TMRF e-Book
Advances in Semantic Computing
(Eds. Joshi, Boley & Akerkar), Vol. 2, pp 1 – 22, 2010

Chapter 1

Uncertainty and Rule Extensions to Description Logics and Semantic Web Ontologies

Jidi Zhao[1]

Abstract The The Semantic Web is an evolving extension of the World Wide Web in which the semantics of the information and resources available on the Web are formally described, making it more machine-interpretable. In the layered architecture of the Semantic Web, the Web ontology language (OWL), built upon the logic-based knowledge formalism Description Logics (DLs), is a W3C recommendation for the ontology layer; the Rule Interchange Format (RIF), built upon the formalism Logic Programs (LP), is a W3C recommendation for the rule layer.
Uncertainty is an intrinsic feature of real-world knowledge and refers to a form of de_ciency or imperfection in the information: the truth of such information is not crisply established, yet can be rigorously formalized. In the last decade, one of the key research directions in the Semantic Web community has been to handle uncertainty, as evidenced by W3Cs Uncertainty Reasoning for the World Wide Web Incubator Group. At the same time, in order to enrich the knowledge representation capabilities of ontologies based on traditional DLs, considerable research e_orts have also been directed towards the integration of ontologies and rules.
In this chapter, we _rst give a comprehensive overview of existing approaches in uncertainty extensions to DLs and OWL based on various uncertainty treatments, mainly including Fuzzy Logic and Probability Theory. We summarize and classify previous work based on (a) the generalized classical description logics, (b) the supported forms and allowed constructors of uncertain knowledge, (c) the underlying fuzzy logics or probabilistic semantics, and (d) their inference problems and reasoning algorithms. Second, we summarize the differences between the two subsets of _rst-order logic formalism, Description Logics and Horn Logic. On top of this, we then review a number of theoretical proposals for integrating ontologies with rules, as well as some practical implementations. By and large, existing approaches for rule extensions to DLs and ontologies can be classi_ed into two categories: the so-called homogeneous approach and the so-called hybrid approach. We explain several existing work as representatives of the homogenous approach, as well as some work following the hybrid approach.

1 DESCRIPTION LOGICS WITH UNCERTAINTY

The famous British author, mathematician, and philosopher, Bertrand Russell, once said, “Everything is vague to a degree you do not realize till you have tried to make it precise.” Uncertainty, which refers to a form of deficiency or imperfection in the information for which the truth is not established definitely [Lakshmanan and Shiri, 2001)], is an intrinsic feature of real-world knowledge.

[1]Faculty of Computer Science, University of New Brunswick, Fredericton, Canada
E-mail: Judy.Zhao AT unb.ca

As shown in (Bacchus, 1990; Lakshmanan and Shiri, 2001; Motro and Smets, 1997; Parsons, 1996), many real-world applications need the capability to handle uncertainty and the modeling of uncertainty and reasoning with it have been challenging issues for over two decades in databases, operations research, and artificial intelligence.

Handling uncertain knowledge is inevitably also a challenge for the Semantic Web and Description Logic community. The need to model and reason with uncertainty has been found in many different Semantic Web contexts, such as matchmaking in Web services (Martin-Recuerda and Robertson, 2005), classification of genes in bioinformatics (Stevens et al., 2007), multimedia annotation (Stamou et al., 2006), and ontology learning (Haase and Völker, 2005). Triggered by such necessities in numerous Web-based applications, W3C founded the Uncertainty Reasoning for the World Wide Web (URW3) Incubator Group (Laskey et al., 2008), which addressed challenges and defined methodologies of reasoning with, and representing uncertain information available through, the World Wide Web and the Semantic Web. According to the latest URW3 draft report, uncertainty is a term intended to include different forms of incomplete knowledge, such as inconclusiveness, vagueness, ambiguity, and others (Laskey et al., 2008).

An OWL ontology is actually a knowledge base composed of a finite set of axioms in Description Logics (DLs). Such a knowledge base can be divided into two components, namely the TBox and the ABox (Baader et al., 2003). The TBox consists of concept inclusions ($C \sqsubseteq D$), concept definitions ($C \equiv D$), role inclusions ($R \sqsubseteq P$), and role definitions ($R \equiv P$), while the ABox consists of concept assertions ($C(a)$) and role assertions ($R(a, b)$), where C, D are concepts, R, P are roles, and a, b are individuals. Intuitively, the TBox refers to the schema of the ontology, whereas the ABox to the instances.

DLs (Baader et al., 2003) are a family of logic-based knowledge representation formalisms designed to represent and reason about the conceptual knowledge of arbitrary domains. Elementary descriptions of a DL are atomic concepts (also called classes) and atomic roles (also called properties or relations). Complex concept descriptions and role descriptions can be built from the elementary constructors according to construction rules.

The basic propositionally closed DL is $\mathcal{ALC}$ in which the letters $\mathcal{AL}$ stand for attributive language and the letter $\mathcal{C}$ for complement (negation of arbitrary concepts). Concepts in $\mathcal{ALC}$ are constructed using boolean operators (conjunction, disjunction, negation) plus restricted quantifiers and atomic roles, as shown in table 1. For example, a complex concept description for "the set of persons all of whose children are doctor or have a child who is a doctor" in $\mathcal{ALC}$ can be written as:

$$Person \sqcap \forall hasChild.(Doctor \sqcup \exists hasChild.Doctor)$$

Table 1: A set of concept constructors for a simple description logic

Constructor Name	**Syntax**
conjunction	$C \sqcap D$
disjunction	$C \sqcup D$
negation	$\neg C$
exists restriction	$\exists R.C$
value restriction	$\forall R.C$

Researchers in the Description Logic community have done a lot of work on more expressive Description Logics. These different Description Logics are distinguished by the kind of concept and role constructors allowed in the Description Logic and the kinds of axiom allowed in the

TBox. Besides $\mathcal{ALC}$, other letters indicate various DL extensions including the following:

- $\mathcal{S}$ often used for ALC extended with transitive roles (R^+).
- $\mathcal{H}$ for role hierarchy.
- $\mathcal{O}$ for nominals (e.g.,$\{Mary, John\}$).
- $\mathcal{I}$ for inverse roles.
- $\mathcal{N}$ for number restrictions.
- $\mathcal{Q}$ for qualified number restrictions (e.g., $\geq$2hasChild.Doctor).
- $\mathcal{R}$ for limited complex role inclusion axioms, role disjointness.
- $\mathcal{F}$ for functional properties.

OWL has three increasingly-expressive sublanguages: OWL-Lite, OWL-DL, and OWL-Full. The OWL-Lite sublanguage is a syntactic variant of the Description Logic $\mathcal{SHIF}(\mathcal{D})$ where (D) means data values or data types, while OWL-DL is almost equivalent to the $\mathcal{SHOIN}(\mathcal{D})$ DL. OWL-Full is the union of OWL syntax and RDF, and known to be undecidable mainly because it does not impose restrictions on the use of transitive properties. Accordingly, an OWL-Lite ontology corresponds to a $\mathcal{SHIF}(\mathcal{D})$ knowledge base, and an OWL-DL ontology corresponds to a $\mathcal{SHOIN}(\mathcal{D})$ knowledge base.

However, the standard DLs and DL based Web languages are limited to dealing with crisp concepts and cannot express uncertain knowledge. Therefore, it has become one of the key research directions for the Semantic Web to extend standard DLs in order to support uncertain knowledge representation and reasoning.

Considerable work has been carried out in integrating uncertain knowledge into DLs in the last decade. The Uncertainty Reasoning for the World Wide Web (URW3) Incubator Group (Laskey et al., 2008) identified several methodologies for dealing with uncertainty, such as, probability, Fuzzy Sets, Belief Functions, Random Sets, and Rough Sets. However, the current literature generally follows only two of them. One is Probabilistic Logic based on Probability Theory, for example the work in (Jaeger, 1994; Koller et al., 1997; Lukasiewicz, 2008), and the other is Fuzzy Logic and Fuzzy Sets, for example the work in (Yen, 1991; Tresp and Molitor, 1998; Straccia, 2001). Although both approaches assign numerical values to concept inclusion axioms and/or instance assertions in a knowledge base, their semantics is quite different from each other. Probabilistic theory refers to a proposition that is either true or false, but due to a lack of information we do not know for certain which one is the case. It represents the probability with which a proposition is true. For example, John can be assumed to be a student with the probability 0.6 and a teacher with the probability 0.4. On the other hand, Fuzzy Logic deal with the vagueness and imprecision of a proposition by using fuzzy predicates, fuzzy quantifiers, or other constructs, which means the proposition itself is only true to a certain degree. For example, John, measuring 1.85m, might be said to be tall with the degree of truth 0.9. Therefore, which approach should be selected really depends on the domain and the requirements of the application.

