way as in the probabilistic case). We refer the reader to [15, 45] for algorithms around possibilistic knowledge bases.

2.3 Many-Valued Logics

In the setting of many-valued logics, the convention prescribing that a proposition is either true or false is changed. A more refined range is used for the function that represents the meaning of a proposition. This is usual in natural language when words are modeled by fuzzy sets. For instance, the compatibility of "tall" in the phrase "a tall man" with some individual of a given height is often graded: The man can be judged not quite tall, somewhat tall, rather tall, very tall, etc. Changing the usual true/false convention leads to a new concept of proposition whose compatibility with a given state of facts is a matter of degree, and can be measured on an ordered scale $\mathcal{S}$ that is no longer $\{0, 1\}$, but e.g. the unit interval $[0, 1]$. This leads to identifying a "fuzzy proposition" ϕ with a fuzzy set of possible states of affairs; the degree of membership of a state of affairs to this fuzzy set evaluates the degree of fit between the proposition and the state of facts

Table 1: Properties of t-norms, s-norms, implication functions, and negation functions.

properties of t-norms "$\wedge$"	properties of s-norms "$\vee$"
$a \wedge 1 = a$	$a \vee 0 = a$
$b \leqslant c$ implies $a \wedge b \leqslant a \wedge c$	$b \leqslant c$ implies $a \vee b \leqslant a \vee c$
$a \wedge b = b \wedge a$	$a \vee b = b \vee a$
$a \wedge (b \wedge c) = (a \wedge b) \wedge c$	$a \vee (b \vee c) = (a \vee b) \vee c$
properties of implication functions "$\rightarrow$"	**properties of negation functions "$\neg$"**
$a \leqslant b$ implies $a \rightarrow c \geqslant b \rightarrow c$	$\neg 0 = 1$
$b \leqslant c$ implies $a \rightarrow b \leqslant a \rightarrow c$	$a \leqslant b$ implies $\neg b \leqslant \neg a$
$0 \rightarrow b = 1$	
$a \rightarrow 1 = 1$	

it refers to. This degree of fit is called *degree of truth* of the proposition ϕ in the interpretation $\mathcal{I}$ (state of affairs). Many-valued logics provide compositional calculi of degrees of truth, including degrees between "true" and "false". A sentence is now not true or false only, but may have a truth degree taken from a *truth space* $\mathcal{S}$, usually $[0,1]$ or $\{\frac{0}{n}, \frac{1}{n}, \ldots, \frac{n}{n}\}$ for an integer $n \geqslant 1$. In the sequel, we assume $\mathcal{S} = [0,1]$.

In the many-valued logic that we consider here, *many-valued formulas* have the form $(\phi \geqslant l)$, where $l \in [0,1]$ [32, 34] (informally, the degree of truth of ϕ is *at least* l). For instance, $(ripe\_tomato \geqslant 0.9)$ says that we have a rather ripe tomato (the degree of truth of $ripe\_tomato$ is at least 0.9).

From the semantical point of view, a *many-valued interpretation* $\mathcal{I}$ maps each basic proposition p_i into $[0,1]$ and is then extended inductively to all propositions by:

$$\begin{array}{lcl} \mathcal{I}(\phi \wedge \psi) & = & t(\mathcal{I}(\phi), \mathcal{I}(\psi)) \\ \mathcal{I}(\phi \vee \psi) & = & s(\mathcal{I}(\phi), \mathcal{I}(\psi)) \\ \mathcal{I}(\phi \rightarrow \psi) & = & i(\mathcal{I}(\phi), \mathcal{I}(\psi)) \\ \mathcal{I}(\neg \phi) & = & n(\mathcal{I}(\phi)) \ , \end{array} \tag{3}$$

where t, s, i, and n are so-called *t-norms*, *s-norms*, *implication functions*, and *negation functions*, respectively, which extend classical Boolean conjunction, disjunction, implication, and negation, respectively, to the many-valued case.

Several t-norms, s-norms, implication functions, and negation functions have been given in the literature to interpret conjunction ($\wedge$), disjunction ($\vee$), negation ($\neg$) and implication ($\rightarrow$), respectively. An important aspect of such functions is that they satisfy some properties that one expects to hold for the connectives; see Table 1. Usually, $\rightarrow$ is defined as *r-implication*, that is, $a \rightarrow b = \sup\{c \mid a \wedge c \leqslant b\}$.

Some t-norms, s-norms, implication functions, and negation functions of various fuzzy logics are shown in Table 2. In fuzzy logic, one usually distinguishes three different logics, namely, Łukasiewicz, Gödel, and Product logic; the popular Zadeh logic is a sublogic of Łukasiewicz logic. Some salient properties of these logics are shown in Table 3. For more properties, see especially [34, 87].

The implication $x \rightarrow y = \max(1-x, y)$ is called Kleene-Dienes implication in the fuzzy logic literature. Note that we have the following inferences: Let $a \geqslant n$ and $a \rightarrow b \geqslant m$. Then, under Kleene-Dienes implication, we infer that if $n > 1 - m$ then $b \geqslant m$. Under r-implication relative to a t-norm $\wedge$, we infer that $b \geqslant n \wedge m$.

Table 2: T-norms, s-norms, implication functions, and negation functions of various fuzzy logics.

	Łukasiewicz logic	Gödel logic	Product logic	Zadeh logic
$x \wedge y$	$\max(x+y-1,0)$	$\min(x,y)$	$x \cdot y$	$\min(x,y)$
$x \vee y$	$\min(x+y,1)$	$\max(x,y)$	$x+y-x \cdot y$	$\max(x,y)$
$x \rightarrow y$	$\begin{cases} 1 & \text{if } x \leqslant y \\ 1-x+y & \text{otherwise} \end{cases}$	$\begin{cases} 1 & \text{if } x \leqslant y \\ y & \text{otherwise} \end{cases}$	$\begin{cases} 1 & \text{if } x \leqslant y \\ y/x & \text{otherwise} \end{cases}$	$\max(1-x,y)$
$\neg x$	$1-x$	$\begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases}$	$\begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases}$	$1-x$

Table 3: Some additional properties of t-norms, s-norms, implication functions, and negation functions of various fuzzy logics.

Łukasiewicz logic	Gödel logic	Product logic	Zadeh logic
$x \wedge \neg x = 0$	$\exists x.\ x \wedge \neg x \neq 0$	$\exists x.\ x \wedge \neg x \neq 0$	$\exists x.\ x \wedge \neg x \neq 0$
$x \vee \neg x = 1$	$\exists x.\ x \vee \neg x \neq 1$	$\exists x.\ x \vee \neg x \neq 1$	$\exists x.\ x \vee \neg x \neq 1$
$\exists x.\ x \wedge x \neq x$	$x \wedge x = x$	$\exists x.\ x \wedge x \neq x$	$x \wedge x = x$
$\exists x.\ x \vee x \neq x$	$x \vee x = x$	$\exists x.\ x \vee x \neq x$	$x \vee x = x$
$\neg\neg x = x$	$\exists x.\ \neg\neg x \neq x$	$\exists x.\ \neg\neg x \neq x$	$\neg\neg x = x$
$x \rightarrow y = \neg x \vee y$	$\exists x.\ x \rightarrow y \neq \neg x \vee y$	$\exists x.\ x \rightarrow y \neq \neg x \vee y$	$x \rightarrow y = \neg x \vee y$
$\neg(x \rightarrow y) = x \wedge \neg y$	$\exists x.\ \neg(x \rightarrow y) \neq x \wedge \neg y$	$\exists x.\ \neg(x \rightarrow y) \neq x \wedge \neg y$	$\neg(x \rightarrow y) = x \wedge \neg y$
$\neg(x \wedge y) = \neg x \vee \neg y$	$\neg(x \wedge y) = \neg x \vee \neg y$	$\neg(x \wedge y) = \neg x \vee \neg y$	$\neg(x \wedge y) = \neg x \vee \neg y$
$\neg(x \vee y) = \neg x \wedge \neg y$	$\neg(x \vee y) = \neg x \wedge \neg y$	$\neg(x \vee y) = \neg x \wedge \neg y$	$\neg(x \vee y) = \neg x \wedge \neg y$

The *degree of subsumption* between two fuzzy sets A and B, denoted $A \rightarrow B$, is defined as $\inf_{x \in X} A(x) \rightarrow B(x)$, where $\rightarrow$ is an implication function. Note that if $A(x) \leqslant B(x)$, for all $x \in [0,1]$, then $A \rightarrow B$ evaluates to 1. Of course, $A \rightarrow B$ may evaluate to a value $v \in (0,1)$ as well. A (binary) *fuzzy relation* R over two countable crisp sets X and Y is a function $R\colon X \times Y \rightarrow [0,1]$. The *inverse* of R is the function $R^{-1}\colon Y \times X \rightarrow [0,1]$ with membership function $R^{-1}(y,x) = R(x,y)$, for every $x \in X$ and $y \in Y$. The *composition* of two fuzzy relations $R_1\colon X \times Y \rightarrow [0,1]$ and $R_2\colon Y \times Z \rightarrow [0,1]$ is defined as $(R_1 \circ R_2)(x,z) = \sup_{y \in Y} R_1(x,y) \wedge R_2(y,z)$. A fuzzy relation R is *transitive* iff $R(x,z) = (R \circ R)(x,z)$.

A many-valued interpretation $\mathcal{I}$ *satisfies* a many-valued formula $(\phi \geqslant l)$ or $\mathcal{I}$ is a *model* of $(\phi \geqslant l)$, denoted $\mathcal{I} \models (\phi \geqslant l)$, iff $\mathcal{I}(\phi) \geqslant l$. Note that $(\neg\phi \geqslant \neg u)$ says that the degree of truth of ϕ is at most u (when $\neg\neg x = x$). The notions of satisfiability, logical consequence, and tight logical consequence for many-valued knowledge bases are then defined in the standard way (as in the probabilistic case). We refer the reader to [31, 32, 34] for algorithms deciding logical consequence.

3 Classical Description Logics

In this section, we recall the expressive description logic $\mathcal{SHOIN}(\mathbf{D})$, which stands behind the web ontology languages OWL DL [49]. Although several XML and RDF syntaxes for OWL-DL exist, in this paper, we use the traditional description logic notation. For explicating the relationship between OWL DL syntax and description logic syntax, see especially [47, 49]. The purpose of this section is to make the paper self-contained. More importantly, it helps in understanding the differences between classical, probabilistic, possibilistic, and fuzzy $\mathcal{SHOIN}(\mathbf{D})$. The reader confident with the $\mathcal{SHOIN}(\mathbf{D})$ terminology may skip this section.

3.1 Syntax

The description logic $\mathcal{SHOIN}(\mathbf{D})$ is a generalization of $\mathcal{SHOIN}$ by concrete datatypes, such as strings and integers, using *concrete domains* [2, 73, 72, 74].

The elementary ingredients are as follows. We assume a set of *data values*, a set of *elementary datatypes*, and a set of *datatype predicates*, each with a predefined arity $n \geqslant 1$. A *datatype* is an elementary datatype or a finite set of data values. A *datatype theory* $\mathbf{D} = (\Delta_{\mathbf{D}}, \cdot_{\mathbf{D}})$ consists of a datatype domain $\Delta_{\mathbf{D}}$ and a mapping $\cdot_{\mathbf{D}}$ that assigns to each data value an element of $\Delta_{\mathbf{D}}$, to each elementary datatype a subset of $\Delta_{\mathbf{D}}$, and to each datatype predicate of arity n a relation over $\Delta_{\mathbf{D}}$ of arity n. We extend $\cdot^{\mathbf{D}}$ to all datatypes by $\{v_1, \ldots\}^{\mathbf{D}} = \{v_1^{\mathbf{D}}, \ldots\}$. For example, over the integers, $\geqslant_{20}$ may be a unary predicate denoting the set of integers greater or equal to 20, and thus $Person \sqcap \exists age.\geqslant_{20}$ may denote a person whose age is at least 20. Let $\mathbf{A}$, $\mathbf{R}_A$, $\mathbf{R}_C$, and $\mathbf{I}$ be pairwise disjoint nonempty finite sets of *atomic concepts*, *abstract roles*, *concrete roles*, and *individuals*, respectively.

A *role* is either an abstract role $R \in \mathbf{R}_A$, the *inverse* R^- of an abstract role $R \in \mathbf{R}_A$, or a concrete role $U \in \mathbf{R}_C$ (note that concrete roles do not have inverses). An *RBox* $\mathcal{R}$ consists of a finite set of transitivity axioms $trans(R)$ and role inclusion axioms of the form $R \sqsubseteq S$, where either $R, S \in \mathbf{R}_A$ or $R, S \in \mathbf{R}_C$. The reflexive and transitive closure of the role inclusion relationships in *RBox* is denoted by $\sqsubseteq^*$. A role not having transitive subroles is a *simple role*.

Concepts are defined by induction, using the following syntactic rules, where A is an atomic concept, $a_1, \ldots, a_n$ are individuals, C, C_1, and C_2 are concepts, R is an abstract role, S is a simple abstract role,

$T, T_1, \ldots, T_n$ are concrete roles, D is an n-ary datatype predicate, and $n \geqslant 0$:

$$\begin{array}{rcl} C & \longrightarrow & \top \mid \bot \mid A \mid \{a_1, \ldots, a_n\} \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \neg C \mid \\ & & \forall R.C \mid \exists R.C \mid (\geqslant n\ S) \mid (\leqslant n\ S) \mid \\ & & \forall T_1, \ldots, T_n.D \mid \exists T_1, \ldots, T_n.D \mid (\geqslant n\ T) \mid (\leqslant n\ T) \end{array}$$

For example, we may write the concept

$$Flower \sqcap \exists hasPetalWidth.(\geqslant_{20mm} \sqcap \leqslant_{40mm}) \sqcap \exists hasColor.Red$$

to informally denote the set of flowers having petal's dimension within $20mm$ and $40mm$, whose color is red. Here, $\geqslant_{20mm}$ and $\leqslant_{40mm}$ are datatype predicates. We use $(= 1\ S)$ to abbreviate $(\geqslant 1\ S) \sqcap (\leqslant 1\ S)$.

