Reasoning with Annotated Description Logic Ontologies

an Stelle einer Habilitationsschrift

vorgelegt an der
Technischen Universität Dresden
Fakultät Informatik

eingereicht von
Dr. rer. nat. Rafael Peñaloza Nyssen

Betreuender Hochschullehrer: Prof. Dr.-Ing. Franz Baader

Dresden, October 2015

Contents

1 Introduction

One fundamental requirement for the development of artificial intelligence and intelligent applications if a formalism for representing and reasoning about the knowledge that is relevant for the application. To ensure correctness of the systems dealing with this knowledge and the transference of knowledge between applications, it is important that the knowledge representation formalisms have clear and well-understood semantics that are free from any ambiguity.

Description Logics (DLs) [BCM$^+$07] have arisen as logic-based knowledge representation (KR) formalisms tailored towards the representation of the conceptual knowledge of the application domain. In DLs, the knowledge is typically encoded in a knowledge base or *ontology* that stores knowledge about the relationships between concepts (terminological knowledge) as well as information about some specific individuals (assertional knowledge). These KR formalisms have been successfully used for representing the knowledge of many real-life application domains, most notably within the biomedical fields. For instance, SNOMED is an ontology for the standard use of medical and clinical terms containing approximately half million axioms written in an inexpressive DL [Spa00]. Perhaps the largest success of DLs to-date is the recommendation by the W3C of the DL-based Web Ontology Language (OWL 2) and its profiles as the standard representation languages for the Semantic Web [W3C09].

The rising popularity of DLs has led to more and larger ontologies being written using these languages. This has in turn had the consequence of showcasing the limitations of formalisms based on classical logic for representing the relevant knowledge of many domains. In fact, it is not difficult to encounter concepts that are intrinsically *vague*, and cannot be defined in any precise manner, and the knowledge provided by domain experts often contains a degree of *uncertainty*, and can contradict other experts. Moreover, while classical DLs treat all axioms from an ontology in the same way, some applications need to distinguish additional properties of these axioms. For instance, some axioms may have an *access restriction*, making them visible to only a few users; if an ontology is built from the combination of different sources, a user may *prefer* to observe the consequences from one source, or *trust* only those consequences that are backed up by a set of different sources; alternatively, some combinations of axioms might have been identified to be harmful during a *debugging* process, and need to be avoided.

To try to handle these and many other situations beyond classical logic, a plethora of formalisms extending classical DLs have been introduced over the years. While they differ greatly in their semantics and other properties, many of these formalisms share a common syntactic backbone: ontologies are built from classical DL ontologies by adding an annotation to each axiom. The differences between the formalisms are characterized by the different interpretations that these annotations are given. Obviously, as the overall goal is still to have an unambiguous knowledge representation formalism, these annotations need to be associated with formal semantics. Our intention is to study the properties, commonalities, and differences of annotated extensions of DLs.

As mentioned already, the semantics of the annotations can differ greatly, depending on the intended application. However, we can in general classify all these approaches into two general groups, depending on whether they modify the semantics of the underlying logic or just how axioms are handled together. We call them the *semantic approach* and the *context-based approach* to interpreting annotated ontologies, respectively.

In the semantic approach the annotated ontology is not interpreted by classical DL semantics anymore. This is the approach usually taken for handling vagueness and uncertainty [LS08, Str01a, Luk08, Jae94, NNS11], since the relations between concepts and individuals cannot be characterized in a precise and absolute manner. For example, some probabilistic logics introduce *multiple-world semantics*, in which the probability of a consequence is given by the proportion of worlds that satisfy it [LS10, Luk08, KP13]. Another example are fuzzy DLs, in which individuals are not anymore either elements of a concept or not, but rather are given an intermediate degree of membership that expresses the imprecise nature of these concepts. Since the formalisms belonging to this approach modify the semantics of the logical component, they need to be studied independently. In this work we focus on formalisms for handling vague knowledge based on mathematical fuzzy logic. We have thus studied the impact of extending DLs with semantics based on different kinds of membership degrees. In our study we have obtained a characterization of the limits of decidability, and the complexity of reasoning in these logics.

In contrast to the semantic approach, in the context-based approach the semantics of the underlying logic remains unchanged, but rather is the reasoning task that is modified. Simply stated, the annotations define a class of relevant sub-ontologies, called the *contexts*, and the reasoning task corresponds a computation on the annotations of the contexts entailing a given consequence (in the classical sense). For example, in the lattice-based context setting, the annotations are ordered via a distributive lattice that expresses dependencies among axioms; specifically, if a context contains an axiom α, then it must also contain all axioms with an annotation greater or equal to the annotation of α. The main reasoning task in this case is to compute a so-called *boundary* for a consequence c: a lattice element that summarizes all the contexts that entail c. In the context-based approach the axioms have a different influence on the outcome of the reasoning task. This is the approach usually taken for handling trust, privacy, provenance, or debugging, among others [BP10a, BP10b, DSSS09, KPHS07, RGL$^+$13]. We study different formalisms based on this approach, from axiom-pinpointing, where the task is only to identify the sets of axioms responsible for a consequence to follow, to a probabilistic DL capable of handling conditional dependencies between axioms. We also study in detail the computational complexity of these formalisms when the logical component is restricted to the light-weight DL $\mathcal{EL}$.

This work focuses on just a few examples of the possible semantics that can be given to annotated ontologies. Nonetheless, many of the lessons learned with these special cases can be generalized or adapted to handle other relevant cases. Moreover, the theoretical results developed during this work have already been used for constructing practical reasoning tools that have shown good runtime behaviour, even when handling very large ontologies.

After some preliminaries presented in the following section, this work provides an overview on ten publications in which we have analysed the properties of different formalisms for interpreting annotated ontologies. We first present our results in the semantic approach, for which we have focused on variants of fuzzy DLs. First we focus on fuzzy DLs with semantics based on t-norms defined over the standard interval $[0, 1]$

of real numbers. In [BDP15], we study in detail the limits of decidability of reasoning in these logics. In particular, we provide a range of inexpressive logics with undecidable reasoning problems, as well as tight complexity bounds for the identified class of decidable logics. One case not covered in [BDP15] corresponds to the case of Gödel semantics with an involutive negation constructor. In [BDP14] that reasoning in this logic is also decidable in exponential time, using a novel automata-based technique. Finally, in [BP13b] we provide the first results on the computational complexity of deciding subsumption in fuzzy $\mathcal{EL}$. These results show that the complexity typically increases for this inexpressive DL.

To regain decidability, we restrict the semantics to allow only finitely many different membership degrees, but allow them to be partially ordered within a lattice, rather than keeping a total order as in the standard case. Using automata-based techniques, we show in [BP13a] that the complexity of reasoning in these finitely-valued fuzzy logics is the same as reasoning in the underlying classical DL. This holds true even if the TBox is restricted to be acyclic, where the complexity usually decreases. In [BP14a] we combine the results from the standard and the finitely-valued semantics to characterize the limits of decidability when infinite lattices are used to describe the membership degrees. In this setting, we characterize infinite families of lattices for which reasoning is decidable and undecidable, respectively.

Section 5 summarizes the formalisms studied within the context-based approach. We start by analysing the complexity of finding the axiomatic causes for a consequence to follow from an ontology. This task, known as axiom-pinpointing, can be used as an auxiliary step in any formalism using the context-based approach. For ontologies written in $\mathcal{EL}$, we show in [PS10b] that almost any reasoning task associated with axiom-pinpointing becomes intractable. Although this does not imply intractability of every context-based approach, it provides a good clue that tractability holds only in very restricted cases. The ideas of axiom-pinpointing are generalized in [BKP12] to consider dependencies between axioms. In essence, the annotations are assumed to form a distributed lattice, and the use of an axiom annotated with an element ℓ automatically implies that all axioms with an annotation larger or equal to ℓ must also be included. This scenario can be used e.g., to control access to some axioms and their consequences. In this work we developed effective algorithms for computing the so-called boundary for a consequence, which summarizes all the labels that define sub-ontologies entailing the consequence.

The work from [CP14a, CP14c] introduces a probabilistic DL in which it is easy to specify conditional and logical dependencies between axioms. Much as for the lattice-based contexts, the annotations are used to specify sets of axioms that must always appear together, or that imply the presence of other axioms. However, these sets of axioms are also associated to a probability distribution, described with the help of a Bayesian network (BN). The reasoning task is to find the probability of observing any context where a desired consequence holds. Although this Bayesian extension can be defined in general for any DL, we study the computational complexity of reasoning in Bayesian $\mathcal{EL}$. With the help of a hypergraph that encodes all the possible derivations of consequences from an ontology, we prove that the complexity of reasoning is governed by the complexity of doing probabilistic inferences in the BN. We also show that reasoning can be decoupled between the logical and the probabilistic components. This latter

Table 1: Syntax and semantics of $\mathcal{ALC}$ constructors and abbreviations

Name	**Syntax**	**Semantics**
negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists r.C$	$\{\delta \in \Delta^{\mathcal{I}} \mid \exists \eta \in C^{\mathcal{I}}.(\delta, \eta) \in r^{\mathcal{I}}\}$
bottom	$\bot$	$\emptyset$
top	$\top$	$\Delta^{\mathcal{I}}$
disjunction	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
implication	$C \to D$	$(\neg C)^{\mathcal{I}} \cup D^{\mathcal{I}}$
value restriction	$\forall r.C$	$\{\delta \in \Delta^{\mathcal{I}} \mid \forall \eta \in \Delta^{\mathcal{I}}.(\delta, \eta) \in r^{\mathcal{I}} \Rightarrow \eta \in C^{\mathcal{I}}\}$

approach does not provide an optimal method in terms of complexity, but suggests an easy way to implement a black-box based reasoner for this logic.

The last paper considered studies an application of the context-based approach for error-tolerant reasoning [LP14b]. The main idea is to exploit the techniques developed for this approach to solve error-tolerant reasoning tasks more efficiently. This is achieved by pre-computing all the repairs for an error in the ontology, and compiling them into an annotated ontology. The methods developed for the lattice-based context setting can be further optimized for handling this special case.

All the publications considered have appeared in top international conferences and journals, and are highlighted using a bold font in the reference key.

2 Classical Description Logics

Description Logics (DLs) [BCM$^+$07] are a family of knowledge representation formalisms specifically designed for representing and reasoning about the knowledge of an application domain in a structured and well-understood manner. The basic notions in any DL are *concepts*, which correspond to unary predicates from first-order logic, and *roles*, which are binary predicates. What differentiates different members of this family are the *constructors* that can be used for building complex concepts and roles from atomic ones. While many constructors, and hence many DLs, have been studied in the literature, for simplicity we focus on the basic DL $\mathcal{ALC}$ and its sublogic $\mathcal{EL}$.

2.1 $\mathcal{ALC}$

Let $\mathsf{N_C}$ and $\mathsf{N_R}$ be two countable, disjoint sets of *concept names* and *role names*, respectively. $\mathcal{ALC}$-concepts are built from these concept and role names through the grammar rule $C ::= A \mid \neg C \mid C \sqcap C \mid \exists r.C$, where $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$. The semantics of these concepts is assigned through interpretations. An *interpretation* is a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty set, called the *domain*, and $\cdot^{\mathcal{I}}$ is an *interpretation function* that maps every concept name $A \in \mathsf{N_C}$ to a set $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, and every role name $r \in \mathsf{N_R}$ to a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. This function is extended to arbitrary concepts as shown in the upper part of Table 1.

In $\mathcal{ALC}$ it is customary to introduce abbreviations, which correspond to other typical constructors in classical logic. These abbreviations are $\bot := A \sqcap \neg A$, for an arbitrary $A \in \mathsf{N_C}$, $\top := \neg\bot$, $C \sqcup D := \neg(\neg C \sqcap \neg D)$, $C \to D := \neg C \sqcup D$, and $\forall r.C := \neg(\exists r.\neg C)$. Using the standard properties of set operations, it is easy to see that the semantics of these abbreviations correspond to those shown in the lower part of Table 1.

The knowledge of an application domain is encoded in an *ontology*, which restricts the class of interpretations that may be taken into account. Ontologies are divided into a TBox, that expresses the relations between the different concepts (terminological knowledge), and an ABox, containing instances of the different concepts and roles (assertional knowledge). Formally, a *TBox* is a finite set of *general concept inclusions* (GCIs) of the form $C \sqsubseteq D$, where C and D are two concepts. The interpretation $\mathcal{I}$ *satisfies* the GCI $C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. $\mathcal{I}$ is a *model* of the TBox $\mathcal{T}$, denoted by $\mathcal{I} \models \mathcal{T}$ iff $\mathcal{I}$ satisfies all the GCIs in $\mathcal{T}$.

Let now $\mathsf{N_I}$ be a countable set, which is disjoint from $\mathsf{N_C}$ and $\mathsf{N_R}$. The elements of $\mathsf{N_I}$ are called *individual names*. For an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, the notion of an interpretation function is extended to map every individual name $a \in \mathsf{N_I}$ to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. An *ABox* is a finite set of *assertions* that are of the form $C(a)$ (*concept assertion*) or $r(a,b)$ (*role assertion*) where $a, b \in \mathsf{N_I}$, $r \in \mathsf{N_R}$, and C is a concept. The interpretation $\mathcal{I}$ satisfies the concept assertion $C(a)$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$; it satisfies the role assertion $r(a,b)$ iff $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$. $\mathcal{I}$ is a *model* of the ABox $\mathcal{A}$ ($\mathcal{I} \models \mathcal{A}$) iff $\mathcal{I}$ satisfies all the assertions in $\mathcal{A}$. An *ontology* is a pair $\mathcal{O} = (\mathcal{T}, \mathcal{A})$, where $\mathcal{T}$ is a TBox and $\mathcal{A}$ is an ABox. The interpretation $\mathcal{I}$ is a *model* of $\mathcal{O}$ ($\mathcal{I} \models \mathcal{O}$) iff it is a model of both, $\mathcal{T}$ and $\mathcal{A}$. We use the term *axiom* to collectively refer to GCIs and assertions.