In each of the following subsections, we overview previous work based on (a) the generalized classical description logics, (b) the supported forms and allowed constructors of uncertain knowledge, (c) the underlying fuzzy logics or probabilistic semantics, and (d) their inference problems and reasoning algorithms. In addition, we also talk a bit about some other approaches in dealing with uncertain knowledge.

1.1 Fuzzy Description Logics

In the following, we will first go over some preliminary knowledge about Fuzzy Set Theory and Fuzzy Logic, introduce the Fuzzy DLs by Yen (1991), Tresp and Molitor (1998), and Straccia (2001), and then describe several other fuzzy extensions to more expressive DLs.

Fuzzy Set Theory was first introduced by Zadeh (1965) as an extension to the classical notion of a set to capture the inherent vagueness (the lack of crisp boundaries of sets). Fuzzy Logic is a form of multi-valued logic derived from Fuzzy Set Theory to deal with reasoning that is approximate rather than precise. In Fuzzy Logic, the degree of truth of a statement can range between 0 and 1, and is not constrained to the two truth values $\{0, 1\}$ or $\{false, true\}$ as in classical predicate logic. Formally, a fuzzy set X with respect to a set of elements Ω (also called a universe) is characterized by a membership function $\mu(x)$ which assigns a value in the real unit interval [0,1] to each element x in X ($x \in X$), notated as $X : \Omega \rightarrow [0, 1]$. $\mu(x)$ gives us a degree of an element x belonging to a set X. Such degrees can be computed based on some specific membership function. A fuzzy relation R over two fuzzy sets X_1 and X_2 is defined by a function $R : \Omega \times \Omega \rightarrow [0, 1]$.

Fuzzy Logic extends the Boolean operations defined on crisp sets and relations in the context of fuzzy sets and fuzzy relations. These operations, such as complement, union, and intersection, are interpreted as mathematical functions over the unit interval [0,1]. The mathematical functions for fuzzy intersection are usually called t-norms ($t(\eta, \theta)$); those for fuzzy union are called s-norms ($s(\eta, \theta)$); those for the fuzzy set complement are called negations ($\neg\eta$); and those for implication between fuzzy sets are called fuzzy implication ($\eta \Rightarrow \theta$); here η, θ define the truth degrees of sets and relations and can range between 0 and 1. These functions usually satisfy certain mathematical properties. Based on different operations, the Fuzzy Logic family includes several so-called t-norm systems, among which, the most widely known are Zadeh Logic, Lukasiewicz Logic, Product Logic, and Gödel Logic. Table 2 summarizes different operations in these fuzzy logics.

Table 2: Fuzzy Operations

	Zadeh Logic	**Lukasiewicz Logic**	**Product Logic**	**Gödel Logic**
t-norm ($t(\eta, \theta)$)	$\min(\eta, \theta)$	$\max(\eta + \theta - 1, 0)$	$\eta \cdot \theta$	$\min(\alpha, \beta)$
s-norm ($s(\eta, \theta)$)	$\max(\eta, \theta)$	$\min(\eta + \theta, 1)$	$\eta + \theta - \eta \cdot \theta$	$\max(\eta, \beta)$
negation ($\neg\eta$)	$1 - \eta$	$1 - \eta$	if η=0 then 1 else 0	if η=0 then 1 else 0
implication ($\eta \Rightarrow \theta$)	$\max(1 - \eta, \theta)$	$\min(1 - \eta + \theta, 1)$	if $\eta \leq \theta$ then 1 else θ/η	if $\eta \leq \theta$ then 1 else θ

Concepts in classical DLs are interpreted as crisp sets. In particular, an individual belonging to a set is either true or false. However, many concepts in real-life are vague in the sense that they do not have precisely defined membership criteria. For instance, the concept of a tall person. It makes sense to say that an individual belongs to the concept *TallPerson* only to a truth degree, which is a real number in the real unit interval [0,1] and depends on the height of the individual. The main idea underlying all fuzzy DLs is to use fuzzy sets and Fuzzy Logic for defining the semantics of the fuzzy extensions of classical DLs. All the interpretations assign fuzzy sets to concepts and roles, i.e., an atomic concept A is interpreted by membership functions of the form $A^I : \triangle^I \rightarrow [0,1]$, and a simple role R by membership functions of the form $R^I : \triangle^I \times \triangle^I \rightarrow [0,1]$, where $\triangle$ is the domain being modeled.

Yen (1991) is the first who combines fuzzy logic with term subsumption languages and proposes a fuzzy extension to a very restricted DL, called $\mathcal{FTSL}^-$. As defined in (Brachman and Levesque, 1984), the corresponding standard DL $\mathcal{FL}^-$ is actually a sublanguage of $\mathcal{ALC}$, and only allows primitive concepts, primitive roles, defined concepts formed from concept intersection, value restriction and existential qualification. Knowledge base in $\mathcal{FTSL}^-$ includes only fuzzy terminological knowledge in the form of $C \sqsubseteq D$, where C and D are two fuzzy concepts. A fuzzy concept here can be either a *primitive concept* defined by a membership function, or a *defined concept* constructed from fuzzy concept intersection and soft value restriction. Yen's approach generalizes the semantics of classical term subsumption languages based on a test score semantics in Fuzzy Logic, namely,

(1) fuzzy concept intersection. The degree an individual x satisfying an intersection of concepts C and D can be computed using the *minimum* operator,

$\mu_{C \sqcap D}(x) = min\{\mu_C(x), \mu_D(x)\}$

(2) soft value restriction. The degree to which a value restriction $\forall R.C$ is satisfied by an individual x is determined by the degree to which the implication is true for a,

$\mu_{\forall R.C}(x) = \inf\{\mu_{R^I(x,y)}(x) \Rightarrow \mu_{C(y)} | y \in \triangle^I\}$

In this formula, $\forall R.C$ corresponds to the First Order Logic formula $\forall x.(R(x,y) \Rightarrow C(y))$, and various fuzzy implication operators can be used to define $\mu_{R^I(x,y)}(x) \Rightarrow \mu_{C(y)}$.

The inference problem Yen is interested in is testing subsumption relationship between fuzzy concepts. A concept D subsumes a concept C if and only if D is a fuzzy superset of C, i.e., given two concepts C, D defined in the fuzzy DL, $C \sqsubseteq D$ is viewed as $\forall x.C(x) \leq D(x)$. Thus, the subsumption relationship itself is a crisp yes/no test.

Yen (1991) also describes a structural subsumption algorithm for $\mathcal{FTSL}^-$, which returns true if C and D are both defined by membership functions and their membership functions satisfy $\forall x.C(x) \leq D(x)$; returns false if only one of the two concepts are defined using membership functions; or in the case of C and D being defined concepts, normalizes the descriptions by recursively replacing defined concepts by their definition and then performs structural comparison on each sub-expression. Such an algorithm is almost identical to the subsumption algorithm for $\mathcal{FL}^-$ DL by Brachman and Levesque (1984).

Tresp and Molitor (1998) consider a more general extension of $\mathcal{ALC}$ to many-valued logics, called $\mathcal{ALC}_{\mathcal{FM}}$. The language $\mathcal{ALC}_{\mathcal{FM}}$ allows constructors including conjunction, disjunction, manipulator, value restriction, and existential qualification in the definition of complex concepts. Its syntax and semantics are summarized in Table 3.

Table 3: Syntax and Semantics of $\mathcal{ALC}_{\mathcal{FM}}$

Constructor	**Syntax**	**Semantics**
concept conjunction	$C \sqcap D$	$(C \sqcap D)^I = \min(C^I(x), D^I(x))$
concept disjunction	$C \sqcup D$	$(C \sqcup D)^I = \max(C^I(x), D^I(x))$
exists restriction	$\exists R.C$	$(\exists R.C)^I(x) = \sup_{y \in \Delta^I}\{\min(R^I(x,y), C^I(y))\}$
value restriction	$\forall R.C$	$(\forall R.C)^I(x) = \inf_{y \in \Delta^I}\{\max(\neg R^I(x,y), C^I(y))\}$
manipulator	$M_i C$	$M_i(C^I(x))$

Note that, different from Yen's work, this work defines the semantics of a value restriction as $(\forall R.C)^I(a) = \inf\{\max(1 - R^I(x,y), C^I(y)|y \in \triangle^I)\}$, since $\forall R.C$ corresponds to the First Order Logic formula $\forall x.(\neg R(x,y) \vee C(y))$. This work also starts addressing the issue of a fuzzy semantics of modifiers M, which are unary operators that can be applied to concepts, such as *mostly*, *more or less*, and *very*. An example is *(very)TallPerson(John)*, which means that "John is a very tall person".

