Web Semantics: Science, Services and Agents on the World Wide Web 6 (2008) 291–308

Contents lists available at ScienceDirect

Web Semantics: Science, Services and Agents on the World Wide Web

journal homepage: www.elsevier.com/locate/websem

# Managing uncertainty and vagueness in description logics for the Semantic Web

Thomas Lukasiewicz [a,1], Umberto Straccia [b,*]

[a] *Computing Laboratory, University of Oxford, Wolfson Building, Parks Road, Oxford OX1 3QD, UK*
[b] *ISTI-CNR, Via G. Moruzzi 1, I-56124 Pisa, Italy*

ARTICLE INFO

*Article history:*
Received 25 October 2006
Received in revised form
22 November 2007
Accepted 20 April 2008
Available online 9 June 2008

*Keywords:*
Description logics
Web ontology languages
Ontologies
Probabilistic uncertainty
Possibilistic uncertainty
Vagueness
Imprecision
Semantic Web

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 in the semantic web community 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 approaches in this context to managing probabilistic uncertainty, possibilistic uncertainty, and vagueness in expressive description logics for the Semantic Web.
© 2008 Elsevier B.V. All rights reserved.

## 1. Introduction

The *Semantic Web* [6,7,33,56] 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-understandable "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 Semantic Web consists of several hierarchical layers, where the *Ontology layer*, in form of the *OWL Web Ontology Language* [56,150] (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* [37] (see especially [116] for an introduction to ontologies including a detailed historical account) 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 [3] with an RDF syntax [56]. More specifically, ontology entailment in OWL Lite and OWL DL reduces to knowledge base (un)satisfiability in the expressive description logics $\mathcal{SHIF}(\mathbf{D})$ and $\mathcal{SHOIN}(\mathbf{D})$ [55,57], 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.

* Corresponding author.
*E-mail addresses:* thomas.lukasiewicz@comlab.ox.ac.uk, lukasiewicz@kr.tuwien.ac.at (T. Lukasiewicz), straccia@isti.cnr.it (U. Straccia).
[1] Alternative address: Institut für Informationssysteme, Technische Universität Wien, Favoritenstraße 9-11, A-1040 Wien, Austria.

1570-8268/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.websem.2008.04.001

However, classical ontology languages and description logics are less suitable in all those domains where the information to be represented comes along with (*quantitative*) *uncertainty* and/or *vagueness* (or *imprecision*). For example, uncertain information may be of the form "John is a teacher with the degree of certainty 0.3 and a student with the degree of certainty 0.7" (roughly, John is either a teacher or a student, but more likely a student), while vague information may be of the form "John is tall with the degree of truth 0.9" (roughly, John is quite tall); see Section 2. Formalisms for dealing with uncertainty and vagueness have started to play an important role in research related to the Web and the Semantic Web. For example, the order in which Google returns the answers to a web search query is computed by using probabilistic techniques. Furthermore, formalisms for dealing with uncertainty and vagueness in ontologies have been successfully applied in ontology matching, data integration, and information retrieval. Vagueness and imprecision also abound in multimedia information processing and retrieval. In addition, handling vagueness is an important aspect of natural language interfaces to the Web. There exists a W3C Incubator Group on *Uncertainty Reasoning for the World Wide Web*, and an important recent forum for approaches to uncertainty in the Semantic Web is the annual *Workshop on Uncertainty Reasoning for the Semantic Web (URSW)* at the *International Semantic Web Conference (ISWC)*.

The rising popularity of description logics and their use, and the need to deal with uncertainty and vagueness, both especially in the Semantic Web, is increasingly attracting the attention of many researchers and practitioners towards description logics able to cope with uncertainty and vagueness. 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 vague/fuzzy theory. A clarifying paper is [26]. 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 [26]. In terms of binary truth values, a bottle is viewed as full or empty. But if one accounts for the quantity of liquid in the bottle, one may, e.g. say that the bottle is "half-full". Under this way of speaking, "full" becomes a fuzzy predicate [155] 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 that only one of the two situations is the true one). Saying 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* (for example, "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 about, e.g. a probability distribution or a possibility distribution over the worlds. For example, 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.

As for the main differences between probability and possibility theory, the probability of an event is the sum of the probabilities of all worlds that satisfy this event, whereas the possibility of an event is the maximum of the possibilities of all worlds that satisfy the event. Intuitively, the probability of an event aggregates the probabilities of all worlds that satisfy this event, whereas the possibility of an event is simply the possibility of the "most optimistic" world that satisfies the event. Hence, although both probability and possibility theory allow for quantifying degrees of uncertainty, they are conceptually quite different from each other. That is, probability and possibility theory represent different facets of uncertainty.

On the other hand, under *vagueness/fuzziness theory* fall all those approaches in which statements (for example, "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, since we are unable to establish whether a statement is completely true or false due to the involvement of vague concepts, such as "ripe", which only have an *imprecise* definition. For example, we cannot exactly say whether a tomato is ripe or not, but rather can only say that the tomato is ripe to some degree. Usually, such statements involve so-called *vague/fuzzy predicates* [155].

Note that all vague/fuzzy statements are truth-functional, that is, the degree of truth of every statement can be calculated from the degrees of truth of its constituents, while uncertain statements cannot be a function of the uncertainties of their constituents [25]. More concretely, in probability theory, only the negation is truth-functional (see Eq. (1)), while in possibility theory, only the disjunction respectively conjunction is truth-functional in possibilities respectively necessities of events (see Eq. (2)). Furthermore, fuzzy logics are based on truly many-valued logical operators, while uncertainty logics are defined on top of standard binary logical operators.

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 Boole in 1854 [11]. 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 [100], Fagin et al. [32], Dubois and Prade et al. [23,28,2,27], Frisch and Haddawy [34], and Lukasiewicz [83,84,86]; see also the survey on sentential probability logic by Hailperin [41]). Recently, nonmonotonic generalizations of probabilistic logic have been developed and explored; see especially [88] for an overview. In this section, for illustrative purposes, we recall only the simple probabilistic logic described in [100].

We first define probabilistic formulas and probabilistic knowledge bases. We assume a set of *basic events* $\Phi = \{p_1, \ldots, p_n\}$ with $n \geq 1$. 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 $\phi$ and $\psi$ 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 $\phi \geq l$, where $\phi$ is an event, and $l$ is a real number from the unit interval [0, 1]. Informally, $\phi \geq l$ says that $\phi$ is true with a probability of at least $l$. For example, *rain_tomorrow* $\geq 0.7$ may express that it will rain tomorrow with a probability of at least 0.7. Notice also that $\neg\phi \geq 1 - u$ encodes that $\phi$ is true with a probability of at most $u$. A *probabilistic knowledge base* $\mathcal{K}$ is a finite set of probabilistic formulas.

Next, we define worlds and probabilistic interpretations. A *world* $I$ associates with every basic event in $\Phi$ 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 $\Phi$. A world $I$ *satisfies* an event $\phi$, or $I$ is a *model* of $\phi$, 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 $Pr : \mathcal{I}_\Phi \to [0, 1]$ 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, that is, the probability function $Pr$ encodes our "uncertainty" about which world is the right one. The *probability* of an event $\phi$ 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 $\phi$ and $\psi$, the following relationships hold*:

$$
\begin{aligned}
&Pr(\phi \wedge \psi) = Pr(\phi) + Pr(\psi) - Pr(\phi \vee \psi);\\
&Pr(\phi \wedge \psi) \leq \min(Pr(\phi), Pr(\psi));\\
&Pr(\phi \wedge \psi) \geq \max(0, Pr(\phi) + Pr(\psi) - 1);\\
&Pr(\phi \vee \psi) = Pr(\phi) + Pr(\psi) - Pr(\phi \wedge \psi);\\
&Pr(\phi \vee \psi) \leq \min(1, Pr(\phi) + Pr(\psi));\\
&Pr(\phi \vee \psi) \geq \max(Pr(\phi), Pr(\psi));\\
&Pr(\neg\phi) = 1 - Pr(\phi);\\
&Pr(\bot) = 0;\\
&Pr(\top) = 1.
\end{aligned} \tag{1}
$$

A probabilistic interpretation *Pr satisfies* a probabilistic formula $\phi \geq l$, or *Pr* is a *model* of $\phi \geq l$, denoted $Pr \models \phi \geq l$, iff $Pr(\phi) \geq l$. 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 $\phi \geq l$ 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 $l = \sup\{r \mid \mathcal{K} \models \phi \geq r\}$.

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. [63] and Hansen et al. [46].

### 2.2. Possibilistic logic

We next recall possibilistic logic; see especially [21]. The main syntactic and semantic differences to probabilistic logic can be summarized as follows. Syntactically, rather than using probabilistic formulas to constrain the probabilities of propositional events, we now use possibilistic formulas to constrain the necessities and possibilities of propositional events. Semantically, rather than having probability distributions on worlds, each of which associates with every event a unique probability, we now have possibility distributions on worlds, each of which associates with every event a unique possibility and a unique necessity. Differently from the probability of an event, which is the sum of the probabilities of all worlds that satisfy that event, the possibility of an event is the maximum of the possibilities of all worlds that satisfy the event. As a consequence, probabilities and possibilities of events behave quite differently from each other (see Eqs. (1) and (2)). These fundamental semantic differences between probabilities and possibilities can also be used as the main criteria for using either probabilistic logic or possibilistic logic in a given application involving uncertainty. In addition, possibilistic logic may especially be used for encoding user preferences, since possibility measures can actually be viewed as rankings (on worlds or also objects) along an ordinal scale.

The semantic differences between probabilities and possibilities are also reflected in the computational properties of possibilistic and probabilistic logic, since reasoning in probabilistic logic generally requires to solve linear optimization problems, while reasoning in possibilistic logic does not, and thus can generally be done with less computational effort. Note that although possibility measures can be viewed as sets of upper probability measures [24], and possibility and probability measures can be translated into each other [20], no translations are known between possibilistic and probabilistic knowledge bases as described here.

We first define possibilistic formulas and knowledge bases. *Possibilistic formulas* have the form $\mathsf{P}\,\phi \geq l$ or $\mathsf{N}\,\phi \geq l$, where $\phi$ is an event, and $l$ is a real number from [0, 1]. Informally, such formulas encode to what extent $\phi$ is *possibly* respectively *necessarily* true. For example, $\mathsf{P}$ *rain_tomorrow* $\geq 0.7$ encodes that it will rain tomorrow is possible to degree 0.7, while $\mathsf{N}$ *father* $\to$ *man* $\geq 1$ 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 $\pi : \mathcal{I}_\Phi \to [0, 1]$. 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 $\pi$ 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 $\phi$ in a possibilistic interpretation $\pi$, 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 $\phi$ is evaluated in the most possible world where $\phi$ is true. The dual notion to the possibility of an event $\phi$ is the *necessity* of $\phi$, denoted $Nec(\phi)$, which is defined by $Nec(\phi) = 1 - Poss(\neg\phi)$. It reflects the lack of possibility of $\neg\phi$, that is, $Nec(\phi)$ evaluates to what extent $\phi$ is certainly true. The following theorem follows immediately from the above definitions.

**Theorem 2.2.** *For all possibilistic interpretations $\pi$ and events $\phi$ and $\psi$, the following relationships hold*:

$$
\begin{aligned}
&Poss(\phi \wedge \psi) \leq \min(Poss(\phi), Poss(\psi));\\
&Poss(\phi \vee \psi) = \max(Poss(\phi), Poss(\psi));\\
&Poss(\neg\phi) = 1 - Nec(\phi);\\
&Poss(\bot) = 0;\\
&Poss(\top) = 1\ (\textit{in the normalized case});\\
&Nec(\phi \wedge \psi) = \min(Nec(\phi), Nec(\psi));\\
&Nec(\phi \vee \psi) \geq \max(Nec(\phi), Nec(\psi));\\
&Nec(\neg\phi) = 1 - Poss(\phi);\\
&Nec(\bot) = 0\ (\textit{in the normalized case});\\
&Nec(\top) = 1.
\end{aligned} \tag{2}
$$

A possibilistic interpretation $\pi$ *satisfies* a possibilistic formula $\mathsf{P}\,\phi \geq l$ (respectively, $\mathsf{N}\,\phi \geq l$), or $\pi$ is a *model* of $\mathsf{P}\,\phi \geq l$ (respectively, $\mathsf{N}\,\phi \geq l$), denoted $\pi \models \mathsf{P}\,\phi \geq l$ (respectively, $\pi \models \mathsf{N}\,\phi \geq l$), iff $Poss(\phi) \geq l$ (respectively, $Nec(\phi) \geq l$). The notions of satisfiability, logical consequence, and tight logical consequence for possibilistic knowledge bases are then defined as usual (in the same way as in the probabilistic case). We refer the reader to [21,53] for algorithms for possibilistic logic.

**Table 1**
Properties for t-norms and s-norms

| Axiom name | T-norm | S-norm |
|---|---|---|
| Tautology/contradiction | $a \otimes 0 = 0$ | $a \oplus 1 = 1$ |
| Identity | $a \otimes 1 = a$ | $a \oplus 0 = a$ |
| Commutativity | $a \otimes b = b \otimes a$ | $a \oplus b = b \oplus a$ |
| Associativity | $(a \otimes b) \otimes c = a \otimes (b \otimes c)$ | $(a \oplus b) \oplus c = a \oplus (b \oplus c)$ |
| Monotonicity | if $b \leq c$, then $a \otimes b \leq a \otimes c$ | if $b \leq c$, then $a \oplus b \leq a \oplus c$ |

**Table 2**
Properties for implication and negation functions

| Axiom name | Implication function | Negation function |
|---|---|---|
| Tautology/contradiction | $0 \rhd b = 1,\ a \rhd 1 = 1,\ 1 \rhd 0 = 0$ | $\ominus 0 = 1, \ominus 1 = 0$ |
| Antitonicity | if $a \leq b$, then $a \rhd c \geq b \rhd c$ | if $a \leq b$, then $\ominus a \geq \ominus b$ |
| Monotonicity | if $b \leq c$, then $a \rhd b \leq a \rhd c$ | |

### 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 example, 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” $\phi$ 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 it refers to. This degree of fit is called *degree of truth* of the proposition $\phi$ 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 $n \geq 1$. In the sequel, we assume $\mathcal{S} = [0, 1]$.

In the many-valued logic that we consider here, *many-valued formulas* have the form $\phi \geq l$ or $\phi \leq u$, where $l, u \in [0, 1]$ [40,42], which encode that the degree of truth of $\phi$ is *at least* $l$ respectively *at most* $u$. For example, *ripe_tomato* $\geq 0.9$ says that we have a rather ripe tomato (the degree of truth of *ripe_tomato* is at least 0.9).

Semantically, a *many-valued interpretation* $\mathcal{I}$ maps each basic proposition $p_i$ into $[0, 1]$ and is then extended inductively to all propositions as follows:

$$\begin{aligned}&\mathcal{I}(\phi \wedge \psi) = \mathcal{I}(\phi) \otimes \mathcal{I}(\psi);\\&\mathcal{I}(\phi \vee \psi) = \mathcal{I}(\phi) \oplus \mathcal{I}(\psi);\\&\mathcal{I}(\phi \rightarrow \psi) = \mathcal{I}(\phi) \rhd \mathcal{I}(\psi);\\&\mathcal{I}(\neg \phi) = \ominus \mathcal{I}(\phi),\end{aligned} \tag{3}$$

where $\otimes$, $\oplus$, $\rhd$, and $\ominus$ are so-called *combination functions*, namely, *triangular norms* (or *t-norms*), *triangular co-norms* (or *s-norms*), *implication functions*, and *negation functions*, respectively, which extend the 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. An important aspect of such functions is that they satisfy some properties that one expects to hold for the connectives; see Tables 1 and 2. Note that in Table 1, the two properties Tautology and Contradiction follow from Identity, Commutativity, and Monotonicity.