The main reasoning problem considered in this logic is ontology consistency; that is, given an ontology $\mathcal{O}$, decide whether there exists a model $\mathcal{I}$ of $\mathcal{O}$. This problem is important because all other standard reasoning problems can be polynomially reduced to consistency [BCM+07]. Consider for example the problem of subsumption. Given a TBox $\mathcal{T}$ and two concepts C, D, we say that C is *subsumed* by D w.r.t. $\mathcal{T}$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ holds in every model $\mathcal{I}$ of $\mathcal{T}$. It is easy to see that C is subsumed by D w.r.t. $\mathcal{T}$ iff the ontology $(\mathcal{T}, \{(C \sqcap \neg D)(a)\})$ is inconsistent; i.e., if it is not possible to find an element of the domain that belongs to C but not to D. The consistency problem in $\mathcal{ALC}$ is ExpTime-complete [Sch91, DM00].

2.2 $\mathcal{EL}$

The DL $\mathcal{EL}$ is a sublogic of $\mathcal{ALC}$ in which the only allowed constructors are top, conjunction, and existential restriction. More formally, in $\mathcal{EL}$, concepts are built through the grammar rule $C ::= A \mid \top \mid C \sqcap C \mid \exists r.C$, where $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$. Notice that, although $\top$ was an abbreviation in $\mathcal{ALC}$, it needs to be explicitly introduced in $\mathcal{EL}$ since the negation constructor is disallowed. The notions of interpretation, ontology, and model are restricted to $\mathcal{EL}$ in the obvious way. One important property of this logic is that every $\mathcal{EL}$ ontology is consistent. Thus, when dealing with $\mathcal{EL}$, one is usually interested in deciding subsumption w.r.t. a TBox.

Concept subsumption w.r.t. an $\mathcal{EL}$ TBox can be decided in polynomial time using a completion algorithm [BBL05]. First notice that it suffices to decide subsumption between two concept names: for any two $\mathcal{EL}$ concepts C, D, C is subsumed by D w.r.t.

Table 2: $\mathcal{EL}$ completion rules

If $A' \sqsubseteq A$	**and** $A \sqsubseteq B$	in $\mathsf{comp}(\mathcal{T})$, then	**add** $A' \sqsubseteq B$	to $\mathsf{comp}(\mathcal{T})$
If $A \sqsubseteq A_1, A \sqsubseteq A_2$,	**and** $A_1 \sqcap A_2 \sqsubseteq B$	in $\mathsf{comp}(\mathcal{T})$, then	**add** $A \sqsubseteq B$	to $\mathsf{comp}(\mathcal{T})$
If $A' \sqsubseteq A$	**and** $A \sqsubseteq \exists r.B$	in $\mathsf{comp}(\mathcal{T})$, then	**add** $A' \sqsubseteq \exists r.B$	to $\mathsf{comp}(\mathcal{T})$
If $A' \sqsubseteq \exists r.B', B' \sqsubseteq A$,	**and** $\exists r.A \sqsubseteq B$	in $\mathsf{comp}(\mathcal{T})$, then	**add** $A' \sqsubseteq B$	to $\mathsf{comp}(\mathcal{T})$

the TBox $\mathcal{T}$ iff A is subsumed by B w.r.t. $\mathcal{T} \cup \{A \sqsubseteq C, D \sqsubseteq B\}$, where $A, B \in \mathsf{N_C}$ are two concept names not occurring in $\mathcal{T}$. The completion algorithm first transforms the TBox $\mathcal{T}$ into normal form. A TBox is in normal form if all its GCIs have one of the following shapes:

$$A \sqsubseteq B, \quad A_1 \sqcap A_2 \sqsubseteq B, \quad A \sqsubseteq \exists r.B, \quad \text{or} \quad \exists r.A \sqsubseteq B,$$

where $A_1, A_2, A, B \in \mathsf{N_C} \cup \{\top\}$, and $r \in \mathsf{N_R}$. Any $\mathcal{EL}$ TBox can be transformed to a normalized one, which is equivalent w.r.t. the relevant subsumption relations, in linear time. Let $\mathcal{T}'$ be the TBox obtained from the normalization of $\mathcal{T}$. A partial logical closure is computed through an iterative application of the completion rules from Table 2 starting with $\mathsf{comp}(\mathcal{T}) := \mathcal{T}'$. For simplicity, we assume that the obvious tautologies $A \sqsubseteq A$ and $A \sqsubseteq \top$ belong to $\mathsf{comp}(\mathcal{T})$ for all concept names appearing in $\mathcal{T}'$. To ensure termination, these completion rules are only applied if they actually add a new GCI to $\mathsf{comp}(\mathcal{T})$. Since they can only add GCIs of a restricted shape to $\mathsf{comp}(\mathcal{T})$, only quadratically many rule applications (in $|\mathcal{T}'|$) are possible before the procedure terminates. If $\mathsf{comp}(\mathcal{T})$ is the TBox obtained from $\mathcal{T}$ after no more rules can be applied, then for every two concept names A, B occurring in $\mathcal{T}$ we have that A is subsumed by B w.r.t. $\mathcal{T}$ iff $A \sqsubseteq B \in \mathsf{comp}(\mathcal{T})$.

2.3 Annotated Ontologies

It has been extensively argued that classical logic in general, and classical DLs in particular, are not fully suited for representing all the facets of the knowledge within an application domain. Depending on the scope of the ontology, it may be relevant to extend it with some non-classical features.

In the bio-medical domains, knowledge is rarely precise and certain. For example, when trying to describe a finding in a patient, it is not uncommon to encounter vague terms like *fast* (as in fast growth) or *high* (as in high temperature), where it is impossible to define a precise point where a temperature becomes high, or the growth-speed is fast. On the other hand, measurements made for a finding, or the consequences of a treatment typically have an associated uncertainty that arises from unobservable or unforeseen factors. This has motivated the study of formalisms for handling vagueness [SKP07, Str01a, SSP$^+$07, MSS$^+$12] and uncertainty [LS08, QJPD11, KP08, dFL08, LS10, Jae94, Luk08] in ontologies.

Considering the Semantic Web, a large ontology might be obtained combining the knowledge provided by different sources found over the web. As some of these sources might be more trustworthy than others, it makes sense to try to order the axioms, and their consequences, by some preference relation. A user can then limit her views to avoid consequences that she does not trust. Similarly, if some knowledge is restricted to some users, one should disallow these users to access any implicit consequence of that restricted knowledge.

Clearly, handling vagueness, uncertainty, preferences, or access restrictions, are just four examples of desiderata of knowledge representation formalisms that have been studied in the literature. For each of them, different methods need to be developed. However, many of these approaches share a common syntactic approach: ontologies are expressed using classical DL GCIs and assertions, extended with an annotation that refers to the extension considered. Thus, for example, when dealing with uncertainty an annotated GCI $\langle A \sqsubseteq B : p\rangle$ with $p \in [0, 1]$ may express the probability with which the axiom holds, while for access control the label ℓ in $\langle A \sqsubseteq B : \ell\rangle$ expresses the security level required to access this GCI.

Formally, let Λ be a set, whose elements are called *labels* or annotations. A Λ-*annotated GCI* is an expression of the form $\langle C \sqsubseteq D : \lambda\rangle$, where C and D are two concepts and $\lambda \in \Lambda$. A Λ-*annotated TBox* is a finite set of Λ-annotated GCIs. Analogously, Λ-*annotated concept and role assertions* are of the form $\langle C(a) : \lambda\rangle$ and $\langle r(a, b) : \lambda\rangle$, respectively. A Λ-*annotated ABox* is a finite set of Λ-annotated assertions, and a Λ-*annotated ontology* is a pair $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ where $\mathcal{T}$ is a Λ-annotated TBox and $\mathcal{A}$ is a Λ-annotated ABox. When the set of annotations Λ is clear from the context, we will often omit it, and speak of, e.g. an annotated ontology.

Each formalism interprets these annotations in a different manner. We classify the different methods for interpreting them in two main approaches: the *semantic* and the *context-based* approach. In a nutshell, the semantic approach is characterized by a change in the semantics of the underlying logic. For example, as explained in Section 3, in fuzzy description logics the interpretation of a concept is not anymore a subset of the domain $\Delta^{\mathcal{I}}$, but rather a function from $\Delta^{\mathcal{I}}$ to a set of membership degrees. In the context-based approach, the semantics of the underlying logic does not change. In this case, the labels in the axioms are used to divide the ontology into sub-ontologies (also called *contexts*). The information of the contexts entailing a given consequence is then combined according to the specific formalism used. In the case of access control, each context corresponds to a privacy level of the axioms. By combining all the contexts that can derive a consequence, we obtain the privacy level of that consequence; that is, the access rights that a user must have in order to observe this consequence. We cover these general approaches, and some of their instances, in more detail in the following sections.

As mentioned before, throughout this work we will focus mainly on the two DLs $\mathcal{ALC}$ and $\mathcal{EL}$. The specific properties of these logics will be necessary for some of our results. The latter is especially true when considering the complexity of reasoning in annotated extensions of $\mathcal{EL}$. However, many other DLs exist, which use other concept constructors. Most of our results from Section 4 can be extended to these DLs, and any other ontology language with a monotonic entailment relation without any major changes.

3 The Semantic Approach

The first approach that we consider for reasoning with annotated ontologies is the semantic approach. In this approach, the semantics of the underlying logical formalism is modified to handle the required non-classical extensions.

The semantic approach is most commonly used for handling vagueness and uncertainty, for example in fuzzy or probabilistic extensions of DLs [LS08]. In the former case, concepts and roles are interpreted as *fuzzy* sets and *fuzzy* binary relations, respectively. Thus, the whole notions of interpretations and models need to be adapted accordingly. In the latter case, typically the notion of an interpretation does not change, but rather the conditions under which an interpretation is a model of the ontology. This can depend on e.g. the proportion of elements of the domain that satisfy a property, if statistical probabilities are used, or a probability distribution over several interpretations in the case of subjective probabilities [KP13]. This is also the approach used for some possibilistic DLs [QJPD11, Hol95].

For the rest of this section we focus on fuzzy DLs only. One of the characterizing factors that defines a fuzzy DL is the set of *membership degrees* that defines its semantics. First, we show that for fuzzy DLs defined over the standard chain $[0,1]$, deciding consistency of an ontology is undecidable, even for very restricted logics. Afterwards, we show that if only finitely many membership degrees are used, then the problem is decidable, and for expressive logics not harder than classical reasoning. This holds even if the membership degrees are not arranged in a total order, but in a lattice. At the end of the section we provide conditions that ensure decidability of fuzzy DLs based on infinite lattices.

3.1 Fuzzy Description Logics

Fuzzy description logics (FDLs) extend classical DLs by allowing a more fine-grained membership relation of elements to concepts and roles. In these logics, the elements of the interpretation domain belong to a concept to some degree, which is typically a number in the interval $[0,1]$. Following the ideas from mathematical fuzzy logic [Háj01], the constructors are interpreted using a t-norm and its associated operators [KMP00].

A t-norm is an associative, commutative, and monotonic (on both arguments) binary operator $\otimes : [0,1] \times [0,1] \to [0,1]$ that has neutral element 1. It is a *continuous* t-norm if it is continuous as a function. For the rest of this section, we will only consider continuous t-norms, and will call them simply t-norms for brevity. The t-norm operator is used in mathematical fuzzy logic to interpret the conjunction.

Every continuous t-norm $\otimes$ defines a unique *residuum* operator $\Rightarrow$ that satisfies, for all $x, y, z \in [0,1]$, that $x \otimes y \leq z$ iff $y \leq (x \Rightarrow z)$. This operator can be defined by $x \Rightarrow y = \sup\{z \in [0,1] \mid x \otimes z \leq y\}$. The residuum is used to generalize the logical implication to fuzzy logics. With this residuum, we can define the *residual negation* $\ominus$ given by $\ominus x := x \Rightarrow 0$ for all $x \in [0,1]$. As suggested by its name, $\ominus$ is used as a generalization of logical negation. To interpret disjunctions, we use the *t-conorm* $\oplus$ of the t-norm $\otimes$, which is defined by $x \oplus y := 1 - ((1-x) \otimes (1-y))$. In some cases, it is useful to consider also the *involutive negation* operator defined by $\sim x := 1 - x$ for all $x \in [0,1]$. Notice that the involutive negation is not expressible in terms of the other

Table 3: Gödel, Łukasiewicz, and product t-norms and their operators

Name	$x \otimes y$	$x \oplus y$	$x \Rightarrow y$	$\ominus x$
Gödel (G)	$\min\{x, y\}$	$\max\{x, y\}$	$\begin{cases} 1 & \text{if } x \leq y \\ y & \text{otherwise} \end{cases}$	$\begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases}$
Łukasiewicz ($\mathsf{Ł}$)	$\max\{x + y - 1, 0\}$	$\min\{x + y, 1\}$	$\min\{1 - x + y, 1\}$	$1 - x$
product ($\mathsf{\Pi}$)	$x \cdot y$	$x + y - x \cdot y$	$\begin{cases} 1 & \text{if } x \leq y \\ y/x & \text{otherwise} \end{cases}$	$\begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases}$

operators introduced above.

The three main t-norms are known as the Gödel (G), Łukasiewicz ($\mathsf{Ł}$), and product ($\mathsf{\Pi}$) t-norms. These t-norms and their associated operators are listed in Table 3. These t-norms are fundamental in the sense that all other continuous t-norms are isomorphic to ordinal sums of copies of these three t-norms; for details see [KMP00, MS57]. Of particular importance are those t-norms whose ordinal sum description has one initial component that is isomorphic to the Łukasiewicz t-norm. These t-norms are said to *start with Łukasiewicz*. Notice that in the Łukasiewicz t-norm, the residual and the involutive negation coincide. One consequence of this fact is that, for every t-norm that starts with Łukasiewicz, the residual negation also behaves as an involutive operator in some closed subinterval $[0, p] \subseteq [0, 1]$. A t-norm *contains Łukasiewicz* if at least one of the components in its ordinal sum decomposition is isomorphic to the Łukasiewicz t-norm.