In both of the work by Yen (1991) and Tresp and Molitor (1998), knowledge bases include only fuzzy terminological knowledge. But different from Yen's work, Tresp and Molitor are interested in determining fuzzy subsumption between fuzzy concepts, i.e., given concepts C, D, they want to know to which degree C is a subset of D. Such a problem is reduced to the problem of determining an adequate evaluation for an extended ABox which corresponds to a solution for a system of inequations. The degree of subsumption between concepts is then determined as the minimum of all values obtained for some specific variable. Tresp and Molitor (1998) present a sound and complete reasoning algorithm for $\mathcal{ALC}_{\mathcal{FM}}$ which basically is an extension of each completion rule in classical tableau algorithm for standard ALC. Such an algorithm generates systems of inequations by applying the completion rules, and, after adopting some treatment of the manipulators, min and max operators to remove them from the inequations, uses linear programming methods to solve the inequations and to determine the minimum value.

Another fuzzy extension of $\mathcal{ALC}$ is due to Straccia (2001). In this work, the interpretation of the Boolean operators and the quantifiers is also based on Zadeh logic which defines conjunction as minimum, disjunction as maximum, negation as $\lambda x.(1 - x)$, universal quantifier as infimum, and existential quantifier as supremum. The work by Yen (1991) and Tresp and Molitor (1998) includes fuzzy terminological knowledge, but no fuzzy assertional knowledge, while Straccia's work (Straccia, 2001) allows for both fuzzy terminological and fuzzy assertional knowledge. That is, the ABox assertions are equipped with a degree from [0,1]. Thus in this context, one may also wants to find out to which degree other assertions follow from the ABox, which is called a fuzzy entailment problem, $KB \models \langle \alpha \geq n \rangle$ or $KB \models \langle \alpha \leq n \rangle$). A decision algorithm for such fuzzy entailment problem in this fuzzy extension of $\mathcal{ALC}$ is presented based on completion rules and is reduced to the unsatisfiability problem of a set of fuzzy constraints. For example, $KB \models \langle \alpha \geq n \rangle$ if and only if $KB \cup \{\langle \alpha < n \rangle\}$ is not satisfiable. Similar to Yen, Straccia (2001) is interested in crisp subsumption of fuzzy concepts with the result being a crisp yes or no instead of a fuzzy subsumption relationship. He further reduces the subsumption problem to the fuzzy entailment problem: $C \sqsubseteq D$ is true, if and only if for all $n > 0$, we have $\langle C(a) \geq n \rangle \models \langle D(a) \geq n \rangle$, where a is a new individual and n is a truth degree in [0,1].

In recently years, more research on fuzzy Description Logics has been put forward. The underlying idea of a later work by Straccia in (Straccia, 2006b) is essentially the same as (Straccia, 2001), except that now the set of possible truth values is a complete lattice rather than [0, 1]. Although Straccia (2005) addresses the syntax and semantics for more expressive fuzzy DLs under Zadeh logic, no reasoning algorithm for the fuzzy subsumption between fuzzy concepts is given in his work. Sánchez and Tettamanzi (2006) consider modifiers in a fuzzy extension of the description logic $\mathcal{ALCQ}$ under Zadeh logic, but the knowledge base in their work only consists of the TBox. They also present an algorithm which calculates the satisfiability interval for a fuzzy concept in fuzzy $\mathcal{ALCQ}$. Stoilos et al. (2007) focuses on a fuzzy extensions of an expressive classical DL, $\mathcal{SHIN}$, which improves the knowledge expressiveness of $\mathcal{ALC}$ by allowing constructors including number restriction, inverse role, transitive role, and role hierarchy, but again, its semantics is still only based on Zadeh logic.

Table 4: Comparison of fuzzy Description Logics

	Classical DL	**Fuzzy Terminology**	**Fuzzy Assertion**	**Concept Subsumption**	**Fuzzy Instance Entailment**	**Fuzzy Logics**	**Fuzzy Modifiers**	**Reasoning Algorithm**
Yen (1991)	$\mathcal{FL}^-$	Yes	No	Crisp	No	Zadeh	No	Structural Comparison
Tresp and Molitor (1998)	$\mathcal{ALC}$	Yes	No	Fuzzy	No	Zadeh	Yes	Tableau
Straccia (2001)	$\mathcal{ALC}$	Yes	Yes	Crisp	Yes	Zadeh	No	Tableau
Straccia (2005)	$\mathcal{SHOIN}$	Yes	Yes	Fuzzy	Yes	Zadeh	No	No
Hájek (2005)	$\mathcal{ALC}$	No	Yes	No	Yes	arbitrary continuous t-norms	No	Tableau
Sánchez and Tettamanzi (2006)	$\mathcal{ALCQ}$	Yes	No	No	Yes	Zadeh	Yes	Tableau
Stoilos et al. (2007)	$\mathcal{SHIN}$	Yes	Yes	Crisp	Yes	Zadeh	No	Tableau
Bobillo and Straccia (2007)	$\mathcal{ALC}$(D)	Yes	Yes	Fuzzy	Yes	product	No	Tableau
Zhao et al. (2009)	$\mathcal{ALCHIN}$	Yes	Yes	Fuzzy	Yes	Zadeh, Lukasiewicz, product, Gödel	No	Tableau

Bobillo and Straccia (2007) consider concrete domains and provide an algorithm for fuzzy $\mathcal{ALC}(\mathcal{D})$ under product semantics. The first work considering a fuzzy DL under arbitrary t-norms is due to Hájek (Hájek, 2005, 2006) with the underlying DL language as $\mathcal{ALC}$. These works in particular provide algorithms for deciding whether $C \sqsubseteq D \geq 1$ is satisfiable. Zhao et al. (2009) recently propose an interval-based, norm-parameterized fuzzy Description Logic, $f\mathcal{ALCHIN}$, based on various fuzzy logics as shown in Table 2, and present an algorithm for consistency checking of knowledge bases in the proposed language.

Table 4 summarizes various fuzzy DLs we discussed above, comparing their underlying classical DLs, their support of uncertain knowledge and inference problems, their underlying Fuzzy Logics, and the availability of reasoning algorithms.

Another interesting inference problem for fuzzy description logics is *top-k retrieval* for conjunctive query answering. In fuzzy description logics, a tuple may satisfy a query to a degree instead of giving classical *true* or *false* answers. Therefore, given a conjunctive query over a fuzzy description logic knowledge base, it is of interest to compute only the top-k answers. While this problem has been well researched in relational databases, little work is known for the case of fuzzy DL knowledge bases, with the exception of (Straccia, 2006a), which deals with the problem of finding the top-k results over knowledge bases in a fuzzy generalization of DL-Lite (Calvanese et al., 2005).

There is also some work on translating fuzzy DLs into classical DLs (Lu et al., 2005, 2007; Straccia, 2004; Li et al., 2005). The main idea underlying such an approach is to translate a fuzzy assertion into a crisp assertion. For example, the assertion "a is an instance of C to degree at least n" of the form $C(a) \geq n$ is translated into $C_n(a)$. Furthermore, such an approach use a concept inclusion axiom of the form $C_{n1} \sqsubseteq C_{n2}$ (n1>n2) is used to relate different C_ns, with the intended meaning to encode that whenever an individual is an instance of C to degree at least n1, then it is also an instance of C to degree at least n2. The advantage of such an approach is that, in this way, classical DL reasoners can be directly used to do reasoning on the fuzzy KBs. However, the translation can become quadratic in the size of the fuzzy knowledge base, leaving alone the presence of fuzzy modifiers and fuzzy concrete domains. For large KBs, such a translation is unrealistic and the performance becomes extremely slow.

1.2 Probabilistic Description Logics

The probabilistic approach is based on classical probability theory and Probabilistic Logics. There has been quite some work proposed to extend various Description Logics with probability.

Some works extend the syntax of the description language with some degrees and the corresponding semantics with probability theory. They embed probabilistic information as part of the knowledge base. For example, Dürig and Studer (D́urig and Studer, 2005, 2008) present a 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. A probabilistic interpretation $\mathcal{I}$ consists of the domain Δ^I and an interpretation function $\cdot^I$ which maps language elements to some probabilistic value in the unit interval [0,1]. The semantics of concept descriptions are defined in a straightforward way. For instance, with the independence assumption, the semantics of concept conjunction is defined as $(C \sqcap D)^I(a) = C^I(a) \cdot D^I(a)$, and the semantics of concept disjunction is defined as $(C \sqcup D)^I(a) = C^I(a) + D^I(a) - C^I(a) \cdot D^I(a)$.