**Table 3**
Combination functions of various fuzzy logics

| | Łukasiewicz logic | Gödel logic | Product logic | Zadeh logic |
|---|---|---|---|---|
| $a \otimes b$ | $\max(a+b-1, 0)$ | $\min(a, b)$ | $a \cdot b$ | $\min(a, b)$ |
| $a \oplus b$ | $\min(a+b, 1)$ | $\max(a, b)$ | $a+b-a \cdot b$ | $\max(a, b)$ |
| $a \rhd b$ | $\min(1-a+b, 1)$ | $\begin{cases}1 & \text{if } a \leq b\\ b & \text{otherwise}\end{cases}$ | $\min(1, b/a)$ | $\max(1-a, b)$ |
| $\ominus a$ | $1-a$ | $\begin{cases}1 & \text{if } a = 0\\ 0 & \text{otherwise}\end{cases}$ | $\begin{cases}1 & \text{if } a = 0\\ 0 & \text{otherwise}\end{cases}$ | $1-a$ |

**Table 4**
Some additional properties of combination functions of various fuzzy logics

| Property | Łukasiewicz logic | Gödel logic | Product logic | Zadeh logic |
|---|---|---|---|---|
| $x \otimes \ominus x = 0$ | + | + | + | – |
| $x \oplus \ominus x = 1$ | + | – | – | – |
| $x \otimes x = x$ | – | + | – | + |
| $x \oplus x = x$ | – | + | – | + |
| $\ominus \ominus x = x$ | + | – | – | + |
| $x \rhd y = \ominus x \oplus y$ | + | – | – | + |
| $\ominus(x \rhd y) = x \otimes \ominus y$ | + | – | – | + |
| $\ominus(x \otimes y) = \ominus x \oplus \ominus y$ | + | + | + | + |
| $\ominus(x \oplus y) = \ominus x \otimes \ominus y$ | + | + | + | + |

Usually, the implication function $\rhd$ is defined as *r-implication*, that is, $a \rhd b = \sup\{c | a \otimes c \leq b\}$.

Some t-norms, s-norms, implication functions, and negation functions of various fuzzy logics are shown in Table 3[42]. 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. $\min(x, y) = x \wedge (x \rightarrow y)$ and $\max(x, y) = (x \rightarrow y) \rightarrow y$. Some salient properties of these logics are shown in Table 4. For more properties, see especially [42,102].

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

Note that implication functions and t-norms are also used to define the degree of subsumption between fuzzy sets and the composition of two (binary) fuzzy relations. A *fuzzy set* $R$ over a countable crisp set $X$ is a function $R : X \rightarrow [0, 1]$. The *degree of subsumption* between two fuzzy sets $A$ and $B$, denoted $A \sqsubseteq B$, is defined as $\inf_{x \in X} A(x) \rhd B(x)$, where $\rhd$ is an implication function. Note that if $A(x) \leq B(x)$, for all $x \in [0, 1]$, then $A \sqsubseteq B$ evaluates to 1. Of course, $A \sqsubseteq B$ 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 $R : X \times Y \rightarrow [0, 1]$. The *inverse* of $R$ is the function $R^{-1} : Y \times X \rightarrow [0, 1]$ 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 $R_1 : X \times Y \rightarrow [0, 1]$ and $R_2 : Y \times Z \rightarrow [0, 1]$ is defined as $(R_1 \circ R_2)(x, z) = \sup_{y \in Y} R_1(x, y) \otimes R_2(y, z)$. A fuzzy relation $R$ is *transitive* iff $R(x, z) \geq (R \circ R)(x, z)$.

A many-valued interpretation $\mathcal{I}$ *satisfies* a many-valued formula $\phi \geq l$ (respectively, $\phi \leq u$) or $\mathcal{I}$ is a *model* of $\phi \geq l$ (respectively, $\phi \leq u$), denoted $\mathcal{I} \models \phi \geq l$ (respectively, $\mathcal{I} \models \phi \leq u$), iff $\mathcal{I}(\phi) \geq l$ (respectively, $\mathcal{I}(\phi) \leq u$). The notions of satisfiability, logical consequence, and tight logical consequence for many-valued knowledge bases are then defined in the standard way (in the same way as in the probabilistic case). We refer the reader to [39,40,42] for algorithms for many-valued logics.

## 3. Classical description logics

In this section, we recall the expressive description logic $\mathcal{SHOIN}(\mathbf{D})$ [57], which stands behind the web ontology languages OWL DL [55,56]. The purpose of this section is to make the paper self-contained. It also 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 expressive description logic $\mathcal{SHOIN}(\mathbf{D})$ is a generalization of $\mathcal{SHOIN}$ by datatypes, such as strings and integers, using *concrete domains* [4,93,92]

The elementary ingredients are as follows. We assume a set of *data values*, a set of *elementary datatypes*, and a set of *datatype predicates*, where each datatype predicate has a predefined arity $n \geq 1$. 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, $\geq_{20}$ may be a unary predicate denoting the set of integers greater or equal to 20, and thus $Person \sqcap \exists age.\geq_{20}$ may denote a person whose age is at least 20. Let $\mathbf{A}$, $\mathbf{R}_A$, $\mathbf{R}_D$, and $\mathbf{I}$ be pairwise disjoint sets of *atomic concepts*, *abstract roles*, *datatype 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 datatype role $T \in \mathbf{R}_D$ (note that datatype roles do not have inverses). We use $\mathbf{R}_A^-$ to denote the set of all inverses of abstract roles in $\mathbf{R}_A$.

An *RBox* $\mathcal{R}$ consists of a finite set of *transitivity axioms* $\mathrm{Trans}(R)$, where $R \in \mathbf{R}_A$, and *role inclusion axioms* $R \sqsubseteq S$, where either $R, S \in \mathbf{R}_A \cup \mathbf{R}_A^-$ or $R, S \in \mathbf{R}_D$.

We next define the notion of a simple abstract role. For abstract roles $R \in \mathbf{R}_A$, we define $\mathrm{Inv}(R) = R^-$ and $\mathrm{Inv}(R^-) = R$. Let $\sqsubseteq_{\mathcal{R}}^{\star}$ denote the reflexive and transitive closure of $\sqsubseteq$ on $\bigcup\{\{R \sqsubseteq S, \mathrm{Inv}(R) \sqsubseteq \mathrm{Inv}(S)\} \mid R \sqsubseteq S \in \mathcal{R},\ R, S \in \mathbf{R}_A \cup \mathbf{R}_A^-\}$. An abstract role $S$ is *simple* relative to $\mathcal{R}$ iff for each abstract role $R$ such that $R \sqsubseteq_{\mathcal{R}}^{\star} S$, it holds that (i) $\mathrm{Trans}(R) \notin \mathcal{R}$ and (ii) $\mathrm{Trans}(\mathrm{Inv}(R)) \notin \mathcal{R}$. Informally, an abstract role $S$ is simple iff it is neither transitive nor has transitive subroles.

*Concepts* are defined by induction as follows. Each $A \in \mathbf{A}$ is a concept, $\bot$ and $\top$ are concepts, and if $a_1, \ldots, a_n \in \mathbf{I}$, then $\{a_1, \ldots, a_n\}$ is a concept (called *oneOf*). If $C$, $C_1$, $C_2$ are concepts and $R \in \mathbf{R}_A \cup \mathbf{R}_A^-$, then $(C_1 \sqcap C_2)$, $(C_1 \sqcup C_2)$, and $\neg C$ are concepts (called *conjunction*, *disjunction*, and *negation*, respectively), as well as $\exists R.C$, $\forall R.C$, $\geq n\,R$, and $\leq n\,R$ (called *exists*, *value*, *atleast*, and *atmost restriction*, respectively) for an integer $n \geq 0$. If $D$ is an $n$-ary datatype predicate and $T, T_1, \ldots, T_n \in \mathbf{R}_D$, then $\exists T_1, \ldots, T_n.D$, $\forall T_1, \ldots, T_n.D$, $\geq n\,T$, and $\leq n\,T$ are concepts (called *datatype exists*, *value*, *atleast*, and *atmost restriction*, respectively) for an integer $n \geq 0$. For example, we may write the concept

$$Flower \sqcap \exists hasPetalWidth.\geq_{20\,\mathrm{mm}} \sqcap \exists hasPetalWidth.\leq_{40\,\mathrm{mm}} \sqcap \exists hasColor.Red$$

to denote the set of flowers having petal's dimension within 20 mm and 40 mm (where we assume that every flower has exactly one associated petal width) whose color is red. Here, $\geq_{20\,\mathrm{mm}}$ and $\leq_{40\,\mathrm{mm}}$ are datatype predicates. We eliminate (and add) parentheses as usual, and we often use $= 1R$ to abbreviate $(\geq 1R) \sqcap (\leq 1R)$.

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

An *ABox* $\mathcal{A}$ is a finite set of *concept membership axioms* $a : C$, *role membership axioms* $(a, b) : R$ (respectively, $(a, v) : T$), *equality axioms* $a = b$, and *inequality axioms* $a \neq b$, where $C$ is a concept, $R \in \mathbf{R}_A$, $T \in \mathbf{R}_D$, $a, b \in \mathbf{I}$, and $v$ is a data value. 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 $T \in \mathbf{R}_D$ 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 (where $R^{\mathcal{I}}(x) = \{y \mid (x, y) \in R^{\mathcal{I}}\}$, and $\#X$ denotes the cardinality of the set $X$):

$$
\begin{aligned}
&(R^-)^{\mathcal{I}} = \{(y, x) \mid (x, y) \in R^{\mathcal{I}}\};\\
&\top^{\mathcal{I}} = \Delta^{\mathcal{I}};\\
&\bot^{\mathcal{I}} = \emptyset;\\
&\{a_1, \ldots, a_n\}^{\mathcal{I}} = \{a_1^{\mathcal{I}}, \ldots, a_n^{\mathcal{I}}\};\\
&(C_1 \sqcap C_2)^{\mathcal{I}} = C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}};\\
&(C_1 \sqcup C_2)^{\mathcal{I}} = C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}};\\
&(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}};\\
&(\forall R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(x) \subseteq C^{\mathcal{I}}\};\\
&(\exists R.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(x) \cap C^{\mathcal{I}} \neq \emptyset\};\\
&(\geq nR)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \#R^{\mathcal{I}}(x) \geq n\};\\
&(\leq nR)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \#R^{\mathcal{I}}(x) \leq n\};\\
&(\forall T_1, \ldots, T_n.d)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid T_1^{\mathcal{I}}(x) \times \ldots \times T_n^{\mathcal{I}}(x) \subseteq d^{\mathcal{I}}\};\\
&(\exists T_1, \ldots, T_n.d)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid T_1^{\mathcal{I}}(x) \times \ldots \times T_n^{\mathcal{I}}(x) \cap d^{\mathcal{I}} \neq \emptyset\}.
\end{aligned}
$$

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: (1) $\mathcal{I} \models \mathrm{Trans(R)}$ iff $R^{\mathcal{I}}$ is transitive, (2) $\mathcal{I} \models R \sqsubseteq S$ iff $R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$, (3) $\mathcal{I} \models C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, (4) $\mathcal{I} \models a : C$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$, (5) $\mathcal{I} \models (a, b) : R$ iff $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$, (6) $\mathcal{I} \models (a, v) : T$ iff $(a^{\mathcal{I}}, v^{\mathbf{D}}) \in T^{\mathcal{I}}$, (7) $\mathcal{I} \models a = b$ iff $a^{\mathcal{I}} = b^{\mathcal{I}}$, (8) $\mathcal{I} \models a \neq b$ iff $a^{\mathcal{I}} \neq b^{\mathcal{I}}$. We say $\mathcal{I}$ *satisfies* $E$, or $\mathcal{I}$ *is a model of* $E$, iff $\mathcal{I} \models E$. We say $\mathcal{I}$ *satisfies* a set of axioms $\mathcal{E}$, or $\mathcal{I}$ *is a model of* $\mathcal{E}$, denoted $\mathcal{I} \models \mathcal{E}$, iff $\mathcal{I} \models E$ for all $E \in \mathcal{E}$. An interpretation $\mathcal{I}$ *satisfies* a knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{R}, \mathcal{A})$, or $\mathcal{I}$ *is a model of* $\mathcal{K}$, denoted $\mathcal{I} \models \mathcal{K}$, iff $\mathcal{I}$ satisfies each component $\mathcal{T}$, $\mathcal{R}$, and $\mathcal{A}$. A knowledge base $\mathcal{K}$ is *satisfiable* iff it has a model $\mathcal{I}$. An axiom $E$ is a *logical consequence* of $\mathcal{K}$, denoted $\mathcal{K} \models E$, iff every model of $\mathcal{K}$ satisfies $E$. A concept $C$ is *satisfiable* relative to $\mathcal{K}$ iff $\mathcal{K}$ has a model $\mathcal{I}$ such that $C^{\mathcal{I}} \neq \emptyset$.

**Example 3.1** *(Car Example).* Consider the following excerpt of a simple ontology about cars. Let $\mathcal{R} = \emptyset$ and let the TBox $\mathcal{T}$ contain the following axioms (where *maker* and *topType* are abstract roles, while *passenger_capacity* and *max_speed* are datatype roles with the natural numbers $\mathbb{N}$ respectively kilometers per hour km/h as datatypes; the datatype predicate $\geq_{245\,\mathrm{km/h}}$ is true if the value is at least 245 km/h):

$$
\begin{aligned}
&(\geq 1\,maker) \sqsubseteq Car; && \top \sqsubseteq \forall maker.Maker;\\
&(\geq 1\,passenger\_capacity) \sqsubseteq Car; && \top \sqsubseteq \forall passenger\_capacity.\mathbb{N};\\
&(\geq 1 max\_speed) \sqsubseteq Car; && \top \sqsubseteq \forall max\_speed.\mathrm{km/h};\\
&Car \sqsubseteq (= 1\,maker) \sqcap (= 1\,passenger\_capacity) \sqcap (= 1 max\_speed);\\
&Roadster \sqsubseteq Cabriolet \sqcap \exists passenger\_capacity.=_2;\\
&Cabriolet \sqsubseteq Car \sqcap \exists topType.SoftTop;\\
&SportsCar = Car \sqcap \exists max\_speed.\geq_{245\,\mathrm{km/h}}.
\end{aligned}
$$

Informally, the roles *maker*, *passenger_capacity*, and *max_speed* relate cars to a car maker, a natural number for its passenger capacity, and a value in kilometers per hour for its maximum speed, respectively. Furthermore, roadsters are cabriolets with the passenger capacity two, cabriolets are cars with a soft top, and sports cars are exactly cars with a maximum speed of at least 245 km/h.