Given a t-norm $\otimes$, an element $x \in (0, 1]$ is a *zero-divisor* of $\otimes$ iff there exists some $y \in (0, 1]$ such that $x \otimes y = 0$. From the three fundamental t-norms, it is easy to see that only the Łukasiewicz t-norm has zero-divisors. In fact, every $x \in (0, 1)$ is a zero-divisor of this t-norm. Moreover, it can be shown that a t-norm has zero-divisors iff it starts with Łukasiewicz. Another relevant property is idempotency. An element $x \in [0, 1]$ is *idempotent* w.r.t. $\otimes$ iff $x \otimes x = x$. The t-norm $\otimes$ is called idempotent iff every element $x \in [0, 1]$ is idempotent w.r.t. $\otimes$. There exists only one idempotent t-norm, namely the Gödel t-norm.

Since many of the standard dualities between logical constructors do not hold for fuzzy logics, the family of fuzzy DLs is larger than those of classical DLs. In particular, many different languages can be considered as a fuzzy extension of the classical $\mathcal{ALC}$, depending on the class of constructors allowed. We introduce the fuzzy DL $\otimes$-$\Im\mathcal{ALC}$, which allows all the constructors expressible in classical $\mathcal{ALC}$ and then describe some of its sublogics. To avoid unnecessary repetitions, we will define all notions only for the larger logic; they are restricted to the different sublogics in the obvious way.

A fuzzy DL is characterized by two components: the t-norm $\otimes$ used for defining its semantics, and the class of constructors allowed for building complex concepts. In the following, let $\otimes$ be an arbitrary, but fixed, continuous t-norm. In $\otimes$-$\Im\mathcal{ALC}$, concepts are built through the grammar rule

$$C ::= A \mid \top \mid \bot \mid \boxminus C \mid \neg C \mid C \sqcap C \mid C \to C \mid \exists r.C \mid \forall r.C,$$

Table 4: Constructors of some relevant fuzzy DLs

Name	$\top$	$\sqcap$	$\exists$	$\forall$	$\bot$	$\to$	$\boxminus$	$\neg$
$\mathcal{EL}$	✓	✓	✓					
$\mathfrak{N}\mathcal{EL}$	✓	✓	✓		(✓)		✓	
$\mathcal{ELC}$	✓	✓	✓		(✓)			✓
$\mathfrak{I}\mathcal{AL}$	✓	✓	✓	✓	✓	✓	(✓)	
$\mathcal{ALC}$	✓	✓	✓	✓	(✓)			✓

Table 5: Semantics of fuzzy DL constructors

$$
\begin{aligned}
\top^{\mathcal{I}}(\delta) &:= 1\\
\bot^{\mathcal{I}}(\delta) &:= 0\\
(\boxminus C)^{\mathcal{I}}(\delta) &:= \ominus\, C^{\mathcal{I}}(\delta)\\
(\neg C)^{\mathcal{I}}(\delta) &:= {\sim} C^{\mathcal{I}}(\delta)\\
(C \sqcap D)^{\mathcal{I}}(\delta) &:= C^{\mathcal{I}}(\delta) \otimes D^{\mathcal{I}}(\delta)\\
(C \to D)^{\mathcal{I}}(\delta) &:= C^{\mathcal{I}}(\delta) \Rightarrow D^{\mathcal{I}}(\delta)\\
(\exists r.C)^{\mathcal{I}}(\delta) &:= \sup_{\eta\in\Delta^{\mathcal{I}}} \big(r^{\mathcal{I}}(\delta,\eta) \otimes C^{\mathcal{I}}(\eta)\big)\\
(\forall r.C)^{\mathcal{I}}(\delta) &:= \inf_{\eta\in\Delta^{\mathcal{I}}} \big(r^{\mathcal{I}}(\delta,\eta) \Rightarrow C^{\mathcal{I}}(\eta)\big)
\end{aligned}
$$

where $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$. Different sublogics are created by restricting the set of constructors allowed as shown in Table 4. In the table, a checkmark ✓expresses that the constructor is allowed in the respective logic; if it appears within parenthesis (✓), then it can be expressed as an abbreviation from other constructors in the same logic; i.e., it does not need to be introduced explicitly. An ontology from the logic $\otimes$-$\mathfrak{I}\mathcal{ALC}$ is simply a $(0,1]$-annotated ontology, where concepts are allowed to use all the constructors from $\mathfrak{I}\mathcal{ALC}$.

A *fuzzy interpretation* is a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty set called the *domain*, and $\cdot^{\mathcal{I}}$ is the *interpretation function* that maps every individual name $a \in \mathsf{N_I}$ to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, every concept name $A \in \mathsf{N_C}$ to a function $A^{\mathcal{I}} : \Delta^{\mathcal{I}} \to [0,1]$ (known as a fuzzy set), and every role name $r \in \mathsf{N_R}$ to a function $r^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \to [0,1]$ (fuzzy binary relation). Intuitively, the fuzzy interpretation provides a membership degree of every element of the domain to belong to each concept name; the higher the degree, the more this element belongs to the concept, with 0 and 1 corresponding to the classical membership degrees. Similarly, the degree associated to a pair of elements in the interpretation of a role name r expresses how much these elements are related via r.

The interpretation function is extended to complex concepts inductively using the operators associated to the t-norm $\otimes$ as described in Table 5 for every $\delta \in \Delta^{\mathcal{I}}$. Notice that the semantics of existential and value restrictions require the computation of suprema and infima over the possibly infinite domain. To avoid issues with such an infinite computation, it is customary to restrict reasoning to a special class of so-called witnessed interpretations [BS09, Háj05]. The interpretation $\mathcal{I}$ is *witnessed* if for every concept C, role name r, and $\delta \in \Delta^{\mathcal{I}}$ there exist $\eta, \eta' \in \Delta^{\mathcal{I}}$ such that

$$(\exists r.C)^{\mathcal{I}}(\delta) = r^{\mathcal{I}}(\delta, \eta) \otimes C^{\mathcal{I}}(\eta), \quad \text{and}$$
$$(\forall r.C)^{\mathcal{I}}(\delta) = r^{\mathcal{I}}(\delta, \eta') \Rightarrow C^{\mathcal{I}}(\eta').$$

This means that the suprema and infima in the semantics of existential and value restrictions are actually maxima and minima, respectively. Without this restriction, the value of $(\exists r.C)^{\mathcal{I}}(\delta)$ might, e.g. be 1 without x actually having a single r-successor with degree 1 that belongs to C with degree 1. Such a behaviour is usually unwanted in DLs, where an existential restriction is intended to express the existence of an adequate successor. Unless explicitly mentioned otherwise, we consider only witnessed interpretations for the rest of this section.

The witnessed interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ *satisfies* the annotated GCI $\langle C \sqsubseteq D : p \rangle$ if for all $\delta \in \Delta^{\mathcal{I}}$, it holds that $C^{\mathcal{I}}(\delta) \Rightarrow D^{\mathcal{I}}(\delta) \geq p$. It *satisfies* the annotated assertion $\langle C(a) : p \rangle$ (respectively $\langle r(a, b) : p \rangle$) if $C^{\mathcal{I}}(a^{\mathcal{I}}) \geq p$ (resp., $r^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}) \geq p$). It is a *model* of the ontology $\mathcal{O}$ if it satisfies all the axioms in $\mathcal{O}$. An ontology is *consistent* if it has a model.

The first doubts of the decidability of ontology consistency in fuzzy DLs, when GCIs are included, arose after it was shown that the existing reasoning procedures were incorrect [BBS11]. Up to that point, existing (usually tableau-based) reasoning algorithms produced a finite model of the ontology [BS07, BS09, SB07]. The work from [BBS11] proved that, if GCIs are allowed, then one can build a consistent ontology that has no finite models; such ontologies would be classified as inconsistent by the methods mentioned above. This triggered the work in [BP11a], where it is shown that ontology consistency is undecidable in a slight extension of Π-$\mathcal{ALC}$. The quest for the limits of decidability in fuzzy DLs continued for a couple of years, where undecidability was shown for a growing class of languages [BP11a, BP11b, BP11c, CS13]. Overall, the main culprit for undecidability turns out to be the existence of an involutive operator. Indeed, if $\otimes$ is a t-norm that starts with Łukasiewicz, then ontology consistency is undecidable already in $\otimes$-$\mathfrak{N}\mathcal{EL}$ which, in addition to the constructors from $\mathcal{EL}$, allows only the residual negation $\boxminus$. If instead of the residual negation the involutive negation is used, i.e., in the logic $\otimes$-$\mathcal{ELC}$, then undecidability arises for any non-idempotent t-norm. As discussed before, this shows undecidability for all except one continuous t-norm, where the Gödel t-norm is the only remaining case.

Rather than proving all these undecidability results independently, a general proof method was proposed in [BP12b]. This general method abstracts the ideas used in previous undecidability proofs and characterizes a series of simple properties that, together, yield a reduction from the Post correspondence problem [Pos46]. Thus, these properties yield sufficient conditions for the consistency problem to be undecidable. The framework was further extended in [BDP15] and instantiated to obtain the undecidability results described above, among others.

Conversely, it was shown that the problem becomes decidable if the involutive oper-

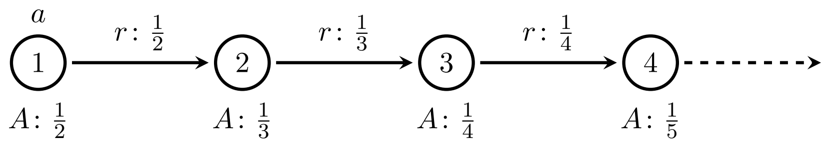

Figure 1: The model $\mathcal{I}_1$ of the G-$\mathcal{ALC}$ ontology $\mathcal{O}_1$

ators are excluded. More precisely, for any t-norm $\otimes$ that does not have zero-divisors (i.e., does not start with Łukasiewicz), ontology consistency in $\otimes$-$\mathfrak{I}\mathcal{AL}$ is decidable in exponential time [BDP12]. In fact, the result from that paper is much stronger than this complexity result. It is shown that to decide consistency of a $\otimes$-$\mathfrak{I}\mathcal{AL}$ ontology it suffices to check consistency of the classical $\mathcal{ALC}$ ontology obtained from removing all the annotations from the axioms. These, and many other decidability and undecidability results results are presented in full detail in [BDP15]. In particular, the decidability results are extended to the much more expressive DL $\mathcal{SROIQ}$, which extends $\mathfrak{I}\mathcal{ALC}$ with several additional concept constructors, as well as axioms restricting the interpretations of roles.

One important case not covered in [BDP15] is the decidability of consistency in fuzzy DLs under Gödel semantics that include the involutive negation as a constructor. Since the Gödel t-norm is idempotent, it was largely believed that reasoning in G-$\mathfrak{I}\mathcal{ALC}$ could be restricted to finitely many membership degrees, which means that it is decidable (see Section 3.2). This property was originally shown by Straccia [Str98, Str01b]. However, the proof relies on a different semantics, called the Zadeh semantics, where the implication $x \Rightarrow y$ is not interpreted as the Gödel residuum, but rather as $\max\{1-x, y\}$. Motivated by this result, following papers using the Gödel semantics directly restricted the membership degrees to a finite set [BDGRS09, BDGRS12].

It turns out that restricting reasoning to a finite set of membership degrees does affect the expressivity of the logic. Consider for example the G-$\mathcal{ALC}$ ontology $\mathcal{O}_1 = (\mathcal{T}_1, \mathcal{A}_1)$ where $\mathcal{T}_1 = \{\langle \forall r.A \sqsubseteq A : 1\rangle, \langle \exists r.\top \sqsubseteq A : 1\rangle\}$ and $\mathcal{A}_1 = \{\langle \neg A(a) : 0.5\rangle\}$. This ontology is consistent; the interpretation $\mathcal{I}_1 = (\mathbb{N}, \cdot^{\mathcal{I}_1})$ with $a^{\mathcal{I}} := 1$, $A^{\mathcal{I}_1}(n) := r^{\mathcal{I}_1}(n, n+1) := \frac{1}{n+1}$ for all $n \in \mathbb{N}$, and $r^{\mathcal{I}_1}(n, m) := 0$ if $m \neq n+1$ (see Figure 1) is a model of $\mathcal{O}_1$. However, it can be seen that every model $\mathcal{I}$ of $\mathcal{T}_1$ that uses only finitely many different membership degrees is such that $A^{\mathcal{I}}(n) = 1$ for all $n \in \mathbb{N}$. The main reason for this behaviour arises from the properties of the Gödel residuum, where $x \Rightarrow y$ is either 1, if $x \leq y$ or y otherwise. The first GCI in $\mathcal{T}_1$ ensures that for every $\delta \in \Delta^{\mathcal{I}}$ there must exist an $\eta \in \Delta^{\mathcal{I}}$ such that $r^{\mathcal{I}}(\delta, \eta) \Rightarrow A^{\mathcal{I}}(\eta) \leq A^{\mathcal{I}}(\delta)$. If $A^{\mathcal{I}}(\delta) < 1$ for some δ, then this will produce an infinite sequence of successors with strictly decreasing membership degrees, but always greater than 0. But if only finitely many membership degrees are allowed, such a chain can never be produced. In particular, this means that a finitely-valued model $\mathcal{I}$ of $\mathcal{T}_1$ cannot be a model of $\mathcal{A}_1$. Thus, $\mathcal{O}_1$ is inconsistent whenever reasoning is restricted to finitely-valued interpretations.

To decide consistency of G-$\mathfrak{I}\mathcal{ALC}$ ontologies, a new reasoning procedure needed to be devised. The main insight required is that for building a model of a G-$\mathfrak{I}\mathcal{ALC}$ ontology, the specific membership degrees used are not as relevant as the order among them. For example, if we change the interpretation function of the model $\mathcal{I}_1$ described above to any mapping that satisfies $A^{\mathcal{I}_1}(n+1) < r^{\mathcal{I}_1}(n, n+1) \leq A^{\mathcal{I}_1}(n)$ for all $n \in \mathbb{N}$, then we would still obtain a model of $\mathcal{O}_1$. Thus, rather than trying to build a model directly, one can try to produce an abstract representation of a family of models, where only the order of the membership degrees of the different elements is explicitly stated. Moreover, it suffices to consider only forest-shaped models, where the order is local; that is, for each element of the domain, it suffices to express the relationship among the membership degrees to the different concepts at that element and those at its (only) direct predecessor in the forest. If all these local orders can be satisfied, then the density of the real numbers guarantees that at least one model can be built from it. Notice that although infinitely many degrees might be necessary to actually construct the model, only finitely many local orders are relevant.