A *TBox* $\mathcal{T}$ is a finite set of concept inclusion axioms $C \sqsubseteq D$, where C and D are concepts. We often use $C = D \in \mathcal{T}$ in place of $\{C \sqsubseteq D, D \sqsubseteq C\} \subseteq \mathcal{T}$. A simple abstract role S is *functional* if the interpretation of the role S (see below) is always functional. A functional role S can always be obtained from an abstract role by means of the axiom $\top \sqsubseteq (\leqslant 1\ S)$. Therefore, whenever we say that a role is functional, we assume that $\top \sqsubseteq (\leqslant 1\ S)$ is in the TBox.

An *ABox* $\mathcal{A}$ is a finite set of *concept assertion axioms* $a\colon C$, *role assertion axioms* $(a, b)\colon R$, and *individual equality* (resp., *inequality*) *axioms* $a \approx b$ (resp., $a \not\approx b$). A *knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ consists of a TBox $\mathcal{T}$, an RBox $\mathcal{R}$, and an ABox $\mathcal{A}$.

3.2 Semantics

An *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ relative to a datatype theory $\mathbf{D} = (\Delta_{\mathbf{D}}, \cdot^{\mathbf{D}})$ consists of a nonempty *abstract domain* $\Delta^{\mathcal{I}}$, disjoint from $\Delta_{\mathbf{D}}$, and an *interpretation function* $\cdot^{\mathcal{I}}$ that assigns to each $a \in \mathbf{I}$ an element in $\Delta^{\mathcal{I}}$, to each $C \in \mathbf{A}$ a subset of $\Delta^{\mathcal{I}}$, to each $R \in \mathbf{R}_A$ a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, to each $T \in \mathbf{R}_C$ a subset of $\Delta^{\mathcal{I}} \times \Delta_{\mathbf{D}}$, and to every data value, datatype, and datatype predicate the same value as $\cdot^{\mathbf{D}}$. The mapping $\cdot^{\mathcal{I}}$ is extended to all roles and concepts as usual:

$$\begin{array}{rcl} (S^-)^{\mathcal{I}} & = & \{(y, x) \mid (x, y) \in S^{\mathcal{I}}\} \\ \top^{\mathcal{I}} & = & \Delta^{\mathcal{I}} \\ \bot^{\mathcal{I}} & = & \emptyset \\ \{a_1, \ldots, a_n\}^{\mathcal{I}} & = & \{a_1{}^{\mathcal{I}}, \ldots, a_n{}^{\mathcal{I}}\} \\ (C_1 \sqcap C_2)^{\mathcal{I}} & = & C_1{}^{\mathcal{I}} \cap C_2{}^{\mathcal{I}} \\ (C_1 \sqcup C_2)^{\mathcal{I}} & = & C_1{}^{\mathcal{I}} \cup C_2{}^{\mathcal{I}} \\ (\neg C)^{\mathcal{I}} & = & \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\ (\forall R.C)^{\mathcal{I}} & = & \{x \in \Delta^{\mathcal{I}} : R^{\mathcal{I}}(x) \subseteq C^{\mathcal{I}}\} \\ (\exists R.C)^{\mathcal{I}} & = & \{x \in \Delta^{\mathcal{I}} : R^{\mathcal{I}}(x) \cap C^{\mathcal{I}} \neq \emptyset\} \\ (\geqslant n\ S)^{\mathcal{I}} & = & \{x \in \Delta^{\mathcal{I}} : \#S^{\mathcal{I}}(x) \geqslant n\} \\ (\leqslant n\ S)^{\mathcal{I}} & = & \{x \in \Delta^{\mathcal{I}} : \#S^{\mathcal{I}}(x) \leqslant n\} \end{array}$$

and similarly for the other constructs, where $R^{\mathcal{I}}(x) = \{y \mid (x, y) \in R^{\mathcal{I}}\}$ and $\#X$ denotes the cardinality of the set X. In particular,

$$(\exists T_1, \ldots, T_n.d)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} : [T_1{}^{\mathcal{I}}(x) \times \ldots \times T_n{}^{\mathcal{I}}(x)] \cap d^{\mathcal{I}} \neq \emptyset\} .$$

The *satisfaction* of an axiom E in an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, denoted $\mathcal{I} \models E$, is defined as follows: $\mathcal{I} \models C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, $\mathcal{I} \models R \sqsubseteq S$ iff $R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$, $\mathcal{I} \models T \sqsubseteq U$ iff $T^{\mathcal{I}} \subseteq U^{\mathcal{I}}$, $\mathcal{I} \models trans(R)$ iff $R^{\mathcal{I}}$

is transitive, $\mathcal{I} \models a\colon C$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$, $\mathcal{I} \models (a,b)\colon R$ iff $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$, $\mathcal{I} \models (a,c)\colon T$ iff $(a^{\mathcal{I}}, c^{\mathcal{I}}) \in T^{\mathcal{I}}$, $\mathcal{I} \models a \approx b$ iff $a^{\mathcal{I}} = b^{\mathcal{I}}$, $\mathcal{I} \models a \not\approx b$ iff $a^{\mathcal{I}} \neq b^{\mathcal{I}}$. We say a concept C is *satisfiable* iff there is an interpretation $\mathcal{I}$ such that $C^{\mathcal{I}} \neq \emptyset$. For a set of axioms $\mathcal{E}$, we say $\mathcal{I}$ *satisfies* $\mathcal{E}$ iff $\mathcal{I}$ satisfies each element in $\mathcal{E}$. We say $\mathcal{I}$ is a *model* of E (resp., $\mathcal{E}$) iff $\mathcal{I} \models E$ (resp., $\mathcal{I} \models \mathcal{E}$). $\mathcal{I}$ *satisfies* (is a *model* of) a knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{R}, \mathcal{A})$, denoted $\mathcal{I} \models \mathcal{K}$, iff $\mathcal{I}$ is a model of each component $\mathcal{T}$, $\mathcal{R}$, and $\mathcal{A}$.

An axiom E is a *logical consequence* of a knowledge base $\mathcal{K}$, denoted $\mathcal{K} \models E$, iff every model of $\mathcal{K}$ satisfies E. According to [47], the entailment, subsumption and the concept satisfiability problem can be reduced to knowledge base satisfiability problem (e.g., $(\mathcal{T}, \mathcal{R}, \mathcal{A}) \models a\colon C$ iff $(\mathcal{T}, \mathcal{R}, \mathcal{A} \cup \{a\colon \neg C\})$ unsatisfiable, also, C is satisfiable iff $\{a\colon C\}$ is satisfiable), for which decision procedures and reasoning tools exists (e.g., RACER [30], FACT [46], and Pellet [89]).

Example 3.1 *(Car Example)* Let us consider the following excerpt of a simple ontology about cars. Let $\mathcal{R} = \emptyset$ and let the TBox $\mathcal{T}$ contain the following axioms:

$$Car \sqsubseteq (=1\ maker) \sqcap (=1\ passenger) \sqcap (=1\ speed)$$

$$\begin{array}{ll} (=1\ maker) \sqsubseteq Car & \top \sqsubseteq \forall maker.Maker \\ (=1\ passenger) \sqsubseteq Car & \top \sqsubseteq \forall passenger.\mathbb{N} \\ (=1\ speed) \sqsubseteq Car & \top \sqsubseteq \forall speed.Km/h \end{array}$$

$$\begin{array}{rcl} Roadster & \sqsubseteq & Cabriolet \sqcap \exists passenger.\{2\} \\ Cabriolet & \sqsubseteq & Car \sqcap \exists topType.SoftTop \\ SportsCar & = & Car \sqcap \exists speed.\geqslant_{245km/h} . \end{array}$$

Here, the value for $speed$ ranges over the datatype of kilometers per hour Km/h, while the value for $passengers$ ranges over the concrete domain of natural numbers $\mathbb{N}$. The concrete predicate $\geqslant_{245km/h}$ is true if the value is at least $245km/h$.

The ABox $\mathcal{A}$ contains the following assertions:

$$\begin{array}{l} mgb\colon Roadster \sqcap \exists maker.\{mg\} \sqcap \exists speed.\leqslant_{170km/h} \\ enzo\colon Car \sqcap \exists maker.\{ferrari\} \sqcap \exists speed.>_{350km/h} \\ tt\colon Car \sqcap \exists maker.\{audi\} \sqcap \exists speed.=_{243km/h} . \end{array}$$

Consider the knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{R}, \mathcal{A})$. It is then easily verified that, e.g.,

$$\begin{array}{ll} \mathcal{K} \models Roadster \sqsubseteq Car & \mathcal{K} \models mg\colon Maker \\ \mathcal{K} \models enzo\colon SportsCar & \mathcal{K} \models tt\colon \neg SportsCar . \end{array}$$

4 Probabilistic Uncertainty and Description Logics

In this section, we recall an important probabilistic generalization of $\mathcal{SHOIN}(\mathbf{D})$ towards sophisticated formalisms for reasoning under probabilistic uncertainty in the Semantic Web, called P-$\mathcal{SHOIN}(\mathbf{D})$, which has recently been introduced in [71] (note that [71] and [28] also introduce closely related probabilistic generalizations of the description logics $\mathcal{SHIF}(\mathbf{D})$ and $\mathcal{SHOQ}(\mathbf{D})$, which stand behind the web ontology languages OWL Lite and DAML+OIL, respectively). The syntax of P-$\mathcal{SHOIN}(\mathbf{D})$ uses the notion of a conditional constraint from [66] to express probabilistic knowledge in addition to the axioms of $\mathcal{SHOIN}(\mathbf{D})$. Its semantics is based on the notion of lexicographic entailment in probabilistic default reasoning [67, 69], which is a probabilistic generalization of the sophisticated notion of lexicographic entailment by Lehmann [57] in default reasoning from conditional knowledge bases. This semantics allows for expressing both terminological probabilistic knowledge about concepts and roles, and also assertional probabilistic knowledge about instances of concepts and roles. It naturally interprets terminological and assertional probabilistic knowledge as statistical knowledge about concepts and roles and as degrees of belief about instances of concepts and roles, respectively, and allows for deriving both statistical knowledge and degrees of belief. As an important additional feature, it also allows for expressing default knowledge about concepts (as a special case of terminological probabilistic knowledge), which is semantically interpreted as in Lehmann's lexicographic default entailment [57].

The notion of probabilistic lexicographic entailment [67, 69] is a formalism for reasoning from statistical knowledge and degrees of belief, which has very nice features. In particular, it shows a similar behavior as reference-class reasoning in a number of uncontroversial examples. But it also avoids many drawbacks of reference-class reasoning: It can handle complex scenarios and even purely probabilistic subjective knowledge as input, and conclusions are drawn in a global way from all the available knowledge as a whole. Furthermore, it also has very nice nonmonotonic properties, which are essentially inherited from Lehmann's lexicographic entailment. In particular, it realizes an inheritance of properties along subclass relationships, where more specific properties override less specific properties, without showing the problem of inheritance blocking (where properties are not inherited to subclasses that are exceptional relative to some other properties). As for general nonmonotonic properties, probabilistic lexicographic entailment satisfies (probabilistic versions of) the rationality postulates by Kraus, Lehmann, and Magidor [56], the property of rational monotonicity, and some irrelevance, conditioning, and inclusion properties. All these quite appealing features carry over to the probabilistic description logic P-$\mathcal{SHOIN}(\mathbf{D})$. See especially [69] for further details and background on the notion of probabilistic lexicographic entailment.

4.1 Syntax

We now introduce the notion of a probabilistic knowledge base. It is based on the language of conditional constraints [66], which encode interval restrictions for conditional probabilities over concepts. Every probabilistic knowledge base consists of (i) a PTBox, which is a classical (description logic) knowledge base along with probabilistic terminological knowledge, and (ii) a collection of PABoxes, which encode probabilistic assertional knowledge about a certain set of individuals. To this end, we partition the set of individuals $\mathbf{I}$ into the set of *classical individuals* $\mathbf{I}_C$ and the set of *probabilistic individuals* $\mathbf{I}_P$, and we associate with every probabilistic individual a PABox. That is, probabilistic individuals are those individuals in $\mathbf{I}$ for which we explicitly store some probabilistic assertional knowledge in a PABox.

We first define conditional constraints as follows. We assume a finite nonempty set $\mathcal{C}$ of *basic classification concepts* (or *basic c-concepts* for short), which are (not necessarily atomic) concepts in $\mathcal{SHOIN}(\mathbf{D})$ that are free of individuals from $\mathbf{I}_P$. Informally, they are the relevant description logic concepts for defining probabilistic relationships. The set of *classification concepts* (or *c-concepts*) is inductively defined as follows. Every basic c-concept $\phi \in \mathcal{C}$ is a c-concept. If ϕ and ψ are c-concepts, then $\neg\phi$ and $(\phi \sqcap \psi)$ are also c-concepts. We often write $(\phi \sqcup \psi)$ to abbreviate $\neg(\neg\phi \sqcap \neg\psi)$, as usual. A *conditional constraint* is an expression of the form $(\psi|\phi)[l, u]$, where ϕ and ψ are c-concepts, and l and u are reals from $[0, 1]$. Informally, $(\psi|\phi)[l, u]$ encodes that the probability of ψ given ϕ lies between l and u.