The ABox $\mathcal{A}$ contains the following concept membership axioms:

$$mgb : Roadster \sqcap \exists maker.\{mg\} \sqcap \exists max\_speed.{\leq_{170\,\mathrm{km/h}}};$$
$$enzo : Car \sqcap \exists maker.\{ferrari\} \sqcap \exists max\_speed.{>_{350\,\mathrm{km/h}}};$$
$$tt : Car \sqcap \exists maker.\{audi\} \sqcap \exists max\_speed.{=_{243\,\mathrm{km/h}}}.$$

It is then not difficult to verify that some logical consequences of the above knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ are given as follows:

$$\mathcal{K} \models Roadster \sqsubseteq Car; \quad \mathcal{K} \models mg : Maker;$$
$$\mathcal{K} \models enzo : SportsCar; \quad \mathcal{K} \models tt : \neg SportsCar.$$

### 3.3. Main reasoning problems

The main reasoning problems in $\mathcal{SHOIN}(\mathbf{D})$ are deciding the logical consequence of concept inclusion axioms (CSub), concept membership axioms (CMem), and role membership axioms from knowledge bases (RMem), deciding the satisfiability of concepts relative to knowledge bases (CSat), and deciding the satisfiability of knowledge bases (KBSat). Note that (i) CSat and KBSat can be reduced to each other, (ii) CMem and RMem are special cases of CSub (in $\mathcal{SHOIN}(\mathbf{D})$), and (iii) CSat and CSub can be reduced to each other. The above problems are all decidable in $\mathcal{SHOIN}(\mathbf{D})$ if all number restrictions in $\mathcal{K} = (\mathcal{T}, \mathcal{R}, \mathcal{A})$ are restricted to simple abstract roles w.r.t. $\mathcal{R}$ [58]. Decision procedures are given in [143,57], and reasoning tools for $\mathcal{SHOIN}(\mathbf{D})$ are, e.g., FaCT++ [145,54] and Pellet [105].

## 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 been introduced in [90].

Note that any other classical description logic can be similarly extended by probabilistic uncertainty. In particular, closely related probabilistic generalizations of *DL-Lite* and 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) have been introduced in [90,36]. The syntax and semantics of such an extension can be defined in the same way as for P- $\mathcal{SHOIN}(\mathbf{D})$. Furthermore, if the chosen classical description logic allows for decidable knowledge base satisfiability, then also the main reasoning tasks in the probabilistic extension are all decidable. Note that to allow for probabilistic role membership axioms (encoding that "$R(a, b)$ (respectively, $U(a, v)$) holds with a probability between $l$ and $u$"), the extended classical description logic should have the oneOf (respectively, datatype oneOf) construct.

The syntax of the probabilistic description logic P- $\mathcal{SHOIN}(\mathbf{D})$ uses the notion of a conditional constraint from [84] 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 [85,87], which is a probabilistic generalization of the sophisticated notion of lexicographic entailment by Lehmann [72] in default reasoning from conditional knowledge bases. Due to this semantics, P- $\mathcal{SHOIN}(\mathbf{D})$ 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 [72].

The notion of probabilistic lexicographic entailment [85,87] is an entailment relation for reasoning from statistical knowledge and degrees of belief, which has very nice features [87]. In particular, it shows a similar behavior as reference-class reasoning in a number of uncontroversial examples.[1] But it also avoids many drawbacks of reference-class reasoning (which are pointed out in [5,87]): differently from reference-class reasoning, probabilistic lexicographic entailment can handle complex scenarios and even purely probabilistic subjective knowledge as input, and probabilistic lexicographic entailment draws conclusions in a global way from all the available knowledge as a whole. Furthermore, probabilistic lexicographic entailment also has very nice nonmonotonic properties, which are essentially inherited from Lehmann's lexicographic entailment [72]. 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). For example, under probabilistic lexicographic entailment, the default knowledge (1) "generally, cars do not have a red color" and (2) "generally, sports cars have a red color", and the probabilistic knowledge (3) "cars have four wheels with a probability of at least 0.9" imply that sports cars have four wheels with a probability of at least 0.9. That is, the property of having four wheels with a probability of at least 0.9 is inherited from cars down to sports cars, even though sports cars are exceptional cars relative to the property of having a red color. As for general nonmonotonic properties, probabilistic lexicographic entailment satisfies (probabilistic versions of) the rationality postulates by Kraus et al. [69], the property of rational monotonicity, and some irrelevance, conditioning, and inclusion properties. For example, as for the property of irrelevance, under probabilistic lexicographic entailment, the above sentence (3) implies that also red cars have four wheels with a probability of at least 0.9. That is, the property of having a red color is irrelevant to the property of having four wheels with a probability of at least 0.9. All these quite appealing features carry over to the probabilistic description logic P- $\mathcal{SHOIN}(\mathbf{D})$. See especially [87] for further details and background on 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 [84], 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

[1] Reference-class reasoning [110,70,71,106] is one of the most influential entailment relations for reasoning from statistical knowledge and degrees of belief. The main idea behind it is to equate the degrees of belief about a particular individual with the statistics of a *reference class*, which is informally defined as a set of individuals that contains the particular individual and about which we have some statistics. If there are several reference classes with conflicting statistics, then the narrowest one and its statistics are preferred. Even though reference-class reasoning has also been criticized in the literature, there are several uncontroversial examples, where it describes exactly the expected inference results.

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 $\phi$ and $\psi$ 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 $\phi$ and $\psi$ are c-concepts, and $l$ and $u$ are reals from $[0, 1]$. Informally, $(\psi|\phi)[l, u]$ encodes that the probability of $\psi$ given $\phi$ lies between $l$ and $u$.

We next define PTBoxes, PABoxes, and probabilistic knowledge bases as follows:

- A *PTBox* $PT = (T, P)$ consists of a classical (description logic) knowledge base $T$ and a finite set of conditional constraints $P$;
- 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 $o \in \mathbf{I}_P$.

Note that the meaning of a conditional constraint $(\psi|\phi)[l, u]$ depends on whether it belongs to $P$ or to $P_o$ for some probabilistic individual $o \in \mathbf{I}_P$:

- Each $(\psi|\phi)[l, u]$ in $P$ informally encodes that "generally, if an object belongs to $\phi$, then it belongs to $\psi$ with a probability in $[l, u]$". For example, $(\exists R.\{o\}|\phi)[l, u]$ in $P$, where $o \in \mathbf{I}_C$ and $R \in \mathbf{R}_A$, encodes that "generally, if an object belongs to $\phi$, then it is related to $o$ by $R$ with a probability in $[l, u]$".
- Each $(\psi|\phi)[l, u]$ in $P_o$, where $o \in \mathbf{I}_P$, informally encodes that "if $o$ belongs to $\phi$, then $o$ belongs to $\psi$ with a probability in $[l, u]$". For example, $(\exists R.\{o'\}|\phi)[l, u]$ 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 $\phi$, then $o$ is related to $o'$ by $R$ with a probability in $[l, u]$".

So, 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 $o$.

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 $P$).

Consequently, a conditional constraint $(\psi|\phi)[1, 1]$ in $P$ encodes "generally, if an object belongs to $\phi$, then it also belongs to $\psi$", while $(\psi|\phi)[1, 1]$ in $P_o$ encodes "if $o$ belongs to $\phi$, then $o$ also belongs to $\psi$". The latter is equivalent to the implication $o : \phi \Rightarrow o : \psi$, while the former is in general not equivalent to $\phi \sqsubseteq \psi$.

**Example 4.1** *(Car Example continued).* 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) $(\neg\exists\, HasColor.\{red\}|Car)[1, 1]$,
(2) $(\exists\, HasColor.\{red\}|SportsCar)[1, 1]$,
(3) $(HasFourWheels|Car)[0.9, 1]$.

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 $P_{John's\,car}$:

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

### 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 and lexicographic entailment, respectively, in probabilistic default reasoning [85,87].

#### 4.2.1. Preliminaries

We now define (possible) objects and probabilistic interpretations, which are certain sets of basic c-concepts respectively 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 $\{i : \phi \mid \phi \in o\} \cup \{i : \neg\phi \mid \phi \in \mathcal{C} \setminus o\}$ 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 (respectively, does not belong) to every c-concept $\phi \in o$ (respectively, $\phi \in \mathcal{C} \setminus o$). 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 $T \cup \{i : \phi \mid \phi \in o\} \cup \{i : \neg\phi \mid \phi \in \mathcal{C} \setminus o\}$ 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 $\phi$, 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 $Pr : \mathcal{O}_\mathcal{C} \to [0, 1]$ 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 $\phi$ in a probabilistic interpretation $Pr$ denoted $Pr(\phi)$, is the sum of all $Pr(o)$ such that $o \models \phi$. For c-concepts $\phi$ and $\psi$ 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 [85,87].

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 $F \in P \setminus (P_0 \cup \cdots \cup P_{i-1})$ that are tolerated under *T* by $P \setminus (P_0 \cup \cdots \cup P_{i-1})$. Informally, 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 $(\neg\exists\, HasColor.\{red\}|Car)[1, 1]$ and $(\exists\, HasColor.\{red\}|SportsCar)[1, 1]$ when reasoning about sports cars is naturally resolved by preferring the latter to the former. We call the above (unique) 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 (i) $PT = (T, P)$ is consistent and (ii) $T \cup P_o$ is satisfiable for every probabilistic individual $o \in \mathbf{I}_P$. Informally, (ii) says that the strict knowledge in *T* must be compatible with the strict degrees of belief in $P_o$, for every probabilistic individual *o*. Observe that (i) involves *T* and *P*, while (ii) involves *T* and $P_o$, for every probabilistic individual *o*. This separate treatment of *P* and the $P_o$'s is due to the fact that *P* represents probabilistic terminological knowledge, while each $P_o$ represents probabilistic assertional knowledge (about *o*).

**Example 4.2** *(Car Example continued).* 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 $o \in \mathbf{I}_P = \{John's\ car\}$. Observe that the z-partition of $(T, P)$ is given by $(P_0, P_1)$, where $P_0 = \{(\psi \mid \phi)[l, u] \in P \mid \phi = Car\}$ and $P_1 = \{(\psi|\phi)[l, u] \in P \mid \phi = SportsCar\}$.

#### 4.2.3. Lexicographic entailment

The notion of lexicographic entailment for probabilistic knowledge bases is based on lexicographic entailment in probabilistic default reasoning [85,87]. 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 (unique) z-partition $(P_0, \ldots, P_k)$ of $(T, P)$ (see Section 4.2.2) 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 $i < j \leq k$. 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 $\mathcal{F} \Vdash\!\sim^{lex} (\psi|\phi)[l, u]$ 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 $\mathcal{F} \Vdash\!\sim^{lex}_{tight} (\psi|\phi)[l, u]$ under *PT*, iff *l* (respectively, *u*) is the infimum (respectively, 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 $\mathcal{F} \Vdash\!\sim^{lex} (\psi|\phi)[l, u]$ and $\mathcal{F} \Vdash\!\sim^{lex}_{tight} (\psi|\phi)[1, 0]$ 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 $PT \Vdash\!\sim^{lex} F$, iff $\emptyset \Vdash\!\sim^{lex} F$ under *PT*. We say *F* is a *tight lex-consequence* of *PT*, denoted $PT \Vdash\!\sim^{lex}_{tight} F$, iff $\emptyset \Vdash\!\sim^{lex}_{tight} F$ under *PT*. A conditional constraint *F* for a probabilistic individual $o \in \mathbf{I}_P$ is a *lex-consequence* of $\mathcal{K}$, denoted $\mathcal{K} \Vdash\!\sim^{lex} F$, iff $P_o \Vdash\!\sim^{lex} F$ under $PT = (T, P)$. We say *F* is a *tight lex-consequence* of $\mathcal{K}$, denoted $\mathcal{K} \Vdash\!\sim^{lex}_{tight} F$, iff $P_o \Vdash\!\sim^{lex}_{tight} F$ under $PT = (T, P)$.

**Example 4.3** *(Car Example continued).* 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 $PT = (T, P)$:

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

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:

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

### 4.3. Main reasoning problems

The main reasoning problems in P- $\mathcal{SHOIN}(\mathbf{D})$ are summarized by the following decision and computation problems (where every lower and upper bound in the PTBox $PT = (T, P)$, the probabilistic knowledge base $\mathcal{K} = (T, P, (P_o)_{o \in \mathbf{I}_P})$, and the set of conditional constraints $\mathcal{F}$ is rational):

PTBox Consistency (PTCon): Given a PTBox $PT = (T, P)$, decide whether $PT$ is consistent.

Probabilistic Knowledge Base Consistency (PKBCon): Given a probabilistic knowledge base $\mathcal{K} = (T, P, (P_o)_{o \in \mathbf{I}_P})$, decide whether $\mathcal{K}$ is consistent.

Tight Lexicographic Entailment (TLexEnt): Given a PTBox $PT = (T, P)$, a finite set of conditional constraints $\mathcal{F}$, and two c-concepts $\phi$ and $\psi$, compute the rational numbers $l, u \in [0, 1]$ such that $\mathcal{F} \Vdash\!\sim^{lex}_{tight} (\psi|\phi)[l, u]$ under $PT$.

Some important special cases of TLexEnt are given as follows: (PCSub) given a consistent PTBox $PT$ and two c-concepts $\phi$ and $\psi$, compute the rational numbers $l, u \in [0, 1]$ such that $PT \Vdash\!\sim^{lex}_{tight} (\psi|\phi)[l, u]$; (PCRSub) given a consistent PTBox $PT$, a c-concept $\phi$, a classical individual $o \in \mathbf{I}_C$, and an abstract role $R \in \mathbf{R}_A$, compute the rational numbers $l, u \in [0, 1]$ such that $PT \Vdash\!\sim^{lex}_{tight} (\exists R.\{o\}|\phi)[l, u]$; (PCMem) given a consistent probabilistic knowledge base $\mathcal{K}$, a probabilistic individual $o \in \mathbf{I}_P$, and a c-concept $\psi$, compute $l, u \in [0, 1]$ such that $\mathcal{K} \Vdash\!\sim^{lex}_{tight} (\psi|\top)[l, u]$ for $o$; and (PRMem) given a consistent probabilistic knowledge base $\mathcal{K}$, a classical individual $o' \in \mathbf{I}_C$, a probabilistic individual $o \in \mathbf{I}_P$, and an abstract role $R \in \mathbf{R}_A$, compute $l, u \in [0, 1]$ such that $\mathcal{K} \Vdash\!\sim^{lex}_{tight} (\exists R.\{o'\}|\top)[l, u]$ for $o$.

Another important decision problem in P-$\mathcal{SHOIN}(\mathbf{D})$ is Probabilistic Concept Satisfiability (PCSat): Given a consistent PTBox $PT$ and a c-concept $\phi$, decide whether $PT \not\Vdash\!\sim^{lex} (\phi|\top)[0, 0]$. This problem is reducible to CSat (see Section 3.3), since $(T, P) \not\Vdash\!\sim^{lex} (\phi|\top)[0, 0]$ iff $T \not\Vdash\!\sim \phi \sqsubseteq \bot$.

There exists an algorithm for deciding whether a PTBox (respectively, 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. More specifically, one has to solve a sequence of solvability problems of systems of linear constraints, whose variables are computed by deciding classical knowledge base satisfiability in $\mathcal{SHOIN}(\mathbf{D})$ (see [90] for further details). This shows that the two consistency problems in P-$\mathcal{SHOIN}(\mathbf{D})$ are both decidable. Furthermore, there is a similar 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 (see [90]). Thus, also lexicographic entailment in P- $\mathcal{SHOIN}(\mathbf{D})$ is computable. As for the computational complexity, deciding the two consistency problems in P-$\mathcal{SHOIN}(\mathbf{D})$ is complete for the complexity class NEXP, while computing tight intervals under lexicographic entailment in P- $\mathcal{SHOIN}(\mathbf{D})$ belongs to $\mathrm{FP}^{\mathrm{NEXP}}$[90].

Although there is no implementation of P- $\mathcal{SHOIN}(\mathbf{D})$ to date, there are already implementations of its predecessor P- $\mathcal{SHOQ}(\mathbf{D})$ (see [99]) and of a probabilistic description logic based on probabilistic default reasoning as in [85,87] (see [66]).

### 4.4. Main applications

As pointed out in [14,15], there is a plethora of applications with an urgent need for handling probabilistic knowledge in ontologies, especially in areas like medicine, biology, defense, and astronomy. Furthermore, there are strong arguments for the critical need of dealing with probabilistic uncertainty in ontologies in the Semantic Web, some of which are briefly summarized as follows.