Putting all these insights together, it is possible construct an automaton that verifies that such a well-structured forest-shaped model can be built, by performing linearly many emptiness tests. This automaton has exponentially many states, measured in the size of the input ontology. Thus G-$\mathfrak{I}\mathcal{ALC}$ ontology consistency is decidable in exponential time, matching the complexity of classical $\mathcal{ALC}$. In [BDP14] this idea is taken one step further. There it is shown that the exponential upper bound still holds even if ABoxes are extended to allow for arbitrary order assertions. *Order assertions* are of the form $\langle \alpha \bowtie \gamma \rangle$, where α is a concept or role assertion, γ is an assertion or a constant in $[0, 1]$, and $\bowtie \in \{>, <, =, \geq, \leq\}$. The semantics of order assertions is the obvious one.

Overall, the results presented in the two papers [BDP15, BDP14] provide a full classification of the limits of decidability of ontology consistency for fuzzy extensions of the DL $\mathcal{ALC}$, over the standard chain $[0, 1]$. Some of these results have been further strengthened. As mentioned above, in the case of t-norms without zero-divisors, decidability has been shown for the more expressive $\otimes$-$\mathcal{SROIQ}$, using a similar technique. For t-norms that start with Łukasiewicz, conversely, undecidability holds even if all the axioms are annotated with the constant 1; i.e., undecidability is not a consequence of the annotations, but rather of the extended semantics. Finally, it is known that using the Gödel t-norm does not increase the complexity of reasoning even in inexpressive sub-logics like $\mathcal{EL}$ [MSS$^+$12], and $\mathcal{FL}_0$ with cyclic TBoxes [BLP14].

If we consider other fuzzy extensions of $\mathcal{EL}$, the picture is less clear. The first attempt to study the complexity of subsumption in $\otimes$-$\mathcal{EL}$ under any t-norm that is not idempotent appeared in [BP13b]. In it, it is shown that for any t-norm $\otimes$ that *contains* at least one Łukasiewicz component, this problem is coNP-hard. On the other hand, a variant of the problem in which the goal is only to decide whether subsumption holds to some positive degree exhibits a dichotomy similar to the one found for consistency in more expressive fuzzy DLs: the problem is polynomial for all t-norms without zero divisors, and coNP-hard for all other t-norms. Unfortunately, matching upper bounds have not been found yet. In fact, it is conjectured that general subsumption is at least ExpTime-hard for t-norms that contain Łukasiewicz.

3.2 Finitely Valued Semantics

A different restriction that can be used for regaining decidability of ontology consistency is to consider only finitely many membership degrees in the semantics of the logic.

Rather than simply restricting to a finite subset of the interval $[0, 1]$, we allow for the membership degrees to be partially ordered, forming a lattice. In order to interpret the different constructors, the lattice needs to be extended with two operators, forming a so-called residuated lattice [DK93, GJKO07].

A *residuated lattice* is an algebraic structure $(L, \vee, \wedge, \mathbf{0}, \mathbf{1}, \otimes, \Rightarrow)$ over the *carrier set* L, where $(L, \vee, \wedge, \mathbf{0}, \mathbf{1})$ is a bounded lattice with minimum element $\mathbf{0}$ and maximum $\mathbf{1}$, $\otimes$ is a monotonic, associative and commutative binary operator on L that has $\mathbf{1}$ as unit (called *t-norm*), and $\Rightarrow$ is a binary operator, called *residuum*, such that $\ell_1 \otimes \ell_2 \leq \ell_3$ iff $\ell_2 \leq \ell_1 \Rightarrow \ell_3$ holds for all $\ell_1, \ell_2, \ell_3 \in L$. As in the previous section, the t-norm $\otimes$ is used to interpret conjunction and the residuum interprets the implication. The interpretation of other constructors, such as the residual negation, is obtained from these operators in an analogous manner. To interpret the involutive negation, we need to further extend the residuated lattice with a *(De Morgan) negation*, which is an involutive and antitonic unary operator $\sim$ that satisfies the De Morgan laws $\sim(\ell_1 \vee \ell_2) = \sim\ell_1 \wedge \sim\ell_2$ and $\sim(\ell_1 \wedge \ell_2) = \sim\ell_1 \vee \sim\ell_2$ for all $\ell_1, \ell_2 \in L$.

Given a finite De Morgan residuated lattice L, an L-$\mathfrak{I}\mathcal{ALC}$ ontology is simply an L-annotated ontology where concepts are built using all the constructors from $\mathfrak{I}\mathcal{ALC}$. The semantics of this logic is defined as in the previous section, except that the range of the interpretations is now restricted to the set L, rather than the interval $[0, 1]$. More precisely, an interpretation is a pair $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where $\Delta^\mathcal{I}$ is a non-empty set and $\cdot^\mathcal{I}$ is the interpretation function that maps every $a \in \mathsf{N_I}$ to an element $a^\mathcal{I} \in \Delta^\mathcal{I}$, every $A \in \mathsf{N_C}$ to a function $A^\mathcal{I} : \Delta^\mathcal{I} \to L$ and every $r \in \mathsf{N_R}$ to a function $r^\mathcal{I} : \Delta^\mathcal{I} \times \Delta^\mathcal{I} \to L$. This interpretation is extended to arbitrary concepts as in the previous section, where $\otimes$, $\Rightarrow$, and $\sim$ are now the t-norm, residuum, and De Morgan negation of the lattice L, respectively.

If L is a finite total order, then every L-$\mathfrak{I}\mathcal{ALC}$ ontology $\mathcal{O}$ can be transformed into an equi-consistent classical ontology from the DL $\mathcal{ALCH}$ [Str04].[1] The idea is to create, for every concept name A appearing in $\mathcal{O}$, finitely many (classical) concept names of the form $A_{\geq \ell}$ that intuitively contain all the elements of the domain that belong to A with a degree at least ℓ, and analogously r_ℓ for role names. Using the properties of the specific t-norm and residuum defined for L, the annotated axioms in $\mathcal{O}$ can be similarly translated into classical axioms expressing the relations between the membership degrees of the original finitely valued concepts. Since consistency of $\mathcal{ALCH}$ ontologies is decidable in exponential time, this yields a decidable reasoning method for consistency of L-$\mathfrak{I}\mathcal{ALC}$ ontologies, on finite total orders. Unfortunately this translation, as described in several papers [BS11, BS13, BDGRS09], produces an exponential blow-up on the size of the ontology. Thus, this approach yields a double-exponential decision procedure for consistency of L-$\mathfrak{I}\mathcal{ALC}$ ontologies, which leaves a gap when compared to the ExpTime complexity of classical $\mathcal{ALC}$.

A direct reasoning procedure that avoids this exponential blow-up was originally proposed in [BP11d]. This procedure extends the automata-based decision procedure of $\mathcal{ALC}$ [LS00] to handle the finitely valued semantics. The main idea of the automata-based method is to construct an automaton that accepts some well-structured forest-shaped models of the ontology. More precisely, it is shown that an L-$\mathfrak{I}\mathcal{ALC}$ ontology

[1] $\mathcal{ALCH}$ is an extension of $\mathcal{ALC}$ that can also express inclusions between roles [BCM+07].

$\mathcal{O}$ is consistent if and only if it has a model formed of a set of (potentially infinite) trees whose roots may be arbitrarily interconnected, and these roots are exactly the interpretations of the individual names appearing in $\mathcal{O}$. To decide consistency, one can guess the interpretation of the roots, and use an automaton on infinite trees to verify that the corresponding tree-shaped interpretations can be built from the guessed roots. Then, the ontology is consistent if and only if the language accepted by this automaton initialized on every root node is not empty.

For simplicity, suppose that $\mathcal{O}$ is of the form $(\mathcal{T}, \{\langle C_0(a) : \ell\rangle\})$, where C_0 is an $\mathfrak{I}\mathcal{ALC}$ concept and $\ell \in L$; that is, the ABox consists of only one concept assertion. We denote as $\mathsf{sub}(\mathcal{O})$ the set of all subconcepts that appear in $\mathcal{O}$. A *Hintikka function* is a mapping $H : \mathsf{sub}(\mathcal{O}) \to L$ that is consistent with the semantics of the propositional constructors; e.g. $H(C \sqcap D) = H(C) \otimes H(D)$ for all concepts $C \sqcap D \in \mathsf{sub}(\mathcal{O})$, and such that, for every GCI $\langle C \sqsubseteq D : \ell\rangle \in \mathcal{T}$, it holds that $H(C) \Rightarrow H(D) \geq \ell$. Intuitively, a Hintikka function can be seen as a possible interpretation for all the relevant concepts at some element of the domain, which satisfies the TBox $\mathcal{T}$. The successors in the tree are used to satisfy the existential and value restrictions. Informally, for every existential restriction $\exists r.C \in \mathsf{sub}(\mathcal{O})$, we create one successor that will witness this restriction. If a node is labelled with a Hintikka function H and its corresponding successor is labelled with the Hintikka function H', then these nodes satisfy the *Hintikka condition* if $H(\exists r.C) = H'(r) \otimes H'(C)$. It then holds that $\mathcal{O}$ is consistent iff there is an infinite tree where all the nodes are labelled with Hintikka functions, and each successor relation satisfies the Hintikka condition. To build an automaton that accepts such trees, it suffices to use the Hintikka functions as states, and the Hintikka condition to define the appropriate transition relation.

The automata-based method was later generalized to handle all the constructors in the expressive DL $\mathcal{SHI}$ [BP13a]. Since the size of the automaton is exponential in the size of the ontology $\mathcal{O}$, and emptiness of an automaton can be decided in polynomial time in its size, overall this yields an exponential time procedure for deciding consistency of L-$\mathcal{SHI}$ ontologies, and hence also of L-$\mathfrak{I}\mathcal{ALC}$ ontologies. Beyond these tight complexity bounds, the automata-based approach is also helpful for understanding the complexity of deciding consistency, if the form of the ontology is restricted. For example, following the ideas originally introduced in [BHP08] for classical DLs, it is shown that the complexity decreases to PSpace in L-$\mathfrak{I}\mathcal{ALC}$ and some of its extensions, if the TBox is *acyclic* [BP13a, BP14b].

Although optimal in terms of complexity, automata-based methods are usually impractical, as they exhibit a very bad *best-case* behaviour. In fact, as the exponentially-large automaton needs to be constructed first, the best-case and the worst-case behaviour of this approach coincide. To alleviate this problem, a tableaux-based algorithm was proposed in [BP12a, BP14a]. As usual in tableaux-based approaches, this algorithm tries to produce a model of the ontology by decomposing complex concepts into their subconcepts, until only concept names and role names remain, from which an interpretation can be built. Unfortunately, this algorithm still requires major optimization techniques before it can be used in practice, as nearly every decomposition rule requires a non-deterministic choice.

3.3 Infinite Lattices

When considering finitely valued semantics, there was no need to restrict to a total order among the membership degrees, but rather allowed these degrees to be ordered within a lattice. Then, one natural question is whether the decidability results from Section 3.1 also hold for infinite lattices and, more generally, where do the limits of decidability lie for fuzzy DLs with semantics based on infinitely-many membership degrees organized as a lattice.

To generalize the finitely valued semantics introduced in the previous section to infinite lattices, we weaken the restrictions on L to require only a *complete* De Morgan residuated lattice. It is easy to see that every continuous t-norm over $[0, 1]$ with its residuum yields one such lattice, and that every finite lattice is also complete. Hence, this setting generalizes the formalisms presented in the previous sections. In particular, all the undecidability results from Section 3.1 hold for semantics based on infinite lattices, too.

Recall that ontology consistency in $\otimes$-$\mathfrak{I}\mathcal{AL}$ is decidable iff the t-norm $\otimes$ has no zero-divisors. However, if the involutive negation constructor is allowed, decidability holds only for the Gödel t-norm. Although for the case of infinite lattices, such a direct characterization of decidability is not possible, some of the techniques developed before can be used to provide partial answers [BP14a].

In terms of decidability, it is shown that if L has no zero-divisors, then consistency of L-$\mathfrak{I}\mathcal{AL}$ ontologies is decidable in exponential time. This is shown generalizing the method for $\otimes$-$\mathfrak{I}\mathcal{AL}$, in which the problem was reduced in linear time to consistency of a classical $\mathcal{ALC}$ ontology. As before, the reduction simply removes all the annotations from the fuzzy ontology. It is then shown that this simplification is consistency-preserving.

Undecidability, on the other hand, is not a direct consequence of the presence of zero-divisors. In fact, the work in [BP14a] characterizes an uncountable family of lattices with finitely many zero-divisors for which L-$\mathfrak{I}\mathcal{AL}$ ontology consistency is decidable. Conversely, there are also uncountably many lattices with only one zero-divisor for which the problem is undecidable. Thus, the existence or absence of zero-divisors is not sufficient for predicting decidability of lattice-based fuzzy DLs.

This concludes our study on fuzzy DLs and the semantic approach for reasoning with annotated ontologies. In the next section we switch our attention to the context-based approach.

4 The Context-based Approach

In the context-based approach, the annotations associated to the ontology axioms have the main purpose of dividing the ontology in relevant subontologies. Reasoning consists then on computing a property of all such subontologies that entail a given consequence, in the classical sense. In this setting, it is then relevant to be able to identify the axioms that are responsible for a consequence to follow. This task, known as *axiom-pinpointing*, can also be seen as a special case of the context-based approach, where every subontology is relevant for the task.

We first give a brief introduction to axiom-pinpointing and provide relevant complexity results for the cases of $\mathcal{ALC}$ and $\mathcal{EL}$. Afterwards, we introduce a general context-based approach for the case in which the annotations are well-structured in a distributive lattice. In the end we show how this approach can be used for error-tolerant reasoning and reasoning with conditionally dependent probabilistic axioms.