Some works keep both the syntax and semantics of the classical part of Probabilistic Description Logics the same as in the standard DLs. However, a new language constructor, namely the *conditional constraint* constructor, is added to handle uncertainties. The majority of work in Probabilistic Description Logics falls into this approach. The syntax of these probabilistic description logics uses the notion of a *conditional constraint* from (Lukasiewicz, 1999) to express probabilistic knowledge which encodes interval restrictions for conditional probabilities over concepts. We assume a finite nonempty set $\mathcal{C}$ of *basic classification concepts*, which are atomic concepts and concept descriptions in classical DLs. We can informally look at these basic classification concepts as the relevant DL concepts for defining probabilistic relationships. The set of *classification concepts* is then inductively defined as follows. Every basic classification concept $C \in \mathcal{C}$ is a classification concept. If C and D are classification concepts, then $\neg C$, $C \sqcap D$ and $C \sqcup D$ are also classification concepts. A *conditional constraint* is an expression of the form $(D|C)$ $[l, u]$ which informally encodes that the probability of D given C lies between l and u, where C and D are classification concepts, and l and u are reals from [0, 1]. Note that, although some frameworks such as (Jaeger, 2004, 1994) express probabilistic terminological axiom using exact probability, $(D|C) = u$, this is simply a special case of interval probability, since it can be expressed as $(D|C)\ [u, u]$.

One of the earliest works on probabilistic description logics is due to Heinsohn (1994), 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. However, this work does not allow for assertional (classical or probabilistic) knowledge about concept and role instances and only deals with knowledge bases with a simple TBox.

Koller et al. (1997) present the probabilistic description logic P-Classic, which is a probabilistic generalization of the description logic Classic. Similar to Heinsohn's work (Heinsohn, 1994), it allows for encoding terminological probabilistic knowledge about concepts, roles, and attributes, but it does not support assertional (classical or probabilistic) knowledge about instances of concepts and roles.

Another early approach to probabilistic description logics is due to Jaeger (1994), 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.

Jaeger's recent work (Jaeger, 2004) 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 in this paper is different from the semantics of all the other probabilistic description logics, 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.

A probabilistic generalization of the description logics $\mathcal{SHIF}(\mathcal{D})$ has been introduced in (Giugno and Lukasiewicz, 2002). Lukasiewicz recently introduced a probabilistic generalization of $\mathcal{SHOIN}(\mathcal{D})$ towards sophisticated formalisms for reasoning under probabilistic uncertainty in the Semantic Web (Lukasiewicz, 2008). The semantics of probabilistic description logics in (Giugno and Lukasiewicz, 2002; Lukasiewicz, 2008) is based on the notion of lexicographic entailment in *probabilistic default reasoning* (Lukasiewicz, 2001, 2002), which is a probabilistic generalization of the notion of lexicographic entailment by Lehmann (1995) in *default reasoning* from conditional knowledge bases. Due to this semantics, these DLs allow 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 as statistical knowledge about concepts and roles, and assertional probabilistic knowledge as degrees of belief about instances of concepts and roles. Every probabilistic knowledge base consists of (i) a PTBox, which has a classical DL knowledge base and a finite set of conditional constraints; and (ii) a collection of PABoxes, each of which consists of a finite set of conditional constraints and encodes probabilistic assertional knowledge about an individual. Thus, a probabilistic knowledge base extends a classical knowledge base $\mathcal{T}$ by probabilistic terminological knowledge and probabilistic assertional knowledge.

The reasoning procedures for probabilistic DLs depend on how the probabilistic information is represented in the knowledge base. Unlike fuzzy DLs, none of the existing probabilistic DLs extends tableau-based reasoning procedures from standard DLs. These probabilistic DLs relies heavily on existing algorithms developed for probabilistic reasoning. In Heinsohn's work (Heinsohn, 1994), the main reasoning problems are deciding the consistency of probabilistic terminological knowledge bases and computing logically entailed tight probability intervals. Heinsohn (1994) 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. In Jaeger's work (Jaeger, 1994), 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. Thus, the main reasoning problems of terminological probabilistic consistency and inference are solved by linear programming, while the reasoning problems of assertional probabilistic consistency and inference are solved by an *approximation algorithm*. The main reasoning problems proposed by (Giugno and Lukasiewicz, 2002; Lukasiewicz, 2008) include consistency and lexicographic entailment for probabilistic knowledge bases, which are based on the notions of consistency and lexicographic entailment, respectively, in *probabilistic default reasoning* (Lukasiewicz, 2001, 2002).

In addition, the probabilistic semantics in Koller et al.s' work (Koller et al., 1997) is based on a reduction to Bayesian networks. The knowledge base consists of a set of probabilistic classes (p-classes), where each p-class represents the probabilistic information related to a certain class of individuals. The p-classes are represented using a Bayesian network (Pearl, 1988). 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. Therefore, the inference algorithms developed for Bayesian networks can be directly applied. As an important feature of P-Classic, if the underlying Bayesian network is a polytree, the reasoning problem can be solved in polynomial time. The work by Yelland (2000) proposes a probabilistic extension of a description logic close to $\mathcal{FL}$, whose probabilistic semantics is also based on a reduction to Bayesian networks. The approach allows for encoding terminological probabilistic knowledge about concepts and roles, but it does not support assertional knowledge about instances of concepts and roles. Like in Koller et al.'s work (Koller et al., 1997), the main reasoning problem of determining the exact probabilities for conditionals between concepts is also solved by a reduction to inference in Bayesian networks.

Some other works focus especially on probabilistic generalizations of Web ontology languages. Pool and Aikin (2004) provide a method for representing uncertainty in OWL ontologies, while Fukushige (2004) proposes a basic framework for representing probabilistic relationships in RDF. Nottelmann and Fuhr (2006) 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. (2006), 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.

Special interest has been put on combining the Web ontology language OWL with probabilistic formalisms based on Bayesian networks. For example, Costa and Laskey (2006) suggest a probabilistic generalization of OWL, called PR-OWL, whose probabilistic semantics is based on multi-entity Bayesian networks (MEBNs). Roughly speaking, PR-OWL represents knowledge as parameterized fragments of Bayesian networks.

Ding et al. (2006) 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 preserves the semantics of the original ontology and is consistent with all the given probability constraints.

Yang and Calmet (2005) also 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. An application in risk analysis for insurance and natural disaster management is addressed in their work.

1.3 Other Approaches

There is also some other work on uncertainty management in DLs, in particular possibilistic DLs (Hollunder, 1995; Stamou et al., 2006) since possibility theory is used as the underlying theory. A possibilistic axiom in possibilistic DLs has the form $P\alpha \geq l$ or $N\alpha \geq l$, where α is a classical description logic axiom, and l is a real number from [0, 1]. To define the semantics of a possibilistic DL using possibility theory, the most basic notion is the possibility interpretation. Let Φ denote the set of all classical description logic interpretations. A possibilistic interpretation is a mapping $\Pi : \Phi \rightarrow [0, 1]$. Given an event or formula α, the possibility measure is defined as: $Poss(\alpha) = \max\{\Pi(I)|I \in \Phi, I \models \alpha\}$ and gives the degree to which event α is possible. Furthermore, the necessity measure of event α is defined as $Nec(\alpha) = 1 - Poss(\neg\alpha)$ and characterizes the extent to which an event is necessary or certain to occur. A possibilistic interpretation Π satisfies a possibilistic axiom $P\alpha \geq l$ (respectively, $N\alpha \geq l$), or is a model of it, denoted $\Pi \models P\alpha \geq l$ (respectively, $\Pi \models N\alpha \geq l$), iff $Poss(\alpha) \geq l$ (respectively, $Nec(\alpha) \geq l$).

Hsueh-Ieng (2008) proposes a framework called $\mathcal{ALC}u$, which extends the standard DL $\mathcal{ALC}$ with uncertainty. In her framework, each axiom and each assertion is bounded with three parameters: a number, a conjuction function and a disjunction function. For example, a general concept inclusion (GCI) axiom takes the form $\langle C \sqsubseteq D|\alpha, f_c, f_d\rangle$ where C and D are concept descriptions, α is a real number or an interval in [0,1] refering to the certainty that the axiom holds, f_c is the conjunction function used as the semantics of concept conjunction and part of the role exists restriction, and f_d is the disjunction function used as the semantics of concept disjunction and part of the role value restriction. The expressiveness of this framework is limited by its underlying classical DL $\mathcal{ALC}$. Moreover, this framework can only support certain forms of uncertainty due to its conjunction function and disjunction function, and its tableau-based algorithm. In general, probabilistic reasoning requires extra information about the events, their relationships, and the facts in the world, for example, positive/negative correlation and conditional probability. Furthermore, none of the existing probabilistic frameworks follows tableau-based reasoning procedures from standard DLs for probabilistic reasoning. The $\mathcal{ALC}u$ framework cannot support uncertainty reasoning that is not based on tableau algorithm.

With a few exceptions such as Pronto (Klinov, 2008) and FuzzyDL (Bobillo and Straccia, 2008), little work has gone beyond the phase of theoretical exploration into actual implementations. In addition, FuzzyDL is currently a fuzzy extension only to the DL language $\mathcal{SHIF}$ and has its own syntax other than the wide-spread OWL ontology language.