We next define the notion of a probabilistic knowledge base. A *PTBox* $PT = (T, P)$ consists of a classical (description logic) knowledge base T and a finite set of conditional constraints P. Informally, every conditional constraint $(\psi|\phi)[l, u]$ in P encodes that "generally, if an object belongs to ϕ, then it belongs to ψ

with a probability between l and u". In particular, $(\exists R.\{o\}|\phi)[l,u]$ in P, where $o \in \mathbf{I}_C$ and $R \in \mathbf{R}_A$, encodes that "generally, if an object belongs to ϕ, then it is related to o by R with a probability between l and u". A *PABox* P is a finite set of conditional constraints. A *probabilistic knowledge base* $\mathcal{K} = (T, P, (P_o)_{o \in \mathbf{I}_P})$ relative to $\mathbf{I}_P$ consists of a PTBox $PT = (T, P)$ and one PABox P_o for every probabilistic individual $o \in \mathbf{I}_P$. Informally, every $(\psi|\phi)[l,u]$ in P_o, where $o \in \mathbf{I}_P$, encodes that "if o belongs to ϕ, then o belongs to ψ with a probability between l and u". In particular, $(\exists R.\{o'\}|\phi)[l,u]$ in P_o, where $o \in \mathbf{I}_P$, $o' \in \mathbf{I}_C$, and $R \in \mathbf{R}_A$, expresses that "if o belongs to ϕ, then o is related to o' by R with a probability between l and u". Informally, a probabilistic knowledge base $\mathcal{K} = (T, P, (P_o)_{o \in \mathbf{I}_P})$ extends a classical knowledge base T by probabilistic terminological knowledge P and probabilistic assertional knowledge P_o about every $o \in \mathbf{I}_P$. That is, P represents our *statistical knowledge about concepts*, while every P_o represents our *degrees of belief about the individual* o. Observe that the axioms in T and the conditional constraints in every P_o with $o \in \mathbf{I}_P$ are *strict* (that is, they must always hold), while the conditional constraints in P are *defeasible* (that is, they may have exceptions and thus do not always have to hold), since $T \cup P$ may not always be satisfiable as a whole in combination with our degrees of belief (and then we ignore some elements of P).

Example 4.1 *(Car Example cont'd)* We now extend the classical description logic knowledge base T given in Example 3.1 by terminological default, terminological probabilistic, and assertional probabilistic knowledge to a probabilistic knowledge base $\mathcal{K} = (T, P, (P_o)_{o \in \mathbf{I}_P})$. We assume an additional atomic concept *HasFourWheels* and an additional datatype role *HasColor* between cars and the elementary datatype *colors*, which has a finite set of color names as data values.

The terminological default knowledge (1) "generally, cars do not have a red color" and (2) "generally, sports cars have a red color", and the terminological probabilistic knowledge (3) "cars have four wheels with a probability of at least 0.9", can be expressed by the following conditional constraints in P:

(1) $(\neg\exists\, \mathit{HasColor}.\{\mathit{red}\} \,|\, \mathit{Car})[1,1]$,
(2) $(\exists\, \mathit{HasColor}.\{\mathit{red}\} \,|\, \mathit{SportsCar})[1,1]$,
(3) $(\mathit{HasFourWheels} \,|\, \mathit{Car})[0.9,1]$.

Suppose we want to encode some probabilistic information about John's car (which we have not seen so far). Then, the set of probabilistic individuals $\mathbf{I}_P$ contains the individual *John's car*, and the assertional probabilistic knowledge (4) "John's car is a sports car with a probability of at least 0.8" (we know that John likes sports cars) can be expressed by the following conditional constraint in $P_{\mathit{John's\ car}}$:

(4) $(\mathit{SportsCar} \,|\, \top)[0.8,1]$.

4.2 Semantics

In this section, we define the semantics of P-$\mathcal{SHOIN}(\mathbf{D})$. After some preliminaries, we introduce the notions of consistency and lexicographic entailment for probabilistic knowledge bases, which are based on the notions of consistency resp. lexicographic entailment in probabilistic default reasoning [67, 69].

4.2.1 Preliminaries

We now define (possible) objects and probabilistic interpretations, which are certain sets of basic c-concepts resp. probability functions on the set of all (possible) objects. We also define the satisfaction of classical knowledge bases and conditional constraints in probabilistic interpretations.

A *(possible) object* o is a set of basic c-concepts $\phi \in \mathcal{C}$ such that $\{\phi(i) \mid \phi \in o\} \cup \{\neg\phi(i) \mid \phi \in \mathcal{C} \setminus o\}$ is satisfiable, where i is a new individual. Informally, every object o represents an individual i that is fully specified on $\mathcal{C}$ in the sense that o belongs (resp., does not belong) to every c-concept $\phi \in o$ (resp., $\phi \in \mathcal{C} \setminus o$). We denote by $\mathcal{O}_\mathcal{C}$ the set of all objects relative to $\mathcal{C}$. An object o *satisfies* a classical knowledge base T, or o is a *model* of T, denoted $o \models T$, iff $T \cup \{\phi(i) \mid \phi \in o\} \cup \{\neg\phi(i) \mid \phi \in \mathcal{C} \setminus o\}$ is satisfiable, where i is a new individual. An object o *satisfies* a basic c-concept $\phi \in \mathcal{C}$, or o is a *model* of ϕ, denoted $o \models \phi$, iff $\phi \in o$. The satisfaction of c-concepts by objects is inductively extended to all c-concepts, as usual, by (i) $o \models \neg\phi$ iff $o \models \phi$ does not hold, and (ii) $o \models \phi \sqcap \psi$ iff $o \models \phi$ and $o \models \psi$. It is not difficult to verify that a classical knowledge base T is satisfiable iff an object $o \in \mathcal{O}_\mathcal{C}$ exists that satisfies T.

A *probabilistic interpretation* Pr is a probability function on $\mathcal{O}_\mathcal{C}$ (that is, a mapping $Pr\colon \mathcal{O}_\mathcal{C} \to [0,1]$ such that all $Pr(o)$ with $o \in \mathcal{O}_\mathcal{C}$ sum up to 1). We say Pr *satisfies* a classical knowledge base T, or Pr is a *model* of T, denoted $Pr \models T$, iff $o \models T$ for every $o \in \mathcal{O}_\mathcal{C}$ such that $Pr(o) > 0$. We define the probability of a c-concept and the satisfaction of conditional constraints in probabilistic interpretations as follows. The *probability* of a c-concept ϕ in a probabilistic interpretation Pr denoted $Pr(\phi)$, is the sum of all $Pr(o)$ such that $o \models \phi$. For c-concepts ϕ and ψ such that $Pr(\phi) > 0$, we write $Pr(\psi|\phi)$ to abbreviate $Pr(\phi \sqcap \psi) \,/\, Pr(\phi)$. We say Pr *satisfies* a conditional constraint $(\phi|\psi)[l,u]$, or Pr is a *model* of $(\psi|\phi)[l,u]$, denoted $Pr \models (\psi|\phi)[l,u]$, iff $Pr(\phi) = 0$ or $Pr(\psi|\phi) \in [l,u]$. We say Pr *satisfies* a set of conditional constraints $\mathcal{F}$, or Pr is a *model* of $\mathcal{F}$, denoted $Pr \models \mathcal{F}$, iff $Pr \models F$ for all $F \in \mathcal{F}$. It is not difficult to verify that a classical knowledge base T is satisfiable iff there exists a probabilistic interpretation that satisfies T.

4.2.2 Consistency

The notion of consistency for PTBoxes and probabilistic knowledge bases is based on the notion of consistency in probabilistic default reasoning [67, 69].

We first give some preparative definitions. A probabilistic interpretation Pr *verifies* a conditional constraint $(\psi|\phi)[l,u]$ iff $Pr(\phi) = 1$ and $Pr(\psi) \in [l,u]$, that is, iff $Pr(\phi) = 1$ and $Pr \models (\psi|\phi)[l,u]$. We say Pr *falsifies* $(\psi|\phi)[l,u]$ iff $Pr(\phi) = 1$ and $Pr \not\models (\psi|\phi)[l,u]$. A set of conditional constraints $\mathcal{F}$ *tolerates* a conditional constraint F under a classical knowledge base T iff $T \cup \mathcal{F}$ has a model that verifies F.

A PTBox $PT = (T,P)$ is *consistent* iff (i) T is satisfiable and (ii) there exists an ordered partition $(P_0,\ldots,P_k)$ of P such that each P_i with $i \in \{0,\ldots,k\}$ is the set of all $F \in P_i \cup \cdots \cup P_k$ that are tolerated under T by $P_i \cup \cdots \cup P_k$. Informally, the condition (ii) means that P has a natural ordered partition into collections of conditional constraints of increasing specificities such that every collection is locally consistent. That is, any inconsistencies can be naturally resolved by preferring more specific pieces of knowledge to less specific ones. For example, the inconsistency between $(\neg\exists\, \mathit{HasColor}.\{\mathit{red}\} \mid \mathit{Car})[1,1]$ and $(\exists\, \mathit{HasColor}.\{\mathit{red}\} \mid \mathit{SportsCar})[1,1]$ when reasoning about sports cars is naturally resolved by preferring the latter to the former. We call the above ordered partition $(P_0,\ldots,P_k)$ of P the *z-partition* of PT. A probabilistic knowledge base $\mathcal{K} = (T,P,(P_o)_{o\in\mathbf{I}_P})$ is *consistent* iff $PT = (T,P)$ is consistent and $T \cup P_o$ is satisfiable for all probabilistic individuals $o \in \mathbf{I}_P$. Informally, the latter says that the strict knowledge in T must be compatible with the strict degrees of belief in P_o, for every probabilistic individual o.

Example 4.2 *(Car Example cont'd)* The probabilistic knowledge base $\mathcal{K} = (T,\ P,(P_o)_{o\in\mathbf{I}_P})$ of Example 4.1 is consistent, since $PT = (T,P)$ is consistent, and $T \cup P_o$ is satisfiable for every probabilistic individual $o \in \mathbf{I}_P = \{\mathit{John's\ car}\}$. Observe that the z-partition of (T,P) is given by (P_0,P_1), where $P_0 = \{(\psi|\phi)[l,u] \in P \mid \phi = \mathit{Car}\}$ and $P_1 = \{(\psi|\phi)[l,u] \in P \mid \phi = \mathit{SportsCar}\}$.

There is an algorithm for deciding whether a PTBox (resp., probabilistic knowledge base) in P-$\mathcal{SHOIN}(\mathbf{D})$ is consistent, which is based on a reduction to deciding whether a classical knowledge base in $\mathcal{SHOIN}(\mathbf{D})$ is satisfiable and to deciding whether a system of linear constraints is solvable [71]. This shows that the two consistency problems in P-$\mathcal{SHOIN}(\mathbf{D})$ are both decidable.

4.2.3 Lexicographic Entailment

The notion of lexicographic entailment for probabilistic knowledge bases is based on lexicographic entailment in probabilistic default reasoning [67, 69]. In the sequel, let $\mathcal{K} = (T, P, (P_o)_{o \in \mathbf{I}_P})$ be a consistent probabilistic knowledge base. We first define a lexicographic preference relation on probabilistic interpretations, which is then used to define the notion of lexicographic entailment for sets of conditional constraints under PTBoxes. We finally define the notion of lexicographic entailment for deriving statistical knowledge and degrees of belief about probabilistic objects from PTBoxes and probabilistic knowledge bases, respectively.

We use the z-partition $(P_0, \ldots, P_k)$ of (T, P) to define a lexicographic preference relation on probabilistic interpretations Pr and Pr': We say Pr is *lexicographically preferable* (or *lex-preferable*) to Pr' iff some $i \in \{0, \ldots, k\}$ exists such that $|\{F \in P_i \mid Pr \models F\}| > |\{F \in P_i \mid Pr' \models F\}|$ and $|\{F \in P_j \mid Pr \models F\}| = |\{F \in P_j \mid Pr' \models F\}|$ for all $i < j \leqslant k$. Roughly speaking, this preference relation implements the idea of preferring more specific pieces of knowledge to less specific ones in the case of local inconsistencies. It can thus be used for ignoring the latter when drawing conclusions in the case of local inconsistencies. A model Pr of a classical knowledge base T and a set of conditional constraints $\mathcal{F}$ is a *lexicographically minimal* (or *lex-minimal*) *model* of $T \cup \mathcal{F}$ iff no model of $T \cup \mathcal{F}$ is lex-preferable to Pr.

We define the notion of lexicographic entailment of conditional constraints from sets of conditional constraints under PTBoxes as follows. A conditional constraint $(\psi|\phi)[l, u]$ is a *lexicographic consequence* (or *lex-consequence*) of a set of conditional constraints $\mathcal{F}$ under a PTBox PT, denoted $\mathcal{F} \mathrel{\|\!\sim}^{lex} (\psi|\phi)[l, u]$ under PT, iff $Pr(\psi) \in [l, u]$ for every lex-minimal model Pr of $T \cup \mathcal{F} \cup \{(\phi|\top)[1, 1]\}$. We say $(\psi|\phi)[l, u]$ is a *tight lexicographic consequence* (or *tight lex-consequence*) of $\mathcal{F}$ under PT, denoted $\mathcal{F} \mathrel{\|\!\sim}^{lex}_{tight} (\psi|\phi)[l, u]$ under PT, iff l (resp., u) is the infimum (resp., supremum) of $Pr(\psi)$ subject to all lex-minimal models Pr of $T \cup \mathcal{F} \cup \{(\phi|\top)[1, 1]\}$. Note that $[l, u] = [1, 0]$ (where $[1, 0]$ represents the empty interval) when no such model Pr exists. Furthermore, for inconsistent PTBoxes PT, we define $\mathcal{F} \mathrel{\|\!\sim}^{lex} (\psi|\phi)[l, u]$ and $\mathcal{F} \mathrel{\|\!\sim}^{lex}_{tight} (\psi|\phi)[1, 0]$ under PT for all sets of conditional constraints $\mathcal{F}$ and all conditional constraints $(\psi|\phi)[l, u]$.

We now define which statistical knowledge and degrees of belief follow under lexicographic entailment from PTBoxes PT and probabilistic knowledge bases $\mathcal{K} = (T, P, (P_o)_{o \in \mathbf{I}_P})$, respectively. A conditional constraint F is a *lex-consequence* of PT, denoted $PT \mathrel{\|\!\sim}^{lex} F$, iff $\emptyset \mathrel{\|\!\sim}^{lex} F$ under PT. We say F is a *tight lex-consequence* of PT, denoted $PT \mathrel{\|\!\sim}^{lex}_{tight} F$, iff $\emptyset \mathrel{\|\!\sim}^{lex}_{tight} F$ under PT. A conditional constraint F for a probabilistic individual $o \in \mathbf{I}_P$ is a *lex-consequence* of $\mathcal{K}$, denoted $\mathcal{K} \mathrel{\|\!\sim}^{lex} F$, iff $P_o \mathrel{\|\!\sim}^{lex} F$ under $PT = (T, P)$. We say F is a *tight lex-consequence* of $\mathcal{K}$, denoted $\mathcal{K} \mathrel{\|\!\sim}^{lex}_{tight} F$, iff $P_o \mathrel{\|\!\sim}^{lex}_{tight} F$ under $PT = (T, P)$.