4.1 Axiom-Pinpointing

Axiom-pinpointing is the task of identifying the axioms that are responsible for a consequence to follow [SC03, KPHS07, BPS07]. More precisely, given an ontology $\mathcal{O}$, and a consequence c (e.g., a subsumption relation between two concept names, or inconsistency), in axiom-pinpointing we are interested in finding all the subsets of $\mathcal{O}$ that entail this consequence c. Since we consider only monotonic consequences, it suffices to find only the minimal such sets; all supersets of these will also entail the consequence. In this section we will abuse of the notation and consider $\mathcal{O}$ simply as a set of axioms, without distinguishing between the TBox and the ABox.

Formally, a *MinA* for a consequence c w.r.t. an ontology $\mathcal{O}$ is a subset $\mathcal{M} \subseteq \mathcal{O}$ such that $\mathcal{M}$ entails c, and every strict subset $\mathcal{M}' \subset \mathcal{M}$ does not entail c. MinAs have also been called *justifications* and *MUPS* in the literature [KPHS07, SC03]. The main task of axiom-pinpointing is to identify all the MinAs for a given consequence. These MinAs can be expressed as a family of subontologies, or compactly represented by a so-called pinpointing formula. Let every axiom in $\mathcal{O}$ be annotated by a unique propositional variable. For a subset $\mathcal{O}' \subseteq \mathcal{O}$, let $\mathsf{ann}(\mathcal{O}')$ be the set of annotations of the axioms in $\mathcal{O}'$. A propositional formula ϕ is called a *pinpointing formula* if for every subontology $\mathcal{O}'$ it holds that $\mathcal{O}'$ entails c iff $\mathsf{ann}(\mathcal{O}')$ entails ϕ. The set of all MinAs can be seen as a pinpointing formula in disjunctive normal form: each MinA corresponds to the conjunction of the variables that annotates it. However, other more compact representations are possible.

Since each MinA is a subset of the ontology, there are at most exponentially many of them. Moreover, verifying whether a given subontology $\mathcal{M}$ is a MinA requires only a linear number of entailment tests: verify first that $\mathcal{M}$ entails c, and for every axiom $\alpha \in \mathcal{M}$ check that $\mathcal{M} \setminus \{\alpha\}$ does not entail c. Recall that standard reasoning in $\mathcal{ALC}$ is ExpTime-complete. Then, all the MinAs for a consequence w.r.t. an $\mathcal{ALC}$ ontology can be computed in exponential time. More generally, for any expressive logic where reasoning is at least exponential, axiom-pinpointing is exactly as hard as standard reasoning. For less expressive logics, like $\mathcal{EL}$ where standard reasoning is polynomial, the simple procedure described above produces an exponential blow-up in terms of complexity. In these cases, it is important to search for more effective axiom-pinpointing techniques.

To understand the impact in the complexity that is caused by the computation of MinAs, we have studied the complexity of axiom-pinpointing in $\mathcal{EL}$. In this way, we can abstract from the cost of deciding whether an ontology entails the consequence, as in this logic this steps is polynomial. The complexity of axiom-pinpointing in $\mathcal{EL}$ was studied in detail in [PS10b]. In general, the conclusion obtained from that work is that finding all the MinAs is a hard task in computational complexity terms, even for this light-weight logic.

Perhaps the most relevant hardness result presented in [PS10b] is that deciding whether a set of MinAs is complete (i.e., contains all the MinAs) is coNP-complete. More precisely, given a set $\mathfrak{M}$ of MinAs for c w.r.t. $\mathcal{O}$, deciding whether there exists an additional MinA $\mathcal{M} \notin \mathfrak{M}$ is an NP-complete problem. A direct consequence of this result is that all MinAs cannot be enumerated in *output polynomial* time; that is, in time that is polynomial in the size of the ontology and the number of MinAs. That is, there exist consequences that have polynomially many MinAs, but computing them all requires superpolynomial time (unless PTime = NP).

As a motivation for the context-based approach, we have mentioned handling preferences. Suppose that we provide a total order among the axioms in $\mathcal{O}$ that corresponds to their preference. We might then be interested in finding the *most preferred MinA*. If the preferrence between MinAs is defined using the lexicographical ordering, then deciding whether a given MinA is the most preferred one is also a coNP-complete problem. These and many other complexity results are presented in detail in [PS10b]. One important thing to notice is that hardness arises in some special cases of $\mathcal{EL}$ already. A similar systematic analysis of the complexity of axiom-pinpointing in the family of DL-Lite description logics is presented in [PS10a].

It can be seen that most of the complexity results for axiom-pinpointing are negative, in the sense that they show that these tasks cannot be solved in polynomial time. On the positive side, it has been shown that a compact representation of the pinpointing formula can be built in polynomial time, using automata-based techniques [Pen09, Pen10]. This compact representation can be exploited by some formalisms in the context-based approach.

As mentioned before, axiom-pinpointing forms the bases for the context-based approach for interpreting annotation. We now present a general framework for context-based reasoning in which the annotations are ordered in a lattice.

4.2 Lattice-based Contexts

When considering axiom-pinpointing, we assume that every axiom is independent from all others in the sense that it can appear or be removed from an ontology without affecting the presence of any other axiom. However, for many applications it is necessary to handle some dependencies between axioms. For example, in the case of access control, a user that has access to one axiom at a security level, also has access to all other axioms at that level, and at any other less-restricted level. It thus makes sense to use the annotations to specify these dependencies.

We consider an L-annotated ontology $\mathcal{O}$, where L is an arbitrary, but fixed, finite distributive lattice. Each annotation $\ell \in L$ defines a subontology $\mathcal{O}_\ell$, called the *context* of ℓ, that contains all the axioms whose annotation is greater or equal to ℓ w.r.t. the lattice L. That is, $\mathcal{O}_\ell := \{\alpha \in \mathcal{O} \mid \mathsf{ann}(\alpha) \geq \ell\}$. Intuitively, the order in the lattice expresses a dependency between the axioms: two axioms with the same annotation must always occur together, and if an axiom α is chosen, then all axioms with an annotation larger than or equal to that of α must also be included. In the access control scenario, the elements of L describe different security levels. The larger elements correspond to a more public access (everyone has access to them), while lower elements describe more private or sensible knowledge. As the elements are ordered through a lattice, some of them might be incomparable.

The main reasoning task in this setting is to compute the contexts from which a consequence follows; e.g., the security clearance a user must possess to be able to observe the consequence. By definition, if $\ell \leq \ell'$, then $\mathcal{O}_{\ell'} \subseteq \mathcal{O}_{\ell}$. Thus, it suffices to find only the maximal annotations w.r.t. L, whose corresponding context entails the consequence. Similarly to axiom-pinpointing, one could enumerate all such labels, which could potentially be as many as the width of L. Instead, in [BKP12] we propose to compute only one annotation that expresses all these contexts. More precisely, given a consequence c, we want to compute an annotation $\mathsf{b}(c) \in L$ (called the *boundary* of c) such that for every $\ell \in L$, $\mathcal{O}_\ell$ entails c iff $\ell \leq \mathsf{b}(c)$. Unfortunately, such a boundary may not exist in general. To solve this issue, one can restrict the class of contexts to the ontologies $\mathcal{O}_\ell$ where ℓ is a *join prime* element of L; i.e., for every two elements $m, n \in L$, if $\ell \leq m \vee n$, then $\ell \leq m$ or $\ell \leq n$. Under this restriction, it is shown that the boundary always exists and is unique.

Notice that axiom-pinpointing is a special case of this setting, where L is the set of all monotone propositional formulas over the variables in $\mathsf{ann}(\mathcal{O})$, modulo logical equivalence. In this case, the join prime elements are exactly the conjunctions of propositional variables, which can be seen as subsets of $\mathcal{O}$. Moreover, the pinpointing formula and the boundary in this lattice coincide.

To compute the boundary, it is possible to use a black-box algorithm that makes repeated calls to a standard reasoner. This method would require in the worst case exponentially many calls to the reasoner, which as for axiom pinpointing, means that for expressive logics the complexity of computing the boundary is not greater than the complexity of standard reasoning. Exploiting the properties of the lattice, the computation of the boundary can be optimized in two ways. First, when trying to find one context that entails the consequence, one can ignore all axioms that would require the context to grow beyond the currently known boundary. Second, every time the boundary $\mathsf{b}(c)$ is updated, every context $\mathcal{O}_\ell$ with $\ell \leq \mathsf{b}(c)$ can be removed from the search space as they are already known to entail c. To achieve this, one can remove from $\mathcal{O}$ all axioms whose annotation is less than or equal to $\mathsf{b}(c)$. These optimizations, together with a state-of-the-art standard reasoner, have been shown to behave well in practice, even for large ontologies.

Even with all these optimizations, this black-box algorithm may still need exponential time to compute the boundary w.r.t. an $\mathcal{EL}$ ontology. Adapting the automata-based methods, it is possible to prove that the boundary is still computable in polynomial time for this logic. Notice, however, that the assumption of the lattice being distributive is fundamental for this polynomial upper bound to hold. In fact, if the set of annotations is a lattice, but not distributive, then the automata-based approach yields only a PSpace upper bound for the complexity of computing the boundary, assuming that the lattice operations are easily computable [LP14a].

So far in this section, the annotations in the axioms are used to identify different sets of axioms, and the main task is to idenfity which of these sets entail a given consequence. In the current setting, the lattice was used to express a membership dependency between the axioms. We now extend this idea to express a probabilistic dependency among the elements of the ontology.

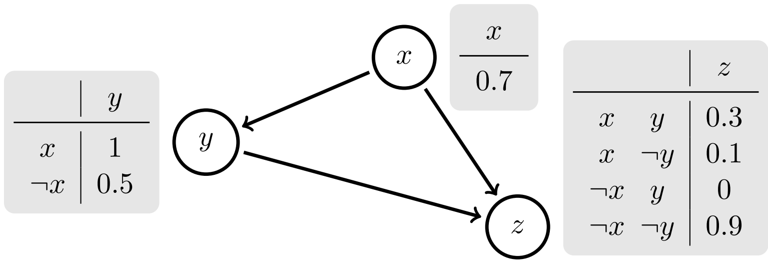

Figure 2: A simple Bayesian network over the variables $\{x, y, z\}$.

4.3 Bayesian Description Logics

While many different probabilistic variants of description logics exist, in most of them it is hard, if not impossible, to express (conditional) probabilistic dependencies between the different axioms in the ontology. Much like in Section 4.2, Bayesian description logics use annotations to divide the ontology into subsets of axioms, which are called *contexts*. The probabilistic component of the logic is given through a Bayesian network, which expresses the joint probability distribution of the contexts in a compact way [Dar09]. From this joint distribution, the conditional dependencies between the contexts, and hence the probability of a consequence, can also be computed.

A *Bayesian network* (BN) is a pair $\mathcal{B} = (G, \Phi)$, where $G = (V, E)$ is a directed acyclic graph (DAG), whose nodes represent Boolean random variables, and Φ is a family of conditional probability distributions containing one distribution $P_{\mathcal{B}}(x \mid \pi(x))$ of x given its parents $\pi(x)$ for every $x \in V$. In this case, we say that $\mathcal{B}$ is a BN *over* V. The DAG G is a graphical representation of a set of conditional independence assumptions: every node from V is independent from its non-descendants, given its parents. Under this assumption, the joint probability distribution (JPD) of V defined by $\mathcal{B}$ is obtained through the *chain rule* $P_{\mathcal{B}}(V) = \prod_{x \in V} P_{\mathcal{B}}(x \mid \pi(x))$. Figure 2 depicts a simple BN over $V = \{x, y, z\}$. This BN expresses, for instance, that the probability of observing variable y given that x was not observed is $P(y \mid \neg x) = 0.5$.

Given a finite set of Boolean variables V, let $\mathsf{con}(V)$ be the set of all consistent sets of literals from V. The elements of $\mathsf{con}(V)$ are called *contexts*. A *Bayesian knowledge base* (KB) is a pair $\mathcal{K} = (\mathcal{O}, \mathcal{B})$, where $\mathcal{O}$ is a $\mathsf{con}(V)$-annotated ontology and $\mathcal{B}$ is a BN over V. The main idea behind this logic is that the ontology expresses knowledge that is certain to hold, in different contexts. Since the elements of V are random variables, the precise context, and hence also its consequences, have an associated uncertainty, expressed by the BN.

Every context $\kappa \in \mathsf{con}(V)$ defines a subontology of $\mathcal{O}$ that contains all axioms that must be true in this context; more precisely, $\mathcal{O}_\kappa := \{\alpha \in \mathcal{O} \mid \mathsf{ann}(\alpha) \subseteq \kappa\}$. We can see a valuation $\mathcal{V}$ of the variables in V as the set of all literals that it maps to true; hence valuations are also contexts from $\mathsf{con}(V)$. Thus, in particular we consider the subontologies $\mathcal{O}_{\mathcal{V}}$, where $\mathcal{V}$ is a valuation of the variables in V.

In Bayesian DLs, we are interested in computing the probability of a consequence to hold. That is, given a consequence c, we want to compute the probability $P_{\mathcal{K}}(c)$ of observing some context κ such that $\mathcal{O}_\kappa$ entails c. It can be shown that it suffices to

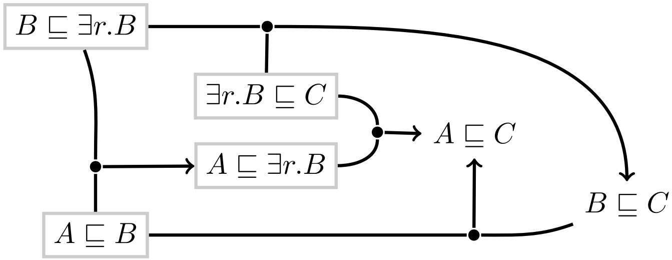

Figure 3: Proof structure of $\mathcal{O}_2$.

deduce the set of all valuations that entail the consequence. The desired probability is then the sum of the probabilities of these valuations, given by the BN $\mathcal{B}$. More precisely, $P_{\mathcal{K}}(c) = \sum_{\mathcal{O}_{\mathcal{V}} \models c} P_{\mathcal{B}}(\mathcal{V})$ [CP14a].