- In addition to being logically related, the concepts of an ontology are generally also probabilistically related. For example, two concepts either may be logically related via a subset or disjointness relationship, or they may show a certain degree of overlap. Probabilistic ontologies allow for quantifying these degrees of overlap, reasoning about them, and using them in semantic-web applications. In particular, probabilistic ontologies are successfully used in information retrieval for an increased recall [146,59] (see also below). The degrees of concept overlap may also be exploited in personalization and recommender systems.
- Rather than consisting of one standardized overall ontology, the Semantic Web will consist of a huge collection of different ontologies. Hence, in semantic-web applications such as automated reasoning and information retrieval, one has to align the concepts of different ontologies, which is called *ontology matching/mapping* [30]. In general, the concepts of two different ontologies do not match exactly, and we have to deal with degrees of concept overlap as above, which are determined by automatic or semi-automatic tools or experts. These degrees of concept overlap are then represented in probabilistic ontologies, which thus allows for inference about the degrees of overlap between other concepts and about probabilistic instance relationships [104,98] (see also Section 4.5.2).
- Like the current Web, the Semantic Web will necessarily contain ambiguous and controversial pieces of information in different web sources. This can be handled via probabilistic data integration by associating with every web source a probability describing its degree of reliability [148,45]. As resulting pieces of data, such a probabilistic data integration process necessarily produces probabilistic facts, that is, probabilistic knowledge at the instance level. Such probabilistic instance relationships can be encoded in probabilistic ontologies and there be enhanced by further classical and/or terminological probabilistic knowledge, which then allows for inference about other probabilistic instance relationships.

An important application for probabilistic ontologies (and thus probabilistic description logics and ontology languages) is especially information retrieval. In particular, Subrahmanian's group [146,59] explores 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 (which is the proportion of documents relevant to a search query in the collection of all returned documents) in information retrieval. In closely related work, Mantay et al. [95] propose a probabilistic least common subsumer operation, which is based on a probabilistic extension of the description logic $\mathcal{ALN}$. They show that applying this approach in information retrieval allows for reducing the amount of retrieved data and thus for avoiding information flood. Another closely related work by Holi and Hyvönen [48,49] 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. Finally, Weikum et al. [151] and Thomas and Sheth [142] describe the use of probabilistic ontologies in information retrieval from a more general perspective.

### 4.5. Other probabilistic ontology languages

To our knowledge, there are no other approaches to probabilistic description logics for the Semantic Web in the literature. Furthermore, although there are several previous approaches to probabilistic description logics without semantic web background, P-$\mathcal{SHOIN}(\mathbf{D})$ is the most expressive probabilistic description logic, both in terms of the generalized classical description logic and in terms of the supported forms of terminological and assertional probabilistic knowledge. That is, previous probabilistic description logics generalize less expressive classical description logics, and they only allow for some facets of the terminological and assertional probabilistic knowledge of this paper, but not for all of them at the same time. There are also several probabilistic extensions of web ontology languages in the literature. In this section, we give an overview of all these approaches.

#### 4.5.1. Probabilistic description logics

Other approaches to probabilistic description logics can be classified according to the generalized classical description logics, the supported forms of probabilistic knowledge, the underlying probabilistic semantics, and the reasoning techniques.

One of the earliest approaches to probabilistic description logics is due to Heinsohn [47], who presents a probabilistic extension of the description logic $\mathcal{ALC}$, which allows to represent terminological probabilistic knowledge about concepts and roles, and which is based on the notion of logical entailment in probabilistic logics, similar to [100,2,34,84]. Heinsohn [47], however, does not allow for assertional (classical or probabilistic) knowledge about concept and role instances. The main reasoning problems are deciding the consistency of probabilistic terminological knowledge bases and computing logically entailed tight probability intervals. Heinsohn proposes a sound and complete global reasoning technique based on classical reasoning in $\mathcal{ALC}$ and linear programming, as well as a sound but incomplete local reasoning technique based on the iterative application of local inference rules.

Another early approach to probabilistic description logics is due to Jaeger [61], who also proposes a probabilistic extension of the description logic $\mathcal{ALC}$, which allows for terminological probabilistic knowledge about concepts and roles, and assertional probabilistic knowledge about concept instances, but does not support assertional probabilistic knowledge about role instances (but he mentions a possible extension in this direction). The entailment of terminological probabilistic knowledge from terminological probabilistic knowledge is based on the notion of logical entailment in probabilistic logic, while the entailment of assertional probabilistic knowledge from terminological and assertional probabilistic knowledge is based on a cross-entropy minimization relative to terminological probabilistic knowledge. The main reasoning problems are terminological probabilistic consistency and inference, which are solved by linear programming, and assertional probabilistic consistency and inference, which are solved by an approximation algorithm.

The recent work by Dürig and Studer [29] presents a further probabilistic extension of $\mathcal{ALC}$, which is based on a model-theoretic semantics as in probabilistic logics, but which only allows for assertional probabilistic knowledge about concept and role instances, and not for terminological probabilistic knowledge. The paper also explores independence assumptions for assertional probabilistic knowledge. The main reasoning problem is deciding the consistency of assertional probabilistic knowledge, but neither an algorithm nor a decidability result is given.

Jaeger's recent work [62] 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. The semantics is different from the semantics of all the other probabilistic description logics in this paper, since it is based on probability distributions over the domain, and not on the more commonly used probability distributions over a set of possible worlds. The paper proposes a sound Gentzen-style sequent calculus for the logic, but it neither proves the completeness of this calculus nor decidability in general.

Koller et al.'s work [68] presents the probabilistic description logic P-Classic, which is a probabilistic generalization (of a variant) of the description logic Classic. Similar to Heinsohn's work [47], it allows for encoding terminological probabilistic knowledge about concepts, roles, and attributes (via so-called p-classes), but it does not support assertional (classical or probabilistic) knowledge about instances of concepts and roles. However, in contrast to [47], its probabilistic semantics is based on a reduction to Bayesian networks. The main reasoning problem is to determine the exact probabilities for conditionals between concept expressions in canonical form. This problem is solved by a reduction to inference in Bayesian networks. As an important feature of P-Classic, the above problem can be solved in polynomial time, when the underlying Bayesian network is a polytree. Note that a recent implementation of P-Classicis described in [65].

Closely related work by Yelland [153] proposes a probabilistic extension of a description logic close to $\mathcal{FL}$, whose probabilistic semantics is also based on a reduction to Bayesian networks, and it applies this approach to market analysis. The approach allows for encoding terminological probabilistic knowledge about concepts and roles, but it does not support assertional (classical or probabilistic) knowledge about instances of concepts and roles. Like in Koller et al.'s work [68], the main reasoning problem is to determine the exact probabilities for conditionals between concepts, which is solved by a reduction to inference in Bayesian networks.

#### 4.5.2. 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 [14], da Costa and Laskey [15], and da Costa et al. [16] suggest a probabilistic generalization of OWL, called PR-OWL, whose probabilistic semantics is based on multi-entity Bayesian networks (MEBNs). The latter are a Bayesian logic that combines first-order logic with Bayesian networks. Roughly speaking, PR-OWL represents knowledge as parameterized fragments of Bayesian networks. Hence, it can encode probability distributions on the interpretations of an associated first-order theory as well as repeated structure.

In [18,19], Ding et al. 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, and it also provides a method for incorporating available probability constraints when constructing the Bayesian network. The generated Bayesian network, which preserves the semantics of the original ontology and which is consistent with all the given probability constraints, supports ontology reasoning, both within and across ontologies, as Bayesian inferences. In [104,19], Ding et al. also describe an application of the BayesOWL approach in ontology mapping.

In closely related work, Mitra et al. [98] describe an implemented technique, called Omen, to enhancing existing ontology mappings by using a Bayesian network to represent the influences between potential concept mappings across ontologies. More concretely, OMEN is based on a simple ontology model similar to RDF Schema. It uses a set of meta-rules that capture the influence of the ontology structure and the semantics of ontology relations, and matches nodes that are neighbors of already matched nodes in the two ontologies.

Yang and Calmet [152] present an integration of the web ontology language OWL with Bayesian networks, called OntoBayes. The approach makes use of probability and dependency-annotated OWL to represent uncertain information in Bayesian networks. The work also describes an application in risk analysis for insurance and natural disaster management. Pool and Aikin [107] also provide a method for representing uncertainty in OWL ontologies, while Fukushige [35] proposes a basic framework for representing probabilistic relationships in RDF. Nottelmann and Fuhr [101] present two probabilistic extensions of variants of OWL Lite, along with a mapping to locally stratified probabilistic Datalog.

Another important work is due to Udrea et al. [147], who present a probabilistic generalization of RDF, which allows for representing terminological probabilistic knowledge about classes and assertional probabilistic knowledge about properties of individuals. They provide a technique for assertional probabilistic inference in acyclic probabilistic RDF theories, which is based on the notion of logical entailment in probabilistic logic, coupled with a local probabilistic semantics. They also provide a prototype implementation of their algorithms.

## 5. Possibilistic uncertainty and description logics

Similar to probabilistic extensions of description logics, possibilistic extensions of description logics have been developed by Hollunder [53] and Dubois et al. [22]. 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 $\mathsf{P}\,\alpha \geq l$ or $\mathsf{N}\,\alpha \geq l$, where $\alpha$ is a classical description logic axiom, and $l$ is a real number from $[0, 1]$. A *possibilistic RBox* (respectively, *TBox*, *ABox*) is a finite set of possibilistic axioms $\mathsf{P}\,\alpha \geq l$ or $\mathsf{N}\,\alpha \geq l$, where $\alpha$ is an RBox (respectively, 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 [53] illustrates possibilistic knowledge bases.

**Example 5.1** *(Car example continued).* 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":

$$\mathcal{T} = \{\mathsf{N}\,\exists owns.Porsche \sqsubseteq richPerson \sqcup carFanatic \geq 0.8,$$
$$\mathsf{P}\,richPerson \sqsubseteq golfer \geq 0.7\}.$$

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":

$$\mathcal{A} = \{\mathsf{N}\,(Tom, 911) : owns \geq 1, \mathsf{N}\,911 : Porsche \geq 1,$$
$$\mathsf{N}\,Tom : \neg carFanatic \geq 0.7\}.$$

### 5.2. Semantics

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

A possibilistic interpretation $\pi$ *satisfies* a possibilistic axiom $\mathsf{P}\,\alpha \geq l$ (respectively, $\mathsf{N}\,\alpha \geq l$), or $\pi$ is a *model* of $\mathsf{P}\,\alpha \geq l$ (respectively, $\mathsf{N}\,\alpha \geq l$), denoted $\pi \models \mathsf{P}\,\alpha \geq l$ (respectively, $\pi \models \mathsf{N}\,\alpha \geq l$), iff $Poss(\alpha) \geq l$ (respectively, $Nec(\alpha) \geq l$). The notion of satisfiability of possibilistic knowledge bases and the notions of logical and tight logical consequences of possibilistic axioms from possibilistic knowledge bases are then defined as usual [53,22].

**Example 5.2** *(Car example continued).* 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" [53], that is,

$$\mathcal{K} \models \mathsf{P}\,Tom : golfer \geq 0.7.$$

### 5.3. Main reasoning problems

The main reasoning problems related to possibilistic description logics are deciding whether a possibilistic knowledge base is satisfiable, deciding whether a possibilistic axiom is a logical consequence of a possibilistic knowledge base, and computing the tight lower and upper bounds entailed by a possibilistic knowledge base for the necessity and the possibility of a classical description logic axiom. As shown by Hollunder [53], deciding logical consequences, and thus also deciding satisfiability and computing tight lower and upper bounds can be reduced to deciding logical consequences in classical description logics. A recent implementation of reasoning in possibilistic description logics using KAON2[2] is reported in [109,108].

### 5.4. Main applications

Liau and Yao [80] report on an application of possibilistic description logics in information retrieval. More concretely, they define a possibilistic generalization of the description logic $\mathcal{ALC}$ and show that it can be used in typical information retrieval problems, such as query relaxation, query restriction, and exemplar-based retrieval. Possibilistic description logics can also be used for handling inconsistencies in ontologies [109,108]. Another important application of possibilistic description logics is the representation of user preferences in the Semantic Web. For example, the recent work by Hadjali et al. [38] shows that possibilistic logic can be nicely used for encoding user preferences in the context of databases.

## 6. Vagueness and description logics

In this section, we define the syntax and the semantics of a fuzzy generalization of $\mathcal{SHOIN}(\mathbf{D})$, called *fuzzy* $\mathcal{SHOIN}(\mathbf{D})$. We recall here

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

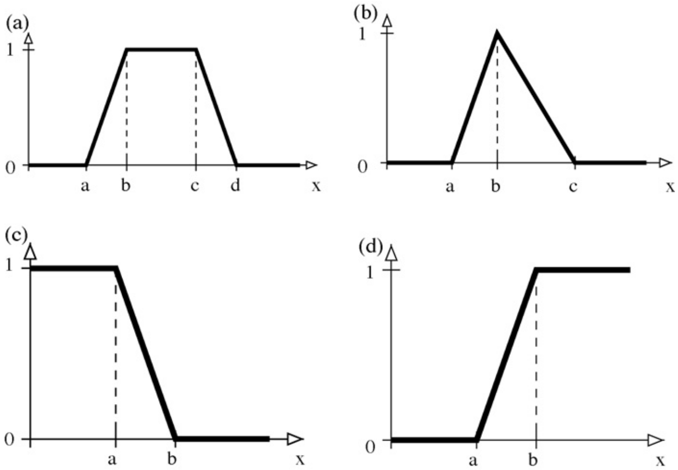


**Fig. 1.** (a) Trapezoidal function $trz(x; a, b, c, d)$, (b) triangular function $tri(x; a, b, c)$, (c) left-shoulder function $ls(x; a, b)$, and (d) right-shoulder function $rs(x; a, b)$.

the semantics given in [131,134], which is based on arbitrary but fixed combination functions $\otimes$, $\oplus$, $\rhd$, and $\ominus$ (see Section 2.3). Note that the language here subsumes the one described in [119], which has been developed in parallel. After describing the main reasoning problems in vague description logics, we summarize their main applications, and we also give an overview on other vague ontology languages.

### 6.1. Syntax

We now define the syntax of fuzzy $\mathcal{SHOIN}(\mathbf{D})$. We first define fuzzy datatype theories and fuzzy modifiers, and then fuzzy axioms and fuzzy knowledge bases.

#### 6.1.1. Fuzzy datatype theories

We have seen that $\mathcal{SHOIN}(\mathbf{D})$ allows to reason with datatypes, such as strings and integers, using the so-called concrete domains. In the fuzzy generalization, concrete domains and thus datatypes may be based on fuzzy sets as well. More specifically, 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 example, like in $\mathcal{SHOIN}(\mathbf{D})$, the datatype predicate $\leq_{18}$ may be a unary crisp predicate over the natural numbers denoting the set of integers smaller than or equal to 18, that is, $\leq_{18} : Natural \rightarrow [0, 1]$ and

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

Then,

$$Minor = Person \sqcup \exists age.\leq_{18} \quad (4)$$

defines persons, whose age is less than or equal to 18, that is, it defines minors.

As for non-crisp fuzzy datatype predicates, we recall that in fuzzy set theory and practice, there are many functions for specifying fuzzy set membership degrees. In particular, the triangular, the trapezoidal, the left-shoulder, and the right-shoulder functions are simple, but most frequently used to specify fuzzy set membership degrees (see Fig. 1). Using these functions, we may then define, for example, $Young : Natural \rightarrow [0, 1]$ to be a fuzzy datatype predicate over the natural numbers denoting the degree of youngness of a person's age. The fuzzy datatype predicate $Young$ may be defined as $Young(x) = ls(x; 10, 30)$. Then,

$$YoungPerson = Person \sqcup \exists age.Young \quad (5)$$

denotes young persons.