Example 4.3 *(Car Example cont'd)* Consider again the probabilistic knowledge base $\mathcal{K} = (T, P, (P_o)_{o \in \mathbf{I}_P})$ of Example 4.1. The following are some (terminological default and terminological probabilistic) tight lex-consequences of $PT = (T, P)$:

$$(\neg\exists\, \mathit{HasColor}.\{\mathit{red}\} \mid \mathit{Car})[1,1],$$
$$(\exists\, \mathit{HasColor}.\{\mathit{red}\} \mid \mathit{SportsCar})[1,1],$$
$$(\mathit{HasFourWheels} \mid \mathit{Car})[0.9,1],$$
$$(\neg\exists\, \mathit{HasColor}.\{\mathit{red}\} \mid \mathit{Roadster})[1,1],$$
$$(\mathit{HasFourWheels} \mid \mathit{SportsCar})[0.9,1],$$
$$(\mathit{HasFourWheels} \mid \mathit{Roadster})[0.9,1]\,.$$

Hence, in addition to the sentences (1) to (3) directly encoded in P, we also conclude "generally, roadsters do not have a red color", "sports cars have four wheels with a probability of at least 0.9", and "roadsters have four wheels with a probability of at least 0.9". Observe here that the default property of not having a red color and the probabilistic property of having four wheels with a probability of at least 0.9 are inherited from cars down to roadsters. Roughly, the tight lex-consequences of $PT = (T, P)$ are given by all those conditional constraints that (a) are either in P, or (b) can be constructed by inheritance along subconcept relationships from the ones in P and are not overridden by more specific pieces of knowledge in P.

The following conditional constraints for the probabilistic individual *John's car* are some (assertional probabilistic) tight lex-consequences of $\mathcal{K} = (T, P, (P_o)_{o \in \mathbf{I}_P})$, which informally say that John's car is a sports car, has a red color, and has four wheels with probabilities of at least 0.8, 0.8, and 0.72, respectively:

$$(\mathit{SportsCar} \mid \top)[0.8,1],$$
$$(\exists\, \mathit{HasColor}.\{\mathit{red}\} \mid \top)[0.8,1],$$
$$(\mathit{HasFourWheels} \mid \top)[0.72,1]\,.$$

There is an algorithm for computing tight intervals under lexicographic entailment in P-$\mathcal{SHOIN}(\mathbf{D})$, which is based on a reduction to deciding classical knowledge base satisfiability in $\mathcal{SHOIN}(\mathbf{D})$ and to solving linear optimization problems [71]. Hence, lexicographic entailment in P-$\mathcal{SHOIN}(\mathbf{D})$ is computable.

4.3 Related Work

To our knowledge, there are no other approaches to probabilistic description logics for the Semantic Web in the literature. However, there are several previous approaches to probabilistic description logics without Semantic Web background. Furthermore, there are several probabilistic extensions of web ontology languages in the literature. Finally, there are important applications of probabilistic description logics and probabilistic web ontology languages in the field of information retrieval. In this section, we give an overview of these approaches.

4.3.1 Probabilistic Description Logics

Other approaches to probabilistic description logics can be classified according to the generalized description logics, the supported forms of probabilistic knowledge, and the underlying probabilistic reasoning formalism. Heinsohn [38] presents a probabilistic extension of the description logic $\mathcal{ALC}$, which allows to represent terminological probabilistic knowledge about concepts and roles, and which is essentially based on probabilistic reasoning in probabilistic logics, similar to [85, 1, 26, 66]. Heinsohn [38], however, does not allow for assertional knowledge about concept and role instances. Jaeger's work [51] proposes another probabilistic extension of the description logic $\mathcal{ALC}$, which allows for terminological and assertional probabilistic knowledge about concepts / roles and about concept instances, respectively, but does not support assertional probabilistic knowledge about role instances (but he mentions a possible extension in this direction). The uncertain reasoning formalism in [51] is essentially based on probabilistic reasoning in probabilistic logics, as the one in [38], but coupled with cross-entropy minimization to combine terminological probabilistic knowledge with assertional probabilistic knowledge. The work by Dürig and Studer [22] presents a further probabilistic extension of $\mathcal{ALC}$, which is based on probabilistic reasoning in probabilistic logics, but which only allows for assertional probabilistic knowledge about concept and role instances, and not for terminological probabilistic knowledge. Jaeger's recent work [52] focuses on interpreting probabilistic concept subsumption and probabilistic role quantification through statistical sampling distributions, and develops a probabilistic version of the guarded fragment of first-order logic. Koller et al.'s work [55] presents a probabilistic generalization of the CLASSIC description logic. Like Heinsohn's work [38], it allows for terminological probabilistic knowledge about concepts and roles, but does not support assertional knowledge about instances of concepts and roles. But, in contrast to [38], it is based on inference in Bayesian networks as underlying probabilistic reasoning formalism. Closely related work by Yelland [123] combines a restricted description logic close to $\mathcal{FL}$ with Bayesian networks, and applies this approach to market analysis. It allows for terminological probabilistic knowledge about concepts and roles, but does not support assertional knowledge about instances of concepts and roles.

4.3.2 Probabilistic Web Ontology Languages

The literature contains several probabilistic generalizations of web ontology languages. Many of these approaches focus especially on combining the web ontology language OWL with probabilistic formalisms based on Bayesian networks. In particular, da Costa [9], da Costa and Laskey [10], and da Costa et al. [11] suggest a probabilistic generalization of OWL, called PR-OWL, which is based on multi-entity Bayesian networks. The latter are a Bayesian logic that combines first-order logic with Bayesian probabilities. Ding et al. [13, 14] propose a probabilistic generalization of OWL, called BayesOWL, which is based on standard Bayesian networks. BayesOWL provides a set of rules and procedures for the direct translation of an OWL ontology into a Bayesian network that supports ontology reasoning, both within and across ontologies, as Bayesian inferences. Ding et al. [88, 14] also describe an application of this approach in ontology mapping. In closely related work, Mitra et al. [84] introduce a technique to enhancing existing ontology mappings by using a Bayesian network to represent the influences between potential concept mappings across ontologies. Yang and Calmet [122] present an integration of the web ontology language OWL with Bayesian networks. The approach makes use of probability and dependency-annotated OWL to represent uncertain information in Bayesian networks. Pool and Aikin [90] also provide a method for representing uncertainty in OWL ontologies, while Fukushige [27] proposes a basic framework for representing probabilistic relationships in RDF. Finally, Nottelmann and Fuhr [86] present two probabilistic extensions of variants of OWL Lite, along with a mapping to locally stratified probabilistic Datalog.

4.3.3 Applications in Information Retrieval

An important research direction deals with the application of probabilistic description logics and probabilistic web ontology languages in enhanced information retrieval techniques. In particular, Mantay et al. [75] propose a probabilistic least common subsumer operation, which is based on a probabilistic extension of the description logic language $\mathcal{ALN}$. They show that applying this approach in information retrieval allows for reducing the amount of retrieved data and thus for avoiding information flood. Closely related work by Holi and Hyvönen [39, 40] shows how degrees of overlap between concepts can be modeled and computed efficiently using Bayesian networks based on RDF(S) ontologies. Such degrees of overlap indicate how well an individual data item matches the query concept, and can thus be used for measuring the relevance in information retrieval tasks. In another closely related work, Udrea et al. [119] explore the use of probabilistic ontologies in relational databases. They propose to extend relations by associating with every attribute a constrained probabilistic ontology, which describes relationships between terms occurring in the domain of that attribute. An extension of the relational algebra then allows for an increased recall in information retrieval. Finally, Weikum et al. [121] and Thomas and Sheth [117] describe the use of probabilistic ontologies in information retrieval from a more general perspective.

5 Possibilistic Uncertainty and Description Logics

Similar to probabilistic extensions of description logics, possibilistic extensions of description logics have been developed by Hollunder [45] and Dubois et al. [16] and especially applied in information retrieval by Liau and Fan [63]. In the sequel, we implicitly assume the description logic $\mathcal{SHOIN}(\mathbf{D})$ as underlying description logic, but any other (decidable) description logic can be used as well.

5.1 Syntax

A *possibilistic axiom* is of the form $(\alpha, \mathrm{P}\, l)$ or $(\alpha, \mathrm{N}\, l)$, where α is a classical description logic axiom, and l is a real number from $[0,1]$. A *possibilistic RBox* (resp., *TBox*, *ABox*) is a finite set of possibilistic axioms $(\alpha, \mathrm{P}\, l)$ or $(\alpha, \mathrm{N}\, l)$, where α is an RBox (resp., TBox, ABox) axiom. A *possibilistic knowledge base* $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$ consists of a possibilistic RBox $\mathcal{R}$, a possibilistic TBox $\mathcal{T}$, and a possibilistic ABox $\mathcal{A}$. The following example from [45] illustrates possibilistic knowledge bases.

Example 5.1 *(Car Example cont'd)* The following possibilistic knowledge base $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$ encodes some possibilistic knowledge about cars and rich people. Let $\mathcal{R} = \emptyset$. The TBox $\mathcal{T}$ represents the possibilistic terminological knowledge that "every person owning a Porsche is either rich or a car fanatic with a necessity of at least 0.8" and "every rich person is a golfer with a possibility of at least 0.7":

$$\begin{aligned}\mathcal{T} \;=\; &\{(\exists owns.Porsche \sqsubseteq richPerson \sqcup carFanatic,\ N\,0.8),\\ &(richPerson \sqsubseteq golfer,\ P\,0.7)\}\,.\end{aligned}$$

Furthermore, the ABox $\mathcal{A}$ expresses the possibilistic assertional knowledge that "Tom owns a 911 with necessity 1", "a 911 is a Porsche with necessity 1", and "Tom is not a car fanatic with a necessity of at least 0.7":

$$\begin{aligned}\mathcal{A} \;=\; &\{((Tom, 911) \colon owns,\ N\,1),\\ &(911 \colon Porsche,\ N\,1),\\ &(Tom \colon \neg carFanatic,\ N\,0.7)\}\,.\end{aligned}$$

5.2 Semantics

Let $\mathcal{I}$ denote the set of all classical description logic interpretations. A *possibilistic interpretation* is a mapping $\pi \colon \mathcal{I} \to [0,1]$. In the sequel, we assume that π is *normalized*, that is, that $\pi(I) = 1$ for some $I \in \mathcal{I}$. The *possibility* of a description logic axiom α in a possibilistic interpretation π, denoted $Poss(\alpha)$, is then defined by $Poss(\alpha) = \max\{\pi(I) \mid I \in \mathcal{I},\, I \models \alpha\}$ (where $\max \emptyset = 0$), and the *necessity* of α, denoted $Nec(\alpha)$, is defined by $Nec(\alpha) = 1 - Poss(\neg\alpha)$.

A possibilistic interpretation π *satisfies* a possibilistic axiom $(\alpha, P\,l)$ (resp., $(\alpha,\, N\,l)$), or π is a *model* of $(\alpha, P\,l)$ (resp., $(\alpha, N\,l)$), denoted $\pi \models (\alpha, P\,l)$ (resp., $\pi \models (\alpha, N\,l)$) iff $Poss(\alpha) \geqslant l$ (resp., $Nec(\alpha) \geqslant l$). The notions of satisfiability, logical entailment, and tight logical entailment for possibilistic knowledge bases are then defined in the standard way. As shown by Hollunder [45], deciding logical consequences and thus also deciding satisfiability and computing tight logical consequences can be reduced to deciding logical consequences in description logics.

Example 5.2 *(Car Example cont'd)* Consider again the possibilistic knowledge base $\mathcal{K}$ of Example 5.2. It is not difficult to verify that $\mathcal{K}$ is satisfiable and logically implies that "Tom is a golfer with a possibility of at least 0.7", that is,

$$\mathcal{K} \quad \models \quad (\mathit{Tom}\colon \mathit{golfer},\ P\,0.7)\,.$$

6 Vagueness and Description Logics

In this section, we define fuzzy $\mathcal{SHOIN}(\mathbf{D})$, using the fuzzy operators of Section 2.3. We recall here the semantics given in [109, 112] (see also [95]).

6.1 Syntax

We have seen that $\mathcal{SHOIN}(\mathbf{D})$ allows to reason with concrete datatypes, such as strings and integers, using so-called concrete domains. In our fuzzy approach, concrete domains may be based on fuzzy sets as well.

6.1.1 Fuzzy Datatype Theories

A *fuzzy datatype theory* $\mathbf{D} = (\Delta_{\mathbf{D}},\, \cdot_{\mathbf{D}})$ is defined in the same way as a classical datatype theory except that $\cdot_{\mathbf{D}}$ now assigns to every n-ary datatype predicate an n-ary fuzzy relation over $\Delta_{\mathbf{D}}$. For instance, as for $\mathcal{SHOIN}(\mathbf{D})$, the predicate $\leqslant_{18}$ may be a unary crisp predicate over the natural numbers denoting the set of integers smaller or equal to 18, i.e., $\leqslant_{18}\colon \mathit{Natural} \rightarrow [0,1]$ and

$$\leqslant_{18}(x) = \begin{cases} 1 & \text{if } x \leqslant 18 \\ 0 & \text{otherwise}\,. \end{cases}$$

So,

$$\mathit{Minor} = \mathit{Person} \sqcap \exists \mathit{age}.\leqslant_{18} \tag{4}$$

defines a person, whose age is less or equal to 18, i.e., it defines a minor.

On the other hand, concerning non crisp fuzzy domain predicates, we recall that in fuzzy set theory and practice, there are many functions for specifying fuzzy set membership degrees. However, the triangular, the trapezoidal, the L-function (left-shoulder function), and the R-function (right-shoulder function) are simple, but most frequently used to specify membership degrees. The functions are defined over the set of non-negative rationals $\mathbb{Q}^+ \cup \{0\}$ (see Fig. 1).