Notice that the set of all propositional formulas over V forms a distributive lattice with the order induced by formula entailment. Moreover, the set of all valuations of the variables in V correspond to the join prime elements of this lattice. Hence, we can use the approach described in Section 4.2 to obtain a boundary for c w.r.t. $\mathcal{O}$ in this lattice. This boundary $\mathsf{b}(c)$ has the property that for every valuation $\mathcal{V}$, $\mathcal{V} \models \mathsf{b}(c)$ if and only if $\mathcal{O}_{\mathcal{V}} \models c$. The probability computation then reduces to adding the probabilities of all valuations that entail $\mathsf{b}(c)$. Since there are potentially exponentially many such valuations, this approach runs in exponential time even if the entailment test at the underlying logic is tractable.

For the Bayesian extension of the light-weight DL $\mathcal{EL}$, the computation of the probability of an entailment can be improved by adapting the completion algorithm. The main idea is to use the completion algorithm to encode the logical entailment test into a Bayesian network of size polynomial on $|\mathcal{O}|$. The probability $\mathcal{P}_{\mathcal{K}}(c)$ can then be computed using standard probabilistic inferences over this BN. The reduction is based on the so-called *proof structure* of $\mathcal{O}$. Essentially, the proof structure of $\mathcal{O}$ is a directed hyper-graph whose nodes are elements of the set $\mathsf{comp}(\mathcal{O})$ and whose hyper-edges express all the possible rule applications that can be performed within $\mathsf{comp}(\mathcal{O})$. For example, if $\{A \sqsubseteq B, B \sqsubseteq C, A \sqsubseteq C\} \in \mathsf{comp}(\mathcal{O})$, then the proof structure will contain a directed hyper-edge from $\{A \sqsubseteq B, B \sqsubseteq C\}$ to $A \sqsubseteq C$, expressing that from the two former axioms the latter is derived by a rule application. Figure 3 depicts the proof structure of the ontology

$$\mathcal{O}_2 := \{\langle A \sqsubseteq B : \{x, y\}\rangle\,,\ \langle B \sqsubseteq \exists r.B : \{\neg z\}\rangle\,,\ \langle A \sqsubseteq \exists r.B : \{\neg x\}\rangle\,,\ \langle \exists r.B \sqsubseteq C : \{y\}\rangle\},$$

where the original axioms are surrounded by a grey box. This hyper-graph can be used to find all the MinAs for a subsumption relation. These MinAs correspond exactly to all the minimal sets of axioms that can reach the given consequence in the proof structure. Notice that the proof structure contains more information than what is obtained by the completion algorithm alone; in particular, it stores all the possible causes for each entailed consequence, while the completion algorithm can preserve at most one. In this section, we are not interested in finding the MinAs for a consequence, but rather in the probability of observing one of them.

To obtain the probability of a consequence appearing in the proof structure w.r.t. a

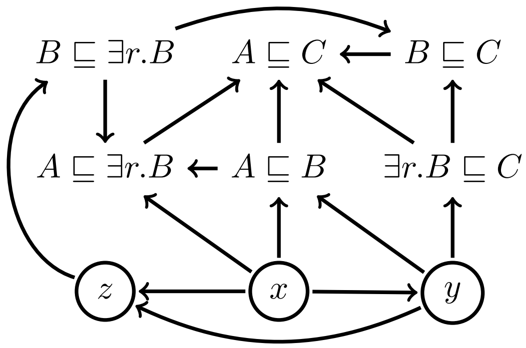

Figure 4: DAG of the BN obtained from $\mathcal{K}_2$.

KB $\mathcal{K} = (\mathcal{O}, \mathcal{B})$, we combine the original BN $\mathcal{B}$ with the proof structure to obtain a new BN expressing the probabilistic information of all the consequences of $\mathcal{O}$. Notice that the proof structure may be cyclic. Thus, as a first step we need to unravel this proof structure into an acyclic hyper-graph. Recall that each hyper-edge represents a possible rule application in the completion algorithm; moreover, every possible derivation of any element in $\mathsf{comp}(\mathcal{O})$ can be obtained after at most $|\mathsf{comp}(\mathcal{O})|$ rule applications. Thus, unraveling the proof structure to at most depth $|\mathsf{comp}(\mathcal{O})|$ suffices for guaranteeing that all the relevant information for deriving each implicit consequence is preserved. This acyclic hyper-graph is then transformed into a DAG by combining together all the hyper-edges that share the same head. A simple conditional probability table is used to guarantee that the parents of this head node are combined in the right way. All the details are explained in [CP14c].

Consider for example the KB $\mathcal{K}_2 = (\mathcal{O}_2, \mathcal{B}_2)$, where $\mathcal{B}_2$ is the BN shown in Figure 2. Notice that the proof structure of $\mathcal{O}_2$ is already acyclic, so there is no need of unraveling it. The graphical component of the BN obtained by the reduction sketched above appears in Figure 4. Notice that using this transformation directly may lead to some nodes in the DAG having many parents, as is the case of the node $A \sqsubseteq C$ in Figure 4, which has four parents. Since the conditional probability tables of a BN grow exponentially on the maximum number of parents of the nodes in its DAG, this might yield an exponential blow-up in the reduction. Fortunately, this blow-up can be avoided by introducing polynomially many auxiliary nodes in such a way that all the nodes that did not belong to the original BN are guaranteed to have at most two parents in the final DAG. Hence, the size of the conditional probability tables is bounded by the size of the tables of the BN from the KB [CP14c].

Let $\mathcal{B}_\mathcal{K}$ be the BN obtained from the KB $\mathcal{K} = (\mathcal{O}, \mathcal{B})$ through this process, and c a consequence of $\mathcal{O}$. It then holds that $P_\mathcal{K}(c) = P_{\mathcal{B}_\mathcal{K}}(x_c)$, where x_c is the node in $\mathcal{B}_\mathcal{K}$ that corresponds to the consequence c. Since the size of $\mathcal{B}_\mathcal{K}$ is polynomial on the size of $\mathcal{O}$ and $\mathcal{B}$, this yields a polynomial-time reduction from the problem of computing the probability of a consequence to probabilistic inferences in a BN. We can use this reduction to find tight complexity bounds for this and other related reasoning problems in the Bayesian extension of $\mathcal{EL}$.

As a last example of the context-based approach we now look at the problem of extracting meaningful consequences from an ontology that is known to contain errors.

4.4 Error-Tolerant Reasoning

The success of DLs as knowledge representation formalisms has meant that more and larger ontologies are being built for representing various knowledge domains. To build these ontologies, domain experts and ontology experts have to interact to formalize the relevant knowledge of the area. Due to misunderstandings between the domain expert and the ontology editor, disagreements between experts, or incorrect translations of notions into the logical knowledge (among many other causes), ontology development and maintenance are very prone to errors. As ontologies get larger, understanding and correcting these errors becomes harder. Moreover, well-managed ontologies usually have long version-publishing cycles. For example, new versions of SNOMED are published only twice per year; this means that one should expect to wait at least one year before an error is corrected.

The goal of an ontology is not only to represent the knowledge of a domain, but also to be able to reason with this knowledge and extract meaningful consequences from it. When an ontology is found to be erroneous, one cannot expect all the applications based on this ontology to stop working and wait until all the errors are corrected. On the other hand, it would be very bad practice to simply ignore the known error and continue using the ontology as if it was correct. The goal of error-tolerant reasoning is to extract meaningful consequences from an ontology while avoiding all the known errors.

A special case of error-tolerant reasoning has been studied previously in the form of *inconsistency-tolerant reasoning* [ABC99, Ber11, BR13, Ros11]. In that setting, the only error considered is the inconsistency of the ontology, and this error is always assumed to be caused by incorrect assertions in the ABox; that is, the TBox is considered to always be correct. We generalize this idea to allow other kinds of errors. For example, an $\mathcal{EL}$ ontology is always consistent, but it might entail an unwanted subsumption relation. Moreover, we do not expect the TBox to be necessarily correct; the error might be caused by some of the terminological axioms of the ontology.

Error-tolerant reasoning is based on the notion of a repair: a maximal sub-ontology that does not entail a consequence. More formally, a *repair* for a consequence c w.r.t. the ontology $\mathcal{O}$ is a subset $\mathcal{R} \subseteq \mathcal{O}$ such that $\mathcal{R}$ does not entail c, and every strict superset $\mathcal{R} \subset \mathcal{R}' \subseteq \mathcal{O}$ entails it. Repairs are the dual notion to MinAs introduced in Section 4.1. In fact, it is well known that the set of all MinAs can be computed from all the repairs and *vice versa* [LS05, SC03]. However, this computation might require super-polynomial time [FK96]. We use the expression $\mathfrak{R}(\mathcal{O}, c)$ to denote the set of all repairs for c w.r.t. $\mathcal{O}$.

Many different error-tolerant reasoning tasks can be defined, depending on the intended application, and the desired properties of the answers. We study the three most common ones, known as brave, cautious, and IAR entailments. Suppose that c is an erroneous consequence of $\mathcal{O}$. A consequence d is *bravely entailed* by $\mathcal{O}$ w.r.t. c if there exists a repair $\mathcal{R} \in \mathfrak{R}(\mathcal{O}, c)$ that entails d. In other words, d is a brave entailment if it possible to remove the error c from $\mathcal{O}$ in such a way that d still holds. This kind of entailments is useful e.g., when trying to understand the relationship between different consequences from an ontology. Notice, however, that brave entailments are not logically closed; it is possible, for example, that $A \sqsubseteq B$ and $B \sqsubseteq C$ are both brave entailments, while $A \sqsubseteq C$ is not. Thus, one must be careful when considering this notion of error-tolerant reasoning.

A stronger notion is that of cautious entailments. We say that d is *cautiously entailed* by $\mathcal{O}$ w.r.t. c if every repair $\mathcal{R} \in \mathfrak{R}(\mathcal{O}, c)$ entails d; that is, if d can still be derived regardless of the repair chosen to remove the error c. Cautious entailments are guaranteed to hold after the process of removing the error c. Thus, they are not affected by the causes of the error, and can be thought to be correct. Finally, an *IAR entailment* is one that follows from the intersection of all the repairs for c w.r.t. $\mathcal{O}$. This notion of entailment was originally introduced in [LLR+10] to regain tractability in inconsistency-tolerant reasoning for an inexpressive DL. It is easy to see that every IAR entailment is also a cautious entailment, and every cautious entailment is also brave, but the converse implications do not hold in general. In contrast to brave entailments, cautious and IAR entailment are also closed under logical deduction.

We have shown that these three kinds of reasoning tasks are unfeasible already for $\mathcal{EL}$. Moreover, cautious and brave entailments cannot be decided in time polynomial on the size of the ontology and the number of repairs. This means that even if the consequence c has only polynomially many repairs, one would still need super-polynomial time to decide whether d is bravely or cautiously entailed [LP14b].

In order to solve these reasoning problems efficiently, we propose to compile the information about all the repairs into an annotated ontology. The idea is that each repair corresponds to one context in this annotated ontology. The set of contexts that entail the consequence d (that is, the boundary of d w.r.t. this annotated ontology) can be easily used to determine whether d is bravely or cautiously entailed. Moreover, IAR entailments can be easily decided through standard reasoning over the sub-ontology composed of those axioms that belong to all contexts. Thus, the main idea proposed for improving the reasoning time for error-tolerant tasks is to reduce the problem to one similar to the lattice-based contexts described in Section 4.2. Notice, however, that the maximality condition in the notion of repairs guarantees that no repair is a subset of another. Thus, in this case, the lattice obtained has a simplified shape that can be exploited for further optimizations of the reasoning tools.

Obviously, the compilation step, in which all the repairs need to be computed in advance, may be a computationally expensive one. Under the assumption that many different error-tolerant reasoning queries are made over a single erroneous ontology, the cost of this computation is soon compensated by the effort saved at each individual error-tolerant reasoning task. Moreover, the compilation can be made off-line, saving the users some waiting time to get answers to their entailment tests. An additional benefit of this approach is that it can be exploited for improving the ontology update process [Thu15].

With this we conclude the section on the context-based approach for reasoning with annotated ontologies. While the reasoning problems described throughout this section are very different, they are all based on the basic task of identifying the contexts that entail a given consequence. Clearly, the list of reasoning problems that belong to the context-based approach is not complete, and one can think of many other problems that can be described using this idea. Many of the methods that have been developed for the special cases presented in this section can be generalized to solve also other reasoning tasks that follow the context-based approach. One remaining task for future work is to describe a general framework that can be used to identify the methods that can be used in specific circumstances.

5 Conclusions

As description logics become better understood and a more popular choice for modelling the knowledge used by practical applications, the limitations of basing these formalisms in classical logic become more apparent. The knowledge expressed in an ontology often needs to be extended with additional information that affects how this knowledge is treated. Examples of such additional information include the origin, or age of an axioms, but also a degree of trust, or the level of certainty that one has that the axiom is correct, to name just a few of the many possible.

One of the main causes for the success of DLs as knowledge representation languages is their formal and well-understood semantics. It is thus important that the annotations are given also a precise meaning that guarantees that the annotated ontology is not ambiguous. Clearly, the meaning of the annotations depends on what they are intended to represent (e.g., provenance, time, probabilities, etc.), which makes it impossible to provide one general semantics for annotated ontologies. Rather than attempting such a task, we have characterized all the different approaches for interpreting annotations into two large classes. In a nutshell, the difference between these classes is whether they modify the underlying logical formalism, or require additional work to be done on top of standard logical reasoning.

The semantic approach refers to all those formalisms in which the logical formalism is affected. As part of this approach, we studied thoroughly the case of fuzzy description logics. These logics change the semantics of classical DLs by interpreting concepts and roles as *fuzzy* sets and binary relations, respectively, as opposed to classical sets and relations. As the computational properties of these logics were not well-understood, our work focused on characterizing the family of fuzzy DLs with decidable reasoning problems, and finding tight complexity bounds for them. Briefly, we showed that these logics become easily undecidable, but in the decidable cases, the complexity is typically not affected by the change in semantics. To the best of our knowledge, the only exception to this rule found so far is the finitely valued $\mathcal{EL}$, in which concept subsumption is decidable, but coNP-hard [BCP14]. We conjecture that this problem is in fact ExpTime-complete.