2 COMBINATION OF DESCRIPTION LOGICS AND LOGIC PROGRAMS

Another important research direction for the Semantic Web is to integrate ontologies with rules. Description Logics (DLs) and Logic Programs (LP) are the two main categories of logic-oriented knowledge representation formalisms for the Semantic Web, which cover different but overlapping areas of knowledge representation. Figure 1 from (Grosof et al., 2003) shows the relationship between DL and LP. To make it first-order, here we limit the expressiveness of LP to Horn Logic (Makowsky, 2002), and often to its function-free (Datalog) subset.

Although DLs and Horn Logic are both based on subsets of first-order logic (FOL) (Koller et al., 1997), there are some noticeable differences between them:

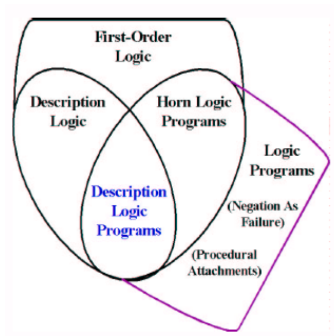

Figure 1: Relationship between DL and LP

- DL cannot represent more than one free variable at a time (Grosof et al., 2003). A Horn Logic rule involving multiple variables cannot be expressed in DLs. For example, FriendshipBetween(?X,?Y) ← Person(?X),Person(?Y).
- DLs cannot directly support n-ary predicates. However, these can be naturally expressed using Horn Logic.
- Traditional expressive DLs support transitive role axioms but they cannot derive values of properties. For example, Father^+ can represent a transitive role. However, in traditional DLs (e.g., the DL language underlying OWL 1.0), Grandfather(?X,?Z) ← Father(?X,?Y), Father(?Y,?Z) cannot be expressed, neither can uncleOf(?X,?Z) ← brotherOf(?X,?Y), parentOf(?Y,?Z).
- DLs cannot describe classes whose instances are related to an anonymous individual via different properties. For example, a person works and lives at the same place implies that he or she is a homeworker. HomeWorker(?X) ← Work(?X, ?Y), Live(?X, ?Z), Loc(?Y,?W), Loc(?Z,?W) (Grosof et al., 2003).
- Horn Logic cannot support negation and does not allow role exists restrictions and disjunctions in the head of a rule (Grosof et al., 2003), which can be easily expressed in DLs. Furthermore, Horn Logic is unable to assert the existence of individuals whose identity might not be known (Grosof et al., 2003). For instance, all persons have a gender, whether the gender is known or unknown. However, this can be easily expressed in DLs : $Person \sqsubseteq hasGender.\top$.
- DLs use the open world assumption (OWA) (Patel-Schneider and Horrocks, 2006), under which a KB is assumed to be incomplete and new facts are allowed to be gradually added into the KB. While Horn Logic uses the closed world assumption (CWA), which assumes that the information that is not currently known to be true in a KB is false. For instance, suppose the only assertion (the term used in DLs) or fact (the term used in Horn Logic) in the KB is that John is a man. If we ask whether Pal is a man in DLs, the answer would be unknown. However, in Horn logic, the answer is false.

- DLs define the domain as a set of objects and relationships between them. In DLs, due to OWA, there can be many models, each of which describes one possible interpretation of the domain. Horn Logic also models the domain as objects and relationships between them. However, due to CWA, it assumes that the only model that can interpret the domain consists of objects and relationships that are explicitly represented in the knowledge base (Patel-Schneider and Horrocks, 2006).

- Different Horn Logic formalisms, including Answer Set Programming (ASP), typically employ a Unique Name Assumption (UNA), i.e., different ground terms denote different objects. However, this does not comply with the view in DLs where no such assumption is made.

In addition, Negation As Failure (NAF) is the traditional operator for inferring negative knowledge from incomplete information, and is peculiar of Logic Programs. As shown in Figure 1, NAF is beyond Horn Logic. On the other hand, DLs do not allow NAF and usually adopt classical negation. For example, consider the following LP knowledge base:

$person(x) \longleftarrow author(x)$

$nonAuthor(x) \longleftarrow not\ author(x)$

$person(John)$

Base on NAF, we can conclude that *nonAuthor(John)* satisfies. However, from the DL counterpart of the LP knowledge,

$Author \sqsubseteq Person$

$\neg Author \sqsubseteq NonAuthor$

$Person(John)$

we cannot draw the conclusion *NonAuthor(John)*.

Therefore, the extension of terminological concepts defined in a DL theory by means of rules defined in a LP theory has nowadays become an important research direction in order to enrich knowledge representation capabilities of traditional DLs and thus the ontology languages such as OWL for the Semantic Web. Figure 2 shows that the Semantic Web consists of several hierarchical layers where each layer is built on top of the others (Berners-Lee et al., 2001). In this layered architecture, the Web ontology language (OWL), built upon the logic-based knowledge formalism Description Logics (DLs), is a W3C recommendation for the ontology layer; the Rule Interchange Format (RIF), built upon the formalism Logic Programs (LP), is a W3C recommendation for the rule layer. Currently, both the ontology layer and the rule layer have received abundant research and reached a sufficient level of maturity. The next step is to move onto the unifying logic layer, representing the combination of the ontology in DLs and rules in LP.

However, as shown in (Levy and Rousset, 1998), the naive combination of even a very simple DL with an arbitrary Horn Logic program is undecidable. Thus the largest obstacle towards combining Description Logics and Logic Programs stems from the issue of decidability. As we will explain later, most of the hybrid approaches indeed provide a notion of safety as a key tool for ensuring decidability.

By and large, the combination of DLs and LP can be classified into two approaches: the so-called homogeneous approach and the so-called hybrid approaches (Mei et al., 2007). In the homogeneous approach, a unique knowledge representation formalism is adopted for both DLs and LP, while different knowledge representation formalisms are used in the hybrid approach. In the following sub sections, we will explain several works following these two approaches.

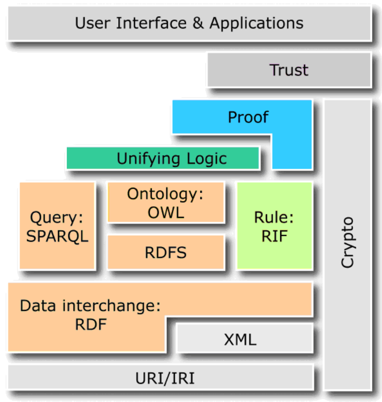

Figure 2: Semantic Web Hierarchical Architecture

2.1 Homogeneous Approach

One formalism that follows the homogeneous approach is the Semantic Web Rules Language (SWRL) (Horrocks et al., 2004). SWRL is an ontology language that integrates OWL with a rule layer built on top of it. SWRL's goal of enhancing description logics with rules is aimed at overcoming some known expressive limitation in ontology languages, which can be easily fixed by adding rules to the ontology. However, the addition of rules is also the main cause of reasoning in SWRL being in general undecidable, that is, there is no algorithm that can compute in finite time whether, for example, an concept assertion is entailed by an ontology in SWRL. Modern DL reasoners such as KAON2 (http://kaon2.semanticweb.org/), Pellet (http://clarkparsia.com/pellet/) (Sirin et al., 2007) and Racer-Pro (Haarslev and Möller, 2001) now increasingly support SWRL or SWRL-like syntax. Their way of dealing with undecidability is to add some kind of restrictions on rules defined in SWRL. Such restrictions are sufficient to make SWRL rules decidable. For example, Pellet applies DL Safety restriction on rules, which is indeed a simple idea to regain decidability: Variables in DL Safe rules bind only to explicitly named individuals in the ontology knowledge base.

Description Logic Programs (DLP) by Grosof et al. (2003) is another work along with the homogeneous approach. DLP, in contrast to SWRL, restricts the syntax of the supported OWL DL fragment to those axioms expressible in Horn rules in order to keep the decidability. From the knowledge expressiveness point of view, DLP falls into the intersection of DL and LP, which is slightly less expressive than $\mathcal{ALC}$ in DL terminology.

OWL 2 is the latest W3C recommendation of the Web Ontology Language. In order to achieve efficiency and adapt itself to different application scenarios, OWL 2 specifies three independent profiles: OWL 2 EL, OWL 2 QL, and OWL 2 RL (Motik et al., 2009). Among these profiles, OWL 2 RL is specifically aimed at applications that require scalable reasoning without sacrificing too much expressive power. The RL acronym reflects the fact that reasoning in this profile can be implemented using standard rule languages and rule-based reasoning engines. Typical ontology reasoning tasks, such as ontology consistency, concept satisfiability, concept subsumption, instance checking, and conjunctive query answering, can thus be solved in time that is polynomial with respect to the size of the ontology. OWL 2 RL is mainly inspired by (DLP) (Grosof et al., 2003) and is defined by placing restrictions on the structure of OWL 2 ontologies.