Using these functions, we may then define, for instance, $\mathit{Young}\colon \mathit{Natural} \rightarrow [0,1]$ to be a fuzzy concrete predicate over the natural numbers denoting the degree of youngness of a person's age. The concrete fuzzy predicate Young may be defined as $\mathit{Young}(x) = L(x; 10, 30)$. So,

$$\mathit{YoungPerson} = \mathit{Person} \sqcap \exists \mathit{age}.\mathit{Young} \tag{5}$$

denotes a young person.

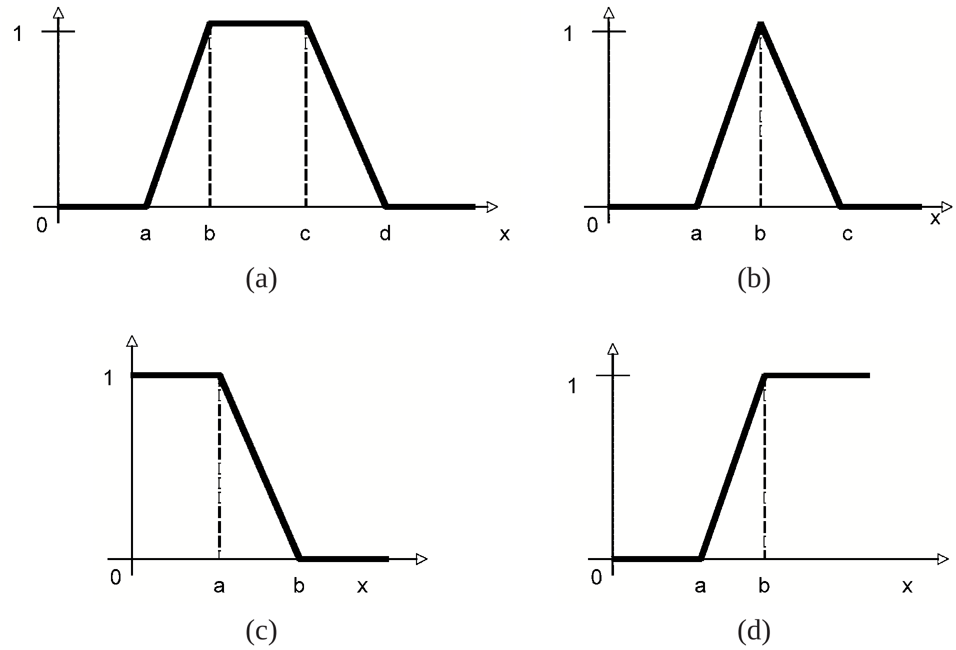

Figure 1: (a) Trapezoidal function; (b) Triangular function; (c) L-function; (d) R-function

6.1.2 Fuzzy Modifiers

We allow modifiers in fuzzy $\mathcal{SHOIN}(\mathbf{D})$. Fuzzy modifiers, like $very$, $more\_or\_less$ and $slightly$, apply to fuzzy sets to change their membership function. Formally, a ***modifier*** is a function $f_m \colon [0,1] \to [0,1]$. For instance, we may define $very(x) = x^2$ and $slightly(x) = \sqrt{x}$. Modifiers have been considered, for instance, in [44, 118]. From a syntactical point of view, if M is a new alphabet for modifier symbols, $m \in M$ is a modifier, and C is a $\mathcal{SHOIN}(\mathbf{D})$ concept, then $m(C)$ is fuzzy concept as well. For instance, by referring to Example 3.1, we may define the concept of sports car as the concept

$$SportsCar = Car \sqcap \exists speed.very(High) \ , \tag{6}$$

where $very$ is a concept modifier, with membership function $very(x) = x^2$, and $High$ is a fuzzy concrete predicate over the domain of speed expressed in kilometers per hour and may be defined as $High(x) = R(x; 80, 250)$.

6.1.3 Fuzzy Knowledge Bases

The syntax of fuzzy $\mathcal{SHOIN}(\mathbf{D})$ concepts is as follows:

$$\begin{array}{rcl} C & \longrightarrow & \top \mid \bot \mid A \mid \{a_1, \ldots, a_n\} \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \neg C \mid m(C) \\ & & \forall R.C \mid \exists R.C \mid (\geqslant n\ S) \mid (\leqslant n\ S) \mid \\ & & \forall T_1, \ldots, T_n.D \mid \exists T_1, \ldots, T_n.D \mid (\geqslant n\ T) \mid (\leqslant n\ T) \, . \end{array}$$

Concerning axioms and assertions, similarly to [103], we define fuzzy axioms as follows: Let be $n \in (0,1]$.

A *fuzzy RBox* $\mathcal{R}$ is a finite set of $\mathcal{SHOIN}(\mathbf{D})$ transitivity axioms $trans(R)$ and *fuzzy role inclusion axioms* of the form $\langle\alpha \geqslant n\rangle$, $\langle\alpha \leqslant n\rangle$, $\langle\alpha > n\rangle$, and $\langle\alpha > n\rangle$, where α is a $\mathcal{SHOIN}(\mathbf{D})$ role inclusion axiom.

A *fuzzy TBox* $\mathcal{T}$ is a finite set of *fuzzy concept inclusion axioms* $\langle\alpha \geqslant n\rangle$, $\langle\alpha \leqslant n\rangle$, $\langle\alpha > n\rangle$, and $\langle\alpha < n\rangle$, where α is a $\mathcal{SHOIN}(\mathbf{D})$ concept inclusion axiom.

A *fuzzy ABox* $\mathcal{A}$ consists of a finite set of *fuzzy concept* and *fuzzy role assertion axioms* of the form $\langle\alpha \geqslant n\rangle$, $\langle\alpha \leqslant n\rangle$, $\langle\alpha > n\rangle$, or $\langle\alpha < n\rangle$, where α is a $\mathcal{SHOIN}(\mathbf{D})$ concept or role assertion. As for the crisp case, $\mathcal{A}$ may also contain a finite set of individual (in)equality axioms $a \approx b$ and $a \not\approx b$, respectively.

For instance, $\langle a\colon C \geqslant 0.1\rangle$, $\langle(a,b)\colon R \leqslant 0.3\rangle$, $\langle R \sqsubseteq S \geqslant 0.4\rangle$, or $\langle C \sqsubseteq D \leqslant 0.6\rangle$ are fuzzy axioms. Informally, from a semantical point of view, a fuzzy axiom $\langle\alpha \leqslant n\rangle$ constrains the membership degree of α to be at most n (similarly for $\geqslant, >, <$). Hence, $\langle jim\colon\ YoungPerson \geqslant 0.2\rangle$ says that jim is a $YoungPerson$ with degree at least 0.2. On the other hand, a fuzzy concept inclusion axiom of the form $\langle C \sqsubseteq D \geqslant n\rangle$ says that the subsumption degree between C and D is at least n.

A $\mathcal{SHOIN}(\mathbf{D})$ *fuzzy knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ consists of a fuzzy TBox $\mathcal{T}$, a fuzzy RBox $\mathcal{R}$, and a fuzzy ABox $\mathcal{A}$.

6.2 Semantics

The semantics extends [103]. The main idea is that concepts and roles are interpreted as fuzzy subsets of an interpretation's domain. Therefore, $\mathcal{SHOIN}(\mathbf{D})$ axioms, rather being satisfied (true) or unsatisfied (false) in an interpretation, become a degree of truth in $[0,1]$. In the following, we use $\wedge, \vee, \neg$ and $\rightarrow$ in infix notation, in place of a t-norm t, s-norm s, negation function n, and implication function i.

6.2.1 Fuzzy Interpretations

A *fuzzy interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ relative to a fuzzy datatype theory $\mathbf{D} = (\Delta_{\mathbf{D}},\ \cdot_{\mathbf{D}})$ consists of a nonempty set $\Delta^{\mathcal{I}}$ (called the *domain*), disjoint from $\Delta_{\mathbf{D}}$, and of a *fuzzy interpretation function* $\cdot^{\mathcal{I}}$ that coincides with $\cdot_{\mathbf{D}}$ on every data value, datatype, and fuzzy datatype predicate, and it assigns

- to each abstract individual $a \in \mathbf{I}$ an element in $\Delta^{\mathcal{I}}$;
- to each abstract concept $C \in \mathbf{A}$ a function $C^{\mathcal{I}}\colon \Delta^{\mathcal{I}} \rightarrow [0,1]$;
- to each abstract role $R \in \mathbf{R}_A$ a function $R^{\mathcal{I}}\colon \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0,1]$;
- to each abstract functional role $R \in \mathbf{R}_A$ a partial function $R^{\mathcal{I}}\colon \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0,1]$ such that for all $x \in \Delta^{\mathcal{I}}$ there is an unique $y \in \Delta^{\mathcal{I}}$ on which $R^{\mathcal{I}}(x,y)$ is defined;
- to each concrete role $T \in \mathbf{R}_C$ a function $R^{\mathcal{I}}\colon \Delta^{\mathcal{I}} \times \Delta_{\mathbf{D}} \rightarrow [0,1]$;
- to each concrete functional role $T \in \mathbf{R}_C$ a partial function $t^{\mathcal{I}}\colon \Delta^{\mathcal{I}} \times \Delta_{\mathbf{D}} \rightarrow [0,1]$ such that for all $x \in \Delta^{\mathcal{I}}$ there is an unique $v \in \Delta_{\mathbf{D}}$ on which $T^{\mathcal{I}}(x,v)$ is defined;
- to each modifier $m \in M$ the modifier function $f_m\colon [0,1] \rightarrow [0,1]$.

The mapping $\cdot^{\mathcal{I}}$ is extended to roles and concepts as specified in the following table (where $x, y \in \Delta^{\mathcal{I}}$ and $v \in \Delta_{\mathbf{D}}$):

$$\begin{array}{rcl} (S^{-})^{\mathcal{I}}(x,y) & = & S^{\mathcal{I}}(y,x) \\ \top^{\mathcal{I}}(x) & = & 1 \end{array}$$

$$\begin{array}{rcl}
\bot^{\mathcal{I}}(x) & = & 0 \\
\{a_1,\ldots,a_n\}^{\mathcal{I}}(x) & = & \bigvee_{i=1}^n a_i^{\mathcal{I}} = x \\
(C_1 \sqcap C_2)^{\mathcal{I}}(x) & = & {C_1}^{\mathcal{I}}(x) \wedge {C_2}^{\mathcal{I}}(x) \\
(C_1 \sqcup C_2)^{\mathcal{I}}(x) & = & {C_1}^{\mathcal{I}}(x) \vee {C_2}^{\mathcal{I}}(x) \\
(\neg C)^{\mathcal{I}}(x) & = & \neg C^{\mathcal{I}}(x)) \\
(m(C))^{\mathcal{I}}(x) & = & f_m(C^{\mathcal{I}}(x)) \\
(\forall R.C)^{\mathcal{I}}(x) & = & \inf_{y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x,y) \rightarrow C^{\mathcal{I}}(y) \\
(\exists R.C)^{\mathcal{I}}(x) & = & \sup_{y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x,y) \wedge C^{\mathcal{I}}(y) \\
(\geqslant n\ S)^{\mathcal{I}}(x) & = & \sup_{\substack{\{y_1,\ldots,y_n\} \subseteq \Delta^{\mathcal{I}} \\ |\{y_1,\ldots,y_n\}| = n}} \bigwedge_{i=1}^n S^{\mathcal{I}}(x,y_i) \\
(\leqslant n\ S)^{\mathcal{I}}(x) & = & \neg(\geqslant n+1\ S)^{\mathcal{I}}(x) \\
(\forall T_1,\ldots,T_n.D)^{\mathcal{I}}(x) & = & \inf_{y_1,\ldots,y_n \in \Delta_{\mathbf{D}}^{\mathcal{I}}} (\bigwedge_{i=1}^n {T_i}^{\mathcal{I}}(x,y_i)) \rightarrow D^{\mathcal{I}}(y_1,\ldots,y_n) \\
(\exists T_1,\ldots,T_n.D)^{\mathcal{I}}(x) & = & \sup_{y_1,\ldots,y_n \in \Delta_{\mathbf{D}}^{\mathcal{I}}} (\bigwedge_{i=1}^n {T_i}^{\mathcal{I}}(x,y_i)) \wedge D^{\mathcal{I}}(y_1,\ldots,y_n)\,.
\end{array}$$

We comment briefly some points. The semantics of $\exists R.C$

$$(\exists R.C)^{\mathcal{I}}(d) \quad = \quad \sup_{y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x,y) \wedge C^{\mathcal{I}}(y)$$

is the result of viewing $\exists R.C$ as the open first order formula $\exists y.F_R(x,y) \wedge F_C(y)$ (where F is the obvious translation of roles and concepts into first-order logic (FOL)) and the existential quantifier $\exists$ is viewed as a disjunction over the elements of the domain. Similarly,

$$(\forall R.C)^{\mathcal{I}}(x) \quad = \quad \inf_{y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x,y) \rightarrow C^{\mathcal{I}}(y)$$

is related to the open first order formula $\forall y.F_R(x,y) \rightarrow F_C(y)$, where the universal quantifier $\forall$ is viewed as a conjunction over the elements of the domain. However, unlike the classical case, in general, we do not have that $(\forall R.C)^{\mathcal{I}} = (\neg\exists R.\neg C)^{\mathcal{I}}$. For instance, it holds in Łukasiewicz logic, but not in Gödel logic. Also interesting is that (see [35]) the axiom $\top \sqsubseteq \neg(\forall R.A) \sqcap (\neg\exists R.\neg A)$ has no classical model. However, in [35], it is shown that in Gödel logic it has no finite model, but has an infinite model.