In contrast, in the context-based approach the semantics of the logic remain unchanged, but the reasoning problem is modified. In this setting, the annotations are used to define a class of sub-ontologies, called contexts. The reasoning tasks is to identify, and in some cases make computations over the class of all the sub-ontologies that entail (in the classical sense) a given consequence. Within the context-based approach, we have studied several formalisms and applications, including axiom-pinpointing, access-control, a variant of probabilistic knowledge representation, and error-tolerant reasoning. While they may seem very different at first sight, they all share as a core reasoning task the need of finding contexts that entail a consequence. The main differences between these formalisms are how the different contexts are defined, and the additional computation required once these have been identified. Many of the methods that we have developed can be adapted to other interpretations of the contexts without mayor changes.

It is worth noting that, although the two approaches might appear to be very different, the distinction between them is not always clear. For example, we have classified Bayesian DLs as part of the context-based approach because reasoning corresponds to finding the probability (defined externally through a Bayesian network) of the boundary of a consequence. However, the original definition of these logics uses a multiple-world semantics that would situate them as part of the semantic approach [CP14b]. Conversely, fuzzy DLs are an obvious choice for the semantic approach; their semantics require a different interpretation of concepts and roles. Still, for some cases based on the finitely-valued Gödel or Zadeh semantics it is possible to equivalently define these logics using the context-based approach [FP12]. A similar behaviour had been previously observed for possibilistic extensions of DLs [Hol95].

One important feature of the context-based approach as a whole is that it divides the knowledge into two separated components: the logical component, which is in charge of detecting which contexts entail the consequence, and the annotation component, that performs additional computations over these contexts. Due to this separation, the methods developed for one formalism using this approach can usually be adapted to other formalisms in the same approach. This does not mean, however, that it suffices to study only one formalism to solve all the others. The best solutions are developed exploiting the properties of the logical and the annotation components simultaneously, as shown in Sections 4.2 and 4.3. Another advantage of the separation between the logical and the annotation component is that the ideas developed do not apply to DLs only. Indeed, the main assumption required throughout Section 4 is that there is a monotone entailment relation between ontologies and consequences; that is, if the ontology $\mathcal{O}$ entails c, then every superset of $\mathcal{O}$ must also entail c. For any ontological language satisfying this condition, the different annotated extensions can be defined and treated accordingly.

Unfortunately, the methods developed for a formalism in the semantic-approach cannot typically be adapted to others in the same approach. This is caused by the fact that the newly introduced semantics can greatly differ from each other. An obvious example is that the reasoning methods developed for finitely-valued fuzzy DLs in Section 3.2 cannot work for their infinitely valued counterparts, as the latter have been shown to be undecidable (Section 3.1). Thus, in this approach, new reasoning techniques need to be developed for each defined formalism.

Any study of the different ways in which annotation ontologies can be interpreted is necessarily incomplete. There will always exist new ways to interpret and use the annotations associated to the axioms. As such, this work is not intended to be a comprehensive view on all the known semantics for annotations, but rather to provide a deeper view into the properties of some of the cases that have been recently studied. We expect that the lessons learned during this study will be helpful for the study of future formalisms.

While the results presented here are mainly theoretical, first efforts regarding implementation of tools for reasoning in these logics have already been made. A query answering tool for finitely-valued fuzzy DLs was presented in [MPT14, MT14]. Many different tools have been implemented for axiom-pinpointing [Sun08, SV09, Lud14, KPHS07] and reasoning with lattice-ordered contexts [BKP12]. A tool for performing error-tolerant reasoning through the compilation approach was presented in [LP14b]. Finally, a prototypical implementation of the algorithms for reasoning in the Bayesian extension of $\mathcal{EL}$ is currently under development.

Among the many possible paths for future work, it is worth mentioning the combination of logics. The work on Bayesian DLs from Section 4.3 suggests an approach for combining an arbitrary monotonic logic (in this case a DL) with an extension of propositional logic (here, probabilistic logic). In this combination, the two components are detached, and the overall complexity of reasoning is bounded by the most expensive of the components. If this intuition holds in general, then it can be used to produce combined logics satisfying some desirable properties.

References

[ABC99] Marcelo Arenas, Leopoldo Bertossi, and Jan Chomicki. Consistent query answers in inconsistent databases. In *Proceedings of the 18th ACM SIGMOD-SIGACT-SIGART symposium on Principles of Database Systems (PODS 1999)*, pages 68–79. ACM, 1999.

[BBL05] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI'05)*, Edinburgh, UK, 2005. Morgan Kaufmann.

[BBS11] Fernando Bobillo, Félix Bou, and Umberto Straccia. On the failure of the finite model property in some fuzzy description logics. *Fuzzy Sets and Systems*, 172(1):1–12, 2011.

[BCM+07] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2nd edition, 2007.

[BCP14] Stefan Borgwardt, Marco Cerami, and Rafael Peñaloza. Many-valued horn logic is hard. In Thomas Lukasiewicz, Rafael Peñaloza, and Anni-Yasmin Turhan, editors, *Proceedings of the 1st International Workshop on Logics for Reasoning about Preferences, Uncertainty, and Vagueness (PRUV'14)*, volume 1205 of *CEUR Workshop Proceedings*, pages 52–58, 2014.

[BDGRS09] Fernando Bobillo, Miguel Delgado, Juan Gómez-Romero, and Umberto Straccia. Fuzzy description logics under Gödel semantics. *International Journal of Approximate Reasoning*, 50(3):494–514, 2009.

[BDGRS12] Fernando Bobillo, Miguel Delgado, Juan Gómez-Romero, and Umberto Straccia. Joining Gödel and Zadeh fuzzy logics in fuzzy description logics. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 20(4):475–508, 2012.

[BDP12] Stefan Borgwardt, Felix Distel, and Rafael Peñaloza. How fuzzy is my fuzzy description logic? In Bernhard Gramlich, Dale Miller, and Uli Sattler, editors, *Proceedings of the 6th International Joint Conference on Automated Reasoning (IJCAR'12)*, volume 7364 of *Lecture Notes in Artificial Intelligence*, pages 82–96. Springer, 2012.

[BDP14] Stefan Borgwardt, Felix Distel, and Rafael Peñaloza. Decidable Gödel description logics without the finitely-valued model property. In Chitta Baral, Giuseppe De Giacomo, and Thomas Eiter, editors, *Proceedings of the 14th International Conference on Principles of Knowledge Representation and Reasoning (KR 2014)*, pages 228–237. AAAI Press, 2014.

[BDP15] Stefan Borgwardt, Felix Distel, and Rafael Peñaloza. The limits of decidability in fuzzy description logics with general concept inclusions. *Artificial Intelligence*, 218:23–55, 2015.

[Ber11] Leopoldo Bertossi. Database repairing and consistent query answering. *Synthesis Lectures on Data Management*, 3(5):1–121, 2011.

[BHP08] Franz Baader, Jan Hladik, and Rafael Peñaloza. Automata can show PSPACE results for description logics. *Information and Computation*, 206(9–10):1045–1056, 2008.

[BKP12] Franz Baader, Martin Knechtel, and Rafael Peñaloza. Context-dependent views to axioms and consequences of semantic web ontologies. *Journal of Web Semantics*, 12–13:22–40, 2012. Available at http://dx.doi.org/10.1016/j.websem.2011.11.006.

[BLP14] Stefan Borgwardt, José A. Leyva Galano, and Rafael Peñaloza. The fuzzy description logic $\mathsf{G}\text{-}\mathcal{FL}_0$ with greatest fixed-point semantics. In Eduardo Fermé and João Leite, editors, *Proceedings of the 14th European Conference on Logics in Artificial Intelligence (JELIA'14)*, volume 8761 of *Lecture Notes in Artificial Intelligence*, pages 62–76, Funchal, Portugal, 2014. Springer.

[BP10a] Franz Baader and Rafael Peñaloza. Automata-based axiom pinpointing. *Journal of Automated Reasoning*, 45(2):91–129, August 2010.

[BP10b] Franz Baader and Rafael Peñaloza. Axiom pinpointing in general tableaux. *Journal of Logic and Computation*, 20(1):5–34, 2010.

[BP11a] Franz Baader and Rafael Peñaloza. Are fuzzy description logics with general concept inclusion axioms decidable? In *Proceedings of the 2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'11)*, pages 1735–1742. IEEE Computer Society Press, 2011.

[BP11b] Franz Baader and Rafael Peñaloza. GCIs make reasoning in fuzzy DL with the product t-norm undecidable. In Riccardo Rosati, Sebastian Rudolph, and Michael Zakharyaschev, editors, *Proceedings of the 2011 International Workshop in Description Logics (DL2011)*, volume 745 of *CEUR Workshop Proceedings*, pages 37–47, 2011.

[BP11c] Franz Baader and Rafael Peñaloza. On the undecidability of fuzzy description logics with GCIs and product t-norm. In Cesare Tinelli and Viorica Sofronie-Stokkermans, editors, *Proceedings of the 8th International Symposium on Frontiers of Combining Systems (FroCoS'11)*, volume 6989 of *Lecture Notes in Computer Science*, pages 55–70. Springer, 2011.

[BP11d] Stefan Borgwardt and Rafael Peñaloza. Description logics over lattices with multi-valued ontologies. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11)*, pages 768–773. AAAI Press, 2011.

[BP12a] Stefan Borgwardt and Rafael Peñaloza. A tableau algorithm for fuzzy description logics over residuated de morgan lattices. In Markus Krötzsch and Umberto Straccia, editors, *Proceedings of the 6th International Conference on Web Reasoning and Rule Systems (RR'12)*, volume 7497 of *Lecture Notes in Computer Science*, pages 9–24. Springer, 2012.

[BP12b] Stefan Borgwardt and Rafael Peñaloza. Undecidability of fuzzy description logics. In Gerhard Brewka, Thomas Eiter, and Sheila A. McIlraith, editors, *Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR 2012)*, pages 232–242, Rome, Italy, 2012. AAAI Press.

[BP13a] Stefan Borgwardt and Rafael Peñaloza. The complexity of lattice-based fuzzy description logics. *Journal on Data Semantics*, 2(1):1–19, 2013.

[BP13b] Stefan Borgwardt and Rafael Peñaloza. Positive subsumption in fuzzy $\mathcal{EL}$ with general t-norms. In Francesca Rossi, editor, *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*, pages 789–795, Beijing, China, 2013. AAAI Press.

[BP14a] Stefan Borgwardt and Rafael Peñaloza. Consistency reasoning in lattice-based fuzzy description logics. *International Journal of Approximate Reasoning*, 55(9):1917–1938, 2014.

[BP14b] Stefan Borgwardt and Rafael Peñaloza. Finite lattices do not make reasoning in ALCOI harder. In F. Bobillo, R.N. Carvalho, P.C.G. da Costa, C. d'Amato, N. Fanizzi, K.B. Laskey, K.J. Laskey, Th. Lukasiewicz, M. Nickles, and M. Pool, editors, *Uncertainty Reasoning for the Semantic Web III*, volume 8816 of *Lecture Notes in Computer Science*, pages 122–141. Springer, 2014. Revised Selected Papers from the ISWC International Workshops URSW 2011 - 2013.

[BPS07] Franz Baader, Rafael Peñaloza, and Boontawee Suntisrivaraporn. Pinpointing in the description logic $\mathcal{EL}^+$. In *Proceedings of the 30th German Conference on Artificial Intelligence (KI2007)*, volume 4667 of *Lecture Notes in Artificial Intelligence*, pages 52–67, Osnabrück, Germany, 2007. Springer.

[BR13] Meghyn Bienvenu and Riccardo Rosati. Tractable approximations of consistent query answering for robust ontology-based data access. In Francesca Rossi, editor, *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*. AAAI Press, 2013.

[BS07] Fernando Bobillo and Umberto Straccia. A fuzzy description logic with product t-norm. In *Proceedings of the 2007 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'07)*, pages 1–6. IEEE Computer Society Press, 2007.

[BS09] Fernando Bobillo and Umberto Straccia. Fuzzy description logics with general t-norms and datatypes. *Fuzzy Sets and Systems*, 160(23):3382–3402, 2009.

[BS11] Fernando Bobillo and Umberto Straccia. Reasoning with the finitely many-valued Łukasiewicz fuzzy description logic $\mathcal{SROIQ}$. *Information Sciences*, 181:758–778, 2011.

[BS13] Fernando Bobillo and Umberto Straccia. Finite fuzzy description logics and crisp representations. In Fernando Bobillo, Paulo C. G. da Costa, Claudia d'Amato, Nicola Fanizzi, Kathryn Laskey, Ken Laskey, Thomas Lukasiewicz, Matthias Nickles, and Michael Pool, editors, *Uncertainty Reasoning for the Semantic Web II*, volume 7123 of *Lecture Notes in Computer Science*, pages 102–121. Springer, 2013.

[CP14a] İsmail İlkan Ceylan and Rafael Peñaloza. The bayesian description logic BEL. In Stephane Demri, Deepak Kapur, and Christoph Weidenbach, editors, *Proceedings of the 7th International Joint Conference on Automated Reasoning (IJCAR'14)*, volume 8562 of *Lecture Notes in Computer Science*, pages 480–494. Springer, 2014.

[CP14b] İsmail İlkan Ceylan and Rafael Peñaloza. Bayesian Description Logics. In Meghyn Bienvenu, Magdalena Ortiz, Riccardo Rosati, and Mantas Simkus, editors, *Proceedings of the 2014 International Workshop in Description Logics (DL2014)*, volume 1193, pages 447–458. CEUR Workshop Proceedings, 2014.

[CP14c] İsmail İlkan Ceylan and Rafael Peñaloza. Tight complexity bounds for reasoning in the description logic $\mathcal{BEL}$. In Eduardo Fermé and João Leite, editors, *Proceedings of the 14th European Conference on Logics in Artificial Intelligence (JELIA'14)*, volume 8761 of *Lecture Notes in Computer Science*, pages 77–91. Springer, 2014.