#### 6.1.2. Fuzzy modifiers

Fuzzy $\mathcal{SHOIN}(\mathbf{D})$ also supports fuzzy modifiers, an interesting feature of fuzzy logics. Fuzzy modifiers, like *very*, *more_or_less*, and *slightly*, apply to fuzzy sets to change their membership function. Formally, a *fuzzy modifier m* represents a function $f_m : [0, 1] \rightarrow [0, 1]$. For example, we may define $f_{very}(x) = x^2$ and $f_{slightly}(x) = \sqrt{x}$. Fuzzy modifiers have been considered, e.g., in [52,144]. Syntactically, if $\mathbf{M}$ is a new alphabet for fuzzy modifiers, $m \in \mathbf{M}$ is a fuzzy modifier, and $C$ is a concept in fuzzy $\mathcal{SHOIN}(\mathbf{D})$, then $m(C)$ is a concept in fuzzy $\mathcal{SHOIN}(\mathbf{D})$ as well. For example, by referring to Example 3.1, we may define the concept of sports cars as

$$SportsCar = Car \sqcap \exists max\_speed.very(High), \quad (6)$$

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

#### 6.1.3. Fuzzy knowledge bases

We next define fuzzy knowledge bases in fuzzy $\mathcal{SHOIN}(\mathbf{D})$. We first define concepts and fuzzy axioms in fuzzy $\mathcal{SHOIN}(\mathbf{D})$.

Concepts in fuzzy $\mathcal{SHOIN}(\mathbf{D})$ are defined in nearly the same way as concepts in $\mathcal{SHOIN}(\mathbf{D})$, except that we now also allow fuzzy modifiers from a set of fuzzy modifiers $\mathbf{M}$ as unary operators on concepts. More concretely, *Concepts* in fuzzy $\mathcal{SHOIN}(\mathbf{D})$ are defined by induction as follows. Every atomic concept $A \in \mathbf{A}$ is a concept, $\bot$ and $\top$ are concepts, and if $a_1, \ldots, a_n \in \mathbf{I}$, then $\{a_1, \ldots, a_n\}$ is a concept (called *oneOf*). If $C, C_1, C_2$ are concepts, $R \in \mathbf{R}_A \cup \mathbf{R}_A^-$, and $m \in \mathbf{M}$, then $(C_1 \sqcap C_2)$, $(C_1 \sqcup C_2)$, $\neg C$, and $m(C)$ are concepts (called *conjunction*, *disjunction*, *negation*, and *fuzzy modification*, respectively), as well as $\exists R.C$, $\forall R.C$, $\geq nR$, and $\leq nR$ (called *exists*, *value*, *atleast*, and *atmost restriction*, respectively) for an integer $n \geq 0$. If $D$ is an $n$-ary datatype predicate and $T, T_1, \ldots, T_n \in \mathbf{R}_D$, then $\exists T_1, \ldots, T_n.D$, $\forall T_1, \ldots, T_n.D$, $\geq nT$, and $\leq nT$ are concepts (called *datatype exists*, *value*, *atleast*, and *atmost restriction*, respectively) for an integer $n \geq 0$. We eliminate parentheses as usual. For decidability reasons, number restrictions are restricted to simple abstract roles.

We define fuzzy axioms, fuzzy RBoxes, fuzzy TBoxes, fuzzy ABoxes, and fuzzy knowledge bases in fuzzy $\mathcal{SHOIN}(\mathbf{D})$ as follows.

A *fuzzy RBox* $\mathcal{R}$ is a finite set of transitivity axioms Trans$(R)$ in $\mathcal{SHOIN}(\mathbf{D})$ and *fuzzy role inclusion axioms* of the form $\alpha \geq n$, $\alpha \leq n$, $\alpha > n$, and $\alpha < n$, where $\alpha$ is a role inclusion axiom in $\mathcal{SHOIN}(\mathbf{D})$, and $n \in [0, 1]$.

A *fuzzy TBox* $\mathcal{T}$ is a finite set of *fuzzy concept inclusion axioms* $\alpha \geq n, \alpha \leq n, \alpha > n$, and $\alpha < n$, where $\alpha$ is a concept inclusion axiom in $\mathcal{SHOIN}(\mathbf{D})$, and $n \in [0, 1]$.

A *fuzzy ABox* $\mathcal{A}$ consists of a finite set of equality and inequality axioms $a = b$ and $a \neq b$, respectively, and of *fuzzy concept* and *fuzzy role membership axioms* of the form $\alpha \geq n$, $\alpha \leq n$, $\alpha > n$, or $\alpha < n$, where $\alpha$ is a concept or role membership axiom in $\mathcal{SHOIN}(\mathbf{D})$, and $n \in [0, 1]$.

For example, $a : C \geq 0.1$, $(a, b) : R \leq 0.3$, $R \sqsubseteq S \geq 0.4$, and $C \sqsubseteq D \leq 0.6$ are fuzzy axioms. Informally, from a semantical point of view, a fuzzy axiom $\alpha \geq n$ (respectively, $\alpha \leq n$, $\alpha > n$, and $\alpha < n$) constrains the membership degree of $\alpha$ to be at least (respectively, at most, greater than, and less than) $n$. Hence, $jim : YoungPerson \geq 0.2$ says that $jim$ is a $YoungPerson$ with degree at least 0.2. On the other hand, a fuzzy concept inclusion axiom of the form $C \sqsubseteq D \geq n$ says that the subsumption degree between $C$ and $D$ is at least $n$.

A *fuzzy axiom* is a transitivity, a fuzzy concept inclusion, a fuzzy role inclusion, a fuzzy concept membership, a fuzzy role membership, an equality, or an inequality axiom. A *fuzzy knowledge base* $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$ consists of a fuzzy RBox $\mathcal{R}$, a fuzzy TBox $\mathcal{T}$, and a fuzzy ABox $\mathcal{A}$.

### 6.2. Semantics

We now define the semantics of fuzzy $\mathcal{SHOIN}(\mathbf{D})$. The main idea behind it is that concepts and roles are interpreted as fuzzy subsets of an interpretation's domain. Therefore, axioms in fuzzy $\mathcal{SHOIN}(\mathbf{D})$, rather being satisfied (true) or unsatisfied (false) in an interpretation, are associated with a degree of truth in [0, 1]. In the following, let $\otimes$, $\oplus$, $\rhd$, and $\ominus$ be an arbitrary but fixed t-norm, s-norm, implication function, and negation function, respectively (see Table 3 for some specific choices). As such, the semantics is a generalization of Straccia [125] in which Zadeh Logic has been used as specific interpretation of the connectives (see Table 3).

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 individual $a \in \mathbf{I}$ an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$;
- to each atomic concept $A \in \mathbf{A}$ a function $A^{\mathcal{I}} : \Delta^{\mathcal{I}} \to [0, 1]$;
- to each abstract role $R \in \mathbf{R}_A$ a function $R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \to [0, 1]$;
- to each datatype role $T \in \mathbf{R}_D$ a function $T^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathbf{D}} \to [0, 1]$;
- to each modifier $m \in M$ the modifier function $m^{\mathcal{I}} = f_m : [0, 1] \to [0, 1]$.

The mapping $\cdot^{\mathcal{I}}$ is extended to all roles and concepts as follows (where $x, y \in \Delta^{\mathcal{I}}$):

$$
\begin{aligned}
&(R^-)^{\mathcal{I}}(x, y) = R^{\mathcal{I}}(y, x),\\
&\top^{\mathcal{I}}(x) = 1,\\
&\bot^{\mathcal{I}}(x) = 0,\\
&\{a_1, \ldots, a_n\}^{\mathcal{I}}(x) = \begin{cases} 1 & \text{if } x \in \{a_1{}^{\mathcal{I}}, \ldots, a_n{}^{\mathcal{I}}\},\\ 0 & \text{otherwise,}\end{cases}\\
&(C_1 \sqcap C_2)^{\mathcal{I}}(x) = C_1{}^{\mathcal{I}}(x) \otimes C_2{}^{\mathcal{I}}(x),\\
&(C_1 \sqcup C_2)^{\mathcal{I}}(x) = C_1{}^{\mathcal{I}}(x) \oplus C_2{}^{\mathcal{I}}(x),\\
&(\neg C)^{\mathcal{I}}(x) = \ominus C^{\mathcal{I}}(x),\\
&(m(C))^{\mathcal{I}}(x) = m^{\mathcal{I}}(C^{\mathcal{I}}(x)),\\
&(\exists R.C)^{\mathcal{I}}(x) = \sup_{y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x, y) \otimes C^{\mathcal{I}}(y),\\
&(\forall R.C)^{\mathcal{I}}(x) = \inf_{y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x, y) \rhd C^{\mathcal{I}}(y),\\
&(\geq nR)^{\mathcal{I}}(x) = \sup_{y_1,\ldots,y_n \in \Delta^{\mathcal{I}}, |\{y_1,\ldots,y_n\}|=n} \otimes_{i=1}^{n} R^{\mathcal{I}}(x, y_i),\\
&(\leq nR)^{\mathcal{I}}(x) = \inf_{y_1,\ldots,y_{n+1} \in \Delta^{\mathcal{I}}, |\{y_1,\ldots,y_{n+1}\}|=n+1} \left(\otimes_{i=1}^{n+1} R^{\mathcal{I}}(x, y_i)\right) \rhd 0,\\
&(\exists T_1, \ldots, T_n.D)^{\mathcal{I}}(x) = \sup_{y_1,\ldots,y_n \in \Delta^{\mathbf{D}}} \left(\otimes_{i=1}^{n} T_i^{\mathcal{I}}(x, y_i)\right) \otimes D^{\mathbf{D}}(y_1, \ldots, y_n),\\
&(\forall T_1, \ldots, T_n.D)^{\mathcal{I}}(x) = \inf_{y_1,\ldots,y_n \in \Delta^{\mathbf{D}}} \left(\otimes_{i=1}^{n} T_i^{\mathcal{I}}(x, y_i)\right) \rhd D^{\mathbf{D}}(y_1, \ldots, y_n).
\end{aligned}
$$

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

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

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) [3]) and the existential quantifier $\exists$ is viewed as a disjunction over the elements of the domain. Similarly,

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

is related to the open first-order formula $\forall y.F_R(x, y) \to F_C(y)$, 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 example, this holds in Łukasiewicz logic, but not in Gödel logic. Also interesting is that (see [43]) the axiom $\top \sqsubseteq \neg(\forall R.A) \sqcap (\neg\exists R.\neg A)$ has no classical model, but it has a fuzzy one. Indeed, in [43], it is shown that in Gödel logic it has no finite model, but it has an infinite fuzzy model.

Another point concerns the semantics of number restrictions. The semantics of the concept $\geq nR$ is equivalent to

$$(\geq nR)^{\mathcal{I}}(x) = \sup_{\{y_1,\ldots,y_n\} \subseteq \Delta^{\mathcal{I}}} \otimes_{i=1}^{n} R^{\mathcal{I}}(x, y_i) \otimes \otimes_{1 \leq i < j \leq n} y_i \neq y_j,$$

which is the result of viewing $\geq nR$ as the open first-order formula

$$\exists y_1, \ldots, y_n \cdot \wedge_{i=1}^{n} R(x, y_i) \wedge \wedge_{1 \leq i < j \leq n} y_i \neq y_j.$$

That is, there are at least $n$ distinct elements that satisfy to some degree $R(x, y_i)$. This also guarantees that $\exists R.\top \equiv (\geq 1R)$.

Similarly, the semantics of $\leq nR$ is equivalent to

$$(\leq nR)^{\mathcal{I}}(x) = \inf_{\{y_1,\ldots,y_{n+1}\} \subseteq \Delta^{\mathcal{I}}} \otimes_{i=1}^{n+1} R^{\mathcal{I}}(x, y_i) \rhd \left(\otimes_{1 \leq i < j \leq n+1} y_i = y_j\right),$$

which is the result of viewing $\leq nR$ as the open first-order formula

$$\forall y_1, \ldots, y_{n+1} \cdot \wedge_{i=1}^{n+1} R(x, y_i) \to \vee_{1 \leq i < j \leq n+1} y_i = y_j.$$

Note that not necessarily $(\leq nR) \equiv \neg(\geq n + 1R)$ holds. The equivalence is true for Zadeh and Łukasiewicz logic, but neither for Gödel nor for Product logic.

We extend $\cdot^{\mathcal{I}}$ to all non-fuzzy axioms as follows (where $a, b \in \mathbf{I}$ and $v \in \Delta^{\mathbf{D}}$):

$$
\begin{aligned}
&(C \sqsubseteq D)^{\mathcal{I}} = \inf_{x \in \Delta^{\mathcal{I}}} C^{\mathcal{I}}(x) \rhd D^{\mathcal{I}}(x),\\
&(R \sqsubseteq S)^{\mathcal{I}} = \inf_{x,y \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x, y) \rhd S^{\mathcal{I}}(x, y),\\
&(T \sqsubseteq U)^{\mathcal{I}} = \inf_{(x,y) \in \Delta^{\mathcal{I}} \times \Delta^{\mathbf{D}}} T^{\mathcal{I}}(x, y) \rhd U^{\mathcal{I}}(x, y),\\
&(a : C)^{\mathcal{I}} = C^{\mathcal{I}}(a^{\mathcal{I}}),\\
&((a, b) : R)^{\mathcal{I}} = R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}),\\
&((a, v) : T)^{\mathcal{I}} = T^{\mathcal{I}}(a^{\mathcal{I}}, v^{\mathbf{D}}).
\end{aligned}
$$

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 $\forall x.F_C(x) \to F_D(x)$. This definition is clearly different from the approaches in which $C \sqsubseteq D$ is viewed as $\forall x.C(x) \leq D(x)$ (e.g., [125,122]). This latter approach has the effect that the subsumption relationship is a classical {0, 1}relationship, while in the former approach, subsumption is determined up to a certain degree in [0, 1].

We next define what it means that a fuzzy interpretation $\mathcal{I}$ *satisfies* a fuzzy axiom $E$, or $\mathcal{I}$ is a *model* of $E$, denoted $\mathcal{I} \models E$, as follows. We define: (1) $\mathcal{I} \models \mathrm{Trans}(R)$ iff $R^{\mathcal{I}}(x, y) \geq \sup_{z \in \Delta^{\mathcal{I}}} R^{\mathcal{I}}(x, z) \otimes R^{\mathcal{I}}(z, y)$ for all $x, y \in \Delta^{\mathcal{I}}$, (2) $\mathcal{I} \models \alpha\,\theta\,n$, where $\theta \in \{\geq, \leq, >, <\}$, iff $\alpha^{\mathcal{I}}\,\theta\,n$, (3) $\mathcal{I} \models a = b$ iff $a^{\mathcal{I}} = b^{\mathcal{I}}$, and (4) $\mathcal{I} \models a \neq b$ iff $a^{\mathcal{I}} \neq b^{\mathcal{I}}$. We say that a concept $C$ is *satisfiable* iff there exists an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ and an individual $x \in \Delta^{\mathcal{I}}$ such that $C^{\mathcal{I}}(x) > 0$. We say that $\mathcal{I}$ *satisfies* a set of fuzzy axioms $\mathcal{E}$, or $\mathcal{I}$ is a *model* of $\mathcal{E}$, denoted $\mathcal{I} \models \mathcal{E}$, iff $\mathcal{I} \models E$ for all $E \in \mathcal{E}$. We say $\mathcal{I}$ *satisfies* a fuzzy knowledge base $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$, or $\mathcal{I}$ is a *model* of $\mathcal{K}$, denoted $\mathcal{I} \models \mathcal{K}$, iff $\mathcal{I}$ is a model of $\mathcal{T} \cup \mathcal{R} \cup \mathcal{A}$. We say $\mathcal{K}$ is *satisfiable* iff it has a model. A fuzzy axiom $E$ is a *logical consequence* of a fuzzy knowledge base $\mathcal{K}$, denoted $\mathcal{K} \models E$, iff every model of $\mathcal{K}$ satisfies $E$.