Another point concerns the semantics of number restrictions. The semantics of the concept $(\geqslant n\ S)$

$$(\geqslant n\ S)^{\mathcal{I}}(x) \quad = \quad \sup_{\substack{\{y_1,\ldots,y_n\} \subseteq \Delta^{\mathcal{I}} \\ |\{y_1,\ldots,y_n\}| = n}} \bigwedge_{i=1}^n S^{\mathcal{I}}(x,y_i)$$

is the result of viewing $(\geqslant n\ S)$ as the open first order formula

$$\exists y_1,\ldots,y_n. \bigwedge_{i=1}^n S(x,y_i) \wedge \bigwedge_{1 \leqslant i < j \leqslant n} y_i \neq y_j\ .$$

That is, there are at least n distinct elements that satisfy to some degree $S(x,y_i)$. This guarantees us that $\exists S.\top \equiv (\geqslant 1\ S)$. The semantics of $(\leqslant n\ S)$ is defined in such a way to guarantee the classical relationship $(\leqslant n\ S) \equiv \neg(\geqslant n+1\ S)$.

An alternative definition for the $(\geqslant n\ S)$ and the $(\leqslant n\ S)$ constructs may rely on the scalar cardinality of a fuzzy set. However, we prefer to stick on the formulation, which derives directly from its FOL translation.

Finally, the mapping $\cdot^{\mathcal{I}}$ is extended to non-fuzzy axioms as specified in the following table (where $a, b \in \mathbf{I}$):

$$\begin{array}{rcl} (R \sqsubseteq S)^{\mathcal{I}} & = & \inf_{x,y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x,y) \rightarrow S^{\mathcal{I}}(x,y) \\ (T \sqsubseteq U)^{\mathcal{I}} & = & \inf_{x,y \in \Delta^{\mathcal{I}}} T^{\mathcal{I}}(x,y) \rightarrow U^{\mathcal{I}}(x,y) \\ (C \sqsubseteq D)^{\mathcal{I}} & = & \inf_{x \in \Delta^{\mathcal{I}}} C^{\mathcal{I}}(x) \rightarrow D^{\mathcal{I}}(x) \\ (a \colon C)^{\mathcal{I}} & = & C^{\mathcal{I}}(a^{\mathcal{I}}) \\ ((a,b) \colon R)^{\mathcal{I}} & = & R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}) \,. \end{array}$$

Note here that e.g. the semantics of a concept inclusion axiom $C \sqsubseteq D$ is derived directly from its FOL translation, which is of the form $\forall x.F_C(x) \rightarrow F_D(x)$. This definition is clearly different from the approaches in which $C \sqsubseteq D$ is viewed as $\forall x.C(x) \leqslant D(x)$. This latter approach has the effect that the subsumption relationship is a classical $\{0, 1\}$ relationship, while the in former approach subsumption is determined up to a certain degree in $[0, 1]$.

The notion of *satisfaction* of a fuzzy axiom E by a fuzzy interpretation $\mathcal{I}$, denoted $\mathcal{I} \models E$, is defined as follows: $\mathcal{I} \models trans(R)$, iff $\forall x, y \in \Delta^{\mathcal{I}}.R^{\mathcal{I}}(x, y) = \sup_{z \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x, z) \wedge R^{\mathcal{I}}(z, y)$. $\mathcal{I} \models \langle \alpha \geqslant n \rangle$, where α is a role inclusion or concept inclusion axiom, iff $\alpha^{\mathcal{I}} \geqslant n$. Similarly, for the other relations $\leqslant, <$ and $>$. $\mathcal{I} \models \langle \alpha \geqslant n \rangle$, where α is a concept or a role assertion axiom, iff $\alpha^{\mathcal{I}} \geqslant n$. Similarly, for the other relations $\leqslant, <, >$. We say that a concept C is *satisfiable* iff there is an interpretation $\mathcal{I}$ and an individual $x \in \Delta^{\mathcal{I}}$ such that $C^{\mathcal{I}}(x) > 0$. Finally, $\mathcal{I} \models a \approx b$ iff $a^{\mathcal{I}} = b^{\mathcal{I}}$ and $\mathcal{I} \models a \not\approx b$ iff $a^{\mathcal{I}} \neq b^{\mathcal{I}}$.

For a set of fuzzy axioms $\mathcal{E}$, we say that $\mathcal{I}$ *satisfies* $\mathcal{E}$ iff I satisfies each element in $\mathcal{E}$. We say that $\mathcal{I}$ is a *model* of E (resp. $\mathcal{E}$) iff $\mathcal{I} \models E$ (resp. $\mathcal{I} \models \mathcal{E}$). $\mathcal{I}$ *satisfies* (is a *model* of) a fuzzy knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{R}, \mathcal{A})$, denoted $\mathcal{I} \models \mathcal{K}$, iff $\mathcal{I}$ is a model of each component $\mathcal{T}, \mathcal{R}$ and $\mathcal{A}$, respectively.

A fuzzy axiom E is a *logical consequence* of a knowledge base $\mathcal{K}$, denoted $\mathcal{K} \models E$ iff every model of $\mathcal{K}$ satisfies E.

The interesting point is that according to our semantics, e.g., a minor is a young person to a certain degree and is obtained without explicitly mentioning it. This inference can not be achieved in classical $\mathcal{SHOIN}(\mathbf{D})$. Similarly, by referring to Example 3.1, we will have that the car tt will be a sports car to a certain degree. Therefore, unlike Example 3.1, tt is now closely a sport car, *as it should be*. The following two examples highlight these points.

Example 6.1 *(Car Example cont'd)* Example 3.1 illustrates an evident difficulty in defining the class of sport cars. Indeed, it is highly questionable why a car whose speed is $243km/h$ is *not* a sport car anymore. The point is that essentially, the higher the speed the more closely a car is a sports car, which makes the concept of sports car rather a *fuzzy* concept, i.e., *vague* concept, rather than a crisp one. In the next section, we will see how to represent such concepts more appropriately. Let us now reconsider Example 3.1, where all axioms of the TBox and ABox are asserted with degree 1, i.e., are of the form $\langle \alpha \geqslant 1 \rangle$. We replace the definition of $SportsCar$ with Definition (6). Then, we have that (under Łukasiewicz logic)

$$\begin{array}{ll} \mathcal{K} \models \langle SportsCar \sqsubseteq Car \geqslant 1 \rangle & \mathcal{K} \models \langle mgb \colon SportsCar \leqslant 0.63 \rangle \\ \mathcal{K} \models \langle enzo \colon SportsCar \geqslant 1 \rangle & \mathcal{K} \models \langle tt \colon SportsCar \geqslant 0.97 \rangle \,. \end{array}$$

Note how the maximal speed limit of the mgb car ($\leqslant_{170km/h}$) induces an upper limit, 0.53, of the membership degree. Neither this inference is possible in classical $\mathcal{SHOIN}(\mathbf{D})$, nor the one involving tt.

Example 6.2 Consider the knowledge base $\mathcal{K}$ with definitions (4) and (5). Then under Łukasiewicz logic we have that (see [108])

$$\begin{array}{lcl} \mathcal{K} & \models & \langle Minor \sqsubseteq YoungPerson \geqslant 0.6\rangle \\ \mathcal{K} & \models & \langle YoungPerson \sqsubseteq Minor \geqslant 0.4\rangle \end{array}$$

which are relationships not captured with classical $\mathcal{SHOIN}(\mathbf{D})$.

6.2.2 Best Truth Value Bound

Finally, given $\mathcal{K}$ and an axiom α, where α is neither a transitivity axiom, nor an individual (in) equality axiom, it is of interest to compute α's best lower and upper degree value bounds (*Best Truth Value Bound* (BTVB)). The *greatest lower bound* of α w.r.t. $\mathcal{K}$ (denoted $glb(\mathcal{K}, \alpha)$) is

$$glb(\mathcal{K}, \alpha) = \sup\{n \mid \mathcal{K} \models \langle \alpha \geqslant n\rangle\}\ ,$$

while the *least upper bound* of α with respect to $\mathcal{K}$ (denoted $lub(\mathcal{K}, \alpha)$ is

$$lub(\mathcal{K}, \alpha) = \inf\{n \colon \mathcal{K} \models \langle \alpha \leqslant n\rangle\}\ ,$$

where $\sup \emptyset = 0$ and $\inf \emptyset = 1$. Determining the lub and the glb is called the *Best Degree Bound* (BDB) problem. For instance, the consequences in Examples 6.1 and 6.2 are the best possible degree bounds. Furthermore, note that,

$$lub(\Sigma, a\colon C) = \neg glb(\Sigma, a\colon \neg C)\ , \tag{7}$$

i.e., the lub can be determined through the glb (and vice-versa).

Similarly, $lub(\Sigma, (a, b)\colon R) = \neg glb(\Sigma, a\colon \neg\exists R.\{b\})$ holds. Also, note that, $\Sigma \models \langle \alpha \geqslant n\rangle$ iff $glb(\Sigma, \alpha) \geqslant n$, and similarly $\Sigma \models \langle \alpha \leqslant n\rangle$ iff $lub(\Sigma, \alpha) \leqslant n$ hold.

Another similar concept is the *best satisfiability bound* of a concept C and amounts to determine

$$glb(\mathcal{K}, C) = \sup_{\mathcal{I}} \sup_{x \in \Delta^{\mathcal{I}}} \{C^{\mathcal{I}}(x) \mid \mathcal{I} \models \mathcal{K}\}\ .$$

Essentially, among all models $\mathcal{I}$ of the knowledge base, we are determining the maximal degree of truth that the concept C may have over all individuals $x \in \Delta^{\mathcal{I}}$.

Example 6.3 Consider the knowledge base $\mathcal{K}$ in Example 3.1. Assume, that a car seller sells an Audi TT for \$31500, as from the catalog price. A buyer is looking for a sports-car, but wants to pay not more than around \$30000. In classical DLs no agreement can be found. The problem relies on the crisp condition on the seller's and the buyer's price. A more fine grained approach would be (and usually happens in negotiation) to consider prices as concrete fuzzy sets instead. For instance, the seller may consider optimal to sell above \$31500, but can go down to \$30500. The buyer prefers to spend less than \$30000, but can go up to \$32000. We may represent these statements by means of the following axioms (see Figure 2):

$$\begin{array}{lcl} AudiTT & = & SportsCar \sqcap \exists hasPrice.R(x; 30500, 31500) \\ Query & = & SportsCar \sqcap \exists hasPrice.L(x; 30000, 32000) \end{array}$$

Then we may find out that the highest degree to which the concept $C = AudiTT \sqcap Query$ is satisfiable is 0.75 (the possibility that the Audi TT and the query matches is 0.75). That is, $glb(\mathcal{K}, C) = 0.75$ and corresponds to the point where both requests intersects (i.e., the car may be sold at \$31250).

Problems such as determining the glb can be solved by relying on mixed integer linear programming as done in [106, 107] and in the *fuzzyDL* system (accessible from Umberto Straccia's home page).

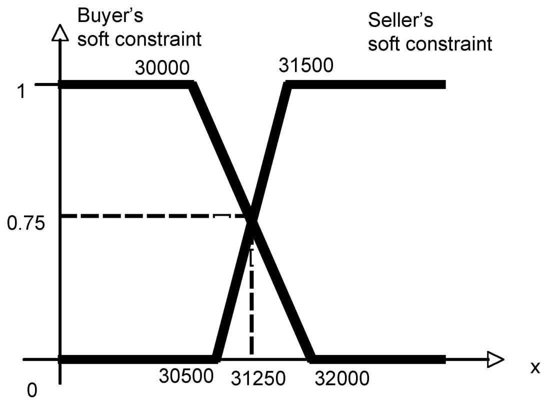

Figure 2: The soft price constraints.

6.3 Related Work

Several ways of extending DLs using the theory of fuzzy logic have been proposed in the literature. The first work is due to Yen [124] who considered a sub-language of $\mathcal{ALC}$, $\mathcal{FL}^-$ [7, 58]. However, it already informally talks about the use of modifiers and concrete domains. Though, the unique reasoning facility, the subsumption test, is a crisp yes/no question. Tresp [118] considered fuzzy $\mathcal{ALC}$ extended with a special form of modifiers, which are a combination of two linear functions: $\min$, $\max$ and $1-x$ membership functions has been considered and a sound and complete reasoning algorithm testing the subsumption relationship has been presented. Similarly to Straccia's work [106, 107], a linear programming oracle is needed.

Assertional reasoning has been considered by Straccia [100, 102, 103], where fuzzy assertion axioms have been allowed in fuzzy $\mathcal{ALC}$ (with $\min$, $\max$ and $1-x$ functions), concept modifiers and fuzzy concrete domains are not allowed however ([102] reports a four-valued variant of fuzzy $\mathcal{ALC}$). He also introduced the BTVB problem and provided a sound and complete reasoning algorithm based on completion rules. In the same spirit, Hölldobler *et al.* [41, 43, 44, 42] extend Straccia's fuzzy $\mathcal{ALC}$ with concept modifiers of the form $f_m(x) = x^\beta$, where $\beta > 0$. A sound and complete reasoning algorithm for the graded subsumption problem, based on completion rules, is presented.

Straccia's works [105, 111, 116] are essentially as [103], except that now the truth space is a complete lattice rather than $[0, 1]$.

Sanchez and Tettamanzi [91, 92, 93] start addressing the issue of alternative semantics of quantifiers in fuzzy $\mathcal{ALC}$ (without the assertional component). Essentially, fuzzy quantifiers allow to state sentences such as $\mathit{FaithfulCustomer} \sqcap (\mathit{Most})\mathit{buys}.\ \mathit{LowCalorieFood}$ denoting "the set of individuals that mostly by low calorie food".

Hajek [35, 36] considers $\mathcal{ALC}$ under arbitrary t-norm and reports, among others, a procedure deciding $\models \langle C \sqsubseteq D \geqslant 1\rangle$ and deciding whether $\langle C \sqsubseteq D \geqslant 1\rangle$ is satisfiable, by a reduction to the propositional BL logic, for which a Hilbert style axiomatization exists [34] (but see also [36] for complexity of rational Pavelka logic and see [5], which reports some complexity results for reasoning in fuzzy DLs).