[CS13] Marco Cerami and Umberto Straccia. On the (un)decidability of fuzzy description logics under Łukasiewicz t-norm. *Information Sciences*, 227:1–21, 2013.

[Dar09] Adnan Darwiche. *Modeling and Reasoning with Bayesian Networks.* Cambridge University Press, 2009.

[dFL08] Claudia d'Amato, Nicola Fanizzi, and Thomas Lukasiewicz. Tractable reasoning with bayesian description logics. In Sergio Greco and Thomas Lukasiewicz, editors, *Proceedings of the Second International Conference on Scalable Uncertainty Management (SUM'08)*, volume 5291 of *Lecture Notes in Computer Science*, pages 146–159. Springer, 2008.

[DK93] Gert De Cooman and Etienne E. Kerre. Order norms on bounded partially ordered sets. *Journal of Fuzzy Mathematics*, 2:281–310, 1993.

[DM00] Francesco M. Donini and Fabio Massacci. Exptime tableaux for $\mathcal{ALC}$. *Artificial Intelligence*, 124(1):87–138, 2000.

[DSSS09] Renata Dividino, Sergej Sizov, Steffen Staab, and Bernhard Schueler. Querying for provenance, trust, uncertainty and other meta knowledge in RDF. *Journal of Web Semantics*, 7(3):204–219, 2009.

[FK96] Michael L. Fredman and Leonid Khachiyan. On the complexity of dualization of monotone disjunctive normal forms. *Journal of Algorithms*, 21(3):618–628, 1996.

[FP12] Weili Fu and Rafael Peñaloza. Adding context to tableaux for DLs. In Yevgeny Kazakov, Domenico Lembo, and Frank Wolter, editors, *Proceedings of the 2012 International Workshop on Description Logics (DL'12)*, volume 846 of *CEUR-WS*, Rome, Italy, 2012.

[GJKO07] Nikolaos Galatos, Peter Jipsen, Tomasz Kowalski, and Hiroakira Ono. *Residuated Lattices: An Algebraic Glimpse at Substructural Logics*, volume 151 of *Studies in Logic and the Foundations of Mathematics*. Elsevier, 2007.

[Háj01] Petr Hájek. *Metamathematics of Fuzzy Logic (Trends in Logic)*. Springer, 2001.

[Háj05] Petr Hájek. Making fuzzy description logic more general. *Fuzzy Sets and Systems*, 154(1):1–15, 2005.

[Hol95] Bernhard Hollunder. An alternative proof method for possibilistic logic and its application to terminological logics. *International Journal of Approximate Reasoning*, 12(2):85–109, 1995.

[Jae94] Manfred Jaeger. Probabilistic reasoning in terminological logics. In Jon Doyle, Erik Sandewall, and Pietro Torasso, editors, *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning (KR 1994)*, pages 305–316. Morgan Kaufmann, 1994.

[KMP00] Erich Peter Klement, Radko Mesiar, and Endre Pap. *Triangular Norms.* Trends in Logic, Studia Logica Library. Springer, 2000.

[KP08] Pavel Klinov and Bijan Parsia. Optimization and evaluation of reasoning in probabilistic description logic: Towards a systematic approach. In Amit P. Sheth, Steffen Staab, Mike Dean, Massimo Paolucci, Diana Maynard, Timothy W. Finin, and Krishnaprasad Thirunarayan, editors, *Proceedings of the 7th International Semantic Web Conference ISWC 2008*, volume 5318 of *Lecture Notes in Computer Science*, pages 213–228. Springer, 2008.

[KP13] Pavel Klinov and Bijan Parsia. Understanding a probabilistic description logic via connections to first-order logic of probability. In Fernando Bobillo, Paulo Cesar G. da Costa, Claudia d'Amato, Nicola Fanizzi, Kathryn B. Laskey, Kenneth J. Laskey, Thomas Lukasiewicz, Matthias Nickles, and Michael Pool, editors, *Uncertainty Reasoning for the Semantic Web II, Revised Selected Papers ffrom URSW 2008-2010 and UniDL 2010*, volume 7123 of *Lecture Notes in Computer Science*, pages 41–58. Springer, 2013.

[KPHS07] Aditya Kalyanpur, Bijan Parsia, Matthew Horridge, and Evren Sirin. Finding all justifications of OWL DL entailments. In *Proceedings of the 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference, ISWC 2007, ASWC 2007*, volume 4825 of *Lecture Notes in Computer Science*, pages 267–280. Springer, 2007.

[LLR+10] Domenico Lembo, Maurizio Lenzerini, Riccardo Rosati, Marco Ruzzi, and Domenico Fabio Savo. Inconsistency-tolerant semantics for description logics. In Pascal Hitzler and Thomas Lukasiewicz, editors, *Proceedings of the 4th International Conference on Web Reasoning and Rule Systems (RR'10)*, volume 6333 of *Lecture Notes in Computer Science*, pages 103–117. Springer, 2010.

[LP14a] Karsten Lehmann and Rafael Peñaloza. The complexity of computing the behaviour of lattice automata on infinite trees. *Theoretical Computer Science*, 534:53–68, May 2014.

[LP14b] Michel Ludwig and Rafael Peñaloza. Error-tolerant reasoning in the description logic EL. In Eduardo Fermé and João Leite, editors, *Proceedings of the 14th European Conference on Logics in Artificial Intelligence (JELIA'14)*, volume 8761 of *Lecture Notes in Computer Science*, pages 107–121. Springer, 2014.

[LS00] Carsten Lutz and Ulrike Sattler. Mary likes all cats. In F. Baader and U. Sattler, editors, *Proceedings of the 2000 International Workshop in Description Logics (DL2000)*, number 33 in CEUR Workshop Proceedings, pages 213–226, Aachen, Germany, August 2000. RWTH Aachen.

[LS05] Mark H. Liffiton and Karem A. Sakallah. On finding all minimally unsatisfiable subformulas. In Fahiem Bacchus and Toby Walsh, editors, *Proceedings* *of the 8th International Conference on Theory and Applications of Satisfiability Testing (SAT 2005)*, volume 3569 of *Lecture Notes in Computer Science*, pages 173–186. Springer, 2005.

[LS08] Thomas Lukasiewicz and Umberto Straccia. Managing uncertainty and vagueness in description logics for the semantic web. *Journal of Web Semantics*, 6(4):291–308, 2008.

[LS10] Carsten Lutz and Lutz Schröder. Probabilistic description logics for subjective uncertainty. In Fangzhen Lin, Ulrike Sattler, and Miroslaw Truszczynski, editors, *Proceedings of the Twelfth International Conference on Principles of Knowledge Representation and Reasoning (KR 2010)*. AAAI Press, 2010.

[Lud14] Michel Ludwig. Just: a tool for computing justifications w.r.t. $\mathcal{EL}$ ontologies. In *Proceedings of the 3rd International Workshop on OWL Reasoner Evaluation (ORE 2014)*, 2014.

[Luk08] Thomas Lukasiewicz. Expressive probabilistic description logics. *Artificial Intelligence*, 172(6-7):852–883, 2008.

[MPT14] Theofilos Mailis, Rafael Peñaloza, and Anni-Yasmin Turhan. Conjunctive query answering in finitely-valued fuzzy description logics. In Roman Kontchakov and Marie-Laure Mugnier, editors, *Proceedings of the 8th International Conference on Web Reasoning and Rule Systems (RR'14)*, volume 8741 of *Lecture Notes in Computer Science*, pages 124–139. Springer, 2014.

[MS57] Paul S. Mostert and Allen L. Shields. On the structure of semigroups on a compact manifold with boundary. *Annals of Mathematics*, 65(1):117–143, 1957.

[MSS$^+$12] Theofilos Mailis, Giorgos Stoilos, Nikolaos Simou, Giorgos B. Stamou, and Stefanos Kollias. Tractable reasoning with vague knowledge using fuzzy $\mathcal{EL}^{++}$. *Journal of Intelligent Information Systems*, 39(2):399–440, 2012.

[MT14] Theofilos Mailis and Anni-Yasmin Turhan. Employing DL-Lite$_R$-reasoners for fuzzy query answering. In Thepchai Supnithi and Takahira Yamaguchi, editors, *Proceedings of the 4th Joint International Semantic Technology Conference (JIST2014)*, volume 8943 of *Lecture Notes in Computer Science*, pages 63–78. Springer, 2014.

[NNS11] Mathias Niepert, Jan Noessner, and Heiner Stuckenschmidt. Log-linear description logics. In Toby Walsh, editor, *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11)*, pages 2153–2158. AAAI Press, 2011.

[Pen09] Rafael Peñaloza. Using tableaux and automata for pinpointing in el. In Valentin Goranko, editor, *TABLEAUX 2009 Wokshop on Tableaux versus Automata as Logical Decision Methods (AutoTab'09)*, 2009.

[Pen10] Rafael Peñaloza. Using sums-of-products for non-standard reasoning. In A.-H. Dediu, H. Fernau, and C. Martín-Vide, editors, *Proceedings of the 4th International Conference on Language and Automata Theory and Applications (LATA 2010)*, volume 6031 of *Lecture Notes in Computer Science*, pages 488–499. Springer, 2010.

[Pos46] Emil L. Post. A variant of a recursively unsolvable problem. *Bulletin of the American Mathematical Society*, 52(4):264–268, 1946.

[PS10a] Rafael Peñaloza and Barış Sertkaya. Complexity of axiom pinpointing in the DL-Lite family of description logics. In Helder Coelho, Rudi Studer, and Michael Wooldridge, editors, *Proceedings of the 19th European Conference on Artificial Intelligence, (ECAI 2010)*, volume 215 of *Frontiers in Artificial Intelligence and Applications*, pages 29–34. IOS Press, 2010.

[PS10b] Rafael Peñaloza and Barış Sertkaya. On the complexity of axiom pinpointing in the $\mathcal{EL}$ family of description logics. In Fangzhen Lin, Ulrike Sattler, and Miroslaw Truszczynski, editors, *Proceedings of the Twelfth International Conference on Principles of Knowledge Representation and Reasoning (KR 2010).* AAAI Press, 2010.

[QJPD11] Guilin Qi, Qiu Ji, Jeff Z. Pan, and Jianfeng Du. Extending description logics with uncertainty reasoning in possibilistic logic. *International Journal on Intelligent Systems*, 26(4):353–381, 2011.

[RGL+13] Yuan Ren, Gerd Gröner, Jens Lemcke, Tirdad Rahmani, Andreas Friesen, Yuting Zhao, Jeff Z. Pan, and Steffen Staab. Process refinement validation and explanation with ontology reasoning. In Samik Basu, Cesare Pautasso, Liang Zhang, and Xiang Fu, editors, *Proceedings of the 11th International Conference on Service-Oriented Computing (ICSOC'13)*, volume 8274 of *Lecture Notes in Computer Science*, pages 515–523. Springer, 2013.

[Ros11] Riccardo Rosati. On the complexity of dealing with inconsistency in description logic ontologies. In Toby Walsh, editor, *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11)*, pages 1057–1062. AAAI Press, 2011.

[SB07] Umberto Straccia and Fernando Bobillo. Mixed integer programming, general concept inclusions and fuzzy description logics. In Martin Stepnicka, Vilém Novák, and Ulrich Bodenhofer, editors, *Proceedings of the 5th EUSFLAT Conference on New Dimensions in Fuzzy Logic and Related Technologies*, pages 213–220. Universitas Ostraviensis, 2007.

[SC03] Stefan Schlobach and Ronald Cornet. Non-standard reasoning services for the debugging of description logic terminologies. In Georg Gottlob and Toby Walsh, editors, *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 355–362. Morgan Kaufmann, 2003.

[Sch91] Klaus Schild. A correspondence theory for terminological logics: Preliminary report. In John Mylopoulos and Raymond Reiter, editors, *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI'91)*, pages 466–471. Morgan Kaufmann, 1991.

[SKP07] Stefan Schlobach, Michel C. A. Klein, and Linda Peelen. Description logics with approximate definitions - precise modeling of vague concepts. In Manuela M. Veloso, editor, *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07)*, pages 557–562, 2007.

[Spa00] K. Spackman. Managing clinical terminology hierarchies using algorithmic calculation of subsumption: Experience with SNOMED-RT. *Journal of the American Medical Informatics Association*, 2000. Fall Symposium Special Issue.

[SSP$^+$07] Giorgos Stoilos, Giorgos B. Stamou, Jeff Z. Pan, Vassilis Tzouvaras, and Ian Horrocks. Reasoning with very expressive fuzzy description logics. *Journal of Artificial Intelligence Research*, 30:273–320, 2007.

[Str98] Umberto Straccia. A fuzzy description logic. In *Proceedings of the 15th Natonal Conference on Artificial Intelligence (AAAI 1998)*, pages 594–599. AAAI Press, 1998.

[Str01a] Umberto Straccia. Reasoning within fuzzy description logics. *Journal of Artificial Intelligence Research*, 14:137–166, 2001.

[Str01b] Umberto Straccia. Reasoning within fuzzy description logics. *Journal of Artificial Intelligence Research*, 14:137–166, 2001.

[Str04] Umberto Straccia. Transforming fuzzy description logics into classical description logics. In *Proceedings of the 9th European Conference on Logics in Artificial Intelligence (JELIA'04)*, Lecture Notes in Computer Science, pages 385–399. Springer, 2004.

[Sun08] Boontawee Suntisrivaraporn. Empirical evaluation of reasoning in lightweight DLs on life science ontologies. In *Proceedings of the 2nd Mahasarakham International Workshop on AI (MIWAI'08)*, 2008.

[SV09] Roberto Sebastiani and Michele Vescovi. Axiom pinpointing in lightweight description logics via horn-sat encoding and conflict analysis. In *Proceedings of the 22nd International Conference on Automated Deduction (CADE-22)*, volume 5663 of *Lecture Notes in Computer Science*, pages 84–99. Springer, 2009.

[Thu15] Aparna Saisree Thuluva. Iterative ontology update with minimum change. Master's thesis, Dresden University of Technology, Germany, 2015.

[W3C09] W3C OWL Working Group. OWL 2 web ontology language document overview. W3C Recommendation, 2009. `http://www.w3.org/TR/owl2-overview/`.