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

$$\mathcal{K} \models SportsCar \sqsubseteq Car \geq 1, \quad \mathcal{K} \models mgb : SportsCar \leq 0.28,$$
$$\mathcal{K} \models enzo : SportsCar \geq 1, \quad \mathcal{K} \models tt : SportsCar \geq 0.92.$$

Note that the maximal speed limit of the *mgb* car ($\leq_{170\,\mathrm{km/h}}$) induces the upper limit 0.28 of the membership degree.

**Example 6.2.** Consider the knowledge base $\mathcal{K}$ with the definitions in (4) and (5). Then, under Łukasiewicz logic, we have that (see [129])

$$\mathcal{K} \models Minor \sqsubseteq YoungPerson \geq 0.6,$$
$$\mathcal{K} \models YoungPerson \sqsubseteq Minor \geq 0.4,$$

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

An interesting point here is that according to the semantics of fuzzy $\mathcal{SHOIN}(\mathbf{D})$, e.g., a minor is a young person to a certain degree and is obtained without explicitly mentioning it. This inference cannot be achieved in classical $\mathcal{SHOIN}(\mathbf{D})$. Similarly, referring to Example 3.1, in fuzzy $\mathcal{SHOIN}(\mathbf{D})$, the car *tt* is a sports car to a certain degree. Therefore, unlike Example 3.1, *tt* is now closely a sports car, *as it should be*.

### *6.3. Main reasoning problems*

In addition to the standard problems of deciding the satisfiability of fuzzy knowledge bases, deciding the satisfiability of concepts relative to fuzzy knowledge bases, and deciding logical consequences of fuzzy axioms from fuzzy knowledge bases, two other important reasoning problems are the best truth value bound (BTVB) problem and the best satisfiability bound problem, which we describe in the following.

Given a fuzzy knowledge base $\mathcal{K}$ and a classical axiom $\alpha$, where $\alpha$ is neither a transitivity axiom nor an equality or inequality axiom, it is of interest to compute $\alpha$'s best lower and upper truth value bounds (*best truth value bound*). The *greatest lower bound* of $\alpha$ relative to $\mathcal{K}$, denoted $glb(\mathcal{K}, \alpha)$, is defined by

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

while the *least upper bound* of $\alpha$ relative to $\mathcal{K}$, denoted $lub(\mathcal{K}, \alpha)$, is defined by

$$lub(\mathcal{K}, \alpha) = \inf\{n \mid \mathcal{K} \models \alpha \leq n\},$$

where $\sup \emptyset = 0$ and $\inf \emptyset = 1$. For example, the logical consequences in Examples 6.1 and 6.2 contain the best truth value bounds. Furthermore, note that

$$lub(\mathcal{K}, a : C) = \ominus\, glb(\mathcal{K}, a : \neg C), \quad (7)$$

that is, the *lub* can be determined through the *glb* (and vice versa).

Similarly, $lub(\mathcal{K}, (a, b) : R) = \ominus\, glb(\mathcal{K}, a : \neg\exists R.\{b\})$. Note also that $\mathcal{K} \models \alpha \geq n$ iff $glb(\mathcal{K}, \alpha) \geq n$, and $\mathcal{K} \models \alpha \leq n$ iff $lub(\mathcal{K}, \alpha) \leq n$.

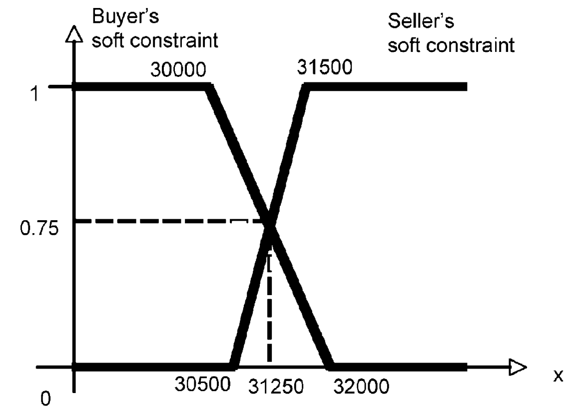


**Fig. 2.** The soft price constraints.

Finally, the *best satisfiability bound* of a concept $C$ relative to $\mathcal{K}$, denoted $glb(\mathcal{K}, C)$, is defined by

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

Intuitively, among all models $\mathcal{I}$ of $\mathcal{K}$, we determine 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 \$31 500, as from the catalog price. A buyer is looking for a sports car, but wants to pay not more than around \$30 000. In classical description logics 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 to consider prices as concrete fuzzy sets instead. For example, the seller may consider optimal to sell above \$31 500, but can go down to \$30 500. The buyer prefers to spend less than \$30 000, but can go up to \$32 000. We may represent these statements by means of the following axioms (see Fig. 2):

$$AudiTT = SportsCar \sqcap \exists hasPrice.rs(x; 30\,500, 31\,500),$$
$$Query = SportsCar \sqcap \exists hasPrice.ls(x; 30\,000, 32\,000).$$

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 AudiTT and the query matches is 0.75). That is, $glb(\mathcal{K}, C) = 0.75$ and corresponds to the point where both requests intersect (that is, the car may be sold at \$31 250).

Problems such as determining the greatest lower bound of an axiom can be solved by relying on mixed integer linear programming (MILP) (see, e.g., [9,128,130,140,139]). Roughly, the basic idea is as follows. Consider a fuzzy knowledge base $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$. To determine the greatest lower bound of an axiom, we combine appropriate description logic tableaux rules with methods developed in the context of *many-valued logics* [40]. For example, to determine, e.g., $glb(\mathcal{K}, a : C)$, we consider an expression of the form $a : C \leq x$, where $x$ is a $[0, 1]$-valued variable. Then, we construct a tableaux for $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A} \cup \{a : C \leq x\})$ in which the application of satisfiability preserving rules generates new assertion axioms together with *inequations* over $[0, 1]$-valued variables. These inequations have to hold in order to respect the semantics of the description logic constructors. Finally, to determine the greatest lower bound, we *minimize* the original variable $x$ such that all constraints are satisfied.[3] Interestingly, under Łukasiewicz and Zadeh logic, we end up with a *mixed integer linear programming* optimization problem, while under the Product logic, we end up with a *mixed integer quadratically constrained programming (MIQCP)* optimization problem [111]. Similarly, $glb(\mathcal{K}, C_1 \sqsubseteq C_2)$ is the minimal value of $x$ such that $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A} \cup \{a : C_1 \sqcap \neg C_2 \geq 1 - x\})$ is satisfiable, where $a$ is new individual. Therefore, the greatest lower bound problem can be reduced to the minimal satisfiability problem of a fuzzy knowledge base. Furthermore, $glb(\mathcal{K}, C)$ is determined by the maximal value of $x$ such that $(\mathcal{R}, \mathcal{T}, \mathcal{A} \cup \{a : C \geq x\})$ is satisfiable, where $a$ is a new individual. In summary,

$glb(\mathcal{K}, a : C) = \min x$ such that $(\mathcal{R}, \mathcal{T}, \mathcal{A} \cup \{a : C \leq x\})$ satisfiable,
$glb(\mathcal{K}, C_1 \sqsubseteq C_2) = \min x$ such that $(\mathcal{R}, \mathcal{T}, \mathcal{A} \cup \{a : C_1 \sqcap \neg C_2 \geq 1 - x\})$ satisfiable,
$glb(\mathcal{K}, C) = \max x$ such that $(\mathcal{R}, \mathcal{T}, \mathcal{A} \cup \{a : C \geq x\})$ satisfiable.

### 6.4. Main applications

Fuzzy logic has numerous practical applications in general (see, e.g., [67]). Related to fuzzy description logics, we point out that they have first been proposed for *logic-based information retrieval* [96], which originated from the idea to annotate textual documents with graded description logic sentences, which goes back to [97]. The idea has been reconsidered in [120,141,156]. In particular, (i) Zhang et al. [156] describe a semantic portal that is based on fuzzy description logics; (ii) Li et al. [76] present an improved semantic search model by integrating inference and information retrieval and an implementation in the security domain; (iii) Straccia and Visco [141] report on a multimedia information retrieval system based on a fuzzy DLR-Lite description logic, which is capable to deal with hundreds of thousands of images. D'Aquin et al. [17] provide a use case in the medical domain, where fuzzy concrete domains are used to identify tumor regions in X-ray images. Agarwal and Lamparter [1] use fuzzy description logics to improve searching and comparing products in electronic markets. They provide a more expressive search mechanism that is closer to human reasoning and that aggregates multiple search criteria to a single value (ranking of an offer relative to the query), thus enabling a better selection of offers to be considered for the negotiation. Liu et al. [81] use a fuzzy description logic to model the management part in project selection tasks.

### 6.5. Other vague ontology languages

There are several extensions of description logics respectively ontology languages using the theory of fuzzy logic in the literature. They can be classified according to (a) the description logic respectively ontology language that they generalize, (b) the allowed fuzzy constructs, (c) the underlying fuzzy logics, and (d) their reasoning algorithms.

The first work is due to Yen [154], who proposes a fuzzy extension of a very restricted sublanguage of $\mathcal{ALC}$, called $\mathcal{FL}^-$ [12,73]. The work includes fuzzy terminological knowledge, but no fuzzy assertional knowledge, and it is based on Zadeh logic. It already informally talks about the use of fuzzy modifiers and fuzzy concrete domains. Though, the unique reasoning facility, the subsumption test, is a crisp yes/no questioning. Tresp and Molitor [144] consider a more general extension of fuzzy $\mathcal{ALC}$. Like Yen, they also allow for fuzzy terminological knowledge along with a special form of fuzzy modifiers (which are a combination of two linear functions), but no fuzzy assertional knowledge, and they assume Zadeh logic as underlying fuzzy logic. The work also presents a sound and complete reasoning algorithm testing the subsumption relationship using a linear programming oracle.

Another fuzzy extension of $\mathcal{ALC}$ is due to Straccia [123,125], who allows for both fuzzy terminological and fuzzy assertional knowledge, but not for fuzzy modifiers and fuzzy concrete domains, and again assumes Zadeh logic as underlying fuzzy logic. Straccia [123,125] also introduces the best truth value bound problem and provides a sound and complete reasoning algorithm based on completion rules. In [124], Straccia reports a four-valued variant of fuzzy $\mathcal{ALC}$. In the same spirit, Hölldobler et al. [50,51] extend Straccia's fuzzy $\mathcal{ALC}$ with concept modifiers of the form $f_m(x) = x^\beta$, where $\beta > 0$, and present a sound and complete reasoning algorithm (based on completion rules) for the graded subsumption problem.

Straccia's works [127,133,138] are essentially as [125], except that now the set of possible truth values is a complete lattice rather than [0, 1].

Sanchez and Tettamanzi [112–114] consider a fuzzy extension of the description logic $\mathcal{ALCQ}$ (without assertional component) under Zadeh logic, and they start addressing the issue of a fuzzy semantics of quantifiers. Essentially, fuzzy quantifiers allow to state concepts such as *FaithfulCustomer* ⊓ (*Most*)*buys*.*LowCalorieFood* encoding "the set of all individuals that mostly by low calorie food". An algorithm is presented, which calculates the satisfiability interval for a fuzzy concept.

Hájek [43,44] considers a fuzzy extension of the description logic $\mathcal{ALC}$ under arbitrary t-norms. He provides in particular algorithms for deciding whether $C \sqsubseteq D \geq 1$ is a tautology and whether $C \sqsubseteq D \geq 1$ is satisfiable, which are based on a reduction to the propositional BL logic for which a Hilbert-style axiomatization exists [42](but see also [44] for the complexity of rational Pavelka logic, and see [10] for some complexity results on reasoning in fuzzy description logics).

Straccia [126] provides a translation of fuzzy $\mathcal{ALC}$ (with general concept inclusion axioms) into classical $\mathcal{ALC}$. The translation is modular, and thus expected to be extendable to more expressive fuzzy description logics as well. The main idea is to translate a fuzzy assertion of the form $a : C \geq n$ into a crisp assertion $a : C_n$, with the intended meaning "$a$ is an instance of $C$ to degree at least $n$". Then, concept inclusion axioms are used to correctly relate the $C_n$'s. For example, $C_{0.7} \sqsubseteq C_{0.6}$ is used to encode that whenever an individual is an instance of $C$ to degree at least 0.7, then it is also an instance of $C$ to degree at least 0.6. The translation is at most quadratic in the size of the fuzzy knowledge base. Note that the translation does not yet work in the presence of fuzzy modifiers and fuzzy concrete domains. Bobillo et al. [8] extend the approach to a variant of fuzzy $\mathcal{SHOIN}$. The idea has further been considered in the works [78,79], which essentially provide a crisp language in which expressions of, e.g., the form $a : \forall R_{0.8}.C_{0.9}$ are allowed, with the 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". The idea has further been extended to a distributed variant of fuzzy description logics in [82].

In [94], a fuzzy extension (based on Zadeh logic) of CARIN [74] is provided, which combines fuzzy description logics with non-recursive Horn rules.

Other extensions of fuzzy description logics concern their integration with fuzzy logic programs, which however goes beyond the scope of the present paper (see, e.g., [138,135,133,89,91,149]). An interesting extension is due to Kang et al. [64], who extends fuzzy description logics by comparison operators, e.g., to state that

[3] Informally, suppose the minimal value is $\bar{n}$ (if no such value exists, then $\mathcal{K}$ is not satisfiable). We know then that for any interpretation $\mathcal{I}$ satisfying $\mathcal{K}$ such that $(a : C)^\mathcal{I} < \bar{n}$, the starting set is unsatisfiable, and thus $(a : C)^\mathcal{I} \geq \bar{n}$ holds. This means that $glb(\mathcal{K}, a : C) = \bar{n}$.

"Tom is taller than Tim". Another interesting extension is proposed by Dubois et al. [22], who combine fuzzy description logics with possibility theory. Essentially, since $a : C \geq n$ 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 [22].

We recall that usually the semantics used for fuzzy description logics is based on Zadeh logic, but where the concept inclusion is crisp, that is, $C \sqsubseteq D$ is viewed as $\forall x.C(x) \leq D(x)$. In [52,144], a calculus for fuzzy $\mathcal{ALC}$ [115] with fuzzy modifiers and simple TBoxes under Zadeh logic is reported. No indication for the BTVB problem is given. Straccia [123,125] reports a calculus for fuzzy $\mathcal{ALC}$ and simple TBoxes under Zadeh logic and addresses the BTVB problem. How the satisfiability problem and the BTVB problem can be reduced to classical $\mathcal{ALC}$, and thus can be solved by means of tools like FaCT and RACER is shown in [126]. Results providing a tableaux calculus for fuzzy $\mathcal{SHIN}$ under Zadeh logic (but only allowing for a restricted form of concept inclusion axioms, which are called *fuzzy inclusion introductions* and *fuzzy equivalence introductions*), by adapting similar techniques as for the classical counterpart, are shown in [120,121]. Fuzzy general concept inclusion axioms under Zadeh logic can be managed as described in [122]. Also interesting is the work [77], which provides a tableau for fuzzy $\mathcal{SHI}$ with general concept inclusion axioms. Finally, the reasoning techniques for classical $\mathcal{SHOIN}(\mathbf{D})$ [57] can be extended to [125], as [120,121,118,117] already show.