Straccia [104] provides a translation of fuzzy $\mathcal{ALC}$ with GCI's into classical $\mathcal{ALC}$. The translation is modular and, thus, it is expected that it can be extended to more expressive DLs as well. The idea is to translate a fuzzy assertion of the form $\langle a\colon C \geqslant n\rangle$ into a crisp assertion $a\colon C_n$ with intended meaning "a is instance of C to degree at least n". It then uses GCI's to correctly relate the C_n. For instance, $C_{0.7} \sqsubseteq C_{0.6}$ is used to say that whenever an individual is instance of C to degree at least 0.7 then it is also instance of C to degree at least 0.6. The translation is at most quadratic in the size of the knowledge base. The idea has further been considered by [61, 62], which essentially provide a crisp language in which expressions of the form e.g. $a\colon \forall R_{0.8}.C_{0.9}$ are allowed, with intended meaning: "if a has an R-successor to degree at least 0.8 then this successor is also an instance of C to degree at least 0.9".

Other extension of fuzzy DLs, mainly concern their integration with logic programming, which we however do not report here (see, e.g. [116, 113, 111]). Also, Kang *et al.* [54] extends fuzzy DLs by allowing comparison operators, e.g., allowing to state that "Tom is more tall than Tim". Another interesting extension is [16], which combines fuzzy DLs with possibility theory. Essentially, as $\langle a\colon C \geqslant n\rangle$ is Boolean (either an interpretation satisfies it or not), we can build on top of it an uncertainty logic, which is based on possibility theory in [16].

From a reasoning point of view, no calculus exists yet checking satisfiability of fuzzy $\mathcal{SHOIN}(\mathbf{D})$ knowledge bases, though there exist an implementation for fuzzy $\mathcal{SHIF}(\mathbf{D})$ (the *fuzzyDL* system) supporting Zadeh semantics, Łukasiewicz semantics and classical semantics.

Usually, the semantics used for fuzzy DLs follows the so-called Zadeh semantics, but where the concept inclusion is crisp, i.e., $C \sqsubseteq D$ is viewed as $\forall x.C(x) \leqslant D(x)$. [44, 118] report a calculus for the case of $\mathcal{ALC}$ [94] with modifiers and simple TBox under Zadeh semantics. No indication for the BTVB problem is given. [100, 103] reports a calculus for $\mathcal{ALC}$ and simple TBox under Zadeh semantics and addresses the BTVB problem. [104] shows how the satisfiability problem and the BTVB problem can be reduced to classical $\mathcal{ALC}$ and, thus, can be resolved by means of a tools like FACT and RACER. [96, 97] show results providing a tableaux calculus for fuzzy $\mathcal{SHIN}$ without GCIs and under the Zadeh semantics, by adapting similar techniques developed for the classical counterpart. Fuzzy GCIs under Zadeh semantics can be managed as described in [98]. Ultimately, we expect that the techniques developed for classical $\mathcal{SHOIN}(\mathbf{D})$ can be extended to the work of [103] as [96, 97] already show. Also interesting is the work [60], which provides a tableaux for fuzzy $\mathcal{SHI}$ with GCI's.

On the other hand side, fuzzy tableaux algorithms under Zadeh semantics, seem not to be suitable to be adapted to other semantics, such as Łukasiewicz logic. Even more problematic is the fact that they are unable to deal with fuzzy concrete domains. However, despite these negative results, recently [107, 106] reports a calculus for $\mathcal{ALC}(\mathbf{D})$ whenever the connectives, the modifiers and the concrete fuzzy predicates are representable as a bounded Mixed Integer Linear Program (MILP). For instance, Łukasiewicz logic satisfies these conditions as well as the membership functions for concrete fuzzy predicates we have presented in this paper. Additionally, modifiers should be a combination of linear functions. In that case the calculus consists of a set of constraint propagation rules and an invocation to an oracle for MILP. The method has been extended to fuzzy $\mathcal{SHIF}(\mathbf{D})$ (the DL behind OWL-Lite) and a reasoner, called *fuzzyDL*, has been implemented and is available from Straccia's Web page (though, a paper describing the algorithm has not yet been published). *FuzzyDL* supports more features than we have described in this work, whose description go beyond the scope of this work. The use of MILP for reasoning in fuzzy DLs is not surprising as their use for automated deduction in many-valued logics is well- known [31, 32].

A new problem for fuzzy DLs is the top-k retrieval problem. While in classical semantics a tuple satisfies a query or does not satisfy the query, in fuzzy DLs a tuple may satisfy a query to degree. Hence, for instance, given a conjunctive query over a fuzzy DLs knowledge base, it is of interest to compute just the top-k answers. While in relational databases this problem is a current research area (see, e.g. [23, 50, 59]), almost nothing is known for the case of first-order knowledge bases in general (but see [114]) and DLs in particular. The only works we are aware of are [110, 115] dealing with the problem of finding the top-k result over a DL-Lite [8] knowledge bases.

We conclude by pointing out that fuzzy DLs has first been proposed for *logic-based information retrieval*. [81] summarizes many previous works on the same argument [82, 83, 76, 77, 78, 79, 78, 80, 96, 99, 101, 101, 102], which originated from the idea to annotated textual documents with graded DL sentences [82]. Other applications are [64] and [12].

7 Conclusions

Handling uncertainty and vagueness has started to play an important role in ontologies and description logics for the Semantic Web. In this paper, we have first provided a brief introduction to uncertainty and vagueness at the propositional level. We have then given an overview of probabilistic uncertainty, possibilistic uncertainty, and vagueness in expressive description logics for the Semantic Web.

An interesting topic of future research is the integration of the above forms of uncertainty and vagueness in a single description logic for the Semantic Web. Another issue for future research is the integration of probabilistic, possibilistic, and fuzzy description logics with rule-based languages for the Semantic Web.

References

[1] S. Amarger, D. Dubois, and H. Prade. Constraint propagation with imprecise conditional probabilities. In *Proceedings UAI-1991*, pages 26–34. Morgan Kaufmann, 1991.

[2] F. Baader and P. Hanschke. A schema for integrating concrete domains into concept languages. In *Proceedings IJCAI-1991*, pages 452–457. Morgan Kaufmann, 1991.

[3] T. Berners-Lee. *Weaving the Web*. Harper, San Francisco, 1999.

[4] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *The Scientific American*, 284(5):34–43, 2001.

[5] P. Bonatti and A. Tettamanzi. Some complexity results on fuzzy description logics. In V. Di Gesù, F. Masulli, and A. Petrosino, editors, *Fuzzy Logic and Applications*, volume 2955 of *LNCS*, pages 19–24. Springer, 2006.

[6] G. Boole. *An Investigation of the Laws of Thought, on which are Founded the Mathematical Theories of Logic and Probabilities*. Walton and Maberley, London, 1854. (reprint: Dover Publications, New York, 1958).

[7] R. J. Brachman and H. J. Levesque. The tractability of subsumption in frame-based description languages. In *Proceedings AAAI-1984*, pages 34–37. AAAI Press, 1984.

[8] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. *DL-Lite*: Tractable description logics for ontologies. In *Proceedings AAAI-2005*, pages 602–607. AAAI Press / MIT Press, 2005.

[9] P. C. G. da Costa. *Bayesian semantics for the Semantic Web*. PhD thesis, George Mason University, Fairfax, VA, USA, 2005.

[10] P. C. G. da Costa and K. B. Laskey. PR-OWL: A framework for probabilistic ontologies. In *Proceedings FOIS-2006*, 2006.

[11] P. C. G. da Costa, K. B. Laskey, and K. J. Laskey. PR-OWL: A Bayesian ontology language for the Semantic Web. In *Proceedings URSW-2005*, pages 23–33, 2005.

[12] M. d'Aquin, J. Lieber, and A. Napoli. Towards a semantic portal for oncology using a description logic with fuzzy concrete domains. In E. Sanchez, editor, *Fuzzy Logic and the Semantic Web*, Capturing Intelligence, pages 379–393. Elsevier, 2006.

[13] Z. Ding and Y. Peng. A probabilistic extension to ontology language OWL. In *Proceedings HICSS-2004*, 2004.

[14] Z. Ding, Y. Peng, and R. Pan. BayesOWL: Uncertainty modeling in Semantic Web ontologies. In Z. Ma, editor, *Soft Computing in Ontologies and Semantic Web*, volume 204 of *Studies in Fuzziness and Soft Computing*. Springer, 2006.

[15] D. Dubois, J. Lang, and H. Prade. Possibilistic logic. In D. M. Gabbay, C. J. Hogger, and J. A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3, pages 439–513. Clarendon Press, Oxford, 1994.

[16] D. Dubois, J. Mengin, and H. Prade. Possibilistic uncertainty and fuzzy features in description logic: A preliminary discussion. In E. Sanchez, editor, *Capturing Intelligence: Fuzzy Logic and the Semantic Web*. Elsevier, 2006.

[17] D. Dubois and H. Prade. On fuzzy syllogisms. *Computational Intelligence*, 4(2):171–179, 1988.

[18] D. Dubois and H. Prade. Can we enforce full compositionality in uncertainty calculi? In *Proceedings AAAI-1994*, pages 149–154. AAAI Press, 1994.

[19] D. Dubois and H. Prade. Possibility theory, probability theory and multiple-valued logics: A clarification. *Ann. Math. Artif. Intell.*, 32(1–4):35–66, 2001.

[20] D. Dubois, H. Prade, L. Godo, and R. L. de Màntaras. Qualitative reasoning with imprecise probabilities. *J. Intell. Inf. Syst.*, 2:319–363, 1993.

[21] D. Dubois, H. Prade, and J.-M. Touscas. Inference with imprecise numerical quantifiers. In Z. W. Ras and M. Zemankova, editors, *Intelligent Systems*, chapter 3, pages 53–72. Ellis Horwood, 1990.

[22] M. Dürig and T. Studer. Probabilistic ABox reasoning: Preliminary results. In *Proceedings DL-2005*, pages 104–111, 2005.

[23] R. Fagin. Combining fuzzy information: An overview. *SIGMOD Rec.*, 31(2):109–118, 2002.

[24] R. Fagin, J. Y. Halpern, and N. Megiddo. A logic for reasoning about probabilities. *Inf. Comput.*, 87:78–128, 1990.

[25] D. Fensel, W. Wahlster, H. Lieberman, and J. Hendler, editors. *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*. MIT Press, 2002.

[26] A. M. Frisch and P. Haddawy. Anytime deduction for probabilistic logic. *Artif. Intell.*, 69(1–2):93–122, 1994.

[27] Y. Fukushige. Representing probabilistic knowledge in the Semantic Web. In *Proceedings of the W3C Workshop on Semantic Web for Life Sciences*, 2004.

[28] R. Giugno and T. Lukasiewicz. P-$\mathcal{SHOQ}(\mathbf{D})$: A probabilistic extension of $\mathcal{SHOQ}(\mathbf{D})$ for probabilistic ontologies in the Semantic Web. In *Proceedings JELIA-2002*, volume 2424 of *LNCS*, pages 86–97. Springer, 2002.

[29] N. Guarino. Formal ontology, conceptual analysis and knowledge representation. *Int. J. Hum.-Comput. Stud.*, 43(5–6):625–640, 1995.

[30] V. Haarslev and R. Möller. RACER system description. In *Proceedings IJCAR-2001*, volume 2083 of *LNCS*, pages 701–706. Springer, 2001.

[31] R. Hähnle. Many-valued logics and mixed integer programming. *Ann. Math. Artif. Intell.*, 12(3–4):231–263, 1994.

[32] R. Hähnle. Advanced many-valued logics. In D. M. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic, 2nd Edition*, volume 2. Kluwer, Dordrecht, Holland, 2001.

[33] T. Hailperin. *Sentential Probability Logic: Origins, Development, Current Status, and Technical Applications*. Associated University Presses, London, UK, 1996.

[34] P. Hájek. *Metamathematics of Fuzzy Logic*. Kluwer, 1998.

[35] P. Hájek. Making fuzzy description logics more expressive. *Fuzzy Sets and Systems*, 154(1):1–15, 2005.

[36] P. Hájek. What does mathematical fuzzy logic offer to description logic? In E. Sanchez, editor, *Capturing Intelligence: Fuzzy Logic and the Semantic Web*. Elsevier, 2006.

[37] P. Hansen, B. Jaumard, G.-B. D. Nguetsé, and M. P. de Aragão. Models and algorithms for probabilistic and Bayesian logic. In *Proceedings IJCAI-1995*, pages 1862–1868. Morgan Kaufmann, 1995.

[38] J. Heinsohn. Probabilistic description logics. In *Proceedings UAI-1994*, pages 311–318. Morgan Kaufmann, 1994.

[39] M. Holi and E. Hyvönen. A method for modeling uncertainty in Semantic Web taxonomies. In *Proceedings WWW-2004*, pages 296–297. ACM Press, 2004.

[40] M. Holi and E. Hyvönen. Modeling degrees of conceptual overlap in Semantic Web ontologies. In *Proceedings URSW-2005*, pages 98–99, 2005.

[41] S. Hölldobler, T. D. Khang, and H.-P. Störr. A fuzzy description logic with hedges as concept modifiers. In *Proceedings InTech/VJFuzzy-2002*, pages 25–34, 2002.

[42] S. Hölldobler, N. H. Nga, and T. D. Khang. The fuzzy description logic $\mathcal{ALC}_{FLH}$. In *Proceeedings DL-2005*, 2005.

[43] S. Hölldobler, H.-P. Störr, and T. D. Khang. The fuzzy description logic ALC_{FH} with hedge algebras as concept modifiers. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 7(3):294–305, 2003.

[44] S. Hölldobler, H.-P. Störr, and T. D. Khang. The subsumption problem of the fuzzy description logic ALC_{FH}. In *Proceedings IPMU-2004*, pages 243–250, 2004.

[45] B. Hollunder. An alternative proof method for possibilistic logic and its application to terminological logics. *Int. J. Approx. Reasoning*, 12(2):85–109, 1995.

[46] I. Horrocks. Using an expressive description logic: FaCT or fiction? In *Proceedings KR-1998*, pages 636–649. Morgan Kaufmann, 1998.

[47] I. Horrocks and P. Patel-Schneider. Reducing OWL entailment to description logic satisfiability. *J. Web Sem.*, 1(4):345–357, 2004.

[48] I. Horrocks and P. F. Patel-Schneider. Reducing OWL entailment to description logic satisfiability. In *Proceedings ISWC-2003*, volume 2870 of *LNCS*, pages 17–29, 2003.