Quite some practical work has used rule engines and rule systems to support the reasoning tasks of ontology in Description Logics and description logic languages. For example, Jena (http://jena.sourceforge.net/), Sesame (http://www.openrdf.org/), Oracle 11g (http://www.oracle.com/technology/tech/semantic-technologies/), SWI-Prolog (http://www.swi-prolog.org/), and FLORA-2 (http://flora.sourceforge.net/). The main underlying idea of these work is to transform and map the Semantic data in RDF, RDFS or fragments of OWL into rules and facts in rule languages and then perform the reasoning. A recent work by Meditskos and Bassiliades from the *Handbook of Research on Emerging Rule-based Languages and Technologies* gives an overview of these rule systems and rule languages and describes their support for ontology languages including RDF, RDFS, and OWL. We would like to point the readers interested in this topic to their work (Meditskos and Bassiliades, 2009) for in-depth details.

2.2 Hybrid Approach

Opposed to the homogeneous approach, the so-called hybrid approach concentrates on the possibility of combining rule sets under traditional logic programming semantics with a Description Logics knowledge base.

First, we define a *hybrid knowledge base* $\mathcal{KB} =< \mathcal{DL}, \mathcal{LP} >$ as the combination of a ontology knowledge base expressed in a Description Logic language and a rule knowledge base expressed in a Logic Program language.

The hybrid approach includes several categories. In this section, we only review two of them: the loose coupling and the tight integration. Roughly speaking, in loose coupling frameworks, the rule knowledge base $\mathcal{LP}$ and the ontology knowledge base $\mathcal{DL}$ are treated as separate and independent components. An interface mechanism is then defined to fulfill the exchange of knowledge between the two components. The decidability of the hybrid knowledge base $\mathcal{KB} =< \mathcal{DL}, \mathcal{LP} >$ is then guaranteed by the particular design of the interfacing mechanism and the restriction of data flow between the two components. In some cases, only unidirectional data flow is allowed, e.g., the rule knowledge base can import data from the ontology knowledge base, but not vice versa. Such loose coupling frameworks are particularly suitable for practical integration on top of existing reasoners for the two sides. Representatives of loose coupling frameworks include $\mathcal{AL}$+*log* (Donini et al., 1998), dl-programs (Eiter et al., 2008), $Datalog^{DL}$ (Mei et al., 2007), and defeasible logic coupled with description logic bases (Rosati, 2006b).

$\mathcal{AL}$+*log* by Donini et al. (1998) is a hybrid integration of Datalog and the Description Logic ALC. The DL queries in the bodies of the $\mathcal{AL}$+*log* rules are restricted to concept assertion queries and concept satisfaction queries; role assertion queries are not allowed. Moreover, the variables of the DL queries in the body of an $\mathcal{AL}$+*log* rule must also appear in the non-DL atoms of the body or in the head. Therefore the DL queries are in fact adding constraints on the variables. Furthermore, AL-queries in $\mathcal{AL}$+*log* are conjunctions of atomic formulae built with non-DL predicates. Query answering in $\mathcal{AL}$+*log* is decidable. The query answering algorithm described in (Donini et al., 1998) first applies backward chaining based on SLD resolution to construct DL queries for a given AL-query, and then uses an ALC tableau reasoner to check whether each DL query is entailed by the DL component of the $\mathcal{AL}$+*log* knowledge base.

Here we use an example from (Donini et al., 1998) to illustrate AL query answering in $\mathcal{AL}$+*log*. Consider the knowledge base defined in table 5, where the two columns refer to the DL component (also called structural component in (Donini et al., 1998)) and the LP component (also called relational component) respectively, and the two cells in the second row refer to the ABox of the DL component and the facts of the LP component respectively. We denote DL predicates and variables by uppercase names, and denote non-DL predicates and constants

Table 5: An Example Knowledge Base in $\mathcal{AL}$+*log*

DL component	**LP component**
$FP \sqsubseteq FM$;	$curr(X, Z) \longleftarrow$
$NFP \equiv FP \sqcap \neg\exists TC.CO$;	$exam(X, Y), subject(Y, Z), ST(X), CO(Y), TP(Z)$; (R1)
$AC \sqcup BC \equiv CO$;	$mayDoThesis(X, Y) \longleftarrow$
	$curr(X, Z), expert(Y, Z), ST(X), TP(Z), (FM \sqcap \exists TC.AC)(Y)$; (R2)
$AC \sqcap BC \equiv \bot$	$mayDoThesis(X, Y) \longleftarrow$
	$ST(X), NFP(Y)$ (R3)
$FP(john)$;	$exam(paul, ai)$;
$TC(john, ai)$;	$subject(ai, kr)$;
$(FP \sqcap \forall TC.AC)(mary)$;	$subject(ai, lp)$;
$ST(paul)$; $AC(ai)$;	$expert(john, kr)$;
$TP(kr)$; $TP(lp)$	$expert(mary, lp)$

by lowercase names. The symbols in the DL component should be read as follows: FP=Full Professor, FM=Faculty Member, NFP=Non-teaching Full Professor, AC=Advanced Course, BC=Basic Course, TC=Teaching, CO=Course, ST=Student, TP=Topic.

Next we explain how the query $mayDoThesis(paul, mary)$ would be answered by the combination of LP reasoning and DL reasoning as described above. Based on backward chaining, an LP reasoning engine derives the following deductions.

By applying R2, we have
$curr(paul, Z), expert(mary, Z), ST(paul), TP(Z), (FM \sqcap \exists TC.AC)(mary)$.
Further apply R1, we get
$exam(paul, Y), subject(Y, Z), ST(paul), CO(Y), TP(Z), expert(mary, Z), (FM \sqcap \exists TC.AC)(mary)$.
Based on the fact $exam(paul, ai)$, we have
$subject(ai, Z), ST(paul), CO(ai), TP(Z), expert(mary, Z), (FM \sqcap \exists TC.AC)(mary)$.
Then with the fact $subject(ai, lp)$ and $expert(mary, lp)$, we get
$ST(paul), CO(ai), TP(lp), (FM \sqcap \exists TC.AC)(mary)$.

On the next step, a DL reasoner would be able to prove that all these DL queries are true. Therefore, the AL query $mayDoThesis(paul, mary)$ is true.

Compared to those loose coupling frameworks, formalisms categorized under the tight integration category keep the vocabularies of the DL predicates and the LP predicates distinct as well, but integrate the rule component $\mathcal{LP}$ with the ontology component $\mathcal{DL}$ to a larger extent from the point view of semantics. A tightly integration framework is built on the notion of an integrated model which satisfies both the rule component and the ontology component. An *integrated model* can be often seen as $\mathcal{M} = (\mathcal{M}_{\mathcal{DL}}, \mathcal{M}_{\mathcal{LP}})$, that is, it is composed of two separate models $\mathcal{M}_{\mathcal{DL}}$ and $\mathcal{M}_{\mathcal{LP}}$ sharing the same domain where $\mathcal{M}_{\mathcal{DL}}$ satisfies the corresponding Description Logic theory and $\mathcal{M}_{\mathcal{LP}}$ satisfies the corresponding Logic Program theory.

A representative work following this category is $\mathcal{DL}$+*log* by Rosati (2006a), a general framework for the integration of Description Logics and disjunctive Datalog. $\mathcal{DL}$+*log* has the following specifications:

- Rule predicates and DL predicates are distinct in the knowledge base.
- The interpretation of constants is based on the *standard names assumption*. That is, every first-order interpretation is over the same fixed, countably infinite domain Δ, and

furthermore, each constant in the constants set has a one-to-one correspondence with Δ in every interpretation (this property is also known as the unique name assumption).

- The integrated models for $\mathcal{DL}$+*log* are called nonmonotonic models (NM-models). A NM-model of the knowledge base $\mathcal{KB} =< \mathcal{DL}, \mathcal{LP} >$ has the form of $\mathcal{I}_{\mathcal{DL}} \cup \mathcal{I}_{\mathcal{LP}}$, where $\mathcal{I}_{\mathcal{DL}}$ is a model of the DL predicates, $\mathcal{I}_{\mathcal{LP}}$ of the rules predicates after deleting those DL atoms satisfied by $\mathcal{I}_{\mathcal{DL}}$ in $\mathcal{LP}$.

- $\mathcal{DL}$+*log* allows weak negation applied to rules predicates, and DL predicates appearing in the head of rules.

- $\mathcal{DL}$+*log* realizes a tighter form of interaction between DL knowledge base and LP knowledge base, through a *weak safeness* condition which weakens *DL-safeness* of variables in Datalog rules. That is, each variable in a rule must occur in some positive body atom of the rule, and this atom must have a rule predicate if the variable occurs in an atom with DL predicates in the head of the rule.