On the other hand, fuzzy tableaux algorithms under Zadeh semantics do not seem to be suitable to be adapted to other semantics, such as Łukasiewicz logic. Even more problematic is the fact that they are yet unable to deal with fuzzy concrete domains. Despite these negative results, recently, Straccia [130,128] report a calculus for fuzzy $\mathcal{ALC}(\mathbf{D})$ whenever the connectives, the modifiers, and the fuzzy datatype predicates are representable as bounded mixed integer linear programs (MILPs). For example, Łukasiewicz logic satisfies these conditions as well as the membership functions for fuzzy datatype predicates that 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 $\mathcal{SHIF}(\mathbf{D})$ [139](the description logic behind OWL Lite). The use of MILP for reasoning in fuzzy description logics is not surprising as their use for automated deduction in many-valued logics is well known [39,40]. Bobillo and Straccia [9] provide a calculus for fuzzy $\mathcal{ALC}(\mathbf{D})$ under product semantics.

A very recent problem for fuzzy description logics is the top-$k$ retrieval problem. While in classical semantics, a tuple satisfies or does not satisfy a query, in fuzzy description logics, a tuple may satisfy a query to a degree. Hence, for example, given a conjunctive query over a fuzzy description logic knowledge base, it is of interest to compute only the top-$k$ answers. While in relational databases, this problem is a current research area (see, e.g., [31,60,75]), very few is known for the case of first-order knowledge bases in general (but see [136]) and description logics in particular. The only works that we are aware of are [132,137,141], which deal with the problem of finding the top-$k$ result over knowledge bases in a fuzzy generalization of *DL-Lite* [13](note that [103] is subsumed by [137]).

## 7. Conclusion

Handling uncertainty and vagueness has started to play an important role in ontology languages for the Semantic Web. In this paper, we have first provided a brief introduction to uncertainty and vagueness at the propositional level, and we have summarized the basics of classical description logics for the Semantic Web. We have then described the most prominent approaches to handling probabilistic uncertainty, possibilistic uncertainty, and vagueness in expressive description logics for the Semantic Web, and we have given an overview of related approaches.

There are many important aspects that are open for future research. In particular, an important issue is to develop more scalable formalisms for handling probabilistic uncertainty, possibilistic uncertainty, and vagueness in ontology languages for the Semantic Web, especially those scalable formalisms that are also practically relevant. Another important issue is to provide more implementations, especially of scalable formalisms. It would also be interesting to integrate the above forms of uncertainty and vagueness in a single description logic for the Semantic Web. Another interesting issue for future research is the integration of probabilistic, possibilistic, and fuzzy description logics with rule-based languages for the Semantic Web.

## Acknowledgments

This work has been partially supported by the German Research Foundation (DFG) under the Heisenberg Programme. We thank the reviewers of this paper, whose constructive comments helped to improve our work.

## References

[1] S. Agarwal, S. Lamparter, SMART: a semantic matchmaking portal for electronic markets, in: Proceedings CEC-2005, IEEE Computer Society, 2005, pp. 405–408.
[2] S. Amarger, D. Dubois, H. Prade, Constraint propagation with imprecise conditional probabilities, in: Proceedings UAI-1991, Morgan Kaufmann, 1991, pp. 26–34.
[3] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P.F. Patel-Schneider (Eds.), The Description Logic Handbook: Theory, Implementation, and Applications, Cambridge University Press, 2003.
[4] F. Baader, P. Hanschke, A schema for integrating concrete domains into concept languages, in: Proceedings IJCAI-1991, Morgan Kaufmann, 1991, pp. 452–457.
[5] F. Bacchus, A. Grove, J.Y. Halpern, D. Koller, From statistical knowledge bases to degrees of belief, Artif. Intell. 87 (1996) 75–143.
[6] T. Berners-Lee, Weaving the Web, Harper, San Francisco, 1999.
[7] T. Berners-Lee, J. Hendler, O. Lassila, The Semantic Web, Sci. Am. 284 (5) (2001) 34–43.
[8] F. Bobillo, M. Delgado, J. Gómez-Romero, A crisp representation for fuzzy $\mathcal{SHOIN}$ with fuzzy nominals and general concept inclusions, in: Proceedings URSW-2006, 2006.
[9] F. Bobillo, U. Straccia, A fuzzy description logic with product t-norm, in: Proceedings FUZZ-IEEE-2007, IEEE Computer Society, 2007, pp. 652–657.
[10] P. Bonatti, A. Tettamanzi, Some complexity results on fuzzy description logics, in: V. Di Gesù, F. Masulli, A. Petrosino (Eds.), Fuzzy Logic and Applications, volume 2955 of LNCS, Springer, 2006, pp. 19–24.
[11] 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).
[12] R.J. Brachman, H.J. Levesque, The tractability of subsumption in frame-based description languages, in: Proceedings AAAI-1984, AAAI Press, 1984, pp. 34–37.
[13] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati, DL-Lite: Tractable description logics for ontologies, in: Proceedings AAAI-2005, AAAI Press/MIT Press, 2005, pp. 602–607.
[14] P.C.G. da Costa, Bayesian semantics for the Semantic Web, PhD thesis, George Mason University, Fairfax, VA, USA, 2005.
[15] P.C.G. da Costa, K.B. Laskey, PR-OWL: a framework for probabilistic ontologies, in: Proceedings FOIS-2006, IOS Press, 2006, pp. 237–249.
[16] P.C.G. da Costa, K.B. Laskey, K.J. Laskey, PR-OWL: a Bayesian ontology language for the Semantic Web, in: Proceedings URSW-2005, 2005, pp. 23–33.
[17] M. d'Aquin, J. Lieber, A. Napoli, Towards a semantic portal for oncology using a description logic with fuzzy concrete domains, in: E. Sanchez (Ed.), Fuzzy Logic and the Semantic Web, Capturing Intelligence, Elsevier, 2006, pp. 379–393.
[18] Z. Ding, Y. Peng, A probabilistic extension to ontology language OWL, in: Proceedings HICSS-2004, 2004.
[19] Z. Ding, Y. Peng, R. Pan, BayesOWL: uncertainty modeling in Semantic Web ontologies, in: Z. Ma (Ed.), Soft Computing in Ontologies and Semantic Web, volume 204 of Studies in Fuzziness and Soft Computing, Springer, 2006.
[20] D. Dubois, L. Foulloy, G. Mauris, H. Prade, Probability–possibility transformations, triangular fuzzy sets, and probabilistic inequalities, Reliab. Comput. 10 (4) (2004) 273–297.

[21] D. Dubois, J. Lang, H. Prade, Possibilistic logic, in: D.M. Gabbay, C.J. Hogger, J.A. Robinson (Eds.), in: Handbook of Logic in Artificial Intelligence and Logic Programming, vol. 3, Clarendon Press, Oxford, 1994, pp. 439–513.
[22] D. Dubois, J. Mengin, H. Prade, Possibilistic uncertainty and fuzzy features in description logic: a preliminary discussion, in: E. Sanchez (Ed.), Fuzzy Logic and the Semantic Web, Capturing Intelligence, Elsevier, 2006, pp. 101–114.
[23] D. Dubois, H. Prade, On fuzzy syllogisms, Comput. Intell. 4 (2) (1988) 171–179.
[24] D. Dubois, H. Prade, When upper probabilities are possibility measures, Fuzzy Sets Syst. 49 (1992) 65–74.
[25] D. Dubois, H. Prade, Can we enforce full compositionality in uncertainty calculi? in: Proceedings AAAI-1994, AAAI Press, 1994, pp. 149–154.
[26] D. Dubois, H. Prade, Possibility theory, probability theory and multiple-valued logics: a clarification Ann. Math. Artif. Intell. 32 (1–4) (2001) 35–66.
[27] D. Dubois, H. Prade, L. Godo, R.L. de Màntaras, Qualitative reasoning with imprecise probabilities, J. Intell. Inf. Syst. 2 (1993) 319–363.
[28] D. Dubois, H. Prade, J.-M. Touscas, Inference with imprecise numerical quantifiers, in: Z.W. Ras, M. Zemankova (Eds.), Intelligent Systems, Ellis Horwood, 1990, pp. 53–72 (Chapter 3).
[29] M. Dürig, T. Studer, Probabilistic ABox reasoning: preliminary results, in: Proceedings DL-2005, 2005, pp. 104–111.
[30] J. Euzenat, P. Shvaiko, Ontology Matching, Springer, Heidelberg, Germany, 2007.
[31] R. Fagin, Combining fuzzy information: an overview, SIGMOD Rec. 31 (2) (2002) 109–118.
[32] R. Fagin, J.Y. Halpern, N. Megiddo, A logic for reasoning about probabilities, Inf. Comput. 87 (1990) 78–128.
[33] D. Fensel, W. Wahlster, H. Lieberman, J. Hendler (Eds.), Spinning the Semantic Web: Bringing the World Wide Web to its Full Potential, MIT Press, 2002.
[34] A.M. Frisch, P. Haddawy, Anytime deduction for probabilistic logic, Artif. Intell. 69 (1–2) (1994) 93–122.
[35] Y. Fukushige, Representing probabilistic knowledge in the Semantic Web, in: Proceedings of the W3C Workshop on Semantic Web for Life Sciences, 2004.
[36] R. Giugno, 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, Springer, 2002, pp. 86–97.
[37] N. Guarino, Formal ontology, conceptual analysis and knowledge representation, Int. J. Hum.-Comput. Stud. 43 (5–6) (1995) 625–640.
[38] A. Hadjali, S. Kaci, H. Prade, Database preference queries—a possibilistic logic approach with symbolic priorities, in: Proceedings FoIKS-2008, LNCS, Springer, 2008.
[39] R. Hähnle, Many-valued logics and mixed integer programming, Ann. Math. Artif. Intell. 12 (3–4) (1994) 231–263.
[40] R. Hähnle, Advanced many-valued logics, in: D.M. Gabbay, F. Guenthner (Eds.), in: Handbook of Philosophical Logic, 2nd ed., vol. 2, Kluwer, Dordrecht, Holland, 2001.
[41] T. Hailperin, Sentential Probability Logic: Origins, Development, Current Status, and Technical Applications, Associated University Presses, London, UK, 1996.
[42] P. Hájek, Metamathematics of Fuzzy Logic, Kluwer, 1998.
[43] P. Hájek, Making fuzzy description logics more expressive, Fuzzy Sets Syst. 154 (1) (2005) 1–15.
[44] P. Hájek, What does mathematical fuzzy logic offer to description logic? in: E. Sanchez (Ed.), Fuzzy Logic and the Semantic Web, Capturing Intelligence, Elsevier, 2006, pp. 91–100.
[45] A. Halevy, M. Franklin, D. Maier, Principles of dataspace systems, in: Proceedings PODS-2006, ACM Press, 2006, pp. 1–9.
[46] P. Hansen, B. Jaumard, G.-B.D. Nguetsé, M.P. de Aragão, Models and algorithms for probabilistic and Bayesian logic, in: Proceedings IJCAI-1995, Morgan Kaufmann, 1995, pp. 1862–1868.
[47] J. Heinsohn, Probabilistic description logics, in: Proceedings UAI-1994, Morgan Kaufmann, 1994, pp. 311–318.
[48] M. Holi, E. Hyvönen, A method for modeling uncertainty in Semantic Web taxonomies, in: Proceedings WWW-2004, ACM Press, 2004, pp. 296–297.
[49] M. Holi, E. Hyvönen, Modeling degrees of conceptual overlap in Semantic Web ontologies, in: Proceedings URSW-2005, 2005, pp. 98–99.
[50] S. Hölldobler, T.D. Khang, H.-P. Störr, A fuzzy description logic with hedges as concept modifiers, in: Proceedings InTech/VJFuzzy-2002, 2002, pp. 25–34.
[51] S. Hölldobler, N.H. Nga, T.D. Khang, The fuzzy description logic $\mathcal{ALC}_{\mathcal{FLH}}$, in: Proceeedings DL-2005, 2005.
[52] S. Hölldobler, H.-P. Störr, T.D. Khang, The subsumption problem of the fuzzy description logic $ALC_{FH}$, in: Proceedings IPMU-2004, 2004, pp. 243–250.
[53] B. Hollunder, An alternative proof method for possibilistic logic and its application to terminological logics Int. J. Approx. Reason. 12 (2) (1995) 85–109.
[54] I. Horrocks, Using an expressive description logic: FaCT or fiction? in: Proceedings KR-1998, Morgan Kaufmann, 1998, pp. 636–649.
[55] I. Horrocks, P. Patel-Schneider, Reducing OWL entailment to description logic satisfiability J. Web Sem. 1 (4) (2004) 345–357.
[56] I. Horrocks, P.F. Patel-Schneider, F. van Harmelen, From $\mathcal{SHIQ}$ and RDF to OWL: the making of a web ontology language, J. Web Sem. 1 (1) (2003) 7–26.
[57] I. Horrocks, U. Sattler, A tableaux decision procedure for $\mathcal{SHOIQ}$, in: Proceedings IJCAI-2005, 2005, pp. 448–453 (Extended version to appear in J. Autom. Reason.).
[58] I. Horrocks, U. Sattler, S. Tobies, Practical reasoning for expressive description logics, in: Proceedings LPAR-1999, volume 1705 of LNCS, Springer, 1999, pp. 161–180.
[59] E. Hung, Y. Deng, V.S. Subrahmanian, TOSS: an extension of TAX with ontologies and similarity queries, in: Proceedings ACM SIGMOD 2004, ACM Press, 2004, pp. 719–730.
[60] I.F. Ilyas, W.G. Aref, A.K. Elmagarmid, Supporting top-k join queries in relational databases, in: Proceedings VLDB-2003, Morgan Kaufmann, 2003, pp. 754–765.
[61] M. Jaeger, Probabilistic reasoning in terminological logics, in: Proceedings KR-1994, Morgan Kaufmann, 1994, pp. 305–316.
[62] M. Jaeger, Probabilistic role models and the guarded fragment, in: Proceedings IPMU-2004, 2004, pp. 235–242 (Extended version in Int. J. Uncertain., Fuzz. Knowl.-Based Syst. 14 (1) (2006) 43–60).
[63] B. Jaumard, P. Hansen, M.P. de Aragão, Column generation methods for probabilistic logic, ORSA J. Comput. 3 (1991) 135–147.
[64] D. Kang, B. Xu, J. Lu, Y. Li, Reasoning for a fuzzy description logic with comparison expressions, in: Proceeedings DL-2006, 2006.
[65] A. Kaplunova, R. Möller, Probabilistic LCS in a P-CLASSIC implementation, Technical report, Institute for Software Systems (STS), Hamburg University of Technology, Germany, October 2007 (Available at www.sts.tu-harburg.de/tech-reports/papers.html).
[66] P. Klinov, Introducing Pronto: Probabilistic DL Reasoning in Pellet, September 2007 (Available at clarkparsia.com/weblog/2007/09/27/introducing-pronto/).
[67] G.J. Klir, B. Yuan, Fuzzy Sets and Fuzzy Logic: Theory and Applications, Prentice-Hall, 1995.
[68] D. Koller, A. Levy, A. Pfeffer, P-Classic: a tractable probabilistic description logic, in: Proceedings AAAI-1997, AAAI Press/MIT Press, 1997, pp. 390–397.
[69] S. Kraus, D. Lehmann, M. Magidor, Nonmonotonic reasoning, preferential models and cumulative logics, Artif. Intell. 14 (1) (1990) 167–207.
[70] H.E. Kyburg Jr., The Logical Foundations of Statistical Inference, D. Reidel, Dordrecht, Netherlands, 1974.
[71] H.E. Kyburg Jr., The reference class, Philos. Sci. 50 (1983) 374–397.
[72] D. Lehmann, Another perspective on default reasoning, Ann. Math. Artif. Intell. 15 (1) (1995) 61–82.
[73] H.J. Levesque, R.J. Brachman, Expressiveness and tractability in knowledge representation and reasoning, Comput. Intell. 3 (1987) 78–93.
[74] A.Y. Levy, M.-C. Rousset, Combining Horn rules and description logics in CARIN, Artif. Intell. 104 (1–2) (1998) 165–209.
[75] C. Li, K.C.-C. Chang, I.F. Ilyas, S. Song, RankSQL: query algebra and optimization for relational top-k queries, in: Proceedings ACM SIGMOD-2005, ACM Press, 2005, pp. 131–142.
[76] R. Li, K. Wen, Z. Lu, X. Sun, Z. Wang, An improved semantic search model based on hybrid fuzzy description logic, in: Proceedings of the Workshop on Frontier of Computer Science and Technology (FCST-2006), IEEE Computer Society, 2006, pp. 139–146.
[77] Y. Li, B. Xu, J. Lu, D. Kang, Discrete tableau algorithms for $\mathcal{FSHI}$, in: Proceeedings DL-2006, 2006.
[78] Y. Li, B. Xu, J. Lu, D. Kang, P. Wang, Extended fuzzy description logic $\mathcal{ALCN}$, in: Proceedings KES-2005, volume 3684 of LNCS, Springer, 2005, pp. 896–902.
[79] Y. Li, B. Xu, J. Lu, D. Kang, P. Wang, A family of extended fuzzy description logics, in: Proceedings of the COMPSAC-2005, IEEE Computer Society, 2005, pp. 221–226.
[80] C.-J. Liau, Y.Y. Yao, Information retrieval by possibilistic reasoning, in: Proceedings DEXA-2001, volume 2113 of LNCS, Springer, 2001, pp. 52–61.
[81] O. Liu, Q. Tian, J. Ma, A fuzzy description logic approach to model management in R&D project selection, in: Proceedings PACIS-2004, 2004.
[82] J. Lu, Y. Li, B. Zhou, D. Kang, Y. Zhang, Distributed reasoning with fuzzy description logics, in: Proceedings of the 7th International Conference on Computational Science, 2007, pp. 196–203.
[83] T. Lukasiewicz, Local probabilistic deduction from taxonomic and probabilistic knowledge-bases over conjunctive events, Int. J. Approx. Reason. 21 (1) (1999) 23–61.
[84] T. Lukasiewicz, Probabilistic deduction with conditional constraints over basic events, J. Artif. Intell. Res. 10 (1999) 199–241.
[85] T. Lukasiewicz, Probabilistic logic programming under inheritance with overriding, in: Proceedings UAI-2001, Morgan Kaufmann, 2001, pp. 329–336.
[86] T. Lukasiewicz, Probabilistic logic programming with conditional constraints, ACM Trans. Comput. Log. 2 (3) (2001) 289–339.
[87] T. Lukasiewicz, Probabilistic default reasoning with conditional constraints, Ann. Math. Artif. Intell. 34 (1–3) (2002) 35–88.
[88] T. Lukasiewicz, Weak nonmonotonic probabilistic logics, Artif. Intell. 168 (1–2) (2005) 119–161.
[89] T. Lukasiewicz, Fuzzy description logic programs under the answer set semantics for the Semantic Web, in: Proceedings RuleML-2006, pp. 89–96. IEEE Computer Society, 2006 (Extended version: Fundamenta Informaticae 82 (3) (2008) 289–310).
[90] T. Lukasiewicz, Expressive probabilistic description logics, Artif. Intell. 172 (6–7) (2008) 852–883.
[91] T. Lukasiewicz, U. Straccia, Tightly integrated fuzzy description logic programs under the answer set semantics for the Semantic Web, in: Proceedings RR-2007, volume 4524 of LNCS, Springer, 2007, pp. 289–298.
[92] C. Lutz, Reasoning with concrete domains, in: Proceedings IJCAI-1999, Morgan Kaufmann, 1999, pp. 90–95.
[93] C. Lutz, Description logics with concrete domains—a survey, in: P. Balbiani, N.-Y. Suzuki, F. Wolter, M. Zakharyaschev (Eds.), in: Advances in Modal Logics, vol. 4, King's College Publications, 2003.