[49] I. Horrocks, P. F. Patel-Schneider, and F. van Harmelen. From $\mathcal{SHIQ}$ and RDF to OWL: The making of a web ontology language. *J. Web Sem.*, 1(1):7–26, 2003.

[50] I. F. Ilyas, W. G. Aref, and A. K. Elmagarmid. Supporting top-k join queries in relational databases. In *Proceedings VLDB-2003*, pages 754–765. Morgan Kaufmann, 2003.

[51] M. Jaeger. Probabilistic reasoning in terminological logics. In *Proceedings KR-1994*, pages 305–316. Morgan Kaufmann, 1994.

[52] M. Jaeger. Probabilistic role models and the guarded fragment. In *Proceedings IPMU-2004*, pages 235–242, 2004. Extended version in *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *14*(1), 43–60, 2006.

[53] B. Jaumard, P. Hansen, and M. P. de Aragão. Column generation methods for probabilistic logic. *ORSA J. Comput.*, 3:135–147, 1991.

[54] D. Kang, B. Xu, J. Lu, and Y. Li. Reasoning for a fuzzy description logic with comparison expressions. In *Proceeedings DL-2006*, 2006.

[55] D. Koller, A. Levy, and A. Pfeffer. P-CLASSIC: A tractable probabilistic description logic. In *Proceedings AAAI-1997*, pages 390–397. AAAI Press / MIT Press, 1997.

[56] S. Kraus, D. Lehmann, and M. Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *Artif. Intell.*, 14(1):167–207, 1990.

[57] D. Lehmann. Another perspective on default reasoning. *Ann. Math. Artif. Intell.*, 15(1):61–82, 1995.

[58] H. J. Levesque and R. J. Brachman. Expressiveness and tractability in knowledge representation and reasoning. *Computational Intelligence*, 3:78–93, 1987.

[59] C. Li, K. C.-C. Chang, I. F. Ilyas, and S. Song. RankSQL: Query algebra and optimization for relational top-k queries. In *Proceedings ACM SIGMOD-2005*, pages 131–142. ACM Press, 2005.

[60] Y. Li, B. Xu, J. Lu, and D. Kang. Discrete tableau algorithms for SHI. In *Proceeedings DL-2006*, 2006.

[61] Y. Li, B. Xu, J. Lu, D. Kang, and P. Wang. Extended fuzzy description logic ALCN. In *Proceedings KES-2005*, volume 3684 of *LNCS*, pages 896–902. Springer, 2005.

[62] Y. Li, B. Xu, J. Lu, D. Kang, and P. Wang. A family of extended fuzzy description logics. In *Proc. COMPSAC-2005*, pages 221–226. IEEE Computer Society, 2005.

[63] C.-J. Liau and Y. Y. Yao. Information retrieval by possibilistic reasoning. In *Proceedings DEXA-2001*, volume 2113 of *LNCS*, pages 52–61. Springer, 2001.

[64] O. Liu, Q. Tian, and J. Ma. A fuzzy description logic approach to model management in R&D project selection. In *Proceedings PACIS-2004*, 2004.

[65] T. Lukasiewicz. Local probabilistic deduction from taxonomic and probabilistic knowledge-bases over conjunctive events. *Int. J. Approx. Reasoning*, 21(1):23–61, 1999.

[66] T. Lukasiewicz. Probabilistic deduction with conditional constraints over basic events. *J. Artif. Intell. Res.*, 10:199–241, 1999.

[67] T. Lukasiewicz. Probabilistic logic programming under inheritance with overriding. In *Proceedings UAI-2001*, pages 329–336. Morgan Kaufmann, 2001.

[68] T. Lukasiewicz. Probabilistic logic programming with conditional constraints. *ACM Trans. Comput. Log.*, 2(3):289–339, 2001.

[69] T. Lukasiewicz. Probabilistic default reasoning with conditional constraints. *Ann. Math. Artif. Intell.*, 34(1–3):35–88, 2002.

[70] T. Lukasiewicz. Weak nonmonotonic probabilistic logics. *Artif. Intell.*, 168(1–2):119–161, 2005.

[71] T. Lukasiewicz. Probabilistic description logics for the Semantic Web. Technical Report INFSYS RR-1843-06-04, Institut für Informationssysteme, Technische Universität Wien, June 2006. Submitted for journal publication.

[72] C. Lutz. Reasoning with concrete domains. In *Proceedings IJCAI-1999*, pages 90–95. Morgan Kaufmann, 1999.

[73] C. Lutz. Description logics with concrete domains — A survey. In P. Balbiani, N.-Y. Suzuki, F. Wolter, and M. Zakharyaschev, editors, *Advances in Modal Logics*, volume 4. King's College Publications, 2003.

[74] C. Lutz. NEXP TIME-complete description logics with concrete domains. *ACM Trans. Comput. Logic*, 5(4):669–705, 2004.

[75] T. Mantay, R. Möller, and A. Kaplunova. Computing probabilistic least common subsumers in description logics. In *Proceedings KI-1999*, volume 1701 of *LNCS*, pages 89–100. Springer, 1999.

[76] C. Meghini, F. Sebastiani, and U. Straccia. Designing effective retrieval engines for multimedia document repositories. In *AI*IA 1996 Workshop on Access, Extraction and Integration of Knowledge*, pages 67–70, 1996.

[77] C. Meghini, F. Sebastiani, and U. Straccia. Reasoning about form, content and uncertainty for multimedia document retrieval. In *Proceedings of the 2nd International Workshop on Logic and Uncertainty in Information Retrieval*, pages 57–58, 1996.

[78] C. Meghini, F. Sebastiani, and U. Straccia. Modelling the retrieval of structured documents containing texts and images. In *Proceedings ECDL-1997*, volume 1324 of *LNCS*, pages 325–344. Springer, 1997.

[79] C. Meghini, F. Sebastiani, and U. Straccia. The terminological image retrieval model. In *Proceedings ICIAP-1997*, volume 1311 of *LNCS*, pages 156–163. Springer, 1997.

[80] C. Meghini, F. Sebastiani, and U. Straccia. A system for the fast prototyping of multidimensional image retrieval. In *Proceedings IEEE ICMCS-1999*, volume 2, pages 603–609. IEEE Computer Society, 1999.

[81] C. Meghini, F. Sebastiani, and U. Straccia. A model of multimedia information retrieval. *J. ACM*, 48(5):909–970, 2001.

[82] C. Meghini, F. Sebastiani, U. Straccia, and C. Thanos. A model of information retrieval based on a terminological logic. In *Proceedings ACM SIGIR-1993*, pages 298–307. ACM Press, 1993.

[83] C. Meghini and U. Straccia. A relevance terminological logic for information retrieval. In *Proceedings ACM SIGIR-1996*, pages 197–205. ACM Press, 1996.

[84] P. Mitra, N. F. Noy, and A. Jaiswal. OMEN: A probabilistic ontology mapping tool. In *Proceedings ISWC-2005*, volume 3729 of *LNCS*, pages 537–547. Springer, 2005.

[85] N. J. Nilsson. Probabilistic logic. *Artif. Intell.*, 28(1):71–88, 1986.

[86] H. Nottelmann and N. Fuhr. Adding probabilities and rules to OWL Lite subsets based on probabilistic Datalog. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 14(1):17–42, 2006.

[87] V. Novák. Which logic is the real fuzzy logic? *Fuzzy Sets and Systems*, 157(5):635–641, 2005.

[88] R. Pan, Z. Ding, Y. Yu, and Y. Peng. A Bayesian network approach to ontology mapping. In *Proceedings ISWC-2005*, volume 3729 of *LNCS*, pages 563–577. Springer, 2005.

[89] B. Parsia, E. Sirin, M. Grove, and R. Alford. Pellet. Available at www.mindswap. org/2003/pellet/.

[90] M. Pool and J. Aikin. KEEPER and Protégé: An elicitation environment for Bayesian inference tools. In *Proceedings of the Workshop on Protégé and Reasoning* held at the *7th International Protégé Conference*, 2004.

[91] D. Sanchez and A. Tettamanzi. Generalizing quantification in fuzzy description logics. In *Proceedings 8th Fuzzy Days in Dortmund*, 2004.

[92] D. Sanchez and A. Tettamanzi. Reasoning and quantification in fuzzy description logics. In *Fuzzy Logic and Applications*, volume 3849 of *LNCS*, pages 81–88. Springer, 2005.

[93] D. Sanchez and A. Tettamanzi. Fuzzy quantification in fuzzy description logics. In E. Sanchez, editor, *Capturing Intelligence: Fuzzy Logic and the Semantic Web*. Elsevier, 2006.

[94] M. Schmidt-Schauß and G. Smolka. Attributive concept descriptions with complements. *Artif. Intell.*, 48(1):1–26, 1991.

[95] G. Stoilos, G. Stamou, V. Tzouvaras, J. Pan, and I. Horrocks. Fuzzy OWL: Uncertainty and the Semantic Web. In *Proceedings of the International Workshop of OWL: Experiences and Directions*, 2005.

[96] G. Stoilos, G. Stamou, V. Tzouvaras, J. Z. Pan, and I. Horrock. A fuzzy description logic for multimedia knowledge representation. In *Proceedings of the International Workshop on Multimedia and the Semantic Web*, 2005.

[97] G. Stoilos, G. B. Stamou, V. Tzouvaras, J. Z. Pan, and I. Horrocks. The fuzzy description logic $f\text{-}\mathcal{SHIN}$. In *Proceedings URSW-2005*, pages 67–76, 2005.

[98] G. Stoilos, U. Straccia, G. Stamou, and J. Pan. General concept inclusions in fuzzy description logics. In *Proceedings ECAI-2006*, pages 457–461. IOS Press, 2006.

[99] U. Straccia. Document retrieval by relevance terminological logics. In *Proceedings MIRO-1995*, Workshops in Computing. BCS, 1995.

[100] U. Straccia. A fuzzy description logic. In *Proceedings AAAI-1998*, pages 594–599. AAAI Press / MIT Press, 1998.

[101] U. Straccia. *Foundations of a Logic based approach to Multimedia Document Retrieval*. PhD thesis, Department of Computer Science, University of Dortmund, Dortmund, Germany, June 1999.

[102] U. Straccia. A framework for the retrieval of multimedia objects based on four-valued fuzzy description logics. In F. Crestani and G. Pasi, editors, *Soft Computing in Information Retrieval: Techniques and Applications*, pages 332–357. Physica (Springer), Heidelberg, Germany, 2000.

[103] U. Straccia. Reasoning within fuzzy description logics. *J. Artif. Intell. Res.*, 14:137–166, 2001.

[104] U. Straccia. Transforming fuzzy description logics into classical description logics. In *Proceedings JELIA-2004*, volume 3229 of *LNCS*, pages 385–399. Springer, 2004.

[105] U. Straccia. Uncertainty in description logics: A lattice-based approach. In *Proceedings IPMU-2004*, pages 251–258, 2004.

[106] U. Straccia. Description logics with fuzzy concrete domains. In *Proceedings UAI-2005*, pages 559–567. AUAI Press, 2005.

[107] U. Straccia. Fuzzy ALC with fuzzy concrete domains. In *Proceeedings DL-2005*, pages 96–103, 2005.

[108] U. Straccia. Fuzzy description logics with concrete domains. Technical Report 2005-TR-03, Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, Italy, 2005.

[109] U. Straccia. Towards a fuzzy description logic for the Semantic Web (preliminary report). In *Proceedings ESWC-2005*, volume 3532 of *LNCS*, pages 167–181. Springer, 2005.

[110] U. Straccia. Answering vague queries in fuzzy *DL-Lite*. In *Proceedings IPMU-2006*, pages 2238–2245, 2006.

[111] U. Straccia. Description logics over lattices. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 14(1):1–16, 2006.

[112] 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.

[113] U. Straccia. Fuzzy description logic programs. In *Proceedings IPMU-2006*, pages 1818–1825, 2006.

[114] U. Straccia. Towards top-k query answering in deductive databases. In *Proceedings IEEE SMC-2006*. IEEE Computer Society, 2006.

[115] U. Straccia. Towards top-k query answering in description logics: The case of *DL-Lite*. In *Proc. JELIA-2006*, volume 4160 of *LNCS*, pages 439–451. Springer, 2006.

[116] U. Straccia. Uncertainty and description logic programs over lattices. In E. Sanchez, editor, *Fuzzy Logic and the Semantic Web*, Capturing Intelligence, chapter 7, pages 115–133. Elsevier, 2006.

[117] C. Thomas and A. Sheth. On the expressiveness of the languages for the Semantic Web — Making a case for "A little more". In E. Sanchez, editor, *Fuzzy Logic and the Semantic Web*, Capturing Intelligence, pages 3–20. Elsevier, 2006.

[118] C. Tresp and R. Molitor. A description logic for vague knowledge. In *Proceedings ECAI-1998*, pages 361–365. J. Wiley & Sons, 1998.

[119] O. Udrea, Y. Deng, E. Hung, and V. S. Subrahmanian. Probabilistic ontologies and relational databases. In *Proceedings CoopIS/DOA/ODBASE-2005*, volume 3760 of *LNCS*, pages 1–17. Springer, 2005.

[120] W3C. OWL web ontology language overview, 2004. W3C Recommendation (10 Feb. 2004). Available at www.w3.org/TR/2004/REC-owl-features-20040210/.

[121] G. Weikum, J. Graupmann, R. Schenkel, and M. Theobald. Towards a statistically Semantic Web. In *Proceedings ER-2004*, volume 3288 of *LNCS*, pages 3–17. Springer, 2004.

[122] Y. Yang and J. Calmet. OntoBayes: An ontology-driven uncertainty model. In *Proceedings IAWTIC-2005*, pages 457–463. IEEE Press, 2005.

[123] P. M. Yelland. An alternative combination of Bayesian networks and description logics. In *Proceedings KR-2000*, pages 225–234. Morgan Kaufmann, 2000.

[124] J. Yen. Generalizing term subsumption languages to fuzzy logic. In *Proceedings IJCAI-1991*, pages 472–477. Morgan Kaufmann, 1991.

[125] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.