The reasoning algorithm in $\mathcal{DL}$+*log* consists of three main steps. Such three steps are applied repeatedly until no more interpretation of the ontology component can be found.

1. First, classical tableau-based DL reasoning algorithm inferences an interpretation $\mathcal{I}_{\mathcal{DL}}$ of the ontology component $\mathcal{DL}$.

2. Second, applying the DL-grounding of $\mathcal{LP}$ with respect to the interpretation $\mathcal{I}_{\mathcal{DL}}$. More specifically, (1) deleting all DL atoms which are satisfied in $\mathcal{I}_{\mathcal{DL}}$ and appear in the bodies of the rules in $\mathcal{LP}$; (2) deleting all DL atoms that are not satisfied in $\mathcal{I}_{\mathcal{DL}}$ and appear in the heads of the rules in $\mathcal{LP}$; (3) deleting all rules which have falsified body and/or true head. The resulting $\mathcal{LP}$ thus does not contain any DL predicates.

3. Third, deduct a stable model of $\mathcal{LP}$.

$\mathcal{DL}$+*log* is decidable, given containment between union of conjunctive queries is decidable in the ontology component. Here we illustrate the grounding step in the above reasoning algorithm with an example, where we denote DL predicates and variables by uppercase names, and denote rule predicates and constants by lowercase names. Consider the following knowledge base $\mathcal{KB} =< \mathcal{DL}, \mathcal{LP} >$:

$$\begin{aligned}\mathcal{DL} = \{&MultiLingual \sqsubseteq \neg MonoLingual,\\ &MultiLingual \sqcup MonoLingual \sqsubseteq Author,\\ &Author \sqsubseteq \exists IsAuthorOf,\\ &Author(john)\}\end{aligned}$$

$$\begin{aligned}\mathcal{LP} = \{&novelist(X) \vee scientist(X) \longleftarrow writer(X),\\ &MonoLingual(X) \longleftarrow novelist(X),\\ &MultiLingual(X) \longleftarrow scientist(X),\\ &scientist(X) \longleftarrow writer(X), IsAuthorOf(X,Y), notlikes(X, astrology),\\ &writer(john)\}\end{aligned}$$

Now we have a DL interpretation $\mathcal{I}_{\mathcal{DL}} = \{Author(john), MultiLingual(john)\}$, then we ground the rule component into

$$\begin{aligned}\mathcal{LP}_{\mathcal{I}_{\mathcal{DL}}} = \{&novelist(john) \vee scientist(john) \longleftarrow writer(john),\\ &\longleftarrow novelist(john),\\ &scientist(john) \longleftarrow writer(john), notlikes(john, astrology),\\ &writer(john)\}\end{aligned}$$

In the third step, we can find the interpretation $\mathcal{I}_{\mathcal{LP}} = \{writer(john), scientist(john)\}$ is a stable model. Therefore, we conclude $\{Author(john), MultiLingual(john), writer(john), scientist(john)\}$ is a NM-model for the knowledge base.

3 CONCLUSION

Ongoing Semantic Web applications have revealed a shortcoming of standard Description Logics, i.e., their limited knowledge expressivity. In order to enrich the knowledge representation capabilities of ontologies based on traditional DLs, considerable research efforts in Semantic Web and Description Logic communities have been conducted mainly in two directions. One is uncertainty extension to DLs and ontologies in order to handle uncertain knowledge, the other is rule extension to DLs and ontologies.

In this chapter, we have focused on addressing these two issues and review the most important achievements in these two areas, not only from a theoretical point of view, but also from practical aspects.

Fuzzy Logic and Probability Theory are the two main methodologies to deal with uncertainty. After discussing the differences between these two methodologies, we have explained existing approaches in uncertainty extensions to DLs and OWL following Fuzzy Logic and those approaches following Probability Theory respectively. Previous work has been reviewed and classified based on the generalized classical description logics of these approaches, the supported forms and allowed constructors of uncertain knowledge, the underlying semantics, and their inference problems and reasoning algorithms.

Description Logics (DLs) and Logic Programs (LP) are the two main categories of knowledge representation formalisms for the Semantic Web, which cover different but overlapping areas of knowledge representation. We have addressed the differences between DLs and LP, and then reviewed a number of theoretical proposals for extending terminological concepts defined in a DL theory by means of rules defined in a LP theory, such as the Semantic Web Rules Language (SWRL), Description Logic Programs (DLP), OWL 2 RL, $\mathcal{AL}$+*log*, and $\mathcal{DL}$+*log*. We have also gone over some practical implementations of integrating ontologies with rules.

Some possible issues for future research have arisen from existing work and achievements reviewed in this chapter. From the aspect of each uncertainty treatment methodology in Description Logics, including probabilistic description logic, fuzzy description logic, and several others, it would be interesting to extend existing work to support more expressive DL fragments, for example, the OWL DL counterpart $\mathcal{SHOIN}$. It would also be interesting to study a uniform framework supporting the representation and reasoning of different kinds of uncertainty, probably via a tight integration of the above uncertainty treatment methodologies. Based on the combination of Description Logics and Logic Programs, another important issue would be integrate probabilistic description logic, fuzzy description logic, other uncertainty extended DLs, or a uniform uncertainty DL framework with various uncertainty rules in Logic Programs. Furthermore, from practical aspects, it would be of equal importance to conduct research on the realization of the prior-mentioned theoretical work to support web-based uncertain knowledge representation and reasoning, and meet the requirements in Semantic Web applications.

References

Baader, F., D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider (2003). *The Description Logic Handbook: Theory, Implementation and Applications.* Cambridge, MA: Cambridge University Press.

Bacchus, F. (1990). *Representing and Reasoning with Probabilistic Knowledge: A Logical Approach to Probabilities*. Cambridge, Massachusetts: MIT Press.

Berners-Lee, T., J. Hendler, and O. Lassila (2001). The semantic web. *Scientific American 284*(5), 34–44.

Bobillo, F. and U. Straccia (2007). A fuzzy description logic with product t-norm. In *Proceedings of the IEEE International Conference on Fuzzy Systems (Fuzz IEEE-07)*, pp. 652C–657. IEEE Computer Society.

Bobillo, F. and U. Straccia (2008). fuzzydl: An expressive fuzzy description logic reasoner. In *Proceedings of the 2008 International Conference on Fuzzy Systems (FUZZ-08)*.

Brachman, R. J. and H. J. Levesque (1984). The tractability of subsumption in frame-based description languages. In *Proceedings AAAI-1984*, pp. 34–37. AAAI Press.

Calvanese, D., G. D. Giacomo, D. Lembo, M. Lenzerini, and R. Rosati (2005). Dl-lite: Tractable description logics for ontologies. pp. 602–607.

Costa, P. C. G. and K. B. Laskey (2006). Pr-owl a framework for probabilistic ontologies. *Frontiers in Artificial Intelligence and Applications 150*, 237–249.

Ding, Z., Y. Peng, and R. Pan (2006). *BayesOWL: Uncertainty Modeling in Semantic Web Ontologies*, pp. 3–29. Studies in Fuzziness and Soft Computing. Springer Berlin / Heidelberg.

Donini, F. M., M. Lenzerini, D. Nardi, and A. Schaerf (1998). Al-log: Integrating datalog and description logics. *Journal of Intelligent Information Systems 10*(3), 227–252.

D́urig, M. and T. Studer (2005). Probabilistic abox reasoning: Preliminary results. In *Proceedings of the International Workshop on Description Logics (DL05)*, pp. 104–111.

D́urig, M. and T. Studer (2008). Palc: Extending alc aboxes with probabilities. http://www.iam.unibe.ch/ tstuder/papers/duerigstuder.pdf.

Eiter, T., G. Ianni, T. Krennwallner, and A. Polleres (2008). Rules and ontologies for the semantic web. pp. 1–53.

Fukushige, Y. (2004, October). Representing probabilistic knowledge in the Semantic Web. In *W3C Workshop on Semantic Web for Life Sciences*, Cambridge, MA, USA.

Giugno, R. and T. Lukasiewicz (2002). P- shoq(d): A probabilistic extension of shoq(d) for probabilistic ontologies in the semanticweb. In *Proceedings JELIA-2002, volume 2424 of LNCS*, pp. 86–97. Springer.

Grosof, B. N., I. Horrocks, R. Volz, and S. Decker (2003). Description logic programs: Combining logic programs with description logic. In *Proceedings of the 12th international conference on World Wide Web*, pp. 48–57.

Haarslev, V. and R. Möller (2001). Racer system description. In *Proceedings of the First International Joint Conference on Automated Reasoning (IJCAR 2001), volume 2083 of Lecture Notes in Computer Science (LNCS)*, pp. 701–705. Springer Verlag.