[94] T.P. Mailis, G. Stoilos, G.B. Stamou, Expressive reasoning with Horn rules and fuzzy description logics, in: Proceedings RR-2007, volume 4524 of LNCS, Springer, 2007, pp. 43–57.
[95] T. Mantay, R. Möller, A. Kaplunova, Computing probabilistic least common subsumers in description logics, in: Proceedings KI-1999, volume 1701 of LNCS, Springer, 1999, pp. 89–100.
[96] C. Meghini, F. Sebastiani, U. Straccia, A model of multimedia information retrieval, J. ACM 48 (5) (2001) 909–970.
[97] C. Meghini, F. Sebastiani, U. Straccia, C. Thanos, A model of information retrieval based on a terminological logic, in: Proceedings ACM SIGIR-1993, ACM Press, 1993, pp. 298–307.
[98] P. Mitra, N.F. Noy, A. Jaiswal, OMEN: a probabilistic ontology mapping tool, in: Proceedings ISWC-2005, volume 3729 of LNCS, Springer, 2005, pp. 537–547.
[99] T.H. Naeth, Analysis of the average-case behavior of an inference algorithm for probabilistic description logics, Diplomarbeit, TU Hamburg-Harburg, Germany, February 2007.
[100] N.J. Nilsson, Probabilistic logic, Artif. Intell. 28 (1) (1986) 71–88.
[101] H. Nottelmann, N. Fuhr, Adding probabilities and rules to OWL Lite subsets based on probabilistic datalog, Int. J. Uncertain. Fuzz. Knowl.-Based Syst. 14 (1) (2006) 17–42.
[102] V. Novák, Which logic is the real fuzzy logic? Fuzzy Sets Syst. 157 (5) (2005) 635–641.
[103] J.Z. Pan, G. Stamou, G. Stoilos, E. Thomas, Expressive querying over fuzzy DL-Lite ontologies, in: Proceedings DL-2007, 2007.
[104] R. Pan, Z. Ding, Y. Yu, Y. Peng, A Bayesian network approach to ontology mapping, in: Proceedings ISWC-2005, volume 3729 of LNCS, Springer, 2005, pp. 563–577.
[105] B. Parsia, E. Sirin, M. Grove, R. Alford, Pellet (Available at www.mindswap.org/2003/pellet/).
[106] J.L. Pollock, Nomic Probabilities and the Foundations of Induction, Oxford University Press, Oxford, 1990.
[107] M. Pool, 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.
[108] G. Qi, J.Z. Pan, Q. Ji, Extending description logics with uncertainty reasoning in possibilistic logic, in: Proceedings ECSQARU-2007, LNCS, Springer, 2007.
[109] G. Qi, J.Z. Pan, Q. Ji, A possibilistic extension of description logics, in: Proceedings DL-2007, 2007.
[110] H. Reichenbach, Theory of Probability, University of California Press, Berkeley, CA, 1949.
[111] H. Salkin, M. Kamlesh, Foundations of Integer Programming, North-Holland, 1988.
[112] D. Sanchez, A. Tettamanzi, Generalizing quantification in fuzzy description logics, in: Proceedings 8th Fuzzy Days in Dortmund, 2004.
[113] D. Sanchez, A. Tettamanzi, Reasoning and quantification in fuzzy description logics, in: Fuzzy Logic and Applications, volume 3849 of LNCS, Springer, 2005, pp. 81–88.
[114] D. Sanchez, A. Tettamanzi, Fuzzy quantification in fuzzy description logics, in: E. Sanchez (Ed.), Fuzzy Logic and the Semantic Web, Capturing Intelligence, Elsevier, 2006, pp. 135–160.
[115] M. Schmidt-Schauß, G. Smolka, Attributive concept descriptions with complements, Artif. Intell. 48 (1) (1991) 1–26.
[116] B. Smith, Ontology, in: L. Floridi (Ed.), Blackwell Guide to the Philosophy of Computing and Information, Blackwell, Oxford, UK, 2003, pp. 155–166.
[117] G. Stoilos, G. Stamou, J. Pan, V. Tzouvaras, I. Horrocks, Reasoning with very expressive fuzzy description logics. J. Artif. Intell. Res. 30 (2007) 273–320.
[118] G. Stoilos, G. Stamou, J. Z. Pan, D2.5.6: fuzzy reasoning extensions, in: Knowledge Web Technical Report, 2007.
[119] G. Stoilos, G. Stamou, V. Tzouvaras, J. Pan, I. Horrocks, Fuzzy OWL: Uncertainty and the Semantic Web, in: Proceedings of the International Workshop of OWL: Experiences and Directions, 2005.
[120] G. Stoilos, G. Stamou, V. Tzouvaras, J.Z. Pan, I. Horrock, A fuzzy description logic for multimedia knowledge representation, in: Proceedings of the International Workshop on Multimedia and the Semantic Web, 2005.
[121] G. Stoilos, G.B. Stamou, V. Tzouvaras, J.Z. Pan, I. Horrocks, The fuzzy description logic f- $\mathcal{SHIN}$, in: Proceedings URSW-2005, 2005, pp. 67–76.
[122] G. Stoilos, U. Straccia, G. Stamou, J. Pan, General concept inclusions in fuzzy description logics, in: Proceedings ECAI-2006, IOS Press, 2006, pp. 457–461.
[123] U. Straccia, A fuzzy description logic, in: Proceedings AAAI-1998, AAAI Press/MIT Press, 1998, pp. 594–599.
[124] U. Straccia, A framework for the retrieval of multimedia objects based on four-valued fuzzy description logics, in: F. Crestani, G. Pasi (Eds.), Soft Computing in Information Retrieval: Techniques and Applications, Physica (Springer), Heidelberg, Germany, 2000, pp. 332–357.
[125] U. Straccia, Reasoning within fuzzy description logics, J. Artif. Intell. Res. 14 (2001) 137–166.
[126] U. Straccia, Transforming fuzzy description logics into classical description logics, in: Proceedings JELIA-2004, volume 3229 of LNCS, Springer, 2004, pp. 385–399.
[127] U. Straccia, Uncertainty in description logics: a lattice-based approach, in: Proceedings IPMU-2004, 2004, pp. 251–258.
[128] U. Straccia, Description logics with fuzzy concrete domains, in: Proceedings UAI-2005, AUAI Press, 2005, pp. 559–567.
[129] 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.
[130] U. Straccia, Fuzzy $\mathcal{ALC}$ with fuzzy concrete domains, in: Proceeedings DL-2005, 2005, pp. 96–103.
[131] U. Straccia, Towards a fuzzy description logic for the Semantic Web (preliminary report), in: Proceedings ESWC-2005, volume 3532 of LNCS, Springer, 2005, pp. 167–181.
[132] U. Straccia, Answering vague queries in fuzzy DL-Lite, in: Proceedings IPMU-2006, 2006, pp. 2238–2245.
[133] U. Straccia, Description logics over lattices, Int. J. Uncertain., Fuzz. Knowl.-Based Syst. 14 (1) (2006) 1–16.
[134] U. Straccia, A fuzzy description logic for the Semantic Web, in: E. Sanchez (Ed.), Fuzzy Logic and the Semantic Web, Capturing Intelligence, Elsevier, 2006, pp. 73–90 (Chapter 4).
[135] U. Straccia, Fuzzy description logic programs, in: Proceedings IPMU-2006, 2006, pp. 1818–1825.
[136] U. Straccia, Towards top-k query answering in deductive databases, in: Proceedings IEEE SMC-2006, IEEE Computer Society, 2006.
[137] U. Straccia, Towards top-k query answering in description logics: the case of DL-Lite, in: Proceedings of the JELIA-2006, volume 4160 of LNCS, Springer, 2006, pp. 439–451.
[138] U. Straccia, Uncertainty and description logic programs over lattices, in: E. Sanchez (Ed.), Fuzzy Logic and the Semantic Web, Capturing Intelligence, Elsevier, 2006, pp. 115–133 (Chapter 7).
[139] U. Straccia, Reasoning in *l*- $\mathcal{SHIF}$: an expressive fuzzy description logic under łukasiewicz semantics, Technical Report TR-2007-10-18, Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, Italy, 2007.
[140] U. Straccia, F. Bobillo, Mixed integer programming, general concept inclusions and fuzzy description logics, in: Proceedings EUSFLAT-2007, vol. 2, University of Ostrava, Ostrava, Czech Republic, pp. 213–220, 2007.
[141] U. Straccia, G. Visco, DLMedia: an ontology mediated multimedia information retrieval system, in: Proceeedings DL-2007, 2007.
[142] C. Thomas, A. Sheth, On the expressiveness of the languages for the Semantic Web—making a case for "A little more", in: E. Sanchez (Ed.), Fuzzy Logic and the Semantic Web, Capturing Intelligence, 2006, pp. 3–20.
[143] S. Tobies, Complexity Results and Practical Algorithms for Logics in Knowledge Representation, PhD thesis, RWTH Aachen, Germany, 2001.
[144] C. Tresp, R. Molitor, A description logic for vague knowledge, in: Proceedings ECAI-1998, J. Wiley & Sons, 1998, pp. 361–365.
[145] D. Tsarkov, I. Horrocks, FaCT++ description logic reasoner: system description, in: Proceedings IJCAR-2006, volume 4130 of LNCS, Springer, 2006, pp. 292–297.
[146] O. Udrea, Y. Deng, E. Hung, V.S. Subrahmanian, Probabilistic ontologies and relational databases, in: Proceedings CoopIS/DOA/ODBASE-2005, volume 3760 of LNCS, Springer, 2005, pp. 1–17.
[147] O. Udrea, V.S. Subrahmanian, Z. Majkic, Probabilistic RDF, in: Proceedings IRI-2006, IEEE Systems, Man, and Cybernetics Society, 2006, pp. 172–177.
[148] M. van Keulen, A. de Keijzer, W. Alink, A probabilistic XML approach to data integration, in: Proceedings ICDE-2005, 2005, pp. 459–470.
[149] T. Venetis, G. Stoilos, G. Stamou, S. Kollias, f-DLPs: extending description logic programs with fuzzy sets and fuzzy logic, in: Proceedings FUZZ-IEEE-2007, 2007.
[150] W3C, OWL web ontology language overview, 2004, W3C Recommendation (10 February 2004) (Available at www.w3.org/TR/2004/REC-owl-features-20040210/).
[151] G. Weikum, J. Graupmann, R. Schenkel, M. Theobald, Towards a statistically Semantic Web, in: Proceedings ER-2004, volume 3288 of LNCS, Springer, 2004, pp. 3–17.
[152] Y. Yang, J. Calmet, OntoBayes: an ontology-driven uncertainty model, in: Proceedings IAWTIC-2005, IEEE Press, 2005, pp. 457–463.
[153] P.M. Yelland, An alternative combination of Bayesian networks and description logics, in: Proceedings KR-2000, Morgan Kaufmann, 2000, pp. 225–234.
[154] J. Yen, Generalizing term subsumption languages to fuzzy logic, in: Proceedings IJCAI-1991, Morgan Kaufmann, 1991, pp. 472–477.
[155] L.A. Zadeh, Fuzzy sets, Inform. Control 8 (3) (1965) 338–353.
[156] L. Zhang, Y. Yu, J. Zhou, C. Lin, Y. Yang, An enhanced model for searching in semantic portals, in: Proceedings WWW-2005, ACM Press, 2005, pp. 453–462.