Haase, P. and J. Völker (2005). Ontology learning and reasoning - dealing with uncertainty and inconsistency. In *Proceedings of Uncertainty Reasoning for the Semantic Web*, pp. 45–55.

Hájek, P. (2005). Making fuzzy description logics more expressive. *Fuzzy Sets Syst. 154*(1), 1–15.

Hájek, P. (2006). *What does mathematical fuzzy logic offer to description logic?*, pp. 91–100. Fuzzy Logic and the Semantic Web, Capturing Intelligence. Elsevier.

Heinsohn, J. (1994). Probabilistic description logics. In *Proceedings UAI-1994*, pp. 311–318.

Hollunder, B. (1995). An alternative proof method for possibilistic logic and its application to terminological logics. *International Journal of Approximate Reasoning 12*(2), 85–109.

Horrocks, I., P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean (2004, May). Swrl: A semantic web rule language combining owl and ruleml. http://www.w3.org/Submission/SWRL/.

Hsueh-Ieng, P. (2008, April). *Uncertainty Management for Description Logic-Based Ontologies*. Ph. D. thesis, Concordia University.

Jaeger, M. (1994). Probabilistic reasoning in terminological logics. In *Proc. of the 4th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR94)*, pp. 305–316.

Jaeger, M. (2004). Probabilistic role models and the guarded fragment. In *Proceedings IPMU-2004 (Extended version in Int. J. Uncertain., Fuzz. Knowl.-Based Syst. 14 (1) (2006) 43C60).*, pp. 235–242.

Klinov, P. (2008). Pronto: A non-monotonic probabilistic description logic reasoner. In *The Semantic Web: Research and Applications*, pp. 822–826.

Koller, D., A. Levy, and A. Pfeffer (1997). P-classic: A tractable probabilistic description logic. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97)*, pp. 390–397.

Lakshmanan, L. and N. Shiri (2001). Logic programming and deductive databases with uncertainty: A survey. *Enclyclopedia of Computer Science and Technology 45*, 153–176.

Laskey, K. J., K. B. Laskey, P. C. G. Costa, M. M. Kokar, T. Martin, and T. Lukasiewicz (05 March, 2008). W3c incubator group report. Technical Report http://www.w3.org/2005/Incubator/urw3/wiki/DraftFinalReport, W3C.

Lehmann, D. (1995). Another perspective on default reasoning. *Annals of Mathematics and Artificial Intelligence 15*, 61.

Levy, A. Y. and M. C. Rousset (1998). Combining horn rules and description logics in carin. *Artificial Intelligence 104*(1-2), 165–209.

Li, Y., B. Xu, J. Lu, D. Kang, and P. Wang (2005). Extended fuzzy description logic alcn. In *Proceedings KES-2005, volume 3684 of LNCS*, pp. 896–902. Springer.

Lu, J., Y. Li, B. Zhou, D. Kang, and Y. Zhang (2007). Distributed reasoning with fuzzy description logics. In *Proceedings of the 7th International Conference on Computational Science*, pp. 196–203.

Lu, J., B. Xu, Y. Li, and D. Kang (2005). A family of extended fuzzy description logics. In *Proceedings of the COMPSAC-2005*, pp. 221–226. IEEE Computer Society.

Lukasiewicz, T. (1999). Probabilistic deduction with conditional constraints over basic events. *Journal of Artificial Intelligence Research 10*, 199–241.

Lukasiewicz, T. (2001). Probabilistic logic programming under inheritance with overriding. In *Proceedings UAI-2001*, pp. 329–336.

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

Lukasiewicz, T. (2008). Fuzzy description logic programs under the answer set semantics for the semantic web. *Fundamenta Informaticae 82*(3), 289–310.

Makowsky, J. (2002). *Theory of Horn clauses - Encyclopaedia of Mathematics*. Springer-Verlag.

Martin-Recuerda, F. and D. Robertson (2005). Discovery and uncertainty in semantic web services. In *Proceedings of Uncertainty Reasoning for the Semantic Web*, pp. 188.

Meditskos, G. and N. Bassiliades (2009). *Rule-based OWL Ontology Reasoning Systems: Implementations, Strengths and Weaknesses*, Chapter VI, pp. 124–147. IGI Global.

Mei, J., H. Boley, J. Li, and V. C. Bharvsar (2007). The datalogdl combination of deduction rules and description logics. *Computational Intelligence 23*(3), 356–372.

Motik, B., B. C. Grau, I. Horrocks, Z. Wu, A. Fokoue, and C. Lutz (2009, October). Owl 2 web ontology language profiles. http://www.w3.org/TR/owl2-profiles/.

Motro, A. and P. Smets (Eds.) (1997). *Uncertainty Management in Information Systems: From Needs to Solutions*. Boston: Springer.

Nottelmann, H. and N. Fuhr (2006). Adding probabilities and rules to owl lite subsets based on probabilistic datalog. *Int. J. Uncertain. Fuzz. Knowl.-Based Syst. 14*(1), 17–42.

Parsons, S. (1996). Current approaches to handling imperfect information in data and knowledge bases. *IEEE Transactions on Knowledge and Data Engineering 8*(3), 353–372.

Patel-Schneider, P. F. and I. Horrocks (2006). A comparison of two modelling paradigms in the semantic web. *Proceeding of the Fifteenth International World Wide Web Conference (WWW 2006)*, 3–12.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA, USA: Morgan Kaufmann.

Pool, M. and J. Aikin (2004). Keeper and protégé: an elicitation environment for bayesian inference tools. In *Proceedings of the Workshop on Protégé and Reasoning*, pp. 1–2.

Rosati, R. (2006a). Dl+log: Tight integration of description logics and disjunctive datalog. In *KR-06*, pp. 68–78.

Rosati, R. (2006b). *Integrating Ontologies and Rules: Semantic and Computational Issues*, pp. 128–151. Reasoning Web, Second International Summer School 2006, volume 4126 of LNCS. Lisbon, Portugal: Springer.

Sánchez, D. and A. G. Tettamanzi (2006). *Fuzzy quantification in fuzzy description logics*, pp. 135–160. Fuzzy Logic and the Semantic Web, Capturing Intelligence. Elsevier.

Sirin, E., B. Parsia, B. C. Grau, A. Kalyanpur, and Y. Katz (2007). Pellet: A practical owl-dl reasoner. *Web Semantics: Science, Services and Agents on the World Wide Web 5*(2), 51–53.

Stamou, G., J. van Ossenbruggen, J. Z. Pan, and G. Schreiber (2006). Multimedia annotations on the semantic web. *IEEE MultiMedia 13*, 86–90.

Stevens, R., M. E. Aranguren, K. Wolstencroft, U. Sattlera, N. Drummond, M. Horridge, and A. Rectora (2007). Using owl to model biological knowledge. *International Journal of Human-Computer Studies 65*(7), 583–594.

Stoilos, G., G. Stamou, J. Z. Pan, V. Tzouvaras, and I. Horrocks (2007). Reasoning with very expressive fuzzy description logics. *Journal of Artificial Intelligence Research 30*, 273–320.

Straccia, U. (2001). Reasoning within fuzzy description logics. *Journal of Artificial Intelligence Research 14*, 137–166.

Straccia, U. (2004). Transforming fuzzy description logics into classical description logics. Volume 3229, Springer-Verlag, pp. 385–399.

Straccia, U. (2005). Towards a fuzzy description logic for the semantic web (preliminary report). In *2nd European Semantic Web Conference (ESWC-05)*, Lecture Notes in Computer Science, pp. 167–181. Springer Verlag.

Straccia, U. (2006a). Answering vague queries in fuzzy dl-lite. In *Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, (IPMU-06)*, pp. 2238–2245.

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

Tresp, C. B. and R. Molitor (1998). A description logic for vague knowledge. In *Proc. of the 13th Eur. Conf. on Artificial Intelligence (ECAI'98)*, pp. 361–365.

Udrea, O., V. Subrahmanian, and Z. Majkić (2006). Probabilistic rdf. *Proceedings IRI-2006, IEEE Systems, Man, and Cybernetics Society*, 172–177.

Yang, Y. and J. Calmet (2005). Ontobayes: An ontology-driven uncertainty model. In *Computational Intelligence for Modelling, Control and Automation*, pp. 457–463.

Yelland, P. M. (2000). An alternative combination of bayesian networks and description logics. In *Proceedings KR-2000*, pp. 225–234.

Yen, J. (1991). Generalizing term subsumption languages to fuzzy logic. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI'91)*, pp. 472–477.

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

Zhao, J., H. Boley, and W. Du (2009, September 16C19). Knowledge representation and consistency checking in a norm-parameterized fuzzy description logic. In *2009 International Conference on Intelligent Computing (ICIC 2009), LNAI 5755*, pp. 111